On the definition and recognition of planar shapes in digital images

Pablo Musé

THÈSE DE DOCTORAT
DE L'ÉCOLE NORMALE SUPÉRIEURE DE CACHAN

Présentée par

Pablo MUSÉ

pour obtenir le grade de

DOCTEUR DE L'ÉCOLE NORMALE SUPÉRIEURE DE CACHAN
Domaine: **Mathématiques Appliquées**

Sujet de la thèse :

Sur la définition et la reconnaissance des formes planes dans les images numériques

On the definition and recognition of planar shapes in digital images

Thèse présentée et soutenue à Cachan le 1er octobre 2004 devant le jury composé de :

M. Frédéric CAO	*Chargé de recherche, IRISA / INRIA Rennes*	*Invité*
M. Vicent CASELLES	*Professeur, Univ. Pompeu Fabra, Barcelone*	*Rapporteur*
M. Patrizio FROSINI	*Professeur, Univ. di Bologna*	*Invité*
M. Daniel HUTTENLOCHER	*Professeur, Cornell University, Ithaca, New York*	*Examinateur*
M. Yves MEYER	*Professeur, ENS de Cachan*	*Président*
M. Lionel MOISAN	*Professeur, Univ. Paris 5*	*Rapporteur*
M. Jean-Michel MOREL	*Professeur, ENS de Cachan*	*Directeur*
M. Gregory RANDALL	*Professeur, Univ. de la República, Montevideo*	*Examinateur*

Centre de Mathématiques et de Leurs Applications
ENS CACHAN / CNRS / UMR 8536
61, avenue du Président Wilson, 94235 CACHAN CEDEX (France)

Remerciements

Je tiens tout d'abord à remercier Yves Meyer de m'avoir fait l'honneur d'être le président du jury de cette thèse.

Je suis également très reconnaissant aux rapporteurs et autres membres du jury : Frédéric Cao, Vicent Caselles, Patrizio Frosini, Daniel Huttenlocher, Lionel Moisan et Gregory Randall. Je les remercie vivement d'avoir accepté de participer à l'évaluation de mon travail. Leurs commentaires et questions ont beaucoup contribué à l'amélioration de la version finale de cette thèse.

Je ne saurais pas comment exprimer ma gratitude envers Jean-Michel Morel. J'ai toujours admiré sa générosité et ses qualités scientifiques exceptionnelles, ainsi que l'esprit d'équipe qu'il a su transmettre à son groupe. Sa grande disponibilité, son enthousiasme et son soutien permanent, m'ont permis de mener à bout cette thèse dans les meilleures conditions.

Un grand merci à Frédéric Sur. Sa gentillesse, sa patience et sa rigueur scientifique ont permis que notre étroite collaboration ait été, tout au long de cette thèse, si agréable et enrichissante.

Je souhaite remercier très chaleureusement Frédéric Cao. Ce travail a aussi bénéficié de sa collaboration remarquable. J'ai beaucoup apprécié nos longues et très éclairantes discussions, et ses conseils et encouragements m'ont été précieux.

Tout au long de ces trois ans j'ai eu le plaisir de travailler avec plusieurs collègues, au sein du groupe de traitement d'images du CMLA et ailleurs. Je les remercie tous, particulièrement Andrés Almansa, Julie Delon, Agnès Desolneux, Yann Gousseau, Saïd Ladjal et José Luis Lisani. Je tiens aussi à remercier Patrizio Frosini et son groupe pour notre collaboration et pour m'avoir si chaleureusement reçu à Bologne, ainsi que Lenny Rudin et Pascal Monasse pour nos discussions et les idées transmises lors de ma visite à Cognitech.

Les conditions de travail au Centre de Mathématiques de l'ENS Cachan ont été excellentes. Je remercie tous les membres du CMLA et le personnel technique et administratif, en particulier Véronique Almadovar, Micheline Brunetti et Pascal Bringas. Je tiens également à remercier les doctorants et tous ceux qui ont contribué à la convivialité du labo.

Je tiens ici à remercier la Faculté d'Ingénierie de la Universidad de la República, à Montevideo, et plus particulièrement mes collègues du département de génie électrique. Je voudrais témoigner ma profonde reconnaissance et mon amitié à Gregory Randall. C'est au sein de son groupe au département

de génie électrique que j'ai découvert le traitement d'images. Son soutien et son aide permanents ont été décisifs au moment de poursuivre mes études en France.

Mes amis de Montevideo, malgré la distance, ont toujours été présents, aussi bien que tous ceux que j'ai eu la chance de rencontrer ici. L'assurance de leur amitié et leur support inconditionnel ont contribué à faire de ce séjour en France une très belle expérience. Je leur adresse toute ma gratitude.

Je souhaite remercier du fond du coeur ma mère, mon père et mes frères. Je ne pourrai jamais rétribuer à mes parents tout ce que je leur dois, et ce n'est que grâce à leur effort, leur confiance en moi et leurs encouragements permanents que cette thèse a pu aboutir. Je voudrais enfin remercier Lucía, pour son soutien constant et sa patience, surtout dans les moments les plus durs.

Contents

Chapter 1

Introduction

– Ainsi, il y a trois sortes de lits ; l'une qui existe dans la nature des choses, et nous pouvons dire, je pense, que Dieu est l'auteur – autrement, qui serait-ce ?... [...] Et Dieu, soit qu'il n'ait pas voulu agir autrement, soit que quelque nécessité l'ait obligé à ne faire qu'un lit dans la nature, a fait celui-là seul qui est réellement le lit ; mais deux lits de ce genre, ou plusieurs, Dieu ne les a jamais produits et ne les produira point.
– Pourquoi donc ? demanda-t-il.
– Parce que s'il en faisait seulement deux, il s'en manifesterait une troisième dont ces deux-là reproduiraient la Forme, et c'est ce lit qui serait le lit réel, non les deux autres.

Platon, La République, Livre X.

1.1 What is a shape?

There are certainly many ways to define what a shape is. And the same remark holds, even if we restrict ourselves to geometric shapes, which are the object of this work. However, no matter what definition is adopted, one can hardly imagine a shape without thinking of comparison, similarity or recognition. It is a fact that in order to recognize a shape, we need some *a priori* knowledge of what that particular shape is. Having that knowledge assumes that, in an earlier stage, we have learned something about this shape that enables us to recognize it. Following this point of view, phenomenologists [Att54, Met75] conceive shape as a subset of an image, digital or perceptual, endowed with some qualities permitting its recognition. Such a perceptual object is called a *planar shape*. In that sense, it is sound to define shapes as any part of an image that can be recognized in another image.

Consider now a recognition process defining a shape. Identify that shape and apply to it, for instance, a rigid transformation. One could say that the resulting shape is a different one. However, according to our perception, this new shape will still be recognized as the original one. In that sense, these two shapes should be considered to be equivalent. Hence, a notion of invariance is underlying to the definition of shape. It is a well known fact that humans recognize shapes undergoing a wide range of

transformations and perturbations, that we will describe later. The ideas presented up to here lead us to a general definition of shape:

DEFINITION 1.1 (GENERAL DEFINITION OF SHAPE) *Let W be a set of reference images ("the world of possible images"). Let $\mathcal{I}$ and $\mathcal{I}'$ be any pair of images in W. We call* shape *any common part between $\mathcal{I}$ and $\mathcal{I}'$,* modulo *a class of invariance.*

From a practical viewpoint, this definition is still too general: "common part" has not yet a precise meaning, and the class of invariance has not been specified.

Let us start by specifying the invariance class. In other words, we will present a set of invariant properties of shapes, according to perceptual principles. We will be naturally led to define a representation of images, that complies with some requirements derived from invariance. This image representation will furnish concrete "shape candidates" or "image parts" to the recognition processes leading to shape definition. Then, in the next subsection, we will come to precise what we mean by "common part" in a recognition process. The conjunction of these two analyses will lead us to a more precise definition of shape.

1.1.1 Invariant properties of shape recognition and a well adapted image representation

Since shapes are defined as the parts of an image being recognized in another image, by identifying what are the perturbations or transformations that, applied to those images, do not change the recognition result, we will be naturally led to a more precise definition of shape. Following Lisani *et al.* [LMMM03], the main classes of perturbations which do not affect recognition are:

1. **Changes in the color and luminance scales (contrast changes).** According to gestaltists Attneave and Wertheimer, shape perception is independent of the grey level scale or of the measured colors:

   "The concentration of information in contours is illustrated by the remarkable similar appearance of objects alike in contour and different otherwise. The "same" triangle, for example, may be either white on black or green on white. Even more impressive is the familiar fact that an artist's sketch, in which lines are substituted for sharp color gradients, may constitute a readily identifiable representation of a person or thing." (Attneave [Att54], 1954).

   "I stand at the window and see a house, trees, sky.
   Theoretically I might say there were 327 brightnesses and nuances of colour. Do I have "327"? No. I have sky, house, and trees. It is impossible to achieve "327" as such. And yet even though such droll calculation were possible and implied, say, for the house 120, the trees 90, the sky 117 – I should at least have this arrangement and division of the total, and not, say, 127 and 100 and 100; or 150 and 177." (Wertheimer [Wer23], 1923).

2. **Occlusions and background modification.** Shape recognition can be performed in spite of occlusion and background, as shown in Figure 1.1. The phenomenology of occlusion was thoroughly studied by Kanizsa [Kan79] and his school. Kanizsa argues that occlusion is always present in every day's vision: most objects we see are partially hidden by other ones. Our perception must therefore perform a recognition of partial shapes. Conversely, if a shape occludes a background, its recognition is invariant to changes in the background. The recognition problem in this condition is known in perception psychology as the *figure-background problem*, studied by Rubin [Rub21]. It is the other face of the occlusion problem: a shape is superimposed to a background, which can be made of various objects: how to extract, to single out, the shape from that clutter? This can also be viewed as a dilemma: do we first extract the shape and then recognize it or, conversely, do we extract it *because* we had it recognized?

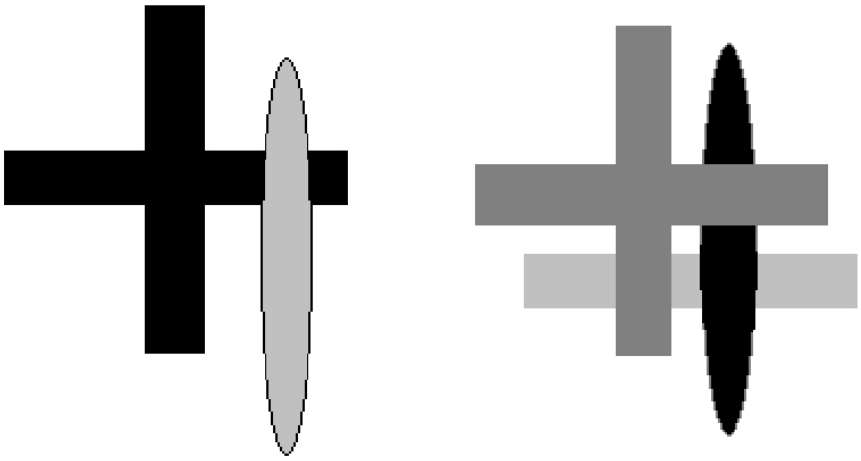


Figure 1.1: *Left: According to the theory of G. Kanizsa and his school, shapes can be recognized even when they undergo several occlusions. Our perception is trained to recognize shapes which are only seeable in part. Here the occluded cross can be easily recovered. Right: the* figure-background *problem. Our perception is adapted to recover a figure on the foreground, independently from the background.*

3. **The classical noise and blur**, inherent to any perception task and to any image generated according to Shannon's theory.

4. **Geometrical distortions or deformations.** The effects of perspective are deeply incorporated in human perception. Humans can recognize objects and shapes under perspective distortion, as long as perspective is not too strong. Recognition is also invariant to elastic deformations, always within some limits.

The four invariant properties we have just described fix some rules or requirements a good image representation should comply with. In the remainder of this subsection we present a derivation of a well adapted image representation, which consists in analyzing these requirements one by one.

1. (a) **The local contrast invariance requirement.** We define a digital image as a function $u(x)$, where $u(x)$ represents the grey level or luminance at x. According to the contrast invariance principle, our first task is to extract from the image a topological information fairly independent from the varying and unknown contrast change function of the optical and/or biological apparatus. One can model such a contrast change function as any continuous increasing function g from $\mathbb{R}^+$ to $\mathbb{R}^+$. The real datum corresponding to the observed u could be as well any image $g(u)$. This simple argument leads to select the level sets of the image [Ser82], or its set of level lines, as a complete contrast invariant image description [CCM99].

   The family of the connected components of the level sets of u, $[u \geqslant \lambda]$, $\lambda \in \mathbb{R}$, is called *upper topographic map*. An image can be reconstructed from its upper level sets by the formula

$$u(x) = \sup\{\lambda, u(x) \geqslant \lambda\}.$$

   We define the level lines as the boundaries of the level sets. There are several frameworks to define the level lines: if u is considered to be a function with bounded variation, the level lines can be defined as a set of nested Jordan curves [ACMM01]. The set of all level lines is called the *topographic map* of the image.

   (b) **The concentration of information requirement.** The local contrast invariance requirement led us to define the set of the image level lines as a complete contrast invariant information. Somewhat in contradiction with this contrast invariance principle, many authors assert, like Attneave, that *"Information is concentrated along contours (i.e., regions where color changes abruptly)"*. One can argue that not all the level lines are really needed to have a complete description in terms of perception. Some of them are due to noise or to small, hardly noticeable, changes in illumination. Thus, it makes sense to prune the tree of level lines by only keeping a selection of the most contrasted level lines. Such a selection (and any other) is not invariant to any contrast change, since it explicitly uses the gradient value. However, it can be shown that it is invariant to affine global contrast changes. Besides, it is probably the most stable selection, in the sense these lines will not vary significantly when "not too strong" contrast changes are applied. A simplification of the level lines tree can be performed by using the method proposed by Desolneux *et al.* [DMM01], which greatly reduces the number of level lines, while preserving the main structures in the image. Figure 1.2 shows an example of such level lines selection (see caption for details).

2. **The occlusion and figure-background requirements.** Even the best adapted choice of level lines is not totally suited to describe image parts. Indeed, when a shape A partially occludes a shape B, the level lines of the resulting image are a concatenation of pieces of the level lines belonging to A and to B. This is shown with a very simple example in Figure 1.3. Even if a shape is not occluded, but simply occludes its own background, there may be no level line

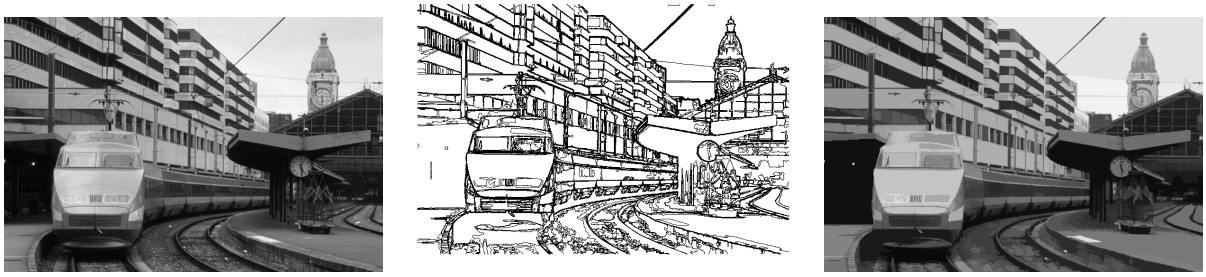


Figure 1.2: *Original image on the left* (83,759 *level lines). Middle: meaningful boundaries* (851 *level lines). Right: reconstruction from the meaningful boundaries. Only* 851 *boundaries remain, while the structure of the image is preserved and perceptual loss is very weak.*

surrounding the whole shape, as shown in figure 1.4. These remarks show us that the Jordan curves of the topographic map, as a whole entity, do not furnish all "shape candidates". In order to overcome this obstacle, a segmentation of these level lines into their parts belonging to different objects is needed. However, since we assume we do not know which are the different objects in the considered images, such a segmentation of level lines is impossible. All we can do is segmenting the level lines in small enough pieces, expecting that most of these pieces will not go through more than one object's boundary.

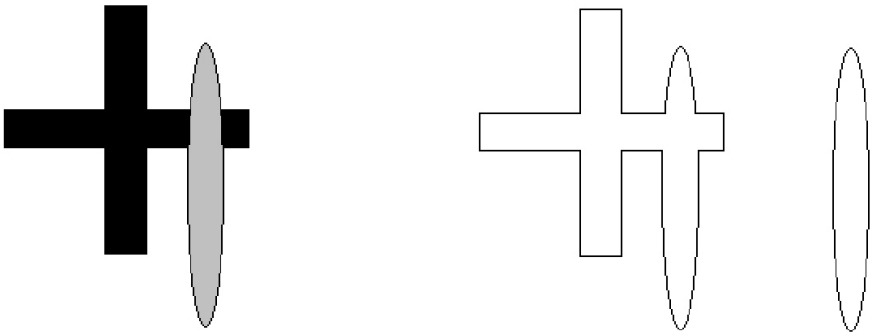


Figure 1.3: *Left: oval occluding a cross, right: the level lines of the resulting image. While the oval's boundaries can be recovered as a full level line, the boundary of the cross concatenates with the oval's boundary. Thus recognition cannot be based on complete level lines, but it can still be based on pieces of level lines.*

3. **The smoothing requirement.** If "common parts" in images subject to noise are still recognized, this means shape information has not been affected by that noise. In that sense, noise can be viewed as introducing details which are much too fine (in relation to the essential shape information) to be perceptually relevant, in terms of recognition. This idea was also pointed out by Attneave in 1954: *"It appears, then, that when some portion of the visual field contains a quantity of information grossly in excess of the observer's perceptual capacity, he treats those*

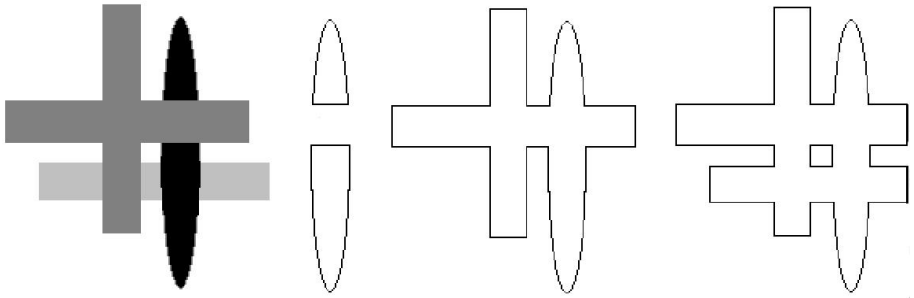


Figure 1.4: *Left: Cross on a background with an oval occluding a rectangle. The cross is wholly in view. All the same, its shape does not appear as a level line because of the background. As in Figure 1.3, one sees that the level lines must be broken into pieces to get clues of each single shape.*

components of information which do not have redundant representation somewhat as a statistician treats "error variance", averaging out particulars and abstracting certain statistical homogeneities." Hence, a correct image representation, which does not get lost in textural details, asks for a previous smoothing. Figure 1.5 illustrates recognition in the presence of noise. The object on the right was obtained by smoothing the one on the left. By comparing them, we immediately recognize the same shape.

Notice that the *smoothing requirement* is somehow consistent with the *concentration of information requirement*.

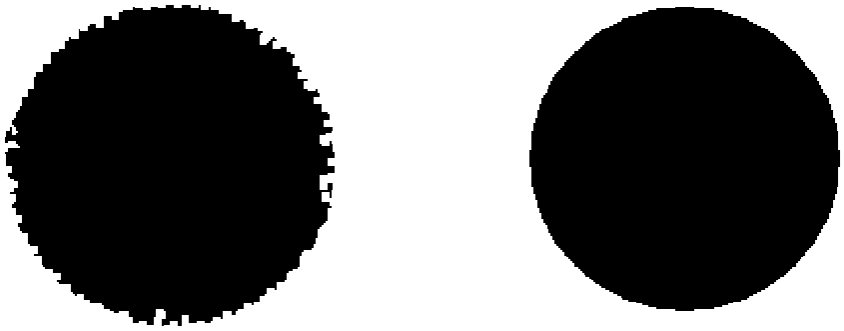


Figure 1.5: *One can immediately see that both objects are disks, with approximately the same size. The second one is obtained from the first by the affine curvature equation [AGLM93]. The main ideas behind such a curvature equation was anticipated by Attneave, who proposed to smooth silhouettes by blurring and then enhancing the resulting image to get a smooth silhouette: "somewhat as if the photograph of the object were blurred and then printed on high-contrast paper".*

4. **Geometric invariance requirements.** Image representations (a set of meaningful level lines, for instance) have to be invariant to weak projective transforms. Allowing invariance to any projective transformation does not make sense, since we cannot recognize shapes under strong

perspectives. Besides, it can be shown that all planar curves within a large class can be mapped arbitrarily close to a circle by projective transformations. This result was reported by Aström in [Åst95], where it is also shown that given a finite set of m Jordan curves $\mathcal{C}_1, \ldots, \mathcal{C}_m$, one can find a Jordan curve $\mathcal{C}$ and m projective transformations $p_1, \ldots, p_m$, such that $p_i(\mathcal{C})$ is arbitrarily close to $\mathcal{C}_i$, for all $i \in \{1, \ldots, m\}$. Hence, in general, schemes based on projective normalization of Jordan curves are not possible. Another argument against general projective invariance is that, despite some interesting attempts [FK95], there is no practical way to define a projective invariant local smoothing. From this viewpoint, affine invariant smoothing is the "best" we can do [AGLM93]. The two arguments we have just presented indicate that affine invariance is a reasonable geometric invariance requirement. Indeed, since we are interested in weak perspective distortions, we can use local affine transformation approximations, for which invariant smoothing is possible. Thus, affine invariant smoothing and affine normalization of meaningful level lines are requested.

Deriving an image representation

To end with this subsection, let us summarize the four invariance requirements, and the constraints they impose to an image representation based on shape information. The *local contrast invariance* led us to define the topographic map as a complete contrast invariant representation. Then, the *concentration of information requirement* asked for selection of a set of meaningful level lines, that are roughly the level lines which are long and contrasted enough (a more precise definition is not necessary for the moment, and will be given later). It follows, from the *occlusion and figure-background requirements*, that small pieces of meaningful level lines are to be considered. Finally, by combining the *smoothing requirement* and the *geometric invariance requirement*, it follows that pieces of meaningful level lines should be smoothed with a local scheme invariant to affine transformations, and then local affine normalization of these pieces has to be performed, leading to the representation we propose. Now we are able to define *shape elements*, elementary structures which are likely to be recognized in images:

Definition 1.2 *We call* shape element *of an image any piece of meaningful level line of the image, affine invariantly smoothed.*

1.1.2 Defining the "common parts" of two images

At the very beginning of this chapter, we defined a *shape* as any common part between two images, *modulo* a class of invariance. The first precision we have to make is that here, "common" does not mean "identical". In the preface to his book "The Statistical Theory of Shape" [Sma96], Christopher G. Small also defines shapes based on invariance: *"In general terms, the shape of an object, data set, or image can be defined as the total of all information that is invariant under translations, rotations, and isotropic rescalings. The two objects can be said to have the same shape if they are similar in the*

sense of Euclidean geometry. For example, all equilateral triangles have the same shape, and so do all cubes. In applications, bodies rarely have exactly the same shape within measurement error. In such cases the variation in shape can often be the subject of statistical analysis." While we do not totally agree with Small's first assertion (in our opinion the Euclidean group is not large enough and some affine transforms or even weak perspectives should be included), we think that the very last sentence clearly represents what we mean by "common". Hence, underlying this notion, there is a *recognition threshold* ε and a resemblance measure (up to invariance) between parts in images $\mathcal{I}$ and $\mathcal{I}'$, such that if this measure is less than ε, the considered parts are said to be "common parts". Besides, it seems natural to model the resemblance measure not only as a function of the considered parts in $\mathcal{I}$ and $\mathcal{I}'$, but also on the context, which is given by the world of possible images W.

Now that we have chosen to represent image parts as sets of *shape elements*, we need a resemblance measure between two such elements $\mathcal{S}$ in $\mathcal{I}$ and $\mathcal{S}'$ in $\mathcal{I}'$, *up to an affine transformation*. This measure, which we will denote by NFA, can be defined as a function $NFA(\mathcal{S}, \mathcal{S}', W)$, where affine normalization of $\mathcal{S}$ and $\mathcal{S}'$ before comparison is implicit. This function is defined to have this property: the lower the NFA is, the more the resemblance between shape elements is confirmed.

1.1.3 A more precise notion of geometric shape

All discussions made up to here motivate the following definition.

DEFINITION 1.3 (ε-MEANINGFUL MATCH) *Let W be a set of reference images ("the world of possible images"). Let $\mathcal{I}$ be any image in W, and $\mathcal{S}$ a shape element from $\mathcal{I}$. Let also $\mathcal{S}'$ be a shape element within images in W. Denote by ε a fixed positive real number. We say $\mathcal{S}$ and $\mathcal{S}'$ match ε-meaningfully if $NFA(\mathcal{S}, \mathcal{S}', W) \leqslant \varepsilon$.*

An ε-meaningful match between two shape elements is a consequence of a recognition process, since shape elements are image structures likely to be recognized, and the fact they match ε-meaningfully means they are among the "common parts" between some images.

Definition 1.3 is much more precise than Definition 1.1, but is general enough to be at the origin of several conceivable applications. Let us briefly describe some of them, that were explored in this work.

Autosimilarity

By autosimilarity we mean identification of shapes that are repeated within an image. Figure 1.6 shows examples of images having the autosimilarity property. Taking $W = \{\mathcal{I}\}$ and $\mathcal{S}, \mathcal{S}'$ both in $\mathcal{I}$, Definition 1.3 leads to the autosimilarity application.

Image comparison

If one takes $W = \{\mathcal{I}, \mathcal{I}'\}$, $\mathcal{S}$ in $\mathcal{I}$ and $\mathcal{S}'$ in $\mathcal{I}'$, Definition 1.3 corresponds to an image comparison problem. Finding the common structures between the two images in figure 1.7 could be a possible ap-

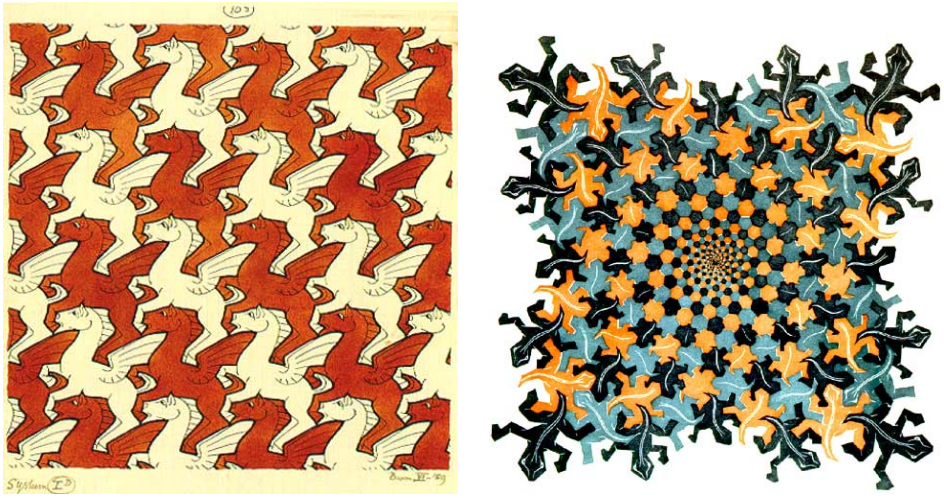


Figure 1.6: *"Pegasus" (left) and "Development II" (right), by M.C. Escher. In the "Pegasus" painting, the tiling is obtained just by translation of shape elements. In "Development II" the underlying transformations are much more complex. The transformations between lizard at close scales seem to be well approximated by similarity transforms (changes in orientation, rotation and scale). It is clearly not the case when lizards are at very different scales.*

plication. At first sight this problem looks rather simple, since we immediately recognize the similar parts between these two images. However, people from the computer vision field know that comparing images like these two paintings is not that easy. Contrast in both images is significantly different, and most part of the boundaries are in fact subjective contours (see the corresponding meaningful level lines in figure 1.8).

Figure 1.7: *Two versions of* Saint Jerôme *by Georges de la Tour.*

Shape recognition applications

If one chooses W to be a generic, representative database of images, and takes $\mathcal{S}$ in $\mathcal{I}$, and $\mathcal{S}'$ in $\mathcal{I}'$ spanning the database W, Definition 1.3 leads to a shape retrieval application. If instead of that

Figure 1.8: *Meaningful level lines of the images in figure 1.7. Many contours are not real but subjective.*

W one takes a database of images containing only variations of $\mathcal{S}$, the resulting application should be "discrimination" or "precise recognition". Let us illustrate the two different approaches with an example. Assume one wants to find the letter "m" in a database composed by: (a) pages scanned from a book, (b) a dictionary of fonts for "m". While both problems admit the same formulation, different choices for the database act as different context information, and recognition should depend on that choice. What we want the recognition method to do is to retrieve all letters "m" in the (a) case, and only the most similar fonts in the (b) case.

1.2 Towards a higher level description of shapes

Looking at figure 1.9, one can recognize on the right image, a detail of Uccello's painting shown on the left. These two images were taken from different websites on the Internet, and present different colors and different compression rates. They are also at very different scales. If we look at their meaningful level lines in figure 1.10, we see that, while some parts are similar, the majority of these level lines are significantly different. Despite of that, shape information in the common parts of the images seems to be almost identical. This analysis shows that some kind of coarser, or more global description of shapes is needed.

Up to here we have only considered recognition processes involving partial shape information, since the notion of ε-meaningful match relies on *shape elements*. Given the amount of differences between the level lines of both images, we can by no means pretend to recover the whole common structure by recognizing *shape elements*, without using any context information. However, even if a few shape elements in common are found, it can be enough to get sure detections of the global common structures, by combining the spatial information furnished by these shape elements. The idea is then to find spatial coherence between shape elements, and hence to define "shapes" as groups of shape elements. In order to do it, we can proceed in a similar way as we did for the definition of ε-meaningful matches.

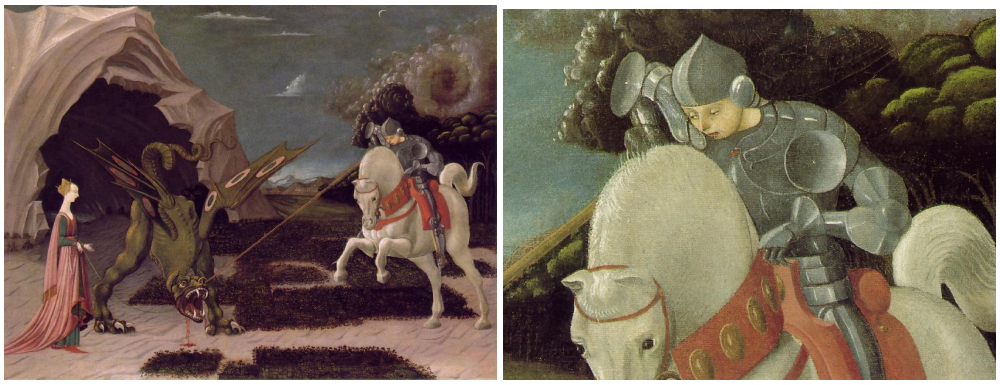


Figure 1.9: *Left: "St. George and the dragon", by Paolo Uccello. Right: detail of the painting.*

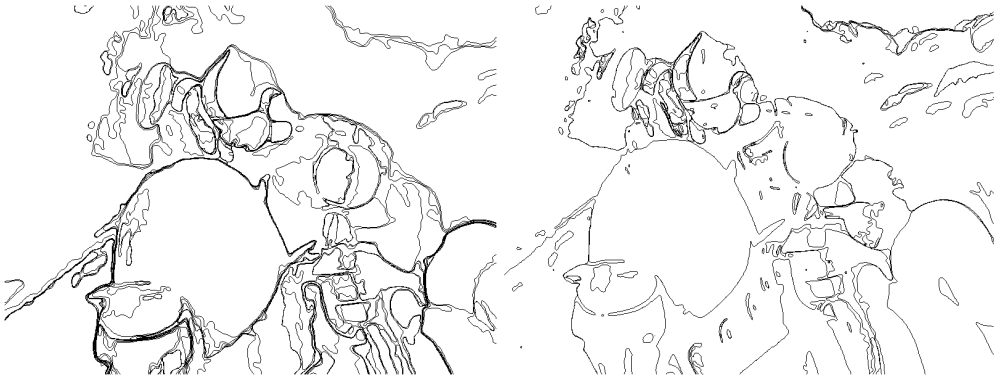


Figure 1.10: *Meaningful level lines of the images in figure 1.9. The right image are the level lines of the detail image. Left: a zoom on the meaningful level lines of the corresponding part from the complete painting. Because of the different scales and the jpeg compression, the level lines differ significantly.*

To begin with, we have to define a measure of resemblance between *groups of shape elements*. It is sound to do it, indirectly, by considering the affine transformations between pairs of pieces of level lines defining shape elements. Hence, instead of defining a measure of resemblance between groups of shape elements, we will define a *spatial coherence measure* on groups of affine transformations. To each ε-meaningful match $(\mathcal{S}, \mathcal{S}')$, we can associate an affine transformation T. The *spatial coherence measure* of a group of transformations G included in $\mathcal{T}$, where $\mathcal{T}$ is the set of all affine transformations associated to ε-meaningful matches, will be denoted by $NFA_g(G)$. This measure function is to be defined to give the following information: the lower is NFA_g, the more the resemblance between two groups of *shape elements* is confirmed. As an indirect resemblance measure involved in a recognition process, a *recognition threshold* ε_g on the *spatial coherence measure* has to be derived, in order to decide if two groups of *shape elements* must be matched. This discussion motivates the following definition.

DEFINITION 1.4 (ε_g-MEANINGFUL GROUP) *Let $\mathcal{I}$, $\mathcal{I}'$ be two images in the "world of images" W. Let $\mathcal{T}$ be the set of all affine transformations associated to the ε-meaningfully matched shape elements between $\mathcal{I}$ and $\mathcal{I}'$. We say a subset $G \subset \mathcal{T}$ is an* ε_g-meaningful group of transformations *if $NFA_g(G) \leqslant \varepsilon_g$.*

Notice that an ε_g-meaningful group of transformations is, by definition, associated to a group A of shape elements in image $\mathcal{I}$ and another group B of shape elements in image $\mathcal{I}'$, such that shape elements in B match ε-meaningfully with shape elements in A, and all matches exhibit a spatial coherence property. Therefore, in the sequel, we will refer indistinctly to *meaningful groups of matches* or *meaningful groups of transformations*, the sense being clear from the context.

1.3 A method to define shapes by recognition

Shapes can be defined by means of recognition processes, as any part of an image that can be recognized in another image. Hence, every shape recognition algorithm that is able to automatically give sure recognition thresholds, can lead to a method to define shapes. Deriving unsupervised decision thresholds involved in shape recognition is one of the central problems of this thesis.

Each step in a shape recognition method is an interesting problem *per se*, and is the object of a deep study in this work. Among these steps, some of them deal with decision thresholds, and we have done significant efforts in order to establish a methodology leading to sure, unsupervised decision thresholds. In this section we briefly describe and illustrate with an example, the main steps of a shape recognition algorithm. Many processes are involved in each of these steps, thus requiring several parameters. The majority of them are related to practical aspects of the method, and can be fixed once for all by numerical arguments. The rest of the parameters are decision parameters, which are the most delicate ones. In the recognition method we propose, all these thresholds are automatically derived by statistical arguments based on perceptual principles, and lead to sure detections. Hence, in that sense, the method we propose is a robust and *parameterless* method, and can then be considered as *a method to define shapes*.

1. **Shape extraction.**
   In section 1.1, following Lisani *et al.* [LMMM03], we presented a representation of image contents based on *shape elements*. This representation is well suited for shape recognition problems, since it is based on perceptual principles. Shape elements were defined as small pieces of level lines, smoothed with an affine invariant filter. Many steps are involved in the extraction of *shape elements*. The first one consists in extracting the topographic map. This can be done with fast a numerical method proposed by Monasse and Guichard [MG00]. The second step consists in selecting a subset of "perceptually significant" level lines from the topographic map. Indeed, as we said in section 1.1, considering all level lines in an image is neither feasible (from a computational viewpoint), nor desirable (because the provided information is highly redundant). We therefore aim at extracting the interesting (*i.e.* constrasted and long "enough") level lines from any image. We propose an algorithm to automatically extract these *meaningful level lines*, which improve the method presented by Desolneux *et al.* in [DMM01]. Figure 1.11 shows that the extracted meaningful level lines outline objects in images, and considerably reduce the amount of information to deal with (the number of level lines is usually reduced by a factor between 100 and 1000, depending on the structure of the image). This step involves a single decision parameter whose value is fixed once for all, based on statistical arguments derived from perceptual principles.

2. **Shape smoothing.**
   Most of the time, the meaningful level lines extracted in the preceding step suffer from image quantization and noise, and smoothing is required (see figure 1.12). Smoothing reduces also the amount of information endowed in the level lines to the "essential" shape information. The Geometric Affine Scale Space [ST93, AGLM93] smoothing is fully convenient here, since it commutates with affine transforms. The affine curve shortening equation characterizes this filter:
   $$\frac{\partial x}{\partial t} = |\mathrm{Curv}(x)|^{\frac{1}{3}}\, \mathbf{n}(x),$$
   where x is a point on a level line, $\mathrm{Curv}(x)$ the curvature and $\mathbf{n}(x)$ the normal to the curve, oriented towards concavity. We use a fast implementation due to L. Moisan [Moi98]. This step is also parameterless, since the scale at which the smoothing is applied is fixed and given by the pixel size.

3. **Shape local invariant encoding.**
   Once meaningful level lines are extracted and smoothed, several shape elements are derived from each of them. Shape elements can be considered invariant to contrast changes, and can deal with occlusion. In order to comply with the geometric invariance requirement (section 1.1.1), shape elements have to be affine normalized with respect to affine transformations (similarity normalization can be enough for many applications). The extraction of shape elements from smoothed, meaningful level lines is based on bitangent lines (a straight line which

is tangent to the curve at two different points) and flat pieces (a piece upon which the curve can be approximated by a segment).

Bitangent lines have been widely used to build local invariant descriptions of curves (Lamdan *et al* [LSW88], Rothwell [Rot95], Lisani [Lis01]). Let us remark that the affine shortening reduces the number of bitangents in the curve (this fact is mathematically proved in [AST98]). Thus, smoothing speeds up the encoding step.

If local description is only based on bitangents, no shape element in convex curves can be extracted. This can be solved by considering also lines given by flat pieces on curves (*i.e.* a curve piece on which the direction of the tangent does not vary too much). Coupled with bitangent lines, flat pieces on curves allow to locally describe nearly all kind of curves.

For computational reasons, each normalized shape element is uniformly subsampled, leading to a representation we will call *code*. To give an idea of orders of magnitude, the target image in figure 1.11 leads to 1759 codes, while the database image leads to 2997 codes. This step only involves a few normalization parameters, which do not have to be tuned by the user since they are fixed once for all, according to requirements of the following shape matching step.

4. **Shape Matching.**
When the three preceding steps are applied to an image, its shape contents is represented by a set of *codes*, corresponding to all normalized shape elements in the image. Then a fundamental problem is to decide whether or not two shape elements, extracted from images within the "world of images" W, are alike. While extraction of boundaries, smoothing and normalization of curves have been widely addressed in several research fields such as image processing or computer vision, the decision step has not been the subject of a deep and systematic study in shape recognition. In this thesis we propose a decision rule that permits to answer *yes* or *no* to the question "does that shape look like a target shape?". This rule is derived from an *a contrario* model of matching process in "random" situations. The general use of *a contrario* models is in keeping with a general detection methodology developed by J.-M. Morel's group [DMM04, DMM03a, DMM00]. The list of *codes* extracted from images in W that match *codes* in a given image $\mathcal{I}$ can thus be drawn up. For each match, a quality or resemblance measure (the number of false alarms, that was denoted by NFA in sections 1.1.2 and 1.1.3) is estimated. The NFA of a match between a target shape element $\mathcal{S}$ and a shape element from W, can be interpreted in statistical terms, as the average number of shape elements in W that look like $\mathcal{S}$ "just by chance". This provides a way to control the number of false alarms, and matching decision can be made by thresholding the NFA. Notice that codes are only matched according to a similarity criterion, and up to this step without taking into account the relative positions of shape elements. See figure 1.13.

5. **Shape grouping.**
Each pair of matching codes (or the corresponding pair of *shape elements*) leads to a single

transformation between the two images, which can be represented as a pattern in a parameter space. The grouping step consists in identifying meaningful clusters among these transformation patterns, based on the ideas we presented in section 1.2. Data clustering methods have been widely studied in many research fields [JMF99]. The goal is to find "natural" groups in a set of data, so that patterns within each clusters are more closely related to one another than to patterns assigned to different clusters. A central problem in clustering methods is *cluster validity assessment*. All clustering algorithms produce clusters, whether they exist or not; do the detected clusters correspond to "natural" clusters? How many "natural" clusters are in the data?

In this work we propose a method to assess the validity of clusters detected by a hierarchical clustering algorithm. This method is also in keeping with a general *a contrario* detection methodology. We build up a *background model* which assumes, roughly speaking, that patterns are thrown in the parameter space at random. This background model is "data-influenced", unlike the one proposed by Desolneux *et al.* in [DMM03a] for the detection of clusters of dots in a plane. Then, clusters are detected *a contrario*, as large deviations from the background model. A number of false alarms NFA_g is associated to each cluster issued from the hierarchical clustering algorithm. The number NFA_g of a group G is a measure of how likely it is that a group showing similar characteristics to G was generated "by chance", as a realization of the background process. The lower is $NFA_g(G)$, the more unlikely G is generated by the background, and hence, the more meaningful is G.

A threshold on $NFA_g(G)$, automatically set once again by statistical arguments, decides if G can be a "natural" cluster or not. Clusters whose NFA_g is below the threshold are called *meaningful clusters* or *meaningful groups*. Consider now the following situation. Assume three meaningful groups A, B and C, are found on a data set, such that A and B are included in C. The problem consist in making a choice between two possible data representations: two separate clusters A and B, or one single cluster C. Decision rules for this problem are known as *local validity rules* in the cluster validity literature [Boc85, Gor99]. In this work we propose a novel local validity rule that, combined with the measure NFA_g, proved to be very efficient in detecting clusters of transformations.

Figure 1.14 illustrates the result of our method when applied to the comparison of the original images in figure 1.11. The validity assessment led to two meaningful clusters of transformations, corresponding to the two groups of shape elements shown in figure 1.14.

6. **Shape recognition verification by registration.**
To each identified group of transformations, we can associate the transformation that provides the best fit between corresponding groups of shape elements (in the least squares sense). Mapping the target image on the database image by this transformation allows to check that two similar objects are retrieved in both images. Figure 1.15 illustrates the result of this registration step.

All these steps will be detailed and widely discussed along the following chapters. To end with this chapter, in the next section we briefly describe the plan of this dissertation.

Figure 1.11: *Two examples of level lines extraction. Left: original images. Middle: a subset of the level lines (quantization step for the gray levels is* 20*). Right: local meaningful level lines, obtained by the algorithm described in Chapter 4. The upper image will be referred as the "target image". It generates* 385 *local meaningful level lines (among* $359,263$*). The lower image will be referred as the "database image". This latest image generate* 577 *local meaningful level lines (among* $462,912$*). We aim at matching shapes in the target image with shapes in the database image. Looking closer to the images, we notice that the images show in fact dissimilarities: the actors' faces are stylized in the database image whereas they correspond to a photography in the target image, the word "Casablanca" slightly differs,* etc.

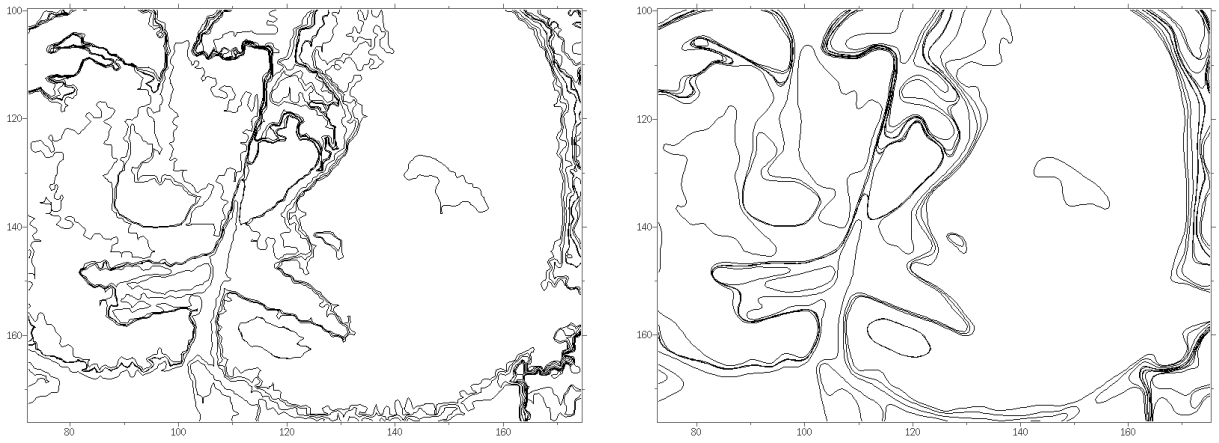


Figure 1.12: *Zoom on the extracted level lines (on the left). Quantization effects can be seen. After a slight smoothing, these effects disappear (on the right).*

Figure 1.13: *Meaningful matches between codes of the target image (on the left) and the database image (on the right). Each piece of shape in the left image matches with a piece of shape in the right image, up to a certain similarity transform. Some false matches can be seen, as predicted by the theory ("AR" matches with the piece of shape on the right of Humphrey Bogart's head for instance).*

Figure 1.14: *Two maximal meaningful clusters are identified in the transformation space. This figure shows the pairs of shape pieces corresponding to each of them (top and bottom). The first group was generated by the shape pieces that belong to the word "Casablanca". The second group was generated by the actors' faces. "Casablanca" and actors' faces respective position and scale is actually not the same in both images. At this step of our algorithm, wrong matches are eliminated since they do not make up a meaningful cluster on their own.*

Figure 1.15: *For each identified transform cluster, an "average" similarity transform is estimated. In order to check the consistency of the method, the target image is mapped over the database image, with each estimated transform. On the left, the average transform corresponding to the first maximal meaningful cluster maps both "Casablanca" word one on the other. On the right, the transform corresponding to the second maximal meaningful group maps the faces. Note the difference in scales.*

1.4 Overview of this thesis

This dissertation is organized in two main parts, separated by a small *intermezzo*. **The first part** deals with the extraction of shape information from images, and with the representation of such information based on the *shape elements* defined in the previous sections. Then, the remainder of this part is devoted to the problem of finding correspondences between shape elements, with a special emphasis on the matching decision problem. A short ***intermezzo*** made of two chapters, explores the matching decision when alternative representations of the shape information are used. **The second part** addresses the recognition of shapes by grouping those matches between shape elements that show some spatial coherence property. As much as it was done in the first part, a major importance is accorded here to decision models. All along this thesis, decision issues involved in the detection of correspondences are analyzed following the general *a contrario* detection methodology.

Part I: The recognition of partial shapes

This part (Chapter 2 to Chapter 9) covers the following topics: shape extraction, shape smoothing, invariant encoding of shapes, shape matching and matching decision.

Chapter 2 presents a large survey of general shape recognition theory. As we already mentioned in the previous section, shape recognition methods consist, in general, of three interdependent main steps: encoding of shapes (possibly an invariant encoding, up to a transformation class), shape comparison, and matching decision. We describe the weaknesses and strengths of frequently used algorithms, then we explain why and how we build up our recognition method.

Chapter 3 is devoted to the topographic map and to the bilinear interpolation of grey level images. The complete image representation given by the topographic map is well suited for shape analysis and recognition, since it is based on the geometrical information of images, and can be embedded in a tree structure. However, since the level lines of digital images (zero order interpolates) suffer from pixelization effect, shapes cannot be accurately described. Higher order interpolates are then to be considered.

Not all the level lines contained in the topographic map of an image are really necessary to have a complete description in terms of perception. **Chapter 4 presents a method to select the most meaningful level lines based on perceptual principles.** The selection and extraction of level lines is based on statistical arguments, leading to a parameter free algorithm. It permits to roughly extract all pieces of level lines of an image, that coincide with pieces of edges. The proposed method aims at improving the original method proposed in [DMM01]. We introduce a multiscale approach that makes the method more robust to noise. A more local algorithm is introduced, taking local contrast variations into account. The contents of this chapter corresponds essentially to the article [CMS04], co-written with F. Cao and F. Sur.

The set of meaningful level lines is a good compromise between compactness and completeness of shape information in an image. However, since these lines may be subject to noise affecting the

essential shape information, a good shape representation asks for a previous smoothing. **In Chapter 5 we describe the affine scale space**, a scale space fully consistent with affine invariant recognition. Moisan's fast implementation of the affine shortening [Moi98] is also presented.
This chapter does not contain original contributions, but we include it here for the sake of completeness of this dissertation. Sections 5.1 and 5.2 closely follow the article *"On the theory of planar shape"* by Lisani, Moisan, Monasse and Morel [LMMM03], and the book in preparation by Guichard, Morel and Ryan [GMR04].

In **Chapter 6 we present an algorithm to detect flat pieces in curves.** Bitangent lines are well-known to be of high interest to build up local invariant curve descriptions [LSW88, Rot95], but they do not enable to encode convex curves (which show none). Moreover, some non-convex curves may show some small oscillations over a straight portion on which numerous bitangent lines are detected, leading to unstable and unreliable invariant descriptors. Another robust direction which is useful to build local invariant descriptions is given by flat pieces on curves (*i.e.* a piece of curve on which the direction of the tangent lines does not vary too much). Coupled with bitangent lines, flat pieces enable to encode nearly all kind of curves.
The work presented in this chapter (joint work with F. Cao, J.-M. Morel and F. Sur) will soon be submitted and available as a preprint.

Chapter 7 presents two methods to encode shapes elements. Both methods build representations invariant up to either similarity or affine transformations. The first one is semi-local, and can deal with occlusions. However, as we will see, it does not allow to encode all of the extracted boundaries. We then introduce a second algorithm to globally encode those boundaries that have not been encoded by the semi-local method. Both algorithms are based on bitangent lines and flat pieces, making the encoding very stable.

While shape comparison, shape matching and shape extraction have been the subject of many researches, the decision step has been rarely addressed. In **Chapter 8 we present a general matching decision framework.** Concerning shape elements, this framework permits to answer *yes* or *no* to the question "does a shape element $\mathcal{S}'$, extracted from the world of images, looks like a target shape element $\mathcal{S}$?", and to measure the confidence level in this answer. A database of shape elements extracted from the world of images being given, with each target shape element $\mathcal{S}$ and each distance δ we associate its *number of false alarms* $NFA(\mathcal{S}, \delta)$, namely the expectation of the number of shape elements at distance δ from $\mathcal{S}$ in the database. Assume that the $NFA(\mathcal{S}, \delta)$ is very small with respect to 1, and that a shape element $\mathcal{S}'$ from the database is found at distance δ from $\mathcal{S}$. This match could not occur just by chance and is therefore a meaningful detection. Its explanation is usually the common origin of both shape elements. This *a contrario* detection framework is in keeping with the general detection methodology followed in this thesis.
The *a contrario* model proposed in this chapter was presented in Icip 2003 [MSCG03]. The contents of this chapter mainly corresponds to the preprint [MSC+04]. Chapters 7 and 8 were also partially extracted from this preprint.

To end up with the first part of this dissertation, **Chapter 9 presents several experiments** that il-

lustrate and validate all the stages of the shape element matching method, proposed in the former chapters.

***Intermezzo*: Meaningful matches based on alternative descriptions**

This short part is independent of the general line followed in this thesis.
The method presented in Chapter 8 can be adapted to shapes described by other features, whenever these features are (almost) statistically independent. When such a set of features is available, low numbers of false alarms can be reached. In **Chapter 10** we address the shape matching problem in this framework, by representing shape elements with sets of features provided by principal components analysis (PCA). Although these features are not necessarily independent, they are at least uncorrelated. Experiments show that a PCA method based on shape elements is not as well adapted as the strategy proposed in Chapter 8.

Chapter 11 addresses also the shape matching problem based on the general framework presented in Chapter 8, but the considered shape features consists here in size functions [FL99, FL01]. The results seem to be still valid in spite of the uncompleteness of the shape description given by size functions. This chapter presents some preliminary results of a work in progress (Galileo project with P. Frosini's group in the University of Bologna; joint work with P. Frosini, M. d'Amico, D. Giorgi, F. Tomassini and F. Sur).

Part II: Shape recognition as a grouping process

This part (Chapters 12 to 15) is devoted to the detection of shapes as groups of shape elements. It covers the following topics: *clustering* or *grouping* of patterns, clustering algorithms and cluster validity analysis. These general techniques from pattern recognition theory are applied here in order to detect groups of spatially coherent matches between shape elements.

In Chapter 12 we present a method to detect natural groups in a data set of patterns, based on hierarchical clustering. A measure of the meaningfulness of clusters (the NFA_g), derived from an *a contrario model* assuming no structure in the data, provides a way to compare clusters, and leads to a cluster validity criterion. This criterion is applied to every cluster in the nested structure. While all clusters passing the validity test are meaningful in themselves, the set of all of them does not necessarily reveals the structure of the data set. However, by selecting a subset of the meaningful clusters, a good data representation can be achieved. We propose a method combining a new local stopping rule (a merging criterion, also derived from the *a contrario model*) with a selection of local maxima of the meaningfulness with respect to inclusion in the nested hierarchical structure.
In Chapter 13, the general method to detect cluster of patterns, developed in Chapter 12, is adapted to the recognition of shapes in images. The first part of this dissertation deals with local representations of shape contents in images. Consequently, common parts between images were described in terms of matched shape elements. The recognition of "global shapes" needs for an integration of the recognized partial shapes. Each pair of matching shape element leads to a unique transformation

between images, which can be represented as a pattern in a transformation space. Hence, spatially coherent meaningful matches correspond to clusters in the transformation space, and their detection can then be formulated as a clustering problem, which can be solved as a particular case of the theory developed in Chapter 12. By this means, shapes can be detected with extremely high levels of confidence.
Chapters 12 and 13 correspond to joint work with F. Cao, J. Delon, A. Desolneux and F. Sur. They will soon be submitted and available as preprints.

Chapter 14 presents several experiments that illustrate the whole recognition method presented in this thesis. The pretty good results validate the theory.

Finally, **in Chapter 15 we present some conclusions**, as well as perspectives and future work.

Part I

The recognition of partial shapes

Chapter 2

Shapes: feature extraction and recognition

Abstract: This chapter is a large survey of general shape recognition theory. Shape recognition algorithms generally consist of three interdependent steps, namely shape encoding (possibly an invariant encoding, up to a transformation class), shape comparison, and matching decision. We explain the weaknesses and advantages of the commonly used algorithms, then we explain why and how we build up our algorithm.

Résumé : Ce chapitre est une revue de la reconnaissance de forme en général. Les algorithmes de reconnaissance de forme consistent en trois étapes interdépendantes : le codage des formes (éventuellement un codage invariant selon certaines classes de transformations), la comparaison des formes, et la décision d'appariement. Nous expliquons les forces et faiblesses des méthodes les plus classiques, et nous expliquons nos choix pour l'algorithme proposé.

2.1 Shape recognition general outline

In its general form, *recognition* can be defined as the ability to identify elements based on prior knowledge. If we restrict the scope to visual recognition, prior knowledge comes necessarily from images. In that sense, in the former chapter, we defined (geometric) shapes as some geometric information from an image, that can be recognized in another image. The geometric information in an image is completely described by its topographic map [CCM99], and, from a perceptual viewpoint [Wer23], the information is in fact concentrated in a reduced subset of level lines. Shapes, as perceived geometric structures, should be contained in this subset of level lines. Assume that, based on some knowledge derived from previous recognition or identification experiences, we have already extracted a large set of shapes, from the "world of images". Then, in such conditions, *shape recognition* is the process aiming at finding out whether a given shape lies or not in a set of shapes, up to a class of invariance imposed by perceptual principles (see chapter 1).

Shape recognition is a complex task, involving several "sub-tasks". In this work, we are particularly

interested in the decision process of shape recognition. We want to derive methods or rules to decide whether two shapes are alike or not. Since the decision process cannot be conceived as a process isolated from the other tasks involved in recognition, a previous analysis of these tasks is needed. Indeed, the decision process widely depends upon the former steps of shape recognition process. Thus, in a method for recognizing shapes, each step is crucial, and any wrong choice for one of them would spoil the final result. General shape recognition methods can be decomposed into three main stages:

1. **Feature extraction.** What enables shape recognition is the fact that shapes have specific characteristic features. Shapes, or shape elements, have to be described in a suitable way. In general, dealing with exhaustive descriptions is computationally unfeasible, and a set of features has to be extracted from shapes. Hence, the feature extraction / selection problem consists in defining a set of features (as small as possible) leading to a high discriminatory power: the more different two shapes are (in a certain sense), the more different their sets of features should be. This requirement can be thought as a *completeness* requirement: two shapes are close if and only if their sets of features are close. As an example, a small set of invariant moments is not a complete representation of a shape, since quite different shapes may have similar moments.

   Shape features can be either global (the value of each feature depends on the whole shape), either local or semi-local (each feature is based on a particular point, or on a part of the shape). Descriptions based on sets of local features are to be preferred, because of the occlusion problem. Features can also be classified according to their degree of geometric invariance. Since shape recognition has to be invariant up to a certain group of geometric deformations, when features are not invariant, the further comparison process has to take invariance into account. We will thus give some details on the various usual shape features.

2. **Matching.** This second stage strongly depends on the feature extraction step. The core of this step is the definition of a distance or dissimilarity measure between features describing shapes. When each shape is represented as a set of n global features, distances between whole sets of n features are to be considered. Among the most commonly used distances are L^p distances, Mahalanobis distance [DHS00] or Hausdorff distance. When shapes are characterized by sequences of n local features, considering distances between the whole set of n features does not make much sense, because they do not exploit the advantage of local features in dealing with occlusion. Hence, dissimilarity measures dealing with subsets of features, such as the partial Hausdorff measure [HKR93, Vel01], or voting schemes may be more adapted to find partial matches between shapes.

   The comparison strategy also depends on whether the considered features are invariant or not. If features are invariant, then comparison between shapes can be achieved as we have just described. If instead the features are not invariant, the matching procedure must take invariance into account by checking many configurations, what makes this approach computationally heavier.

3. **Decision.** This third and last step is crucial, and is usually the Achilles' heel of shape recognition methods. Once two shapes are likely to match, how is it possible to come to a decision? Several authors have proposed probabilistic or deterministic distances, or voting procedures as well, to sort the matches. Now, to our knowledge, the best methods only succeed in ordering the candidates to a match. In other terms, shape recognition methods usually deliver, to any query shape, an ordered list of the shapes which are likely to match with the query. When rejection thresholds exist, they are very specific to a particular shape recognition algorithm, and mainly consist in an arbitrary threshold over the ordering quantity.

It is worth noting that in general, the local (or semi-local) or global nature of features determines if the shape recognition method is local (or semi-local) or global, but this is not always the case. Indeed, while senseless, some methods do not use partial distances or dissimilarity measures (or voting schemes) between the sets of local features, and thus they are not local methods in the sense they do not allow for partial matching. Concerning global features, when used, the corresponding method cannot be but global.

Each of the three described steps is crucial for a proper achievement of a shape recognition task. We will then discuss, in the following sections, the most common methods for each step. Other particular approaches or marginal methods, are described in surveys by Alt *et al.* [AG99], Veltkamp *et al.* [VH01, Vel01], Loncarnic [Lon98] and Dryden [Dry96]. A review of more applied methods, involved in "Content-based image retrieval systems", can be found in [VT00, VC00].

2.2 Feature extraction

As we just said, a shape recogntion method can be based on local (or semi-local) or global features. Before addressing the feature extraction problem, let us make some remarks about the role of shape extraction in shape recognition methods, and recall some issues on geometric invariance requirements for recognition.

2.2.1 Some comments on the shape extraction problem

Prior to feature extraction, shapes must be extracted from images. While this should be the "step o" of every shape recognition method, shape extraction is in fact a rarely addressed problem in the context of shape recognition. Indeed, most works on shape recognition assume shapes were already extracted [GW99], or, at most, they extract them but from images that are not particularly challenging from the shape extraction viewpoint [Mok95, Rot95]. While even if shapes are given the shape recognition problem is hard and deserves a particular analysis, approaches disconnected from the shape extraction issue may mislead to shapes that would be rarely present in real images. For instance, in Mokhtarian's approach [Mok95, MAK96a, MAK96b], shapes are extracted by simply thresholding dark objects over a bright background. The further steps, which will be described

later, are very sound when shapes are curves representing objects boundaries, but will probably fail when objects are occluded (and occlusion is always present in every day's vision, as pointed out by Kanizsa [Kan79]). Rothwell proposes a whole recognition system, based on shapes, applied to the recognition of objects on uniform background [Rot95]. This recognition method is based on considering shapes as Jordan curves corresponding to the objects boundaries. Rothwell's work is very interesting and inspired some of the techniques we propose in this work, but the extraction of shapes is once again neglected. Indeed, Rothwell's method builds object boundaries by extracting edges using Canny's edge detector [Can86]. Canny's filter performs well in Rothwell's framework, where objects are well-contrasted over a uniform background, but in general, it suffers from several problems: while edges are usually thought about as curves, it detects sets of points with an orientation (*edgels*) that have to be connected afterward. Moreover, it requires different thresholds since contrast has no absolute meaning. In addition, Canny's filter is very sensitive to noise (since it uses derivatives of the image) and can only be considered through a multiscale process. The choice of these thresholds depends on the observed image, and is not that easy.

We will not discuss shape extraction further in this chapter. The aim of this subsection was to show that shape recognition should not be addressed independently from the shape extraction problem. The extraction of shapes from images will be analyzed in depth in chapter 4.

2.2.2 On the geometric invariance of features for shape recognition

When shapes are subject to weak perspective distortions, human perception is still able to recognize them. Geometric invariance requirement for shape recognition was already discussed in chapter 1, section 1.1.1. We claimed that in a general setting, affine invariance should be considered, while similarity invariance could be enough for a large class of particular applications. Such a claim was based on the following arguments:

- Projective transformations are shown not to behave well with regard to shape matching, because they permit to map a large class of curves arbitrarily close to a circle, and thus to map a finite number of curves arbitrarily close to a given curve (for example, a rabbit and a duck are "almost" projective equivalent [Åst94, Åst95]).
- Despite some interesting attempts [FK95], there is no practical way to define a projective invariant local smoothing. From this viewpoint, affine invariant smoothing is the "best" we can do [AGLM93].
- Projective transformations can be locally approximated by affine transformations (for which invariant smoothing is well defined), and these approximations are particularly accurate under weak perspective distortion.

Notice that, in general, the affine approximation does not hold unless it is conceived to be local. This is not really a restriction, since locality of shape representation was already required in order to deal with occlusion and with the figure-background problem (see also chapter 1, section 1.1.1).

In what follows we present a few features used to describe shapes. We organize the presentation into global features (each feature is computed over the whole shape) and local or semi-local features (each feature stands for a special point or region in the shape). The geometric invariance of the presented features is also discussed.

2.2.3 Global features

Several recognition methods are global, in the sense the extracted features are computed over the whole shape. Since they mix global and local information, they are sensitive to occlusions (a part of the shape is discarded) or insertion (a part of the shape is added). This makes them inappropriate for general applications, and restrict their use to a very few particular applications, where the observed shapes do not overlap with one another.

Global features are in general scalar numbers computed over the entire shape. In the case of closed curves, Fourier descriptors [KLS89, LC86, PF77, ZR72] or invariant moments [DBM77, Mon99] (following Hu [Hu62]) can be used. Affine invariant scalars for global shape representation can also be derived from wavelet coefficients [KB01, SI99]. Using wavelets allow to capture some local information of the shape, but not to the point to be able to deal with occlusion (the invariant scalars are computed by using coefficients from different scales). Another well-known, moment related, global method is *modal matching*, by Sclaroff and Pentland [SP95]. In the *modal matching* method, a finite element, physical model of solid shapes (given by its boundaries) is considered, and shapes are represented by ordered sets of eigenvalues associated to their models.

An original approach using size functions is proposed by Frosini *et al.* [FL99, FL01]. Size functions can be seen as tools to get information about the topology of any graph. A size function is a mapping from $\mathbb{R}^+ \times \mathbb{R}^+$ into $\mathbb{N}$ that to each couple (x, y) associates the number of connected components of the graph such that $D \leqslant y$ and such that at least one point of it satisfies $D \leqslant x$, where D is some "measuring function" defined on the graph vertices. The provided description is not complete (in the sense a shape could not be reconstructed from its representation), so several size functions (depending on different measuring functions) have to be computed. Applied to shape recognition, size function theory leads to nearly-invariant descriptors, which can be well adapted to perceptual matching since they rely on structural information.

Methods based on moments or Fourier descriptors, as well as size function methods, face the same problem: how to define the relative weights of each moment, or size function, in a shape comparison distance? This choice is in general arbitrary, or based on *ad hoc* arguments. Robustness against noise is another aspect of this problem. Since high order moments (or high frequency *modes* for the modal matching method) represent details or fine information of the shape, they can be contaminated by noise and they should not be considered. But up to what order should moments be considered? As for the weights, this choice is arbitrary or based on *ad hoc* arguments.

A totally different approach for global description of shapes are the so called *normalization methods*. These methods allow the transformation of any element of an equivalence class of shapes under a

group of geometric transforms (up to affine transforms) into a specific one, fixed once for all in each class. As every global method, global normalization methods are not robust against occlusions. They usually rely in the computation of high order moments, what makes them also sensitive to noise. In his PhD thesis [Coh94], T. Cohignac proposes an affine normalization global method, that we discuss in chapter 7. This method presents an inherent drawback: shapes showing a "quasi-symmetry" are represented by an unstable normalization (see chapter 7, figure 7.6 for an example).

Scale-space representation of shapes can also be used to derive invariant representations. One such method can be found in Alvarez *et al.* [AMS02], where shape invariants are based on the evolution of area and perimeter of shapes under the affine scale space. In the seminal work by Mokhtarian and Mackworth [MM92], shapes (Jordan curves corresponding to objects boundaries) are described on the mean curvature motion scale space. A shape is represented as follows. At each scale, the curve is reparametrized by the normalized arclength, and the position of inflexion points (zero-crossings of the curvature) is tracked. If we denote by σ the scale and s the corresponding normalized arclength, the proposed multiscale representation of the shape consists in the set of 2-tuples (s_i, σ_i), corresponding to the position and the scale at which two inflexion points meet and vanish. This representation is similarity invariant. It can also be robust to noise, if one only considers the information given by the scale space for scales larger than an *ad hoc* or abitrarily fixed threshold. At first sight, this method seems to be able to deal with occlusion, since curvature is a local property of curves. This is not the case, however, since at each scale curves are reparametrized by the normalized arclength, and occlusions or insertions can drastically modify the positions of points (s_i, σ_i).

2.2.4 Local and semi-local features

While global features are in general defined to be geometrically invariant up to, at least, rigid transformations, the local or semi-local features defined in the shape recognition literature can be invariant or not.

Commonly used non invariant features are, for instance, sets of edges [Mar82, MH80]. Groups of features are more informative than individual local features, and consequently enhance the matching stages: chained edges [Wol90b] or edgels [OH97] (an edge element with a direction) can be considered.

In order to achieve (geometrical) invariant recognition, non invariant features must be compared by means of strategies dealing with invariance, thus leading to more time consuming algorithms. We will not discuss further non invariant features, since in this work we are concerned with invariant features.

Invariant local features may be computed directly on the image, or after the shape has been extracted. Features can be differential or integro-differential invariants at some special points (like corners [SM97]) or regions (*e.g.* coherent regions [BCGM98, WN99]) of the image. The computation of differential invariants is very unstable, even after smoothing the image, since it involves high order derivatives.

I. Weiss [Wei93] proposes local projective invariants needing fourth order derivatives of curves. This is of course out of range for contours of shapes derived from images. Sato and Cipolla [SC98] propose semi-local quasi-invariants of curves, which do not need high order derivatives. Nevertheless, their affine quasi-invariants need to compute at least second order derivatives, what is still too large to deal with curves extracted from real images. F.S. Cohen *et al.* [CHY95, HC96] propose to approximate curves with B-Splines, leading to a compact representation. This interpolation appears to be robust to noise, and an adequate matching algorithm permits to deal with occlusions. Although this method seems promising, it suffers from the interpolation in itself, which depends on the original sampling of the considered curve.

Most local recognition methods involve curvature extrema of the shapes, which are not only affine invariants of curves, but are certainly, from the perceptual viewpoint, the most salient point of shapes. This was already pointed out by Attneave in his 1954 paper [Att54]: *"Information is concentrated along contours (i.e., regions where color changes abruptly), and is further concentrated at those points on a contour at which its direction changes most rapidly (i.e., at angles or peaks of curvature)."* (See Figure 2.1.) In such local shape recognition methods, shapes are represented by a finite code, composed by the coordinates of curvature extrema points. Two variations based on this general method, leading respectively to a similarity invariant and to a translation-rotation invariant recognition methods, can be found in [AD90, GW99]. Cohignac *et al.* [CLM94] propose a multiscale curvature representation for shape recognition, by considering curvature extrema of surfaces derived from a shape with the affine morphological scale space. This leads, for each shape, to a set of points of interest in $\mathbb{R}^3$.

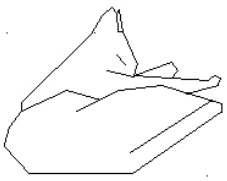


Figure 2.1: *(From [Att54]) Curvature extrema concentrates a large amount of shape information. Quoting Attneave: "Common objects may be represented with great economy, and fairly striking fidelity, by copying the points at which their contours change direction maximally, and then connecting these points appropriately with a straight-edge."*

Up to here we have mainly discussed local invariant features. Since invariants which are too local, such as differential invariants, suffer from noise, and those being too global (*e.g.* moment invariants) suffer from occlusions, a suitable trade-off solution can be semi-local features.

Lamdan *et al.* [LSW88], followed by Rothwell [Rot95, RZFM95], have proposed semi-local descriptors of shapes, invariant up to similarity or affine transformations (Rothwell *et al.* also propose projective invariant representations). These features are based on the description of pieces of non-convex curves lying between two bitangent points (*i.e.* points at which the same straight line is tangent to the curve). Such features are affine invariant and the use of bitangent lines ensures robustness to noise. Lisani *et al.* [Lis01, LMMM03] improved this bitangent method by associating, with each bitangent

to the shape, a local coordinate system and defining a local affine or similarity normalized piece of curve. They have also added to the representation, similar local invariant descriptions based also on tangent lines to the curve at inflexion points, leading to a more complete representation of shapes. We give more details on Lisani's local invariant normalization along the following chapters.

2.3 Features matching

From the point of view of matching procedures, three different feature extraction strategies are to be distinguished: extraction of global features without shape normalization, shape normalization (applied to global or semi-local features) and local invariant feature extraction. The order in which we mention them coincides with an increasing complexity of the subsequent matching procedure.

When shapes are described by sets of global features, the comparison between them is simply achieved by evaluating distances between corresponding vectors of features. When features are not of the same nature, combining them in a matching distance is not trivial. Mahalanobis distance [Sma96, DHS00] attempts to achieve this combination, by taking the covariances between features into account. An *ad hoc* "probability" upon which curves are supposed to match may also be derived [Sch99]. Some global features allow to match shapes based on other criteria than invariance with respect to a projective subgroup. For instance, a lot of work has been done on methods for matching shapes by minimizing the deformation energy involved in aligning one shape with another. One such method is modal matching [SP95], which takes into account a certain physical plausibility of the deformations, and accepts thus a larger class of invariance than geometric groups. Methods minimizing non-rigid energy deformations can also be based on local features, but they do not allow for partial matching, since all features are involved in the deformation energy. As an example, Belongie *et al.* [BMP02] propose to estimate the transformation leading from one shape to another, when each shape is described by some points with a "shape context" (information about the points vicinity). M. Miller, L. Younès and A. Trouvé [MTY02, MTY03] study the orbit of shapes via the action of diffemorphic transformations, allowing by this way non-rigid transformations.

When shapes have been subject to a (global or semi-local) normalization method, matching is simply reduced to a comparison between normalized curves. This comparison is straightforward, since normalization eliminates ambiguities such as the choice of the starting point. Normalized curves can be compared using L^p, Hausdorff or Fréchêt distances [AKW01]. In the normalization method introduced by Lisani *et al.* [Lis01, LMMM03] the curve matching procedure is as follows. Each normalized piece of curve is represented in a hash-table, leading to a fast pre-identification of matches (pre-matches). This identification of pre-matches is made by using a large enough distance threshold, in order to ensure that all true matches are kept, even if some wrong matches pass the test. Then the actual distance between them is computed, and prematches are rejected if this distance is larger than some threshold. As a last step, the matching beyond the initial portion of curves are extended, provided that the distance between the corresponding points in the curves is below the distance threshold. This method gives a very accurate local estimation of the matching of two curves.

If the features are local and invariant, then the matching process consists in comparing two sets of local features (the coordinates of curvature extrema points, for instance). This can be done by considering Procrustean distances [Sma96], or by means of voting schemes. The three most popular voting schemes are Geometric Hashing [LW88, Wol90a, WR97], the Generalized Hough Transform [Bal81] and the Alignment method [HU87]. Given two shapes, the Geometric Hashing method aims at determining if there is a transformed subset of the features from one shape, that matches a subset of the features of the other one. Geometric Hashing algorithm is presented in figure 2.2, for affine transformations. The Generalized Hough Transform method, instead of voting over all possible configurations of shapes, consists in voting over all possible transformations mapping a shape to another one. Alignment method is a similar hashing method [HU87]. Like for all techniques based on histograms in multidimensional spaces, these voting methods are very sensitive to the choice of quantization precision (too large bins may lead to false matches, and too small bins may produce misses). Besides, most of the time, the size of the hash table and the amount of parameters (size of the bins in the voting stage, threshold for the amount of votes in each bin, *etc.*) are crippling. The complexity of these voting schemes increases with the invariance degree; affine invariant shape retrieval in large databases is intractable. All these properties make the local features not suitable for shape retrieval in large databases.

2.4 Decision

A general theory of perceptual recognition thresholds is provided by neuroscientists through the ideal observer theory [Gei03, LKK95, OK04]. A theoretical "ideal" device is designed as the best device performing a classification or recognition task. It often consists on a Bayesian analysis, trained over a data set. Nevertheless, ideal observer depends on the training stage and on some parameters (such as the coefficients of an utility function which penalizes wrong classifications). This theory does not derive any perceptual thresholds, but gives a framework to compare performances of human recognition and classification task.

Up to our knowledge, nobody has yet proposed a generic acceptance / rejection decision method for shape matching. In general, matches with a query shape are, at most, only ranked (for example along a distance, or along some probability [Sch99]). The question of how to fix a decision threshold in order to assess whether or not two shapes are alike, is never addressed. While some works have studied the problem of observing wrong detections due to random ("hallucinating a wrong fit" [Ste95]), and some of them even quantify a false alarms rate [GH91, HJ95, OH97], none of them propose an automatic decision rule.

Let us specify what we mean by "automatic decision rule" for shape matching. Assume we are looking for a query shape $\mathcal{S}$, in a large set of shapes extracted from W, the "world of images". A distance between shapes is available, so that the smaller the distance, the more similar the shapes. The question is: what is the threshold value for that distance to ensure recognition?

Geometric Hashing: A target shape $\mathcal{S}$ is searched in a set of shapes

Preprocessing (off line) For each shape $\mathcal{S}'_i$ in the set of shapes:

(1) Extract local invariant features from $\mathcal{S}'_i$. Assume n such features are found.

(2) For each local basis b_j (*e.g.* a pair of points for similarity transformations, three non-collinear points for affine transformations) of features:

  (a) Compute the quantized coordinates (u, v) of all the remaining features, in the local basis.

  (b) Use the couple (u, v) as an index in a hash table, and write the information (i, b_j) in the corresponding bin (i is the index that identifies $\mathcal{S}'_i$).

Recognition stage (on line) For the target shape $\mathcal{S}$:

(1) Extract local invariant features from $\mathcal{S}$. Assume n such features are found.

(2) Choose arbitrarily a local basis (two or three points, depending on the considered invariance).

(3) Compute the quantized coordinates (u, v) of all the remaining features, in the local basis.

(4) For each of these coordinates, go to the corresponding bin in the hash table, and cast a vote for each pair (i, b_j) inscribed in the bin.

(5) Keep only the pairs (i, b_j) which received more than a certain number of votes: each of this pairs stands for a potential match.

(6) For each potential match, compute the best transformation (in the least squares sense) between all corresponding features, and check if the target features and the features from the corresponding shape, are well aligned. If not, go to (2) and choose another basis.

Figure 2.2: *Geometric hashing algorithm. For affine invariant shape recognition, time complexity for the preprocessing stage is $\mathcal{O}(n^4)$ for each shape in the set of shapes, and, if the access time to the hash table is $\mathcal{O}(1)$, time complexity for the recognition stage is between $\mathcal{O}(m)$ (when the first target basis chosen at random corresponds to a model in the set of shapes) and $\mathcal{O}(m^4)$ (when no basis from the target shape corresponds to a model in the set of shapes).*

Given two shapes and an observed small distance δ between them, there are only two possibilities:

1. Both shapes lie at that distance δ because they 'match' (that is, they are similar because they are two instances of the same "ideal shape").

2. The set of shapes extracted from W is so large, that, just by chance, one of these shapes is close to $\mathcal{S}$ (there is no underlying common cause between them, and they do not correspond to the same "ideal shape").

As we will more precisely discuss in the following chapters, our aim is to evaluate, for any δ, the probability (or rather the expectation) of the second possibility. If this number happens to be very small for two shapes, then the first possibility is certainly a better explanation. Hence, pairing both shapes would make sense, because this match is not likely to happen by chance.

In order to fix this decision threshold, the distribution of distances between shapes from W and $\mathcal{S}$ must be learned. This distribution yields the probability that a shape from W lies at any fixed distance from $\mathcal{S}$. If a match between two shapes is very unlikely to be due to chance then pairing them is highly relevant.

Let us review and discuss some previous methods, similar to the one that we will propose.

2.4.1 Probability of wrong match or number of false alarms?

Some authors have addressed this question of 'wrong matches' occurring purely by chance, but the proposed models do not lead to an automatic recognition criterion. Let us discuss two interesting examples.

Olson and Huttenlocher [OH97] present a method for automatic target recognition under similarity invariance. Objects and images in which the objects are sought are encoded by oriented edges, and compared by using a relaxed Hausdorff distance. Modeling the matching process by a Markov process leads to an estimation of the probability $P_K(t)$ of a false alarm between K consecutive edges for a given transformation t. The authors give an estimate of the probability of a false alarm occurring over the entire image by computing $1 - \prod_t (1 - P_K(t))$, which is used to take a decision. Let us quote the authors: *"One method by which we could use the estimate is to set the matching threshold such that the probability of a false alarm is below some predetermined probability. However, this can be problematic in very cluttered images since it can cause correct instances of targets that are sought to be missed."* This methods raises several problems. Finding a false alarm at a given location is clearly not independent of finding it at a close location. Besides, the real handy quantity to be controlled is not the false alarm probability, but the expected number of false alarms.

Grimson and Huttenlocher [GH91] propose to fix a threshold on the proportion of model features (edges) among image features (considered in the transformation space) upon which the detection is sure. Their main assumption is that the features are uniformly distributed. This can look odd, because images are precisely made of non-uniformly distributed features (for instance, the edgels

are along edges, *etc.*). Let us quote Grimson and Huttenlocher's answer: *"Although the features in an image appear to be clustered (e.g. shadow edges and real edges), this does not necessarily mean that a uniform random distribution is a bad model. Even under a random process, clusters of events will occur. The question is whether the degree of clustering of features more than one would be expected from a random process."* The last sentence is exactly the formulation of an *a contrario* model, whose notion we will soon define. This framework allows the authors to estimate the probability that a certain cluster is due to chance (due to the *"conspiracy of random"* in their words). Fixing a threshold on this probability gives sure detections: rare events are the most significant ones. The ideas developed in [GH91] inspired several papers [AL00, Ols98].

Following Huttenlocher and Grimson's work, X. Pennec [Pen98] presents a method to compute the intrinsic false alarm rate of commonly used methods such as Geometric Hashing and Generalized Hough Transform, by incorporating the uncertainty of measurements. The proposed computation relies on several limitative assumptions (exact model, uniform distribution of features), as in Huttenlocher and Grimson's work. As pointed out by X. Pennec: *"[These limitations] are hardly ever verified in real cases. For instance in medical images of the head, extremal points are not uniformly distributed in the image, but more or less uniformly distributed on the surface of the brain and the skull [...]. A very interesting extension would be to compute the probability of false positives* online*, during the recognition itself. This would allow us to take into account the specific distribution of the model and scene features."* This is precisely what we aim to achieve (see chapter 8).

In the following, we intend to make such probabilistic methods reliable and to give the right matching thresholds.

1) Instead of defining a threshold distance for each given shape $\mathcal{S}$, we define a quantity (namely the Number of False Alarms) that can be thresholded independently of $\mathcal{S}$.

2) This quantity can be interpreted as the expected number of appearances of a shape at a certain distance from $\mathcal{S}$. Even if thresholding this number naturally leads to thresholding the matching distance, we get an additional information about how likely the matching is, and therefore about how sure we are that the matching is correct.

2.4.2 Previous methods based on false alarm rates

A contrario detection frameworks have been widely used in signal processing theory, in which the concept of controlling the detection quality by the "Number of False Alarms" was introduced. Let us give a precise description. In [ACDH99], a method to detect gravitational waves is presented. The basic assumption is that the detector noise is white, stationary, and Gaussian with zero mean and standard deviation σ. The problem consists in separating signal from noise. A signal $(x_i)_{1 \leqslant i \leqslant N}$ of N data samples being given, they count the number of samples whose values exceed a threshold $s \cdot \sigma$.

In the absence of signal, the noise being Gaussian:

$$\Pr(|x_i| \geqslant s\sigma) = 2\int_s^{+\infty} \frac{e^{-x^2/2}}{\sqrt{2\pi}} dx.$$

If N_c is the number of samples above threshold, one has:

$$\Pr(N_c = n) = \binom{N}{n} p^n (1-p)^{N-n},$$

where $p = \text{erfc}(s/\sqrt{2})$, erfc is the complementary error function. Under limit considerations, if $\mu_c = Np$ and $\sigma_c = \sqrt{Np(1-p)}$, the normalized random variable $\tilde{N}_c = (N_c - \mu_c)/\sigma_c$ is well approximated by a standard normal random variable. The threshold s being set by physical arguments, the following relation between the detection threshold η (the number of samples upon which it is unlikely that the signal was generated by the noise) and the false alarm rate r_{fa} holds:

$$2 \cdot \frac{1}{\sqrt{2\pi}} \int_\eta^{+\infty} \exp\left(\frac{-x^2}{2}\right) dx = r_{fa}.$$

The authors fix r_{fa} by converting it to the *number of false alarms* per hour for their sampling rate, and thus deduce a handy value for η. Of course, the lower the threshold, the higher the number of false alarms, and conversely. To summarize, a detection threshold is deduced by the imposed value of the number of false alarms of the detection algorithm.

Another example where an *a contrario* model is applied in image processing can be found in [CBCN01]. This work aims at detecting small targets in natural images, which are modeled as texture images (the targets are supposed to be rare enough in order not to disturb this model). After a suitable transformation is applied to the original images, the grey level distribution over the resulting images is assumed to be a zero-mean unit-variance Gaussian. Grey levels in small windows are then observed. Although there is no reason why the grey levels over a fixed small window should follow a Gaussian distribution, the authors observe that the Gaussian model fits well the data. In other words, they build a background model for these small windows. A rare window with regards to this model (*i.e.* whose probability is less than a certain threshold) is supposed to correspond to a detection. The detection threshold is fixed in such a way that the false alarms rate is low enough.

An example of target detection in non-Gaussian images is given in [WW96]. The authors model the background of the considered images with a fractal model based on a wavelet analysis. Targets are detected as rare events with regard to this model.

2.5 Conclusion

General shape recognition methods can be decomposed into three main stages: shape feature extraction, matching and decision. But before dealing with shape recognition, shape information has to be extracted from images. Global representations are not well adapted to the recognition problem, since they are not robust to occlusions. From the perceptual viewpoint, shape information is

concentrated along contours (and particularly in high curvature points) [Att54], which are generally detected using edge or edgel detectors, corner detectors, *etc.* These procedures do not give global structures, but only clouds of points. The provided description is consequently quite rough and the only possible matching procedure is geometric hashing, or similar voting techniques. However, these methods show serious drawbacks, such as space and time complexity or numerous thresholds which endanger the robustness of the process. Reducing and organizing the mass of information thus appears as essential: the higher level the features, the more robust the comparison (this statement is accurately formalized in [Pen98], where it is shown that the false alarm rate of matching algorithms decreases when using more discriminative features). For example, chains of edges are much more discriminative than a simple, unstructured collection of edges. Unfortunately, chaining edges is an unstable procedure, depending on several thresholds (see for example [Gir87], or Chapter 4 in Pavlidis' book [Pav80] for an overview of "edge tracing" algorithms). Hence, reducing and organizing the information to get higher level features is very problematic in practice, since each involved step introduces many interdependent thresholds. This makes the acceptance / rejection decision all the more delicate.

To summarize, the alternative is as follows. Either, primitive parameterless extraction procedure (edgels) followed by intensive search like geometric hashing: in that case, because of the computational complexity, we can attain at most rotation-translation invariant recognition. Either, one needs a more sophisticated extraction, followed by a normalization procedure. However, chained edgels, although more discriminative, are not reliable since they are very dependent on parameters fixed "by hand". One way to bypass the chaining problem is to represent the geometric shape information by an appropriate set of meaningful level lines. The parameterless method proposed by Desolneux *et al.* [DMM01] can be applied for that purpose. Sets of pieces of meaningful level lines yield significant, stable shape information, that meet the image representation requirements presented in chapter 1. In [Lis01, LMMM03], Lisani *et al.* represent shape information based on normalized pieces of meaningful level lines (up to similarity or affine transformations), but the decision problem is not addressed, in the sense no automatic decision rule for matching is proposed. All along this thesis, we shall explore decision issues, in order to derive automatic matching decision rules.

Chapter 3

The bilinear tree

Abstract: The topographic map provides a complete representation of an image. This representation is well suited for shape analysis and recognition, since it is based on the geometrical information of images, and can be embedded in a tree structure. However, since the level lines of digital images (zero order interpolates) suffer from pixelization effect, shapes cannot be accurately described. Higher order interpolates are then to be considered. In this chapter, the bilinear interpolation of gray level images is described. Then, following [LMR01], a fast numerical method for extracting the topographic map is presented.

Résumé : La carte topographique fournit une représentation complète des images. Cette représentation est bien adaptée à l'analyse et à la reconnaissance de formes, et peut être présentée comme une structure d'arbre. Cependant, les lignes de niveau des images numériques (des interpolées d'ordre zéro) sont affectées par l'effet de pixelisation, et les formes ne peuvent donc pas être décrites fidèlement. Des interpolations d'ordre supérieur doivent donc être considérées. Dans ce chapitre, nous décrivons l'interpolation bilinéaire des images en niveaux de gris. Ensuite, suivant [LMR01], nous présentons une méthode numérique rapide pour extraire la carte topographique.

3.1 Introduction

It is a well known fact that shape information in images is concentrated along regions where color or gray level changes abruptly [Att54, Mar82]. Since Marr and Hildreth's seminal work on edge detection [MH80], the effort on extracting shape information from images has been mainly concentrated on local methods. Among these methods, which are commonly referred as edge detectors, Canny [Can83] and Canny-Deriche [Der87] filters are certainly the most widely used.

Classical edge detectors suffer mainly from two problems. The first one is that they depend on (at least) a threshold on the contrast, which is hard to estimate and is usually fixed arbitrarily. The difficulty in fixing the contrast threshold is a consequence of the fact that contrast has no absolute meaning. Indeed, as noticed by phenomenologists like Attneave and Wertheimer, shape perception is independent of the gray scale (see Chapter 1). The second problem of these methods is that all they

detect are sets of points with an orientation. Since we do not think of edges as collections of isolated points but as curves, these points have to be connected afterwards, and this procedure is known to be very unstable.

Following [LMMM03], in Chapter 1 we claimed that the set of level lines of a digital image was a natural representation of its shape contents, since it provides topological information invariant to contrast changes. Moreover, no chaining procedure is needed since level lines are already curves. However, the level lines representation of digital images presents essentially two drawbacks: level lines are restricted to lie on the initial grid (*pixelization* effect), and only a small subset of them is relevant. Both these problems are illustrated in Figure 3.1. The selection of meaningful level lines is addressed in Chapter 4, where a method based on perceptual arguments is presented. All the remainder of the current chapter is devoted to present a solution to avoid pixelization effect, proposed by Lisani *et al.* [LMR01].

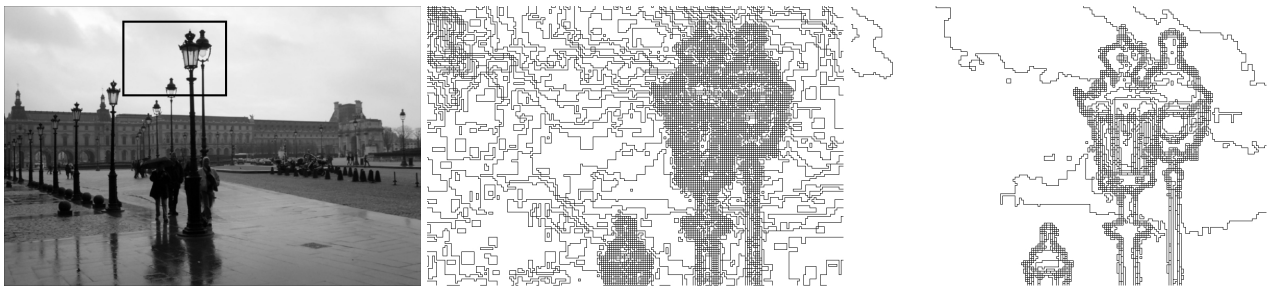


Figure 3.1: *Level lines from a digital image (zero-order interpolation). Left: original image. Middle: all level lines for the small image inside the rectangle (the quantization step for the gray level is* 1*). Right: level lines for a gray level quantization step of* 10*. Level lines are restricted to lie in the original grid, and the majority of them provides no useful information from the perceptual viewpoint.*

The plan of this chapter is as follows. Section 3.2 deals with level lines in digital images and bilinearly interpolated images. Then, in section 3.3, some properties of the tree structure in which level lines are embedded are presented, and a fast algorithm to extract the level lines based on this tree, due to Monasse and Guichard [MG00] is described. This last section was extracted from [LMR01].

3.2 Level lines and bilinear interpolation

3.2.1 Level lines: the topographic map

A family of binary images obtained by thresholding an image at given values provides a complete representation of the image [Mat75, Ser82]. This is equivalent to considering level sets; the (upper) level set of a gray level image $u : \mathbb{R}^2 \to \mathbb{R}$ at the value λ is

$$\chi_\lambda(u) = \{x \in \mathbb{R}^2, \quad u(x) \geqslant \lambda\}.$$

An image can be reconstructed from the whole family of its level sets, by

$$u(x) = \sup\{\lambda \in \mathbb{R}, \quad x \in \chi_\lambda(u)\}.$$

The geometrical information in images can then be reduced to the topological boundaries of connected components of level sets, referred to as *level lines*. The *topographic map* of an image, defined as the collection of all its level lines, gives a complete representation of an image, and verifies two main properties:

- It is invariant with respect to contrast changes. Indeed, if g is an increasing function from $\mathbb{R}$ to $\mathbb{R}$, u and $g(u)$ have the same topographic map,
- It is a hierarchical representation: since level sets are ordered by the inclusion relation (and so are there connected components), the topographic map may be embedded in a tree structure.

The first property is very interesting since it means that level lines are features that do not depend on the contrast in the image, and allow to bypass the thresholding problem from classical local edge detection methods. The second fundamental property is at the origin of the algorithms presented in section 3.3.

In practice, we do not deal with real valued images defined in a closed rectangle of $\mathbb{R}^2$, but with digital images. A digital image u_d is a function defined in a rectangular grid, that takes values in a finite set, typically integer values between 0 and 255. One can think of u_d as a regular sampling of an image u defined in a closed rectangle of $\mathbb{R}^2$, whose grey levels were quantized, followed by a zero-order interpolation with a rectangular element. Each element of the grid is called a *pixel*. Digital images are piecewise constant functions, and as a consequence, level lines from digital images show *pixelization* effects, as shown in Figure 3.1.

Some consequences of this pixelization are:

- Useful invariant features such as inflexion points or curvature extrema, that are extensively used in shape recognition, cannot be computed on pixelized level lines,
- The accuracy of any measure based on the location of the level lines is limited by the pixel size,
- Level lines corresponding to different gray levels may have pieces in common (creating T-junctions). This never happens when dealing with level lines of a continuous image.

Pixelization effect can be avoided by considering higher order (than zero-order) interpolations of digital images. Then, the level lines of the interpolated images can be computed. These level lines have some interesting properties:

- Level lines will be smoother than in the previous case,
- Subpixel accuracy can be achieved when measuring level lines, since they are not restricted to the grid of the digital image,

- Level lines from different gray levels never touch each other, since the considered images are now continuous functions.

Among the possible interpolations, the bilinearly interpolation presents two advantages: it is the most local of continuous interpolations, and it preserves the order between the gray levels of the image. The idea to consider the level lines of the bilinear interpolated image is also present in Digital Morse Theory [CK98].

3.2.2 Level lines and bilinear interpolation

The bilinear interpolate of a digital image u_d, denoted by $\tilde{u}$, can be obtained as the convolution of u_d (considered as a network of Dirac masses concentrated at the centers of pixels) with the function $\Phi(x, y) = \varphi(x)\varphi(y)$, where

$$\varphi(x) = \max(1 - |x|, 0).$$

As $\varphi \geqslant 0$, bilinear interpolation is a convolution with a nonnegative kernel and hence an increasing operator. Since $\varphi(x) < \varphi(0) = 1$, the extrema of $\tilde{u}$ are all located at points on the discrete grid. More precisely, all regional extrema of $\tilde{u}$ contain at least a local extremum in the original grid.

The general form of a bilinear function is $f(x, y) = axy + bx + cy + d$. For each set of four adjacent pixels (which will be called a *Qpixel* from now on, see Figure 3.2), a bilinear function can be determined; parameters a, b, c and d are fixed by the gray levels of the four pixels $(i, j), (i + 1, j), (i, j + 1), (i + 1, j + 1)$. The bilinear interpolate of a *Qpixel* is defined only inside the rectangle delimited by the *Qedgels*, which are the segments between adjacent pixels centers in the *Qpixel* (Figure 3.2). The bilinear interpolation of a digital image is the concatenation of bilinear interpolates of its *Qpixels*. Continuity of the gray levels between contiguous *Qpixels* is guaranteed by the properties of the bilinear interpolation, but higher continuities (e.g., of the gradient) are not preserved at *Qedgels*.

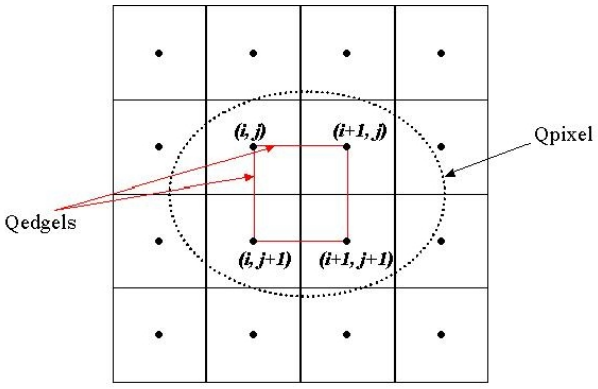
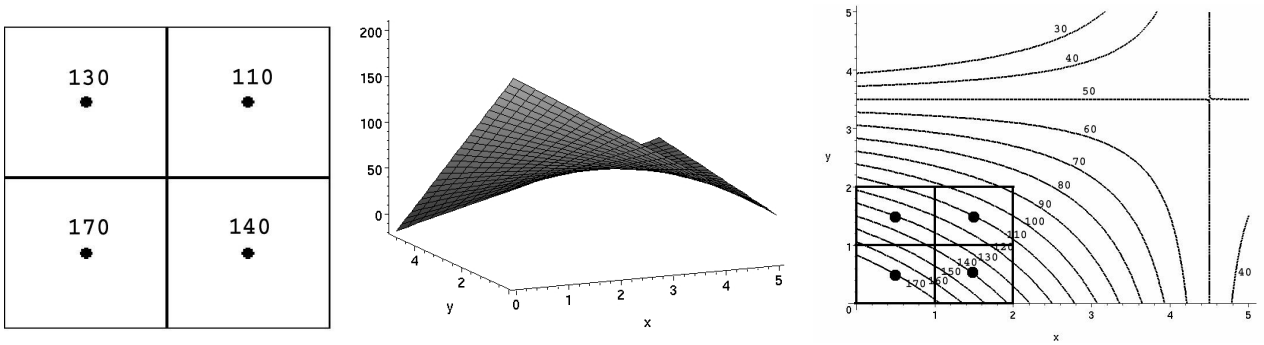


Figure 3.2: *Definition of* Qpixels *and* Qedgels.

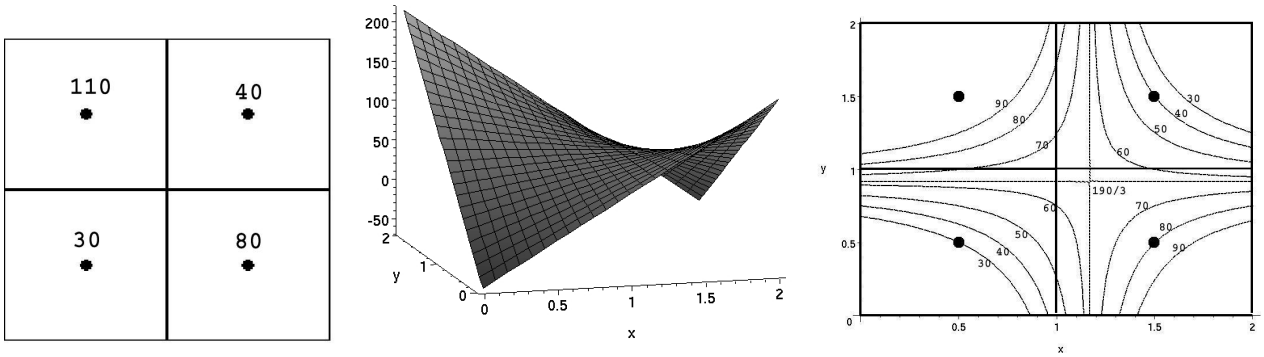
Except for the degenerate case when $a = 0$, the equation for the level line at level λ of the bilinear interpolate of a *Qpixel* can be written as follows:

$$a(x - x_s)(y - y_s) + (\lambda_s - \lambda) = 0,$$

where $x_s = -\frac{c}{a}$, $y_s = -\frac{b}{a}$ and $\lambda_s = d - \frac{bc}{a}$. Level lines are then pieces of hyperbolae, of common axes $x = x_s$ and $y = y_s$. When $\lambda = \lambda_s$ the level line is composed of two perpendicular straight lines that cross at point (x_s, y_s). This singular point where the level line crosses itself is a *saddle point*, and the corresponding gray level λ_s is referred to as *saddle level*. Every bilinear interpolation of a *Qpixel* in the non degenerate case ($a \neq 0$) has an associated saddle point (the center of symmetry of the hyperbola), but it is not always inside the *Qpixel*. Figure 3.3 shows two examples of such bilinear interpolations. The saddle point falls inside the rectangle given by *Qedgels* only when the maximum of values at diagonally opposed pixels is strictly less than the minimum of the other two values.



(a) *Left:* Qpixel. *Middle: bilinear interpolate of the* Qpixel. *Right: some level lines of the bilinear interpolate. The saddle point (level* 50*) falls outside the rectangle defined by the centers of pixels (which is the domain of definition of the bilinear interpolate).*



(b) *Left:* Qpixel *for which the maximum of values at diagonally opposed pixels is strictly less than the minimum of the other two values. Middle: bilinear interpolate of the* Qpixel *on the left. Right: the saddle point of the bilinear interpolate (level* $190/3$*) falls inside the domain of definition.*

Figure 3.3: *Bilinear interpolates of two different Qpixels, for the non degenerate case ($a \neq 0$). The domain of definition of the interpolates is the interior of the rectangle defined by the* Qedgels*. Level lines are pieces of hyperbolae; its saddle point does not fall inside the domain of definition, unless the maximum of values at diagonally opposed pixels is strictly less than the minimum of the other two values (like in (b)), in which case the existence of a saddle point is not an artifact of the interpolation but an intrinsic property of the image.*

In the degenerate case, the equation for the level line at level λ of the bilinear interpolate of a *Qpixel* is given by:

$$bx + cy + (d - \lambda) = 0.$$

Its level lines are then straight lines. Such a case may lead to a pixelization effect in the *Qpixel*. For instance, if $b = 0$, the level lines are vertical straight lines, and if λ is equal to one of the gray values at pixel centers, a level line may follow one of the vertical *Qedgels*. This idea is illustrated with the test image in Figure 3.4. The level lines at gray level 70 follow several *Qedgels* leading to a strong pixelization effect.

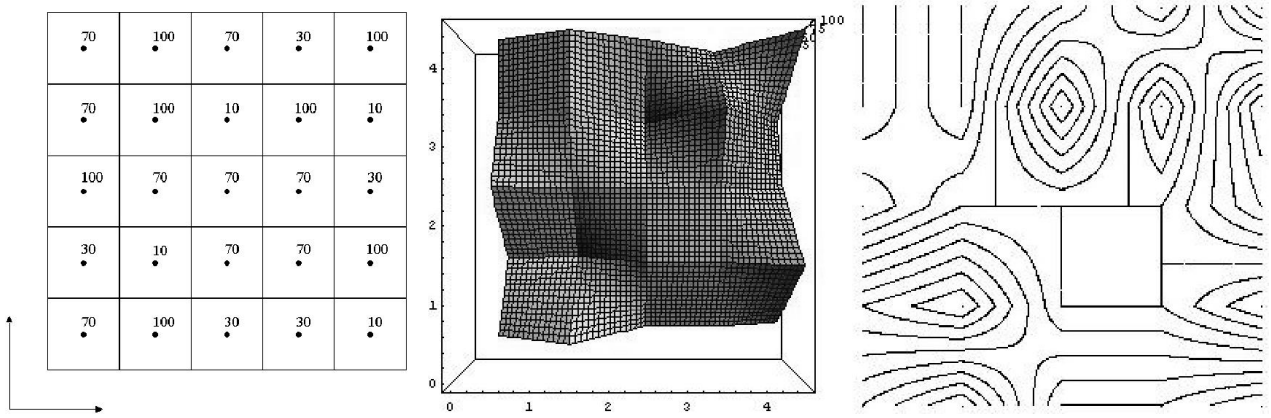


Figure 3.4: *(From [LMR01]) Left: test image. Middle: bilinear interpolate of the test image. Right: level lines for gray levels from* 10 *to* 100*, with step* 10*. Observe how some of the level lines (the ones at gray level* 70*) follow the* Qedgels*, producing some pixelization effect.*

This pixelization effect in bilinear interpolates can be avoided if none of the gray levels of the original digital images is considered. In other words, if none of the gray levels of the level lines is equal to the original levels of the image, the level lines will never cross the centers of the pixels, nor will they follow the *Qedgels*, but always cross them. Choosing the gray levels this way guarantees that no pixelization effect is present in the level lines (moreover, a level line will cross a *Qedgel* only once, at most). This is illustrated in Figure 3.5, for the test image in Figure 3.4.

If the level values of the digital image are in $\{0, \ldots, 255\}$, a simple way to avoid them is to consider non integer values. An example is shown in Figure 3.6, which also summarizes some of the ideas presented up to here.

3.3 Tree of bilinear level lines

In [MG00] a tree of level sets was defined and computed (the FLST). The extraction of this tree was carried out by considering the level sets of the digital image. The information on the inclusion relations between the connected components of level sets of the image (that here will be called *solid shapes*) was coded in this tree, in such a way that a *solid shape* is *child* of another *solid shape* if it is

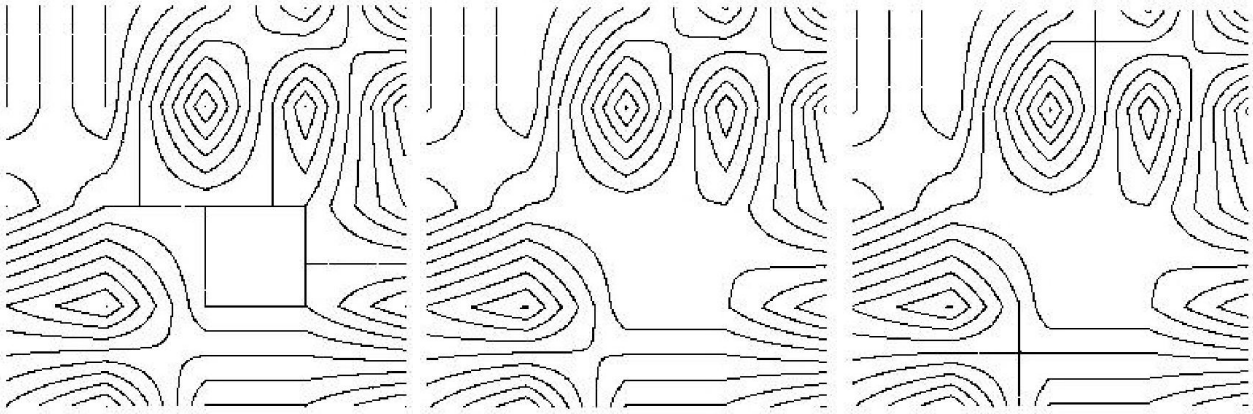


Figure 3.5: *(From [LMR01]) Gray levels for the piecewise bilinearly interpolated image in Figure 3.4. Three different sets of level lines were computed. Left: gray levels from* 10 *to* 100 *with step* 10*. Observe how some of the level lines (the ones at gray level* 70*) follow the Qedgels, producing an effect similar to pixelization. Middle: gray level* 11 *to gray level* 91 *with quantization step* 10*. Pixelization effect no longer arises since level lines are computed at gray levels different to those of the original image. Right: level lines were computed at gray levels different from those of the original image but we get* 90^o *crossings between level lines due to the presence of a saddle point. Nevertheless, these saddle points will always appear inside the Qpixels and the curves never go along the grid of the digital image.*

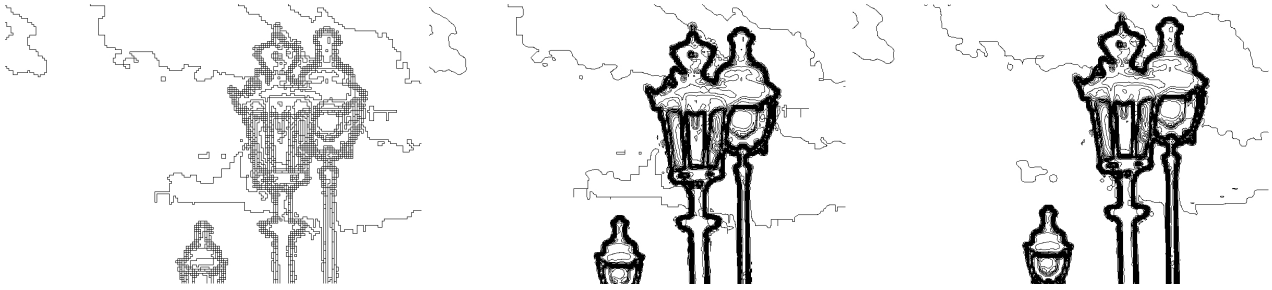


Figure 3.6: *Left: level lines from a digital image (zero-order interpolation). The quantization step for the gray levels is* 10*, starting from* 10*. The pixelization effect creates artificial T-junctions, and do not allow to compute useful features such as inflexion points or curvature extrema. Middle: level lines from the piecewise bilinear interpolated image. The quantization step for the gray levels is* 10*, starting from* 10*. Pixelization effect has been reduced, but some pixelized regions can still be seen. Right: level lines from the piecewise bilinear interpolated image, with a gray level quantization step of* 10*, starting at gray level* 0.5*. These level lines do not suffer from pixelization effects.*

included in its *interior*. The notion of interior depends on the type of level set (upper and lower) and a different connectivity (4 or 8) needs to be used in order to extract the interior of an upper or a lower level set.
The framework in [MG00] is semi-continuous images (see [BCM03]). When dealing with continuous images, the property that level lines are disjoint provides a much simpler argument for the existence of the tree structure, similar to that provided in the pioneering work of Kronrod [Kro50]. The tree structure is however different from that of [CK98] where the order in the tree is driven by the gray level and not by a geometric consideration.
This unpleasant unpairment between upper and lower level sets can be avoided by computing the tree of inclusions between the level lines. In particular, when bilinear level lines are used, since they never touch each other (as pointed out in the previous section) a fast and quite simple algorithm can be devised, based on the crossings of the level lines with the *Qedgels* of the image. This algorithm is not presented here, and can be found in [LMR01].

3.3.1 Properties

In this subsection, some basic results concerning level lines of a bilinear interpolated image are presented. These results will be used for the extraction algorithm.

DEFINITION 3.1 *The interior of a closed level line is the portion of $\mathbb{R}^2$ which is enclosed by the level line. If the curve is open, one needs to "close" it in order to decide which part of the plane is inside or outside. An open curve can be closed along the border of the image but there are two possible paths to follow. Here it is (arbitrarily) decided to choose the shortest path (see Figure 3.7).*

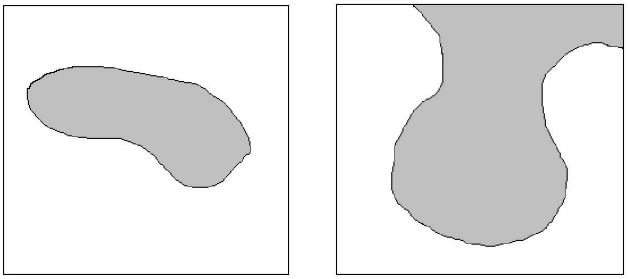


Figure 3.7: *Interior (in gray) of a closed and an open level line. Open level lines are closed following the shortest path along the border of the image.*

PROPOSITION 3.1 *At any level λ, the number of level lines at level λ is finite.*

Indeed, inside each *Qpixel*, there are at most two components of the isolevel set $\tilde{u} = \lambda$. As there is a finite number of *Qpixels*, there is a finite number of level lines at level λ. The consequence is that when considering a finite number of levels, there is a finite number of corresponding level lines.

Proposition 3.2 *If L is a level line at level λ, the image $\tilde{u}$, in the exterior vicinity of L, is either uniformly $< \lambda$ or $> \lambda$.*

Consider a neighborhood U of L, $U = \{x : d(x, L) < \varepsilon\}$, not meeting any other level line at level λ (which is possible due to Proposition 3.1). Then, all points of $U \setminus L$ are at a level different from λ. The union of U and the interior of L is open and connected, so as the exterior of L, hence the intersection of U and the exterior of L is connected (unicoherency of the definition set of $\tilde{u}$, see [Kur92, §41,X]). Since $\tilde{u}$ is continuous, it cannot take values $< \lambda$ and $> \lambda$ on this connected set.
In particular, this implies that the interior of a level line contains a local extremum, a fact that will be exploited in the extraction algorithm.

3.3.2 An algorithm for the extraction of the tree of bilinear level lines (TBLL)

The algorithm that is presented here is a variant of the FLST in [MG00], computing the so called *fundamental* TBLL of the image, describing the topography of the image, which in turn can be used to compute the TBLL corresponding to *any* quantization. This algorithm is proposed in [LMR01] with the name of *Morse Algorithm*, as it extracts level lines at critical points, that is, extrema and saddle points. As for a Morse function [Mil69], these levels correspond to a change in the topology of level lines [CK98].

Definition 3.2 *The* fundamental TBLL *of an image is the tree of level lines passing through a center of pixel or a saddle point.*

Therefore, all level lines containing critical points are in the fundamental TBLL. The interest is that the other level lines can be deduced from these.

Definition 3.3 *A* solid shape *associated to a level line L is the union of L and its interior.*

Therefore, a solid shape is connected, so as its complement. A *Qpixel* or *Qedgel* P is said to be adjacent to the solid shape S if

$$P \setminus S \neq \emptyset \neq P \cap S;$$

in particular, P meets the boundary of S. Since S and its complement are connected, the *Qpixels* adjacent to S can be ordered in a chain, whose each node is 4-adjacent to the following one (two *Qpixels* are 4-adjacent if they share a common *Qedgel*). As two successive *Qpixels* in this chain share a *Qedgel* adjacent to S, one can store either the chain of adjacent *Qpixels* or of *Qedgels*.

Proposition 3.3 *A Qedgel E is adjacent to a solid shape S if and only if exactly one of both extremal points of E is in S. $E \subseteq S$ if and only if both extremal points of E are in S. $E \cap S = \emptyset$ if and only if both extremal points of E are not in S.*

These properties can be easily proved using Proposition 3.2 and the fact that the restriction of the image to E is affine.

A consequence is that the datum of the chain of adjacent *Qpixels* to a solid shape S is equivalent to the knowledge of the centers of pixels in S (see Figure 3.8).

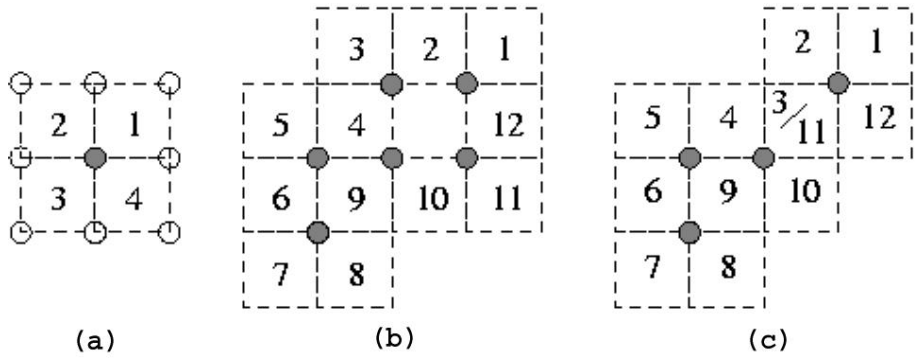


Figure 3.8: *Chains of Qpixels adjacent to solid shapes. Centers of pixels in solid shape are represented as gray dots. a: only one center of pixel in the solid shape. b: a standard configuration. c: 8-connection is made by the intermediary of a saddle point, the Qpixel containing it is present twice in the chain, at positions 3 and 11.*

DEFINITION 3.4 *The* reduced level line $L^\flat$*, associated to a level line* L*, is the boundary of the solid shape associated to* L*.*

$L^\flat$ is therefore a connected subset of L. The advantage of using $L^\flat$ instead of L is:

- $L^\flat$ and L encode the same information from the point of view of the tree structure: the solid shapes associated to them are the same.
- $L^\flat$ and L are identical for all the level lines not passing through a center of pixel or containing a saddle point.
- $L^\flat$ can be naturally described as a curve, keeping the interior at left hand side and the exterior at right hand side.

The term level *line* is thus more appropriate for $L^\flat$ than for L, the latter being even susceptible to contain *Qpixels*. Moreover, $L^\flat$ can be computed (i.e., sampled as a curve) from the knowledge of the chain of the *Qpixels* adjacent to its associated solid shape, see Figure 3.9. A reduced level line is almost a Jordan curve: it can have (a finite number of) double points, occurring at saddle points.

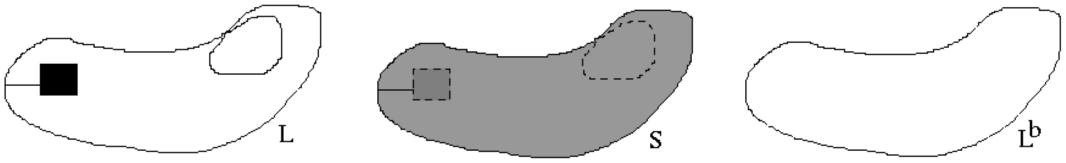


Figure 3.9: *Left: a level line* L*, containing a* Qpixel *and passing trough a saddle point. Middle: the solid shape* S *associated to* L*. Right: the reduced level line* $L^\flat$*, which is the boundary of* S*.*

Computing the fundamental TBLL is interesting because of the following properties:

1. For each level line L'' in the TBLL, there are some solid shapes L and L' in the fundamental TBLL such that L is the parent of L' and L'' is comprised between L and L'.

2. If L is in the fundamental TBLL so as L', a child of L, and if L'' is any level line comprised between L and L' in the TBLL, the chains of *Qpixels* adjacent to the solid shapes associated to L' and L'' are the same.

The second property comes from Proposition 3.3: indeed, the solid shapes associated to L' and L'' contain the same centers of pixels, implying that their adjacent *Qedgels*, and thus *Qpixels*, are the same. This property shows that the knowledge of the chains of 4-adjacent *Qpixels* of elements of the fundamental TBLL permits to deduce the ones in any TBLL.

3.3.3 Extraction of the fundamental TBLL

Proposition 3.2 is used here: extracting interiors of level lines rather than directly the (reduced) level lines themselves.

The first step consists in finding the *Qpixels* containing a saddle point, and the corresponding saddle value. It is the case when the maximum of values at diagonally opposed pixels is strictly less than the minimum of the other two values. In the algorithm described below, a *point* is either a center of pixel or a saddle point. Each has an associated value, thus two images are stored in memory: the values at centers of pixels and the values at saddle points.
To each center of pixel P, a solid shape S_P is associated: the smallest solid shape containing P. Initially, S_P is set to `NULL`.
All centers of pixels are scanned, and each time a local extremum P (comparison with 4-neighbors) at level λ is met, the following steps are performed:

1. Initialize a list $\mathcal{P}$ of points to $\emptyset$ and of neighbor points $\mathcal{N}$ to $\{P\}$.

2. While $\mathcal{N}_\lambda \neq \emptyset$, $\mathcal{N}_\lambda$ being the set of points of $\mathcal{N}$ at level λ, remove $\mathcal{N}_\lambda$ from $\mathcal{N}$ and append it to $\mathcal{P}$, and add to $\mathcal{N}$ the neighbors of $\mathcal{N}_\lambda$ not already in $\mathcal{P}$.

3. If the set of points in $\mathcal{P}$ has no hole and the points in $\mathcal{N}$ are all at level $< \lambda$ or all at level $> \lambda$, create a new solid shape with associated points $\mathcal{P}$; otherwise, put all points of image stored in $\mathcal{P}$ to level λ and exit.

4. Store the new solid shape, follow its boundary to determine the chain of adjacent *Qpixels*.

5. For each point $Q \in \mathcal{P}$, if S_Q is `NULL`, put it to the new solid shape. Otherwise, follow up the tree starting from S_Q, and put the resulting solid shape as child of the new solid shape.

6. Set λ to the closest level of points in $\mathcal{N}$ and go back to step (2).

The meaning of "neighbor" has not been precised yet. The neighbors of a center of pixel P are its 4-neighbors and the saddle points in the *Qpixels* whose one corner is P; the neighbors of a saddle point Q are the centers of pixels being corners of its containing *Qpixel*.

The number of holes is computed locally, with the following rule: two centers of pixels are connected if they are 4 - neighbors, or if they are diagonally opposed in the *Qpixel* and at least one of the other two corners or the saddle point inside the *Qpixel* (provided there is one) is in the set. Then the number of holes is computed by counting patterns of connected neighbors, in a manner similar to that exposed in [KR89, KR90b].

Notice that it is important to order the points in $\mathcal{N}$ by their level. This can be done with a balanced binary tree, similar to the one used in *heap sort* [Sed99].

After the image is scanned, its gray levels have been changed and it becomes uniform. The resulting constant level is the one of the root of the tree.

3.3.4 Computation of a TBLL from the fundamental TBLL

From the knowledge of the fundamental TBLL and a given quantization, it is direct to compute the resulting TBLL. For each solid shape S in the fundamental TBLL, find the levels of the quantization comprised between the level of S (strictly) and the level of its parent S' in the fundamental TBLL. Create a solid shape for each one; the level line passes through the same chain of adjacent *Qpixels* as S'.

For each solid shape created in this manner, their order in the TBLL is the same as the one of their least greater solid shapes in the fundamental TBLL. In that sense, the TBLL is a sampling of the fundamental TBLL, see Figure 3.10.

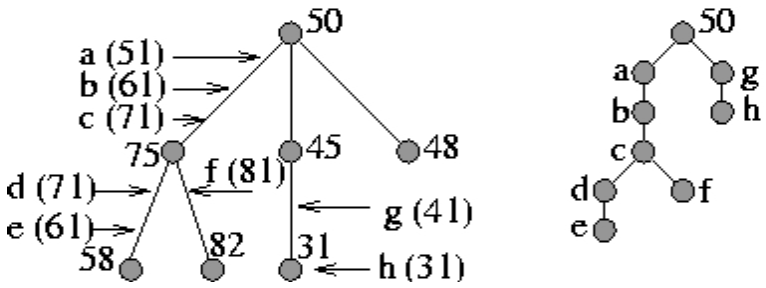


Figure 3.10: *Computing the TBLL of quantization levels* $\{1, 11, 21, ...\}$ *from the fundamental TBLL. Left: fundamental TBLL, showing associated levels. Right: the resulting TBLL is obtained by sampling of the fundamental TBLL.*

Chapter 4

Extracting meaningful curves from images

Abstract: Since the beginning, Mathematical Morphology has proposed to extract shapes from images as connected components of level sets. These methods have proved very efficient in shape recognition and shape analysis. In this chapter, we present an improved method to select the most meaningful level lines (boundaries of level sets) from an image. This extraction can be based on statistical arguments, leading to a parameter free algorithm. It permits to roughly extract all pieces of level lines of an image that coincide with pieces of edges. By this method, the number of encoded level lines is reduced by a factor 100, with almost no loss of shape contents. In contrast to edge detection algorithms or snakes methods, such a level lines selection method delivers accurate shape elements, without user parameter since the parameter selection can be derived from the Helmholtz Principle. This chapter aims at improving the original method proposed in [DMM01]. We give a mathematical interpretation of the model, which explains why some pieces of curve are overdetected. We introduce a multiscale approach that makes the method more robust to noise. A more local algorithm is introduced, taking local contrast variations into account. Finally, we empirically prove that regularity makes detection more robust but does not qualitatively change the results.

Résumé : Depuis sa fondation, la morphologie mathématique a proposé d'extraire les formes des images comme des composantes connexes d'ensembles de niveau. Il a été prouvé que ces méthodes sont très efficaces pour la reconnaissance des formes comme pour leur analyse. Dans ce chapitre, nous présentons une méthode pour sélectionner les lignes de niveau (frontières des ensembles de niveau) les plus significatives. Cette extraction peut être basée sur des arguments statistiques conduisant à un algorithme sans paramètre. Celui-ci permet d'extraire à peu près tous les morceaux de lignes de niveau qui correspondent à un morceau de bord. Par cette méthode, le nombre de lignes de niveau codées est réduit d'un facteur 100, sans presque aucune perte sur le contenu des formes. Au contraire des méthodes de détection de bords ou de contours actifs, une telle sélection des lignes de niveau donne des éléments de forme précis, sans paramètre utilisateur car les paramètres peuvent être calculés par le principe de Helmholtz. L'objectif de ce chapitre est d'améliorer la méthode originale proposée dans [DMM01]. Nous donnons une interprétation mathématique du modèle, qui explique pourquoi certains morceaux de courbes sont surdétectés. Nous introduisons une approche multiéchelle qui

rend la méthode robuste vis-à-vis du bruit. Un algorithme plus local est introduit, prenant en compte les variations locales du contraste. Enfin, nous prouvons de manière empirique que la régularité rend les détections plus robustes mais ne change pas qualitativement les résultats.

This chapter corresponds to the article "Extracting meaningful curves from images" (F. Cao, P. Musé and F. Sur) [CMS04].

4.1 Introduction

Natural images are very complex, and despite the progress of modern computers, we cannot handle the huge amount of information they contain. Thus, the idea of Marr and Hildreth [MH80] that edges provide a good summary of images is still vivid. Since their seminal work, efforts have been carried on local methods. Marr defined edges as zero-crossings of the Laplacian [Mar82], and Haralick [Har84] proposed a more correct definition which is equivalent to the zero-crossings of $D^2u(Du, Du)$ where Du and D^2u are respectively the gradient and the second derivative of the image. In his famous paper [Can86], Canny gives a filter that tries to optimize the edge localization (as a trade-off with signal to noise ratio), but which is equivalent to Haralick's. Although they are technically sound, local methods have an immediate drawback: while edges are usually thought about as curves, these methods detect sets of points with an orientation (*edgels*) that have to be connected afterward. Moreover, they require different thresholds since contrast has no absolute meaning. In addition, they are very sensitive to noise, since they use derivatives of the image. The choice of these thresholds depends on the observed image, and is not that easy. It is also known that edge is not a completely local concept and that it does not rely entirely on contrast. Indeed, following Gestalt Theory [Kan96, Wer23], shapes (and thus edges) result from the collaboration of a small set of perceptual laws (called "partial gestalts" by Desolneux, Moisan and Morel [DMM03a]), and contrast is only one of them. Among others, we can cite alignments, symmetry, convexity, closedness and good continuation.

Other theories, related to edge detection, explicitly use good continuation, which means in this case regularity of curves. The most famous one is certainly the theory of active contours (or snakes) [KWT87], where optimal boundaries result from a compromise between their intrinsic regularity and the extrinsic value of the image contrast along the active contours. The main weaknesses of this theory are the number of parameters and the sensitivity to an initial guess. More recent methods propose to initiate the detection with many contours, most of which will hopefully disappear [CV01]. But again, there is no measure on the certainty of the remaining detected contours.

The Mathematical Morphology school proposed an alternative to the local approaches above. Following morphologists, the image information is completely contained in a family of binary images that are obtained by thresholding the images at given values [Mat75, Ser82]. This is equivalent to considering level sets; the level set of u at the value λ is

$$\chi_\lambda(u) = \{x \in \mathbb{R}^2, \quad u(x) \geqslant \lambda\}. \tag{4.1}$$

Obviously, if we only consider a coarsely quantized set of different grey levels, information is lost, especially in textures. Nevertheless, it is worth noting how large shapes are already present with as few as 5 or 6 levels. As soon remarked by Serra [Ser82], no information is lost at all, since we can reconstruct an image from the whole family of its level sets, by

$$u(x) = \sup\{\lambda \in \mathbb{R}, \quad x \in \chi_\lambda(u)\}.$$

Thus, the level sets do not only give a convenient way to extract information, they provide a complete representation of images. Alternative complete representations are, for instance, Fourier or wavelets decomposition [Mal99]. But while these last ones are very adequate for image compression (they are used in the JPEG 2000 standard), they are not very well adapted in shape analysis, since their basic elements have no immediate perceptual interpretation. (More recent decompositions as bandlets [PM04] or curvelets [SCD02] try to take image geometry more into account, but they are either still too local, or need a preliminary detection step.) On the contrary, morphologists soon remarked that boundaries of level sets fit parts of objects boundaries very well. They call level lines the topological boundaries of connected components of level sets, and topographic map of an image, the collection of all its level lines. The topographic map gives a complete representation of an image and enjoys several important advantages [CCM99]:

- It is invariant with respect to contrast changes. It is not invariant to illumination change, since in this case, the image is really different, although it represents the same scene. However, many level lines still are locally the same.
- It is not as local as sets of edges, since level lines are Jordan curves that are either closed or meet the image borders. (This property requires that the image has bounded variations [ER92]).
- It is a hierarchical representation: since level sets are ordered by the inclusion relation (and so are there connected components), the topographic map may be embedded in a tree structure.
- But most important regarding the main subject of this paper, object contours locally coincide with level lines very well. Basically, level lines are everywhere normal to the gradient as edges. On the other hand, level lines are accurate at occlusions. Whereas, edges detectors usually fail near T-junctions (and additional treatments are necessary), there are several level lines at a junction. The order of the multiple junction coincides with the number of level lines [CCM96]. We shall go back to this in Section4.3.2.

The level sets representation has recently been used, with success, for image simplification and segmentation. In particular, it was shown that it allowed to define multiscale representation of images [MM00, SG00, SS95], while avoiding the main drawbacks of linear scale space theory [Koe84, Wit83], namely an oversmoothing of contours.

We are convinced that level lines may directly give usable curves for any shape recognition algorithm. The main drawback of the topographic map representation is its lack of compactness. First, since it is

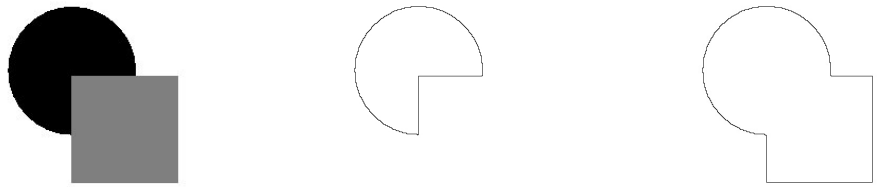


Figure 4.1: *Level lines and T-junction. Depending on the grey level configuration between shapes and background, level lines may follow or not (as on the figure) the objects boundary. In any case, junctions appear where two level lines separate. Here, there are two kinds of level lines: the occluded circle and the shape composed of the union of the circle and the square. The square itself may be retrieved by difference.*

complete, it contains all the texture information. The level lines in textures are usually very complicated, and are not always useful for blind shape recognition. (The opposite may be true, for instance for very accurate image registration). Moreover, because of noise and interpolation, many level lines may follow roughly one and the same contour. Thus, it is useful, for practical computational reasons, to select only the most meaningful level lines.

Recently, Desolneux *et al.* proposed a parameterless algorithm to detect contrasted level lines (called *meaningful boundaries*) in grey level images [DMM01]. Their method, which needs no parameter tuning, relies on a perceptual principle called Helmholtz Principle. Experimentally, meaningful boundaries are often very close to minimizers of any reasonable snake energy [DMM03b]. This adequation of meaningful boundaries and snakes is a bit paradoxical since, unlike snakes, no local regularity is imposed on meaningful boundaries.

However, the algorithm of Desolneux *et al.* raises several questions and objections. The definition of meaningful boundaries has first to be precisely interpreted in mathematical terms. Second, because of noise (and certainly partly because of quantization noise), some edges are missing (lots of them in some low contrasted images). Third, it uses a global information on contrast (the histogram). This yields an overdetection in regions with important contrast and a subdetection in low contrasted regions (it is the so-called blue sky effect). Finally, regularity of edges is not used for the detection.

In this paper, we discuss these objections and propose some answers, with a significant improvement. Our conclusions are the following: the definition of meaningful boundaries themselves does not ensure that they do not contain any undesirable parts. We propose a method to remove those parts. Second, the method can be extended to several scales, and this makes the method more robust to noise. We also propose a method considering contrast in a more local way. If we use more local contrast information, we can remove edges in texture. Whether this is useful or not depends on the application: for very accurate registration, texture-edges can be useful, while they must be useless for shape recognition. (For texture recognition, harmonic analysis methods are certainly more efficient.) Last, we introduce a local and stable measure of regularity of a curve and use it for smooth edges

detection. As already noticed in [Cao03], regularity is often sufficient to detect some very meaningful edges. Nevertheless, general belief is that both regularity and contrast are useful for edge detection. We experimentally check that contrast and regularity are often very redundant. This redundancy is used to make the detection even more robust, but does not change the results of contrast based detection alone. We are also able to tune automatically the relative weight of regularity and contrast, which is a recurrent question in active contours theory.

The plan of this chapter is as follows. In section 4.2, we recall the bases of Helmholtz Principle, the definition of meaningful boundaries of Desolneux, Moisan and Morel. We will justify and discuss this definition, which was not explicitly made in [DMM01]. A multiscale extension is presented in section 4.3. In section 4.4, we describe a procedure that automatically handles local contrast variations. In section 4.5, we explain how both contrast and regularity criteria can naturally be mixed in a probabilistic setting by introducing a measure of regularity on random level lines in section 4.5.1. We conclude in section 4.6.

4.2 Meaningful boundaries

4.2.1 Helmholtz Principle

Helmholtz Principle is a perceptual principle asserting that conspicuous structures may be viewed as exceptions to randomness. The unexpected configurations we must be interested in, are given by the perceptual laws of Gestalt Theory [Kan96, Wer23], as alignments, closedness of sets, parallelism, *etc*. Since this principle is quite general, its formulation may slightly vary from one application to another but, we propose the following formulation. Assume that O_1, ..., O_N are local objects in an image (for instance, O_1, ..., O_N may be edgels, that is points assigned with a direction). We want to find out whether some of these objects must be grouped in a more global structure, with respect to some shared quality. Let us assume that we have K group candidates G_1, ... G_K. Each of the G_i gathers several of the local objects O_n, given in advance. We now consider a quality Q measured from the O_n. Each measure defines a random variable X_n. We wish to determine if the G_i are meaningful groups for the quality Q. We then carry out the following mental experiment: assume that, anything else held equal, the quality Q is independently and identically distributed over the O_n, that is to say the X_n are i.i.d. variables. If no a priori information is available on X_n, we call this hypothesis the *a contrario* model. If we have no a priori information on the X_n, their distribution in the *a contrario* model can be, for instance, their distribution in a white noise image. Assume that for some group G_i, the X_n are equal up to some precision. By definition, we will say that G_i is ε-meaningful, if in the *a contrario* model, the probability that all X_n in G_i are equal up to the observed precision is less than $\frac{\varepsilon}{K}$. As will be seen on a more precise example in the following sections, this definition implies that, in the *a contrario* model, the expected number of ε-meaningful groups is less than ε. In other words, the number of groups appearing by chance is controlled, on the average, by ε.

We refer the reader to [DMM03a] and references therein, for precise applications of this principle.

The following sections are devoted to the application of this principle to the extraction of shape information.

Before going further, let us give a few comments on the above principle. A very important point is that it is discrete by nature. Indeed we consider a finite number of local objects, and we also consider a finite a priori number of group candidates. Moreover, the quality Q is measured with a finite accuracy. Put together, this implies that, under the *a contrario* model, any group has a positive probability of occurence. This probability is decreasing with the size of the group (the number of local events it contains). Moreover, for all $\varepsilon > 0$ the number of ε-meaningful groups is obviously bounded by K (the total number of group candidates). Assume now that, to the previous K group candidates, we add K' new candidates. Then, for a group G_i to be meaningful, its probability of occurence in the random model has now to be smaller. If K' tends to $+\infty$, this probability of occurence must go to zero. This means that the meaningfulness depends on the size of the data. This size also has to be finite, otherwise no group can be ε-meaningful for $\varepsilon > 0$. This is completely compatible with digital image processing where image sampling implies a finite amount of information. Moreover, a digital white noise image should yield no detection. It is thus sound to construct the *a contrario* model such that it is true at least in the case of white noise. In order to be coherent with Shannon's sampling theory, we have to assume that the distance between the objects O_i is larger than the Nyquist distance, namely 2 pixels, for they have to be independent in noise. In the following, we shall say that two points are independent if their distance is larger than 2 pixels.

Even though digital images are discrete by nature, it is often convenient to consider grey level images as functions from $\mathbb{R}^2$ to $\mathbb{R}$, as we will do in the following. In practice, we use a bilinear interpolation, which allows us to define level lines at any level. We also use finite differences scheme to define a contrast value which is consistent with the gradient.

A last important comment is the choice of ε, which is the only decision parameter. Of course the principle can be efficient only if it is robust with respect to ε. In fact, it is often possible to prove (see [DMM03a] and the following of this chapter) that the minimal length of an ε-meaningful boundary depends on the logarithm of ε. In practice, setting $\varepsilon = 1$ means that we have less than one detection in the *a contrario* model. Thus we choose $\varepsilon = 1$, and check a posteriori that changing this value does not change the results, as predicted by the theory. One main reason why this is empirically very stable is that detected structures are (very) large deviations from the *a contrario* model and can be detected with some values of ε even less than 10^{-10}.

4.2.2 Contrasted boundaries

In order to illustrate Helmholtz principle, we recall here the definition of meaningful boundaries given in [DMM01]. It will be also useful since we will discuss this definition in the next sections. Let $u : \mathbb{R}^2 \to \mathbb{R}$ be a differentiable grey level image. Assume that we have a measure of contrast. To simplify we take it here equal to the norm of the gradient. Assume that we know the distribution of

the gradient of u given by

$$H_c(\mu) = P(|Du| > \mu).$$

In fact, we do not really care that this contrast is the gradient norm of a differentiable function, and in practice, we shall take a finite differences approximation of the gradient. In [DMM01], Desolneux, Moisan and Morel proposed the following definition.

DEFINITION 4.1 ([DMM01]) *Let E be a finite set of N_{ll} level lines of u. A level line C is an ε-meaningful boundary if*

$$NFA(C) \equiv N_{ll} H_c(\min_{x \in C} |Du(x)|)^{l/2} < \varepsilon, \tag{4.2}$$

where l is the length of C. This number is called number of false alarms (NFA) of C.

We first remark that ε, N_{ll} and $\min_{x \in C} |Du(x)|$ being fixed, the minimal l such that C is ε-meaningful depends on the logarithm of the other parameters. This was already discussed in [DMM03a], and in practice we can take $\varepsilon = 1$ in all experiments. This definition will be further commented in 4.2.4, and we just shortly describe its implementation. We first need an a priori on the gradient law. The approximation by the empirical histogram is used. That is, we assume that the gradient norm is distributed following the law of the positive random variable X defined by

$$\forall \mu > 0, \quad P(X > \mu) = \frac{\#\{x \in \Gamma,\ |Du(x)| > \mu\}}{\#\{x \in \Gamma,\ |Du(x)| > 0\}}, \tag{4.3}$$

where the symbol $\#$ designs the cardinality of a set, Γ the finite sampling grid, $|Du|$ is computed by finite difference approximation, and we assume that it is constant in each pixel. Moreover, we need a finite and reasonable set of level lines. Since images are assumed continuous, they have an infinite number of level lines. These lines are very redundant since interpolated images are very smooth. Thus, it is soundly assumed that quantized level lines contains all the information of the image. It is perceptually known that beyond a few hundreds of grey levels, we are not able to distinguish intensity differences. So we naturally quantize 8-bits encoded images every integer levels. Dividing the quantization step by 10 will approximately multiply the number of level lines by 10. Thus the number of false alarms of a given line will also be increased by a factor 10, which has nearly no incidence on the detection. For interpolated images, quantization yields a finite number of level lines. This number N_{ll} is dependent on the image; textured images have more level lines than more simple images. To give an order of magnitude, a natural 256×256 image contains between 10^4 and 10^5 level lines.

4.2.3 Maximal boundaries

Since level lines are nested, meaningful boundaries can also be embedded in a tree structure. To make things simple, a level line L_2 is a descendent of another line L_1 in the tree if and only if L_2 is included in the interior of L_1. This is not obvious for continuous images, but was proved by Monasse [Mon00]. Since only quantized grey levels are considered, the tree of level lines is also

quantized and contains only a finite number of nodes. As remarked by Desolneux, Moisan and Morel [DMM00], meaningful boundaries usually appear in parallel groups, because of interpolation. Moreover, since images are made band-limited before sampling, they are blurry and there is a transition layer around objects boundaries of width at least two or three pixels. These boundaries are redundant, and in applications, it may be useful to eliminate some of them. The previous authors first remarked that the meaningful level lines inherit the tree structure of the original tree. The idea is to use this structure to efficiently remove redundant boundaries.

DEFINITION 4.2 ([MONO00]) *A monotone section of a level lines tree is a part of a branch such that each node has a unique son and where grey level is monotone (no contrast reversal). A maximal monotone section is a monotone section which is not strictly included in another one.*

DEFINITION 4.3 ([DMM01]) *We say that a meaningful boundary is maximal meaningful if it has a minimal NFA in a maximal monotone section.*

Figure 4.2 illustrates the fact that the loss of information of maximal meaningful boundaries is negligible, compared to the gain of information compactness.

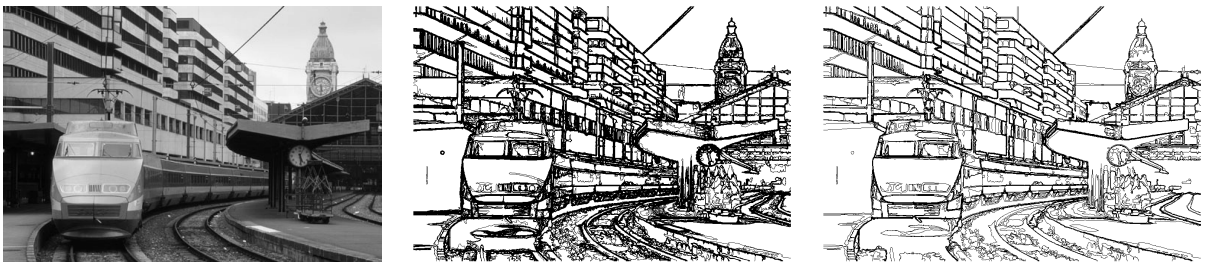


Figure 4.2: *Maximal meaningful boundaries. 1. Original image, 83,759 level lines 2. All meaningful boundaries: 11,505 detections. 3. Maximal meaningful boundaries. Only 883 boundaries remain, while the visual loss is very weak.*

Since meaningful boundaries inherit the tree structure of the topographic map, they can be used to reconstruct an image, thus defining an image operator, see Figure 4.3. It is a connected operator as defined by Salembier and Serra [SS95] (but it is not a filter by reconstruction). It is neither a contrast invariant operator, since it explicitly uses the gradient value (it only commutes with affine global contrast change), nor an idempotent operator (since meaningfulness depends on the number of level lines in the original tree).

As remarked by Salembier *et al.* [SG00], an operator pruning the topographic maps preserves edges very well. Contrarily to local operators as, for instance, the grain filter [Vin93], the meaningful boundary reconstruction does not simply remove leaves of the tree (small level lines) but also inner nodes corresponding to possibly large (but low contrasted) level lines.

Figure 4.3: *Original image on the left (99,829 level lines). Right: reconstruction from the 429 maximal meaningful boundaries. The grey level may not be really significant since, on edges, the maximal meaningful level line has an intermediate level between both sides of the edge. It would be more perceptually adequate to set the grey level to the brighter or darker meaningful level line. Nevertheless, for contrast independent shape recognition purposes, we do not use the grey level value, but only the geometry of level lines. The most important is that we preserved the main geometric structure, while removing the textures.*

4.2.4 Discussion on the definition of meaningful contrasted boundaries

Interpretation of the number of false alarms

In this section, we give a precise interpretation of Definition 4.1, which was not explicit in [DMM01]. Let us first recall the following classical lemma.

Lemma 4.1 *Let X be a real random variable and $H(x) = P(X \geqslant x)$. Then for all $t \in [0,1]$,*

$$P(H(X) < t) \leqslant t.$$

Assume that X is a real random variable described by the inverse repartition function $H(\mu) = P(X \geqslant \mu)$. Assume that u is a random image such that the values $|Du|$ are independent with the same law as X. Let now E be a set of random curves (C_i) in u such that $\#E$ (the cardinality of E) is independent from each C_i. For each i, we note $\mu_i = \min_{x \in C_i} |Du(x)|$. We also assume that we can choose L_i independent points on C_i (points that are afar at least by Nyquist's distance). We can think of the C_i as random walks with independent increments but since we choose a finite number of samples on each curve, the law of the C_i does not really matter. We assume that L_i is independent from the pixels crossed by C_i.
We say that C_i is ε-meaningful if

$$NFA(C_i) = \#E \cdot H(\mu_i)^{L_i} < \varepsilon.$$

PROPOSITION 4.1 *The expected number of ε-meaningful curves in a random set E of random curves is smaller than ε.*

Proof: Let us denote by X_i the binary random variable equal to 1 if C_i is meaningful and to 0 else. Let also $N = \#E$. Let us denote by $\mathbb{E}(X)$ the expectation of a random variable X in the *a contrario* model. We then have

$$\mathbb{E}\left(\sum_{i=1}^{N} X_i\right) = \mathbb{E}\left(\mathbb{E}\left(\sum_{i=1}^{N} X_i | N\right)\right).$$

We have assumed that N is independent from the curves. Thus, conditionally to $N = n$, the law of $\sum_{i=1}^{N} X_i$ is the law of $\sum_{i=1}^{n} Y_i$, where Y_i is a binary variable equal to 1 if $nH(\mu_i)^{L_i} < \varepsilon$ and 0 else. By linearity of expectation,

$$\mathbb{E}\left(\sum_{i=1}^{N} X_i | N = n\right) = \mathbb{E}\left(\sum_{i=1}^{n} Y_i\right) = \sum_{i=1}^{n} \mathbb{E}(Y_i).$$

Since Y_i is a Bernoulli variable,

$$\mathbb{E}(Y_i) = P(Y_i = 1) = P(nH(\mu_i)^{L_i} < \varepsilon) = \sum_{l=0}^{\infty} P(nH(\mu_i)^{L_i} < \varepsilon | L_i = l) P(L_i = l).$$

Again, we have assumed that L_i is independent of the gradient distribution in the image. Thus conditionally to $L_i = l$, the law of $nH(\mu_i)^{L_i}$ is the law of $nH(\mu_i)^l$. Let us finally denote by $(\alpha_1, \cdots, \alpha_l)$ the l (independent) values of $|Du|$ along C_i. We have

$$\begin{aligned}
P\left(nH(\mu_i)^l < \varepsilon\right) &= P\left(H(\min_{1 \leqslant k \leqslant l} \alpha_k) < \left(\frac{\varepsilon}{n}\right)^{1/l}\right) \\
&= P\left(\max_{1 \leqslant k \leqslant l} H(\alpha_k) < \left(\frac{\varepsilon}{n}\right)^{1/l}\right) \text{ since } H \text{ is nonincreasing} \\
&= \prod_{k=1}^{l} P\left(H(\alpha_k) < \left(\frac{\varepsilon}{n}\right)^{1/l}\right) \text{ by independence} \\
&\leqslant \frac{\varepsilon}{n} \text{ from Lemma 4.1.}
\end{aligned}$$

This term does not depend upon l, thus

$$\sum_{l=0}^{\infty} P(nH(\mu_i)^{L_i} < \varepsilon | L_i = l) P(L_i = l) \leqslant \frac{\varepsilon}{n} \sum_{l=0}^{\infty} P(L_i = l) = \frac{\varepsilon}{n}.$$

Hence,

$$\mathbb{E}\left(\sum_{i=1}^{N} X_i | N = n\right) \leqslant \varepsilon.$$

This finally implies $\mathbb{E}\left(\sum_{i=1}^{N} X_i\right) \leqslant \varepsilon$, which exactly means that the expected number of meaningful curves is less than ε. ∎

In this proposition, we have not assumed a priori that the C_i are level lines of u. Indeed, in this case, we cannot certainly assert that the length (number of independent points) of the curve is independent from the values of the gradient along the curve.

Cleaning-up meaningful boundaries

Proposition 4.1 asserts that if a curve is a meaningful boundary, then it cannot be *entirely* generated in white noise (up to ε false detections on the average). On the other hand, can we guarantee that no part of a meaningful boundary is contained in noise ? Or, for a given meaningful boundary, can we give an upper bound of the size of the part of the boundary that is likely to be contained in noise (i.e. a non-edge region) ? To answer this question, we use the a posteriori length distribution

$$P(L \geqslant l | \min_{x \in C} |Du(x)| \geqslant \mu). \tag{4.4}$$

Contrarily to the probability appearing in Definition 4.1, this one penalizes long curves not only through the gradient value. To compute it, we need the a priori distribution $P(L \geqslant l)$ that a level line in noise has a length larger than l. As we do not know this distribution explicitly, we choose to estimate this law empirically. For $l \leqslant 1000$ (to give an order of magnitude), the number of lines whose length is larger than l is still quite large (for images of size about 500×500), and we assume that the distribution is quite correctly estimated for such length. (See Figure 4.4.) For higher values, there are too few level lines. By using Bayes' rule, we derive

$$P(L \geqslant l | \min_{x \in C} |Du(x)| \geqslant \mu) = \frac{\sum_{k=l}^{\infty} P(\min_{x \in C} |Du(x)| \geqslant \mu | L = k) P(L = k)}{\sum_{k=1}^{\infty} P(\min_{x \in C} |Du(x)| \geqslant \mu | L = k) P(L = k)}.$$

(The denominator is nothing but $P(|Du| > \mu)$). By the *a contrario* assumption (independence of

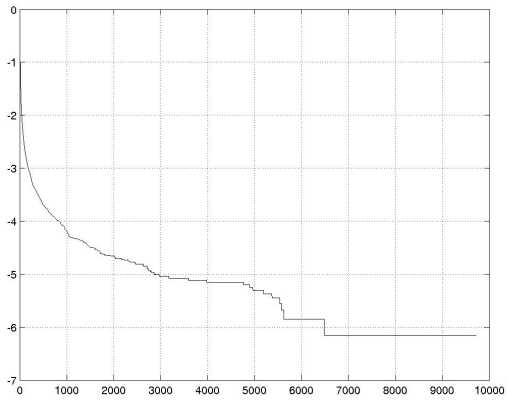


Figure 4.4: *log_{10} of the inverse repartition function of length of level lines in a white noise image. The average length is about 3.5, meaning that most level sets enclose a single pixel.*

the gradient along curves), we can still write

$$p_\mu(l) \equiv P(L \geqslant l | \min_{x \in C} |Du(x)| \geqslant \mu) = \frac{\sum_{k=l}^{\infty} H_c(\mu)^k P(L = k)}{\sum_{k=1}^{\infty} H_c(\mu)^k P(L = k)}. \tag{4.5}$$

Let us now consider an image u with N_{ll} (quantized) level lines. We also denote by N_l the number of all possible sampled subcurves of these level lines. (N_l is the sum of the squared number of independent points of the lines if they are closed).

Assume that C is a piece of level line with L independent points, contained in a non-edge part, described by the noise model. We want to estimate the probability that L is larger than $l > 0$, knowing that $|Du| \geqslant \mu$. This is exactly $p_\mu(l)$, the probability defined in (4.5). As in Proposition 4.1, we can prove that $N_l \cdot p_\mu(l)$ is an upper bound of the expected number of *pieces* of lines of length larger than l with gradient larger than μ. For a fixed μ, let be l such that $N_l \cdot p_\mu(l) \leqslant \varepsilon$. Then, we know that, on the average, we cannot observe more than ε pieces of level line with a length larger than l and a gradient everywhere larger than μ. We make the assumption that a point with a gradient less than μ is located in noise. Let us remove any piece of length l containing such a point. Then all remaining points belongs to a piece of curve with length larger than l with gradient larger than μ, which cannot be due to chance. This yields a clean-up algorithm for boundary detection.

1. Detect meaningful boundaries.
2. For a fixed $\mu > 0$, let $\mathcal{L}(\mu) = \inf\{l,\ N_l \cdot p_\mu(l) < \varepsilon\}$.
3. For any meaningful boundary, remove every subcurve of length $\mathcal{L}(\mu)$ containing a point where $|Du| \leqslant \mu$.

This introduces a parameter, μ. When μ gets larger, $\mathcal{L}(\mu)$ decreases, so that the clean-up removes more numerous but smaller pieces of curves. The choice of μ can be determined by applicative considerations. Detected edges may be used for different purposes, for instance shape recognition or image matching. Letting $|Du|$ less than 1, means that we may detect edges with an accuracy less than one pixel. Thus choosing to eliminate pieces of curves with a gradient larger than $\mu = 1$ for all images is not restrictive. In practice, the remaining pieces of level lines have a gradient much larger than 1 and can be well enough located. We also check that for μ about 1, we obtain values of $\mathcal{L}(\mu)$ less than a few hundreds, which is compatible with the empirical estimation of the a priori length distribution.
Figure 4.5 illustrates the result of a meaningful boundary clean-up.

4.3 Multiscale meaningful boundaries

4.3.1 Meaningful boundaries by downsampling

As previously noted, the contrast measure is an approximation of the gradient by finite differences. More precisely, Desolneux *et al.* [DMM01] use the following scheme:

$$\frac{\partial u}{\partial x} \simeq u_x(i,j) = \frac{1}{2}(u(i+1,j) + u(i+1,j+1) - u(i,j) - u(i,j+1)), \tag{4.6}$$

$$\frac{\partial u}{\partial y} \simeq u_y(i,j) = \frac{1}{2}(u(i,j+1) + u(i+1,j+1) - u(i,j) - u(i+1,j)). \tag{4.7}$$

Using a 2×2 scheme is coherent with the application of Helmholtz principle: points afar from the Nyquist distance have independent values of contrast in white noise. On the other hand, this measure

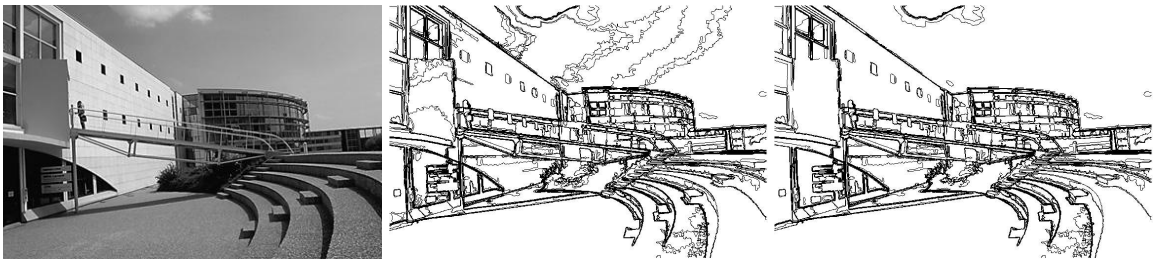


Figure 4.5: *Meaningful boundary clean-up. On the left the original image. In the middle, the meaningful boundaries with local histograms, see section 4.4. Boundaries are found in the sky. They are detected since the gradient in the sky is regular because of the smoothly changing illumination. The gradient value is about* 0.2. *Even though they are not smooth at small scale (they cannot be well located, due to the too small gradient), they are nearly parallel at large scales, which can be explained, a posteriori. Now, these boundaries may not be very useful for shape recognition purposes, because of their bad localization. On the right, the result after the clean-up procedure with a gradient threshold equal to* 1.

is sensitive to noise. This problem was known from Marr and Hildreth [MH80] who considered that edge detection should be multiscale. They compute the zero-crossing of the Laplacian of the image convoluted with Gaussians with different standard deviations. Since edges at larger scale are badly located, they propose to track back the strongest edges to smaller scales, which is not obvious in practice. Smoothing introduces local dependencies between pixels, making the *a contrario* model false in smoothed white noise. Nevertheless, the *a contrario* model still applies if we downsample the image at a lower frequency, given by the amount of smoothing. More precisely, we apply the following algorithm. Consider a set $\{1, 2, ..., 2^{N_s-1}\}$ of N_s dyadic scales. For any level line C, we denote by C^s the curve $\frac{C}{2^s}$, obtained by scaling C by a factor 2^{-s}. We also denote by H^s the empirical contrast distribution of u^s, where u^s is obtained by downsampling u with a factor 2^s, conformly to Shannon's theory. (That is to say, downsampling follows an adequate smoothing, for instance convolution with a prolate function.)

1. Compute the quantized level lines of the image u.

2. For each level line C with l independent points in u, compute μ^s, the minimal value of $|Du^s|$ over all pixels crossed by C^s. Let

$$NFA(C) = N_s \cdot N_{ll} \min_{s\in\{0,\cdots,N_s-1\}} (H^s(\mu^s))^{l/2^s}. \tag{4.8}$$

   We say that C is meaningful if $NFA(C) < \varepsilon$.

Thus, a curve is meaningful if and only if there exists a scale such that it is $\frac{\varepsilon}{N_s}$ meaningful in the sense of the previous section. A direct corollary of the linearity of expectation and of Proposition 4.1 is that the expected number of ε-meaningful multiscale boundaries is less than ε in the *a contrario* model. Note that C^s is not a level line of u^s, but this is not required in Proposition 4.1. Moreover, if C was already $\frac{\varepsilon}{N_s}$-meaningful, then we are sure that C is still detected by the multiscale method. It is

clear that we only consider a small number of dyadic scales (say 3 or 4), else images will only contain a few pixels. Since the detection depends on $\log \varepsilon$, we do not eliminate many lines by considering $\frac{\varepsilon}{N_s}$-meaningful boundaries at each scale. On the other hand, the method should be numerically less sensitive to white noise, since filtering followed by downsampling reduces noise. On Figure 4.6 and 4.7, we show the result of this multiscale method on images with quantization noise and additive Gaussian noise.

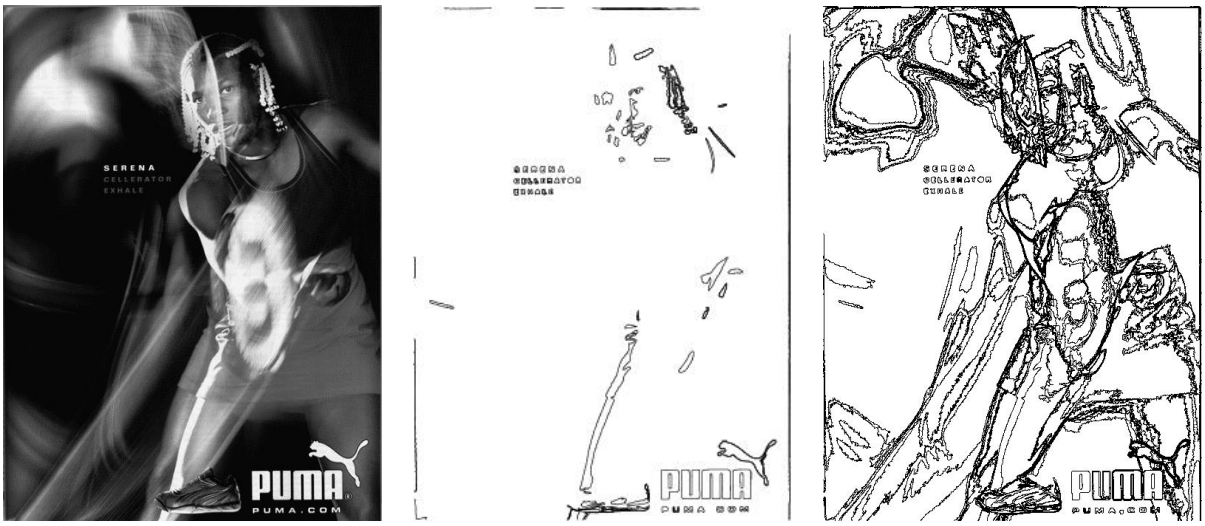


Figure 4.6: *Influence of quantization noise on meaningful boundaries. On the left, the original image is coarsely quantized since it has a very low contrast. This leads to bad gradient estimation and a lot of missing detections (middle). Multiscale detection is less sensitive to quantization noise and leads to more correct detections (right).*

4.3.2 Meaningful boundaries vs. Haralick's detector

In this section, we comment the main differences between the meaningful boundary model and the classical edge detector introduced by Haralick. The meaningful boundaries are based on the topographic map of grey level images, which gives a complete topological representation of grey level images. Caselles, Coll and Morel [CCM96, CCM99] detail all the properties of this representation. A first advantage of this representation is its stability: even with an important amount of noise, many level lines do not change much. Our multiscale approach also makes the detection quite robust. (See Figure 4.7.) A second advantage is its invariance with respect to global contrast change. Meaningful boundaries are not contrast invariant since they use the distribution of contrast, but they are still invariant with respect to affine contrast change. But the main property is the structure of this representation: it is a set of nested curves that are either closed or meet the image boundary. As a consequence, level lines have two of the main properties usually expected in edge detection or image segmentation: they are curves (and not sets of points), and are embedded in a hierarchical structure [FH98, MS89, SM00]. Moreover, away from critical points, level lines coincide with isophotes. As a consequence, for almost any level, the gradient is almost everywhere normal to level lines, which

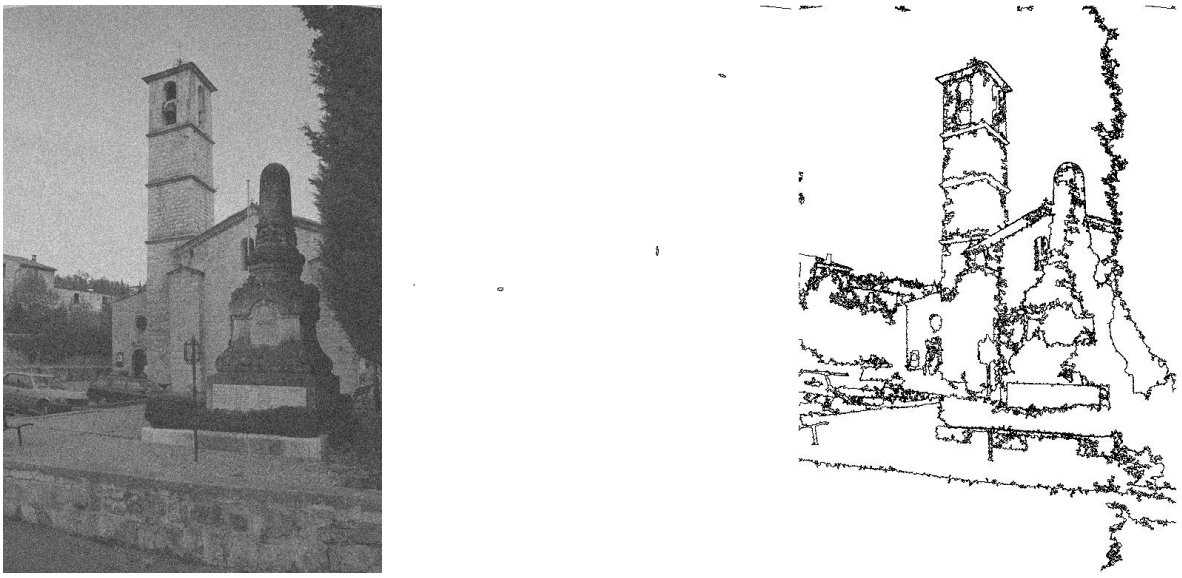


Figure 4.7: *Multiscale meaningful boundaries and noise. Left: image of Figure 4.10 with an additive white Gaussian noise of standard deviation 30. Middle: meaningful boundaries. Since noise dominates the gradient distribution, only 6 small level lines are detected. Right: multi-scale detection using 4 dyadic scales. Textures are not detected, meaning that noisy textures are in this case not different enough from noise to be detected. On the other hand, main structures remain. This allow to empirically check the stability of the topographic map in spite of the important amount of noise.*

makes level lines good candidates for edges.

Following Haralick [Har84], edges are the maxima of the gradient norm in the direction of the gradient, such that the gradient is larger than a given threshold. Thus, for a grey-level image u, they are the zero-crossings of $D^2u(Du, Du)$. Since, this quantity is numerically sensitive to noise, a multiscale strategy *à la* Marr is applied. Thus in practice, u is first convolved with a Gaussian with standard deviation σ (we denote by g_σ this Gaussian and $u_\sigma = g_\sigma * u$) and the points where $D^2u_\sigma(Du_\sigma, Du_\sigma)$ changes sign and $|Du_\sigma| > \mu$ are edges points. Although there have been some attempts to automatically determine the scale parameter σ [Lin98], edge detection widely remains multiscale as predicted by Marr [Mar82], and it is quite difficult to track edges back to small scales. The multiscale meaningful boundaries detection of the previous section allows to consider different scales, while keeping detection thresholds completely automatic. Moreover, the number of scales has a log influence. Haralick's detector provides with a set of points or a few pixels long curves. The way they should be connected is far from obvious and may lead to a very high computational complexity; this problem is structurally handled by level lines. Last but not least, Haralick's operator is inefficient for corners and junctions. Indeed, at those points, the gradient direction is very badly estimated and edges may be severely cut. Additional algorithms are necessary to reconnect pieces of edges. On the opposite, level lines bifurcate at junctions, thus handling the different boundaries. Figure 4.8 illustrates this, by showing the meaningful boundaries and Canny's filter output near two junctions.

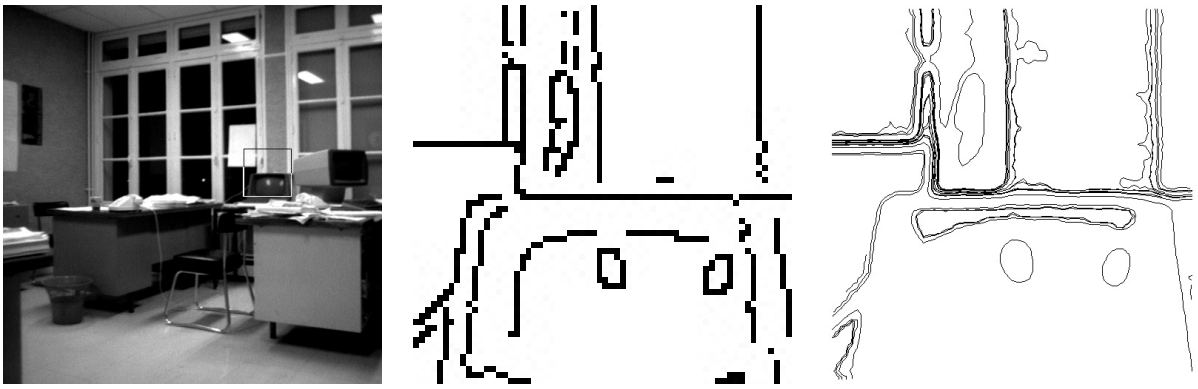


Figure 4.8: *Junction and level lines. On the left, the original image. Middle, Haralick's detector implemented with Canny's filter on the area designated on the left image. Note how the contour is broken at the junction, due to the bad estimate of the gradient direction, and the high number of edge pieces. Right: detailed view of meaningful boundaries on the region. There are two level lines, each corresponding to an edge part.*

4.4 Local boundary detection

In the model presented in the previous sections, the values of the gradient are random variables whose distribution is empirically estimated. It is simply the histogram of the gradient in the image. One can argue that this distribution is too global. This also yields what we call the "blue sky effect". Consider an image containing two parts: a contrasted or textured one (e.g. ground) and a smooth one (e.g. sky). Then, we can observe an overdetection in the ground, and an underdetection in the sky. Indeed, the sky only contributes with small values in the histogram. Thus we tend to detect anything which is more contrasted than the sky, and nearly anything is detected in the ground. On the contrary, the contrasted ground makes the detection more difficult for regions with a small contrast. This is not in agreement with human vision, since we locally adapt our perception of contrast. Objects are masked in contrasted regions, while our accuracy is improved in low contrasted regions (up to some physiological thresholds).
In this section, we address this local adaptivity to contrast. The model, which does not make use of new concepts, is an adaptation of the meaningful boundary model. We first describe the algorithm, then show some experimental results.

4.4.1 Algorithm

Assume that we have detected a closed boundary. Then it divides the image into two connected components: the interior and the exterior of the curve. We can then compute the empirical contrast distribution in the interior on the one hand and in the exterior on the other hand. We then independently detect new meaningful boundaries in each connected component. Then, this procedure is recursively applied. Since the size of the level line tree is finite, it is clear that we end the detection in a finite number of steps.

The situation is actually a bit more complicated. First, this method depends on the order we use

to describe the image boundaries. We simply choose to start with the most meaningful boundaries. Second, boundaries are not always closed. In this case, their endpoints belong to the image border. They still cut the image into two connected components. Unfortunately, there is no clear notion of interior and exterior. A choice is made, but it is purely algorithmic and arbitrary from a perceptual point of view [Mon00]. Thus, we cannot rely on this choice of interior, which conflicts with closed boundaries. However, we can first apply the detection to open boundaries, then to the closed ones. (Open boundaries contain all the closed ones, since level lines are nested.) More precisely, we proceed as follows.

Let us call R_0 the root boundary, that is the (non-meaningful) boundary containing all the image. If C is a boundary, we denote by $\operatorname{Int} C$ its interior.

1. Set $R \leftarrow R_0$. (Local root.)
2. Set $\mathcal{M}$, the set of already stored in R meaningful boundaries. Initially, $\mathcal{M}$ is empty.
3. Let $R' \leftarrow R \setminus \cup_{C \in \mathcal{M}} \operatorname{Int} C$.
4. Compute the histogram of $|Du|$ in R'.
5. Use this histogram and detect the maximal meaningful boundaries included in R'. Let $\mathcal{N}$ be the maximal meaningful boundaries defined by $C \in \mathcal{N}$ if and only if

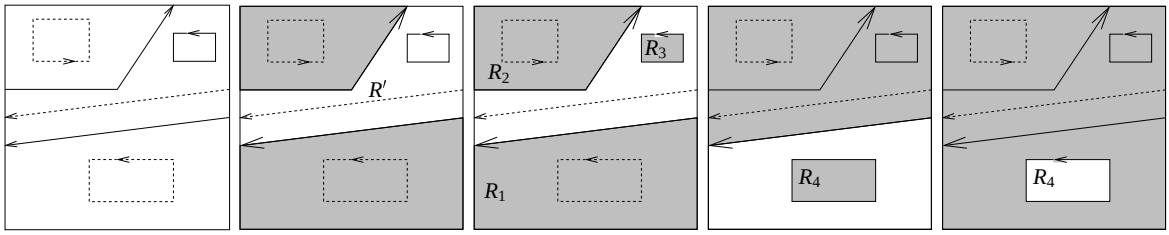
$$\begin{cases} \operatorname{Int}(C') \subsetneq \operatorname{Int}(C) \Rightarrow NFA(C) < NFA(C') \\ \operatorname{Int}(C) \subset \operatorname{Int}(C') \Rightarrow NFA(C) \leqslant NFA(C'). \end{cases} \tag{4.9}$$

   Otherwise said, the boundaries in $\mathcal{N}$ have an optimal NFA. Note that this is stronger than the maximality defined in section 4.2.3 since we go across monotone sections. We call the boundaries in $\mathcal{N}$ the total maximal boundaries. The subtree with root equal to R that remains by keeping only the boundaries in $\mathcal{N}$ has only two levels: the local root R, and $\mathcal{N}$. Since the interior of open boundaries is arbitrary, we do not mix the detection of open and closed boundaries. In practice, this means that if we detect an open meaningful boundary C, we apply the definition of total maximal boundary (4.9) only to open boundaries containing C or contained in C.
6. If $\mathcal{N} \neq \emptyset$, then we have detected new boundaries in the complementary of the already detected ones. Then,
   (a) Set $\mathcal{M} = \mathcal{M} \cup \mathcal{N}$. By construction, all the closed boundaries in $\mathcal{M}$ have disjoint interior.
   (b) return to step 3.
7. If $\mathcal{N} = \emptyset$, there are no new boundaries in the local root and in the complementary of the currently detected boundaries. We then continue the search at lower levels of the tree. For any boundary $C \in \mathcal{M}$,

(a) Store C.

(b) Set $R \leftarrow C$, and $\mathcal{M} \leftarrow \emptyset$.

(c) Return to step 3.



(a) (b) (c) (d) (e)

Figure 4.9: *Example of local search of meaningful boundary. (a) the initial boundaries. They are oriented such that the tangent and the interior normal form a direct frame. We compute the NFA of each boundary. In solid line, we draw the total meaningful ones. Two are open, one is closed. Remark that the interior are disjoint, because of total maximality. While we detect some open curves, we ignore the closed ones. (b) While we detect new meaningful boundaries, we compute the contrast histogram in the complementary of the interior of the open detected boundaries and resume search in this part of the image. In R', the exterior of the detected open boundaries, we detect a total maximal boundary. Remark that this boundary may have been already detected but rejected because of open boundaries. We assume here that no new open boundaries are meaningful. Thus, we keep this closed boundary. (c) We resume the search (with recomputed histogram) in the exterior (white part) of the detected boundaries, until we cannot find new ones. When this is over, we then compute the local contrast histogram in each region R_1, R_2, R_3 and look for boundaries inside them. (d) A boundary R_4 has been detected in R_1. Compute the local histogram in $R_1 \backslash R_4$ and detect boundaries. (e) Finally, we scan for boundaries in R_4 with new local contrast histogram.*

Remark: Each boundary may be tested more than once. Thus, the number of false alarms has to be multiplied by the maximal number of visits of a boundary, which is upper bounded by the level lines tree depth. In fact, each detected boundary often lies in the middle of the local root, and this divides the tree depth by 2. Thus in practice, the maximal number of visits of a boundary is like the logarithm of the initial tree depth. In practice, it is always much smaller than 100.

4.4.2 Experiments on locally contrasted boundaries

In Figure 4.10, we show the difference between the detection with a global contrast histogram and the updated local histogram. To give an idea of the magnitude of the number of false alarms, the boundary delimiting sky and foreground has a NFA equal to 10^{-357}. This means, that, in order to observe such a contrasted line in noise, we need to oberve on the average 10^{357} images. The smaller boundaries around the opening on the top of the tower have NFAs about 10^{-10}.

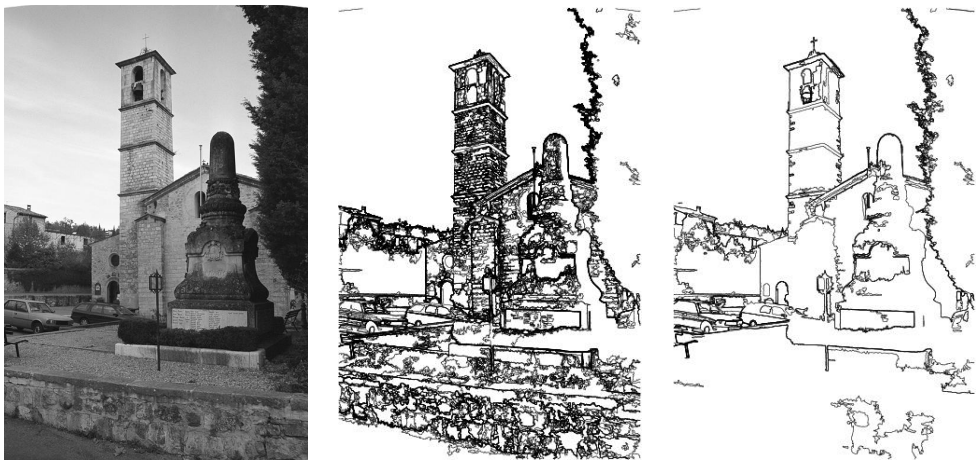


Figure 4.10: *Influence of local contrast. From left to right: original image, maximal meaningful boundaries, local maximal meaningful boundaries. There are 280,000 boundaries in the initial image, 652 in the second one and 193 in the last one. Texture is removed since local contrast (for instance) on the church tower is much more demanding than the global histogram. As the texture is uniform, no level line is a large deviation to the empirical local contrast, yielding no detection. This is very good for shape analysis where we often want to distinguish texture from real shapes.*

Very interestingly, using local contrast removes boundaries in texture. This is logical since the local contrast in textured regions (as on the tower in Figure 4.10) assumes larger values than in the rest of the image. Thus, this decreases the NFA of boundaries and most of them simply disappear in textured regions. This is a masking phenomenon.

Let us explain why this is useful for shape recognition. In general, as we saw in Chapter 2, a shape recognition algorithm can be divided in four steps:

1. extraction of shapes
2. (invariant) encoding
3. comparison: compute some distance between encoded shapes
4. decision: accept or reject pairs of matching shapes

Present and future applications need to compare images in huge databases, where we have no *a priori* that two images, or two shapes should match. Since every procedure in the above methodology is very costly, it is interesting to limit the number of encoded shapes and to try to keep the "most meaningful".

For the time being, there is no general model of shapes [Zhu99]. Nevertheless, we can give empirical observations of what a "good shape" is from a perceptual viewpoint, as we did in Chapter 1. For encoding, a good shape should not be too simple, especially if we are interested in an invariant recognition. For instance, most convex shapes are very alike in affine invariant shape recognition. Assume

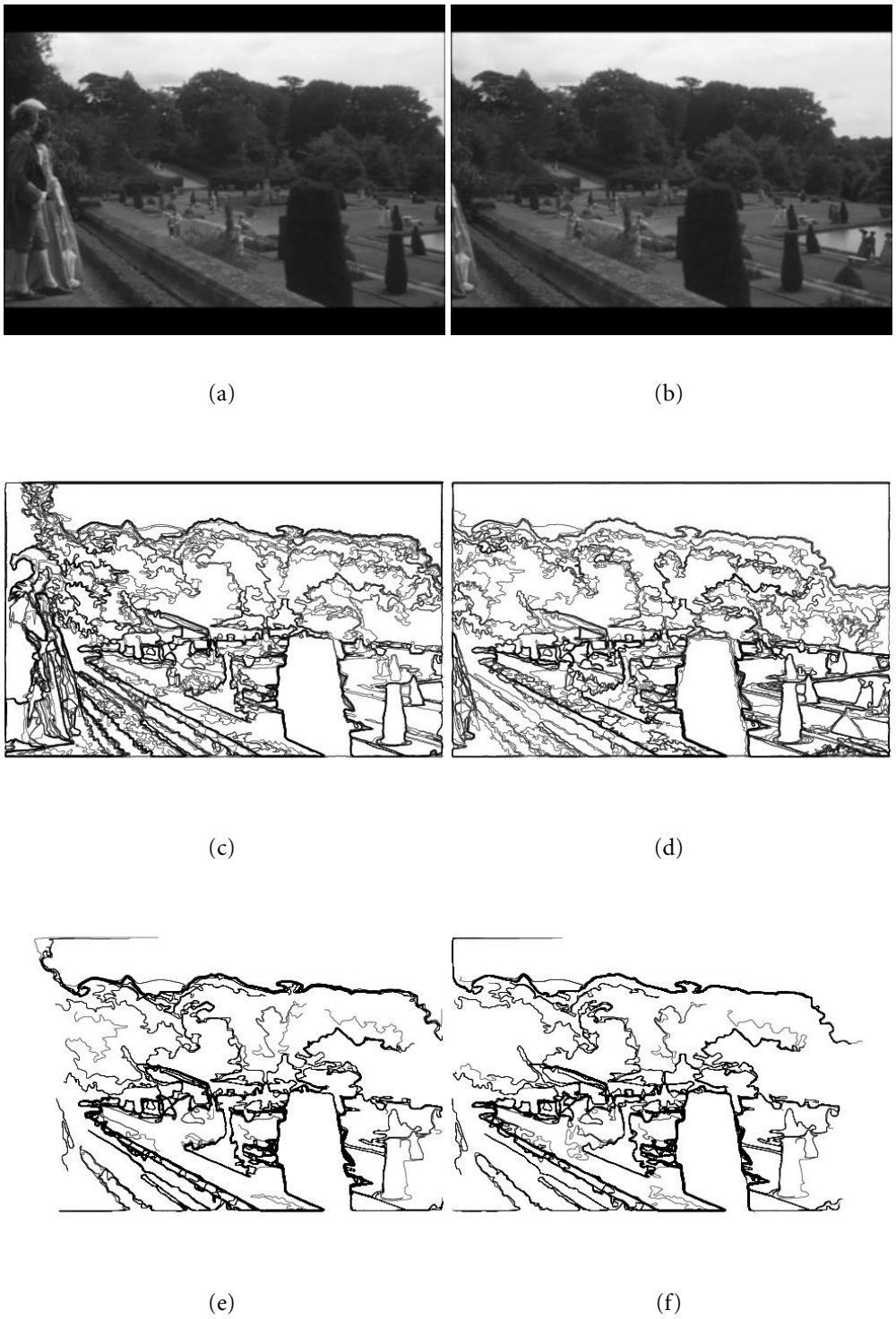
that we have chosen an affine invariant distance between shapes. If we want to be sure that two convex shapes match, the distance between them has to be very small. Indeed, two convex shapes can casually be close to each other, while the probability that it occurs for more complex shapes is very small. On the other hand, a shape should not be too complex, since complexity usually makes the encoding longer and more difficult. Because of occlusions, we usually try to match pieces of shapes. Very complex shapes will be divided in numerous pieces, making computations longer.
Now, it is well known that texture is strongly damaged by compression. Thus level lines in texture may not be reliable when two images come from different sources (with different quality, compression rate, *etc*). Moreover, they are very complex, and yield many encoded pieces of curves. If these curves match for two different images, then those images are certainly the same. Now, the computational cost may be too high for some applications, where we may want to detect a particular shape (a logo for instance) in a database. Thus it may be useful to automatically remove contrasted regions corresponding to texture. This is what the local contrast detection makes in practice.

The argument above is reversed for stereo images registration. In this case, we have the strong *a priori* that the images are close views of the same scene, and the goal is to register them as best as possible. In this application, textures can also give some useful information. (See Figure 4.11).
The effect of local contrast in boundaries detection is twofold: first, textures are eliminated. On the contrary, local contrast should make curves in low contrasted areas more detectable. This is also what we empirically observe: we detect illumination gradient (See Figure 4.5 and 4.12). This can be due to the vicinity of the light source, or to the variation of the orientation of the surface of a three dimensional object with respect to the light source. Such lines do not correspond to the usual notion of shapes (objects). Nevertheless, it is logical to detect them as remarkable structures.

4.5 Meaningful boundaries or snakes?

In [DMM03b], Desolneux, Moisan and Morel compared the MB model with variational snake theory. This may seem a bit weird since the MB model only uses contrast observations along a curve, while snakes are also required to be smooth. In fact, the explanation for natural images is that contrasted boundaries often locally coincide with objects. Thus, they are also incidentally smooth. Whereas smoothness seems to be optional for the detection, it may give a better localization of the contour. In this section, we study the possible influence of smoothness in the detection to see whether or not smoothness is fundamental in the detection. We conclude that, when using smoothness, there are only few additional detections, while the position of the maximal meaningful boundaries may change a little bit. The NFA also significantly decreases. The small number of new detections and the fact that each partial detector can detect most image edges prove *a contrario* that contrast and regularity are not independent in natural images.

An *a contrario* model of regularity has been proposed in [Cao03]. It assumes that the variation of the orientation of the tangent between two samples is a random value uniformly distributed in $(-\pi, \pi)$.



(a) (b)

(c) (d)

(e) (f)

Figure 4.11: *Image registration. (a) and (b) are two images from a movie during a rightward traveling. (c) and (d) are the meaningful boundaries in the previous images. (e) and (f) are the shape elements of (c) and (d) that match with a number of false alarms less than* 10^{-7} *(this result was obtained with the algorithm we propose in Chapter 8, which is based on an* a contrario *definition of shape matching).*

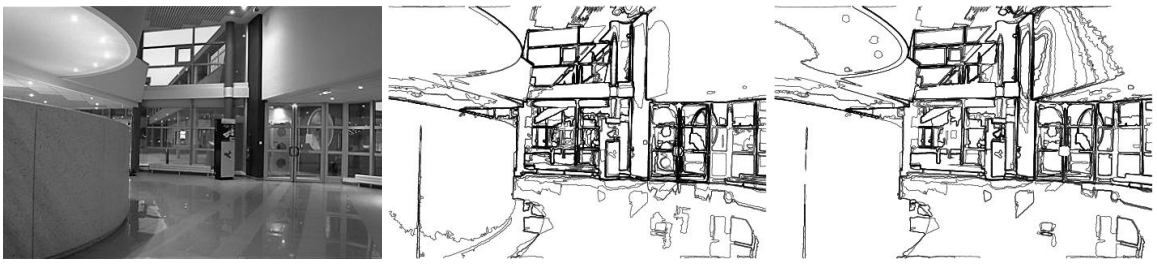


Figure 4.12: *Illumination, local contrast and regularity. Left: original image. Middle: meaningful contrasted boundary. Right: meaningful contrasted and smooth boundary with local contrast. With contrast only, a single boundary appears on the right with the contrast due to illumination. If contrast is localized, then more boundaries are detected. If we also add a regularity constraint (see section 4.5.1 below), there are still more detections. These boundaries are very different from texture since they are nearly convex and parallel. They are eliminated by the cleaning procedure described in section 4.2.4.*

Thus, the implicit a contrario model is random walks with isotropic and independent increments. This model is not really adapted for the following reason. All the curves we detect are level lines, thus boundaries of compact sets. As a consequence, they do not self-intersect. While the local influence is not clearly visible, this implies that long level lines are much more regular than random walks. This logically leads to an overdetection of long level lines because the independence assumption is strongly violated at very long range. The solution we propose is to stick to Helmholtz principle: "no detection in white noise". Thus we have to learn the regularity of level lines in white noise, and use this as the a priori distribution.

4.5.1 Definition of local regularity

Let $l_0 > 0$ be a fixed positive value. Let C be a rectifiable planar curve, parameterized by its length. Let $x = C(s_0) \in C$. With no loss of generality, we assume that $s_0 = 0$.

DEFINITION 4.4 *We call regularity of C at x (at scale l_0) the quantity*

$$R_{l_0}(x) = \frac{\max(|x - C(-l_0)|, |x - C(l_0)|)}{l_0}. \tag{4.10}$$

Of course, this definition really makes sense if the length of C is larger than $2l_0$. This definition of regularity (see Figure 4.13) is related to the Hausdorff dimension of C around x. First, $R_{l_0}(x) \leqslant 1$, with equality if and only if either $C((-l_0, 0))$ or $C((0, l_0))$ is a line segment. On the contrary, if $R_{l_0}(x)$ is small, then the curve is highly curved around x.

We can also interpret $R_{l_0}(x)$ as a function of the local curvature. Indeed, if C is a circle with large enough radius ρ, then

$$R_{l_0}(x) = \operatorname{sinc}\left(\frac{l_0}{2\rho}\right), \text{ where } \operatorname{sinc} x = \frac{\sin x}{x}. \tag{4.11}$$

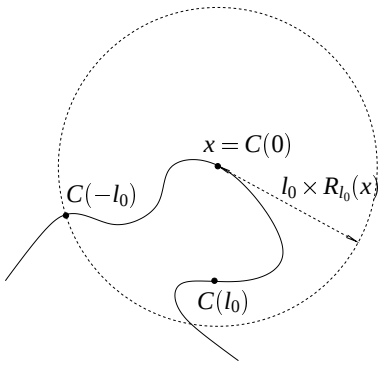


Figure 4.13: *Regularity definition. The regularity at x is obtained by comparing the radius of the circle with l_0. The radius is equal to l_0 if and only if the curve is a straight line. If the curve has a large curvature, the radius will be small compared to l_0.*

This approximation is valid when l_0 is small compared to ρ. In this case, the regularity is a nonincreasing function of the curvature.
This definition is not purely local, but it is also less sensitive to noise compared to differential measures as the curvature. Let

$$\mathcal{H}_{l_0}(r) = P(x \in C,\ C \text{ is a white noise level line and } R_{l_0}(x) > r). \tag{4.12}$$

This distribution only depends on l_0 and can be empirically estimated. Of course, we learn it on level lines whose length is much larger than l_0 in order to avoid quantization effects.

Remark: As expected, the distribution $\mathcal{H}_{l_0}$ is very different in white noise and natural images. In natural images, the histogram of R_{l_0} has a peak at 1, corresponding to real objects boundaries (which often contain alignments). In some textured images, such as paintings, most edges are not real but subjective and this is clearly visible on the histogram of R_{l_0}. See Figure 4.14. The distribution also clearly depends on l_0. When l_0 grows, the histogram mode moves to lower values. However, we obtain the same qualitative behavior as above. In Appendix 4.7, we use these distributions to compute the Hausdorff dimension of white noise level lines. We then quantitatively check that they are much more smooth than (self-intersecting) isotropic random walks.

Again, the choice of l_0 is a natural question. Of course l_0 should be larger than Nyquist distance. It should not be too large either. In experiments we have chosen $l_0 = 10$. But, since NFAs are additive, we may also choose several reasonables values of l_0 (say $l_0 = 5, 10, 20$) and multiply the NFAs by the number of l_0. In practice, changing l_0 influences the number of samples and best NFAs are attained for small l_0.

4.5.2 Meaningful contrasted and smooth boundaries

Now that we have a background model of regularity, we use it to detect regular curves *a contrario*. It is natural to assume, in the background model, that contrast and regularity are independent. Thus

$$P(C \text{ is contrasted and smooth}) = P(C \text{ is smooth}) \times P(C \text{ is contrasted}).$$

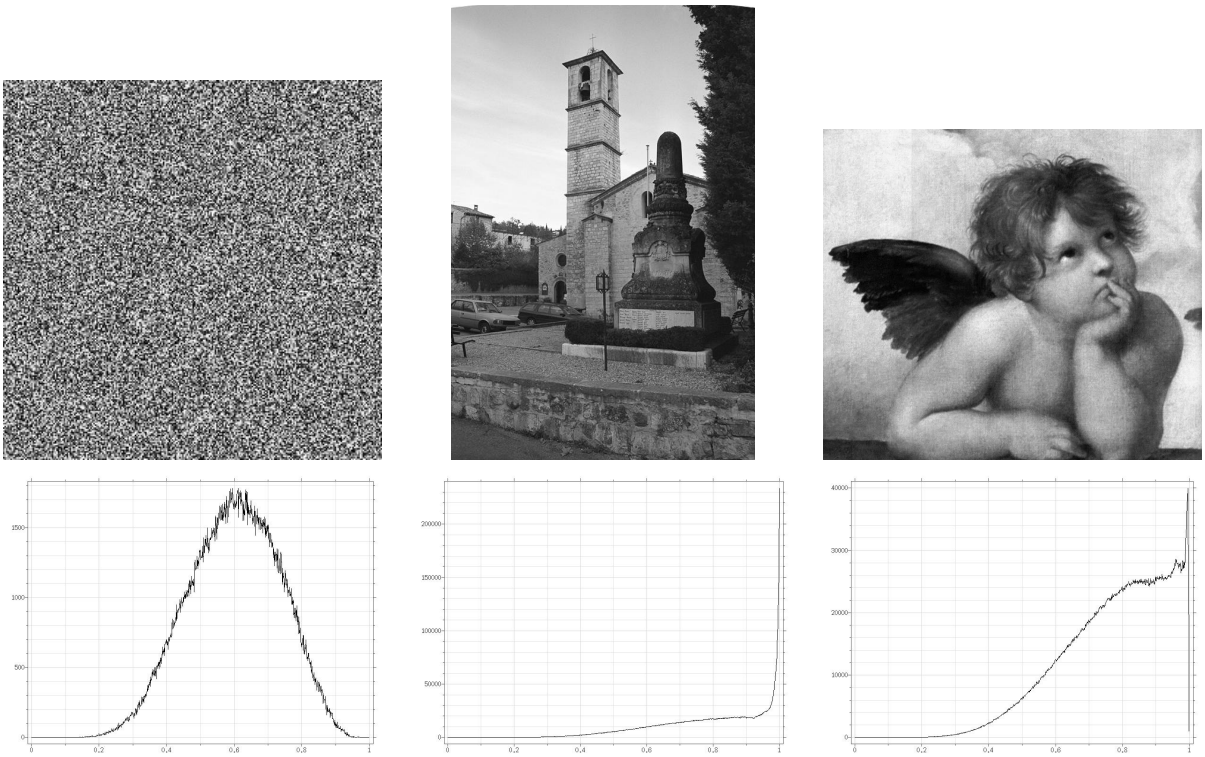


Figure 4.14: *Regularity histograms. Upper row: a white noise image, a scanned photograph and a scanned photograph of a painting. Bottom row: the three regularity histograms for* $l_0 = 10$*. Since its histogram vanishes near* 1*, white noise does not contain any alignments or smooth curves, as foreseen. Nearly all natural images (containing true edges) have a regularity histogram like the second one. The third image contains mostly subjective edges, as it is composed of painted strokes. As a consequence, the regularity histogram is much less concentrated around 1 as for "natural" images. If we now unzoom the three images (with an adequate smoothing before downsampling), then the first histogram remains unchanged (scale invariance), while the other two have regularity histograms like the second one. Indeed, after unzooming, most textures and small scale features disappear, and small gaps get filled.*

Definition 4.5 *Let C be a level line. Let*

$$\nu = \min\{|Du(x)|,\ x \in C\}, \tag{4.13}$$

$$\rho = \min\{|R_l(x)|,\ x \in C\}, \tag{4.14}$$

be respectively the minimal quantized contrast and regularity along C. Let

$$NFA_{cs}(C) = N_{ll} H_c(\nu)^{l/2} \left(\mathcal{H}_{l_0}(\rho)\right)^{l/2l_0}. \tag{4.15}$$

We say that C is a ε-meaningful smooth boundary if $NFA_{cs}(C) < \varepsilon$.

The number of false alarms is the product of the number of level lines and the probability that the contrast and the regularity are simultaneously larger than the observed values along a curve with prescribed length taken in the background model. The probability is computed in the a contrario model where contrast and regularity are independent and local observations are mutually independent. As in section 4.4, this search can be recursively performed by computing local histograms of the gradient.

In experiments, detection results are qualitatively equivalent with or without regularity. On the other hand, NFA may decrease a lot for smooth boundaries. Even though the detection is not changed in one single image, it is still interesting to decrease the NFA as much as possible. Indeed, we may want to detect boundaries not in a single image but in a database (for instance in shape recognition applications). We can consider that any database has a size much less than 10^{15}. Thus, curves with a NFA lower than 10^{-15} in a single image can also be considered as universally meaningful, since they will be detected in any database.

4.5.3 Comparison with active contours

Active contours is one of the most popular techniques of boundary detection. The first works of Kass, Witkin and Terzopoulos [KWT87] have been improved and generalized by many authors. Recent models are more intrinsic, can be expressed implicitly (which ease the possible topological changes of the active contours) and can use image statistics [CKS97, PD02]. In this section, we do not focus on any particular active contour model, but try to compare a generic model with meaningful contrasted and smooth boundaries. Such a comparison has already been made by Desolneux, Moisan and Morel [DMM03b] for meaningful boundaries. Even though these boundaries are only contrast-based, they showed that they are very close to active contours in general and particularly to the model of Kimmel and Bruckstein [KB03]. Since in this chapter we have also introduced a regularity criterion, comparison is even more adequate.

Let us briefly give a generic active contour model: it is a curve that fits shape contours (hence contrast should be large along the contour) and which is also as smooth as possible. The problem usually

assumes a variational formulation. An optimal curve minimizes an energy of the type

$$E(C) = \int_C g(|Du(C(s))| + \lambda h(\mathrm{curv}(C(s)))\, ds, \tag{4.16}$$

where Du is the gradient of a given grey-level image, g is a nonincreasing function, $\mathrm{curv}(C(s))$ is the curvature of C at point $C(s)$, h is a nondecreasing function and s is the arc-length. The optimal curve is a trade-off between the external energy depending on the image gradient, and the internal energy depending only on the curve itself. Such a model can accurately give the position of the contour. However, it has several drawbacks:

- The model assumes that there is a contour: It cannot be used as a detection algorithm. This also explains why active contours are also introduced in Bayesian models, where the real question is: knowing that one object is present, what is the best candidate?
- The initialization is crucial.
- The optimal balance parameter λ (which, for homogeneity reasons, can also be viewed as a scale parameter) is unknown and depends on the image. It has a strong influence on the result.

If we only consider the homogeneity of the different energy terms, we have to minimize a potential of the form $Lg(|Du|) + \lambda Lh(\mathrm{curv}\, C)$, L being the length of the curve.

Let us now consider the meaningful smooth boundary model. A meaningful curve has a small probability to occur in the *a contrario* model. Our regularity measure is a non increasing function of the curvature (see (4.11)). Thus, for a meaningful curve, the quantity

$$(H_c(|Du|)^{L/2}\mathcal{H}_{l_0}(R_{l_0}(C)))^{L/2l_0}$$

is small. Let us now take the logarithm of this expression. We obtain an expression of the type

$$L\left(E_{ext}(|Du|) + E_{int}(\mathrm{curv}\, C)\right),$$

where E_{ext} is a non increasing function of $|Du|$, and E_{int} is a non decreasing function of the curvature. The model is qualitatively alike a snake model. Nevertheless, there are three major differences:

1. There is a quantitative criterion to decide if the curve has to be detected. Contrary to snakes algorithm, meaningful boundaries detection is *not* a minimization algorithm. It is well known in active contours model that the value of the energy of the minimizer has no interpretation. All we can say is that a candidate is better than another one. Our model gives a meaning to the energy-like term. Thus, there is no need for a minimization since we can give thresholds under which a candidate has to be detected.
2. Meaningful boundaries are level lines. Thus, no initialization by hand is needed.
3. We do not have to fix the weight functions g and h as well as the scale parameter λ.

4.5.4 Experiments on smooth meaningful boundaries

In general, adding a regularity criterion does not qualitatively change the result. This is in conformity with the observation of Desolneux et al. in [DMM03b]. Remark also that adding the regularity criterion does not eliminate irregular level lines that were already detected thanks to contrast. Indeed,

$$NFA_{cs}(C) \leqslant N_{ll} H_c(\nu)^{l/2},$$

(with the same notations as in Definition 4.5) since $\mathcal{H}_r(\rho) \leqslant 1$. We can only detect more lines, which is what we want: check whether or not we had misdetections because regularity was not taken into account. Of course, the NFA of smooth boundaries decreases a lot (about 10^{-15}), and this can modify maximal meaningful boundaries. As it was already observed in [Cao03], contrast and regularity are often very redundant, and this explains why the same curves are detected.

Figure 4.15, (INRIA desk) is very geometrical and shows the redundancy between contrast and regularity. Since adding a regularity criterion does not change the results, we could believe that our regularity definition is just wrong and does not bring anything. This is not so. Indeed, we can also define NFA for smooth boundaries, with no care of contrast, as

$$NFA_{reg}(C) = N_{ll} \left(\mathcal{H}_{l_0}(\rho)\right)^{l/2l_0}. \tag{4.17}$$

We retrieve most edges in the desk image with this definition. The conclusions of these experiments are the following: for natural images, there is a strong redundancy between regularity and contrast. Pieces of objects boundaries coincide with pieces of level lines, and they can be detected either by regularity or contrast, or using both criteria.
In Figure 4.16, locally straight structures are also contrasted but the gradient distribution exhibits large values (since the texture variations are important). This explains why contrasted meaningful boundaries lose many lines. In this case, our local regularity criterion allows to characterize this elongated structures.

4.6 Conclusion

In this chapter, we brought a contribution to Desolneux, Moisan and Morel's theory of meaningful boundaries. First, we gave a mathematical interpretation of the model. Basically, it means that a meaningful boundary cannot be generated only by noise. This implies that a meaningful boundary may contain some spurious parts. We proposed an algorithm to remove them. We also proposed a multiscale setting to the theory. As a result, detection is less sensitive to noise, in particular quantization noise. We also presented a method that automatically handle local contrast variations, and do not only use a global measure of contrast. This is very useful for our purpose (shape matching) since it removes texture that usually does not yield stable shape elements. Finally, we discussed the importance of regularity in detection. Our conclusion is that it makes detection more robust, but in natural images, curves that are smooth but not contrasted are empirically quite seldom.

Figure 4.15: *Regularity detectability. Left: original image. In middle, we display the* 204 *detected contrasted smooth boundaries as defined in Definition 4.5. On the right, the* 96 *smooth boundaries, with no contrast information, defined in (4.17). All the main boundaries are already present. Of course, contrast may be the main cause of small NFA, since regularity acts at larger scales. For instance, the window panes have NFA about* 10^{-150} *with contrast and* 10^{-15} *with regularity only (which still make them detectable in any image database). The desk on the bottom right has a NFA equal to* 10^{-60} *with contrast and* 10^{-20} *with regularity, which is already very small.*

Figure 4.16: *Influence of regularity. On the left, the original texture contains a lot of elongated structure. Because the texture shows large contrast variations, meaningfulness is a very strict criterion and contrasted meaningful boundaries miss many details (middle). In this case, local regularity is important and smooth and contrasted boundaries allow to retrieve missing lines.*

Experiments show that this model allows to extract a large number of shape elements from natural images. We cannot pretend to directly extract shapes of images since we believe that many contours are subjective and configurations of the type of Kanizsa's triangle often appear at lower degree. For such contours, all local methods are doomed to fail. However, for practical shape matching by shape elements comparison [LMMM03, MSC+04, MSM03], the MB model with local contrast and cleaning-up automatically eliminates most edges due to texture or small illumination gradient. For our purpose, it is the best compromise between the compactness and the completeness of shape elements dictionaries in natural images.

4.7 Appendix: Numerical estimation of the Hausdorff dimension of a curve

In order to compute the Hausdorff dimension of identically distributed random curves from the histogram of regularity, we proceed as follows. Let C be a curve.

Definition 4.6 *The Hausdorff measure of dimension α is defined by*

$$\lim_{\delta \to 0} \inf_{(B_i)\delta-covering} \sum_i |B_i|^\alpha,$$

where the B_i form a covering of C and $|B_i|$ is the diameter of B_i. The family (B_i) is a δ-covering of C if $C \subset \cup_i B_i$ and for all i, $|B_i| < \delta$.

The problem to estimate this quantity is that it makes no sense to let $\delta \to 0$ for digital curves. Indeed, even for white noise, the precision is bounded from below by Nyquist distance. We assume that the curve is self-similar. This allows to examine it at larger and larger scales, instead of letting δ go to 0. Let us cut a curve with length $L = 2Nl$ in N chunks of length $2l$. We measure the regularity $R_l(i)$ at the middle point x_i of each piece. The balls with radius $R_l l$ nearly form a covering of C. It is not a covering because the endpoint of the curve chunk may not be the most remote point from the center (see (4.10)). Nevertheless, we approximate the measure of C by

$$\mathcal{H}^\alpha(C) \simeq \sum_{i=1}^{N} (2lR_l)^\alpha \simeq 2^{\alpha-1} L l^{\alpha-1} \overline{R_l}^\alpha,$$

where $\overline{R_l}$ is the mean regularity along C. Let us now consider the curve λC with $\lambda > 1$. We can make the same procedure as above with chunks whose length is equal to $2\lambda l$. Thus we evaluate the measure of λC by

$$\mathcal{H}^\alpha(\lambda C) \simeq 2^{\alpha-1} \lambda L (\lambda l)^{\alpha-1} \overline{R_{\lambda l}}^\alpha.$$

But, if we now use pieces of curves of length $2l$, we also obtain

$$\mathcal{H}^\alpha(\lambda C) \simeq 2^{\alpha-1} \lambda L l^{\alpha-1} \overline{R_l}^\alpha.$$

Thus

$$\lambda^{\alpha}\overline{R_{\lambda l}}^{\alpha} = \lambda\overline{R_l}^{\alpha},$$

yielding

$$\log(\overline{R_{\lambda l}}) = \left(\frac{1}{\alpha} - 1\right)\log\lambda + \log\overline{R_l}. \tag{4.18}$$

We can evaluate α by examining the histograms of R_l as a function of l.
For random walks with independent increments, we find $\alpha = 2.02$, whereas the true dimension is 2. For level lines in white noise, we find $\alpha = 1.78$. As expected, the level lines of a white noise image are more regular than random walks.

Chapter 5

Curve smoothing

Abstract: In Chapters 3 and 4, the computation of the tree of bilinear level lines, and the selection of a set of meaningful level lines were addressed. The set of meaningful level lines is a good compromise between the compactness and the completeness of shape information in an image. Moreover, when meaningful level lines are computed from bilinearly interpolated images, they do not suffer from pixelization effect. However, meaningful level lines may be subject to noise, which introduces details that are too much fine in relation to the essential shape information. Hence, a good shape representation asks for a previous smoothing. Since we are interested in affine invariant encoding of pieces of shapes, the smoothing procedure should be invariant to affine transformations. The affine scale space, which is described in this chapter, fits well this requirement. A fast implementation of the affine shortening due to L. Moisan is also presented here.

Résumé : Dans les chapitres 3 et 4 nous nous sommes intéressés au calcul de l'arbre bilinéaire et à l'extraction d'un ensemble de lignes de niveau significatives. Cet ensemble de lignes significatives est un bon compromis entre la concision et la complétude de l'information de forme dans une image. De plus, ces lignes étant extraites des interpolées bilinéaires des images, elles ne présentent presque pas d'effet de pixélisation. Cependant, les lignes de niveau significatives peuvent être affectées par le bruit, qui introduit des détails trop fins par rapport à l'information de forme essentielle. Ainsi, une bonne représentation des formes doit être précédée d'une étape de lissage. Puisque nous sommes intéressés par des codages de formes invariants affine, la procédure de lissage doit être invariante par transformations affines. Le scale-space affine remplit cette condition. Une implémentation rapide due à L. Moisan est également présentée ici.

This chapter does not contain original contributions, but we include it here for the sake of completeness of this dissertation. Sections 5.1 and 5.2 closely follow the article *"On the theory of planar shape"* by Lisani, Moisan, Monasse and Morel [LMMM03], and the book in preparation by Guichard, Morel and Ryan [GMR04].

5.1 Affine invariant mathematical morphology and affine scale space

Shape recognition demands a shape representation that is robust to noise. It is then natural to smooth the shapes before computing the set of features that will be used to represent them. This is particularly true if we want a shape representation that can cope with occlusions, since in that case features have to be local, what makes them all the more sensitive to noise. Consider now the problem of smoothing an unknown shape in order to remove noise (and pixelization effects in digital images), that is, useless information. We immediately realize that we cannot solve this problem by removing all details whose size is below some pre-defined threshold, since we do not know what is noise and what is shape information; the notion of noise, or spurious detail, is relative to the scale at which shapes are observed. Since scale is arbitrary and depends on the observer distance and not on the shapes themselves, shapes (or images, if features are directly extracted from them) should be smoothed at several scales, allowing to remove noise at each scale and hence to extract their main features. This multiscale representation is called *scale space*.

In [AGLM93], Alvarez, Guichard, Lions and Morel proposed an axiomatic development of image smoothing. They first introduce several properties that smoothing operators should exhibit, in order to be consistent with some invariance principles which are inherent to visual recognition. These requirements impose a set of constraints that permits to characterize a class of smoothing operators. Causality (no information is created in the smoothing process), local monotonicity (local inclusion of shapes is preserved, at least for small scales) and the regularity condition that images' second order characteristics locally determine the smoothing process (locality allows to deal with occlusions), characterize scale spaces as well posed, parabolic PDEs. Further requirements of invariance to contrast changes and, and translation-rotation invariance, yield curvature evolution equations. Then they consider three additional constraints: the smoothing operator must be affine invariant (smoothing should not depend on the position of the camera), contrast invariant (shape information in images is invariant to contrast changes) and also reverse contrast invariant (a self-dual operator in the mathematical morphology terminology [Ser82]). This leads to a single PDE, the so-called *affine morphological scale space* (AMSS):

$$\frac{\partial u}{\partial t} = |Du|\mathrm{curv}(u)^{\frac{1}{3}}, \tag{5.1}$$

where Du denotes the gradient of the image, $\mathrm{curv}(u)$ the curvature of the level lines, t denotes the scale parameter and the power $\frac{1}{3}$ is signed, i.e. $s^{\frac{1}{3}} = sign(s)|s|^{\frac{1}{3}}$. This equation is equivalent to the *affine curve shortening* ([ST93]) of all of the level lines of the image, given by the equation

$$\frac{\partial x}{\partial t} = |\mathrm{Curv}(x)|^{\frac{1}{3}}\mathbf{n}, \tag{5.2}$$

where x denotes a point of a level line, $\mathrm{Curv}(x)$ its curvature and $\mathbf{n}$ the signed normal to the curve, always pointing towards the concavity.

The affine invariance property plays a fundamental role in vision, since affine transforms provide fine local approximations to projective transforms, specially for small deformations [Fau93]. However, in order to ensure independence from the observer's viewpoint, smoothing should ideally be

projective invariant, but this is not possible. Indeed, no local and causal scale space can be projective invariant, since asking for affine invariance lets no degree of freedom in the PDE's characterization (see [AGLM93]). Hence, projective invariant local smoothing cannot be achieved unless one of the requirements above is not considered. For instance, the projective evolution of $2D$ curves proposed by Faugeras and Keriven [FK95] does not verifies the maximum principle, that is, the local monotonicity property does not hold; the resulting higher order PDE is unstable and its numerical implementation is still an open problem.

5.1.1 Affine erosions and dilations

Schemes for the affine curve shortening based on affine erosions were introduced by Moisan [Moi98], where it was also shown how to compute exact affine erosion of polygons (the natural digital representation of curves) by concatenating pieces of hyperbolae. A fast algorithm based on affine erosions was first reported in [KM99]. **This subsection presents a different exposition which follows [LMMM03] and the book in preparation by Guichard, Morel and Ryan [GMR04], after original ideas by Moisan [Moi97, Moi98], by briefly recalling some definitions and results**. We refer the reader to [GMR04, LMMM03] for all proofs and for a more detailed exposition. This series of results describe a practical derivation of the affine invariant smoothing, which directly leads to the design of a fast algorithm. Basically, the approach is based on the mathematical morphology formalism and consists in defining an affine distance of a point to a set, which then permits to define affine invariant set erosions and dilations. A result by Guichard and Morel proves that these filters are consistent with Equation (5.1), and yield a natural formal derivation for Moisan's scheme [Moi98].

In what follows, $SL(\mathbb{R}^2)$ denotes the special linear group, defined as the set of 2×2 nonsingular matrices of determinant 1. A set operator T is said to be special affine invariant if $AT = TA$ for every A in $SL(\mathbb{R}^2)$.

"Solid shapes" X are defined, in whole generality, as closed nonempty subsets of $\mathbb{R}^2$. Let $x \in \mathbb{R}^2$ and Δ an arbitrary straight line passing by x. If $x \notin X$, two and only two connected components of $\mathbb{R}^2 \setminus (X \cup \Delta)$ contain x in their boundary. These two sets, denoted by $\mathrm{CA}_1(x, \Delta, X)$, $\mathrm{CA}_2(x, \Delta, X)$ (see Figure 5.1), are called the **chord-arc sets** defined by x, Δ and X, and can be ordered so that $\mathrm{area}(\mathrm{CA}_1(x, \Delta, X)) \leqslant \mathrm{area}(\mathrm{CA}_2(x, \Delta, X))$.

Definition 5.1 *Let X be a "solid shape" and $x \in \mathbb{R}^2, x \notin X$. The* **affine distance of x to X** *is the real number $\delta(x, X) = \inf_\Delta \mathrm{area}(\mathrm{CA}_1(x, \Delta, X))^{1/2}$, $\delta(x, X) = 0$ if $x \in X$.*

Definition 5.2 *The* **affine σ-dilation** *$\tilde{D}_\sigma$ and the* **affine σ-erosion** *$\tilde{E}_\sigma$ are set operators defined on $X \subset \mathbb{R}^2$ by $\tilde{D}_\sigma X = \{x, \delta(x, X) \leqslant \sigma^{1/2}\}$ and $\tilde{E}_\sigma X = \{x, \delta(x, X^c) > \sigma^{1/2}\} = (\tilde{D}_\sigma X^c)^c$.*

Proposition 5.1 *The affine invariant erosions and dilations $\tilde{E}_\sigma$ and $\tilde{D}_\sigma$ are special affine invariant monotone operators.*

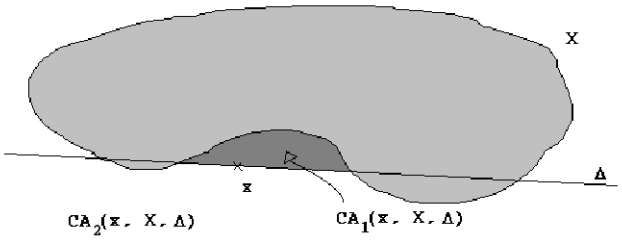


Figure 5.1: *(From [GMR04]) Affine distance.*

It is also clear than the affine invariant erosions and dilations are translation invariant operators. Thus, Matheron theorem (Theorem 7.1 in [GMR04]), which holds for translation invariant monotone operators, can be applied in order to give them a standard form. This leads to the following definition and proposition:

DEFINITION 5.3 *B is an affine structuring element if its interior contains 0, and if there is some $b > 1$ such that for every line Δ containing 0, both connected components of $B \setminus \Delta$ containing 0 in their boundary have an area larger or equal to b (see Figure 5.2). The set of affine structuring elements is denoted by $\mathcal{B}_{\text{aff}}$.*

PROPOSITION 5.2 *For every set X,*

$$\tilde{E}_\sigma X = \bigcup_{B \in \mathcal{B}_{\text{aff}}} \bigcap_{y \in \sigma^{1/2} B} X - y = \{x, \exists B \in \mathcal{B}_{\text{aff}}, x + \sigma^{1/2} B \subset X\}.$$

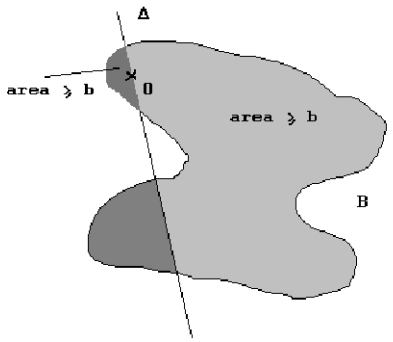


Figure 5.2: *(From [GMR04]) An affine structuring element: all lines passing by 0 divide B into several connected components. All of them which contain 0 in their boundary have area larger or equal to b.*

By Proposition 5.2, x belongs to $\tilde{E}_\sigma X$ if and only if for every straight line Δ, chord-arc sets containing x have an area strictly larger than σ. Conversely, it is also stated that:

Corollary 5.1 *$\tilde{E}_\sigma X$ can be obtained from X by removing, for every straight line Δ, all chord-arc sets contained in X which have an area smaller or equal than σ.*

Definitions 5.2 and 5.3 are equivalent to definition 2 in [Moi98]. In the next subsection we give a result by Guichard and Morel which states the consistency of alternated affine erosions and dilations with the AMSS. Hence, alternating affine erosions and dilations on the set X surrounded by a Jordan curve yields a numerical scheme that computes the affine shortening (5.2) of the curve.

5.1.2 Consistency of the affine erosion/dilation scheme with the affine invariant PDE

The main difficulty for showing the consistency of a fully affine invariant scheme with a PDE lies in its non-locality. Indeed, the set of affine structuring elements contains stretched sets of any size. Guichard and Morel introduce the notion of localizability of structuring elements that follows. This localization is the key point to prove the announced consistency.

Definition and Proposition 5.1 ([GMR04]) *A set of structuring element $\mathcal{B}$ is localizable if it is made of compact connected sets containing 0 and if there exists a constant $c > 0$ such that for every $\rho > c$ we can assert that $\forall B \in \mathcal{B}, \exists B' \in \mathcal{B}, B' \subset D(0,\rho)$ and $B' \subset D_{\frac{c}{\rho}}(B) = \{x, \inf_{y\in B} d(x,y) \leqslant \frac{c}{\rho}\}$, where d denotes the Euclidean distance.*
As a consequence, defining $\mathcal{B}_\sigma = \{\sigma^{1/2}B, B \in \mathcal{B}\}$, one also has :

$$\exists\, c > 0, \forall\, \sigma \leqslant c^{-1}r^2, \forall\, B \in \mathcal{B}_s, \exists\, B' \in \mathcal{B}_\sigma, B' \subset D(0,r) \text{ and } B' \subset D_{\frac{c\sigma}{r}}(B).$$

Proposition 5.3 ([Moi98]) *$\mathcal{B}_{\text{aff}}$ is localizable.*

The localizability of $\mathcal{B}_{\text{aff}}$ proves to be crucial for the following result:

Theorem 5.1 ([GMR04]) *Let $\mathcal{B} = \mathcal{B}_{\text{aff}}$, and set $\mathcal{B}_\sigma = \sigma^{\frac{1}{2}}\mathcal{B}$ its scaled version. Consider the alternate operator $IS_\sigma SI_\sigma$, where for any real valued image $u(x)$ on the plane,*

$$SI_\sigma u(x) = \sup_{B\in\mathcal{B}_\sigma} \inf_{y\in B} u(x+y), \quad IS_\sigma u(x) = \inf_{B\in\mathcal{B}_\sigma} \sup_{y\in B} u(x+y).$$

Then, there exists a constant $c_{\mathcal{B}} > 0$ such that for every C^3 function $u(x)$, one has

$$\lim_{\sigma\to 0} \frac{IS_\sigma SI_\sigma u(x) - u(x)}{\sigma^{\frac{2}{3}}} = c_{\mathcal{B}} |Du| (\text{curv}(u)(x))^{\frac{1}{3}}.$$

This uniform consistency of the alternate operator $IS_\sigma SI_\sigma$ with the affine morphological scale space is a sufficient condition for the convergence of the scheme to the AMSS, thanks to a result by Barles and Souganidis [BS91].

THEOREM 5.2 ([GMR04]) *Let u_0 be a bounded and uniformly continuous function in $\mathbb{R}^2$. Let u_σ be defined by $u_\sigma(x,t) = (IS_\sigma SI_\sigma)^n u_0(x)$ if $n\omega\sigma^{2/3} \leqslant t < (n+1)\omega\sigma^{2/3}$, where $\omega = \frac{1}{2}\left(\frac{3}{2}\right)^{2/3}$. Then, when $\sigma \to 0$, u_σ tends locally uniformly to the unique viscosity solution of*

$$\frac{\partial u}{\partial t} = |Du|(\mathrm{curv}u)^{1/3}.$$

Operators SI_σ and IS_σ correspond respectively to the inf-sup form of affine invariant erosions and dilations (recall that a well known result by Matheron states that every monotone, translation and contrast invariant operator admits an inf-sup form). Thus, applying the alternate scheme $IS_\sigma SI_\sigma$ to u is equivalent to apply alternate affine set erosions and dilations to each level set of u (this holds because these operators are monotonous and continuous). Level lines move following Sapiro and Tannenbaum affine curve shortening. Consequently, the infinitesimal iteration of affine set erosions/dilations (defined in subsection 5.1.1) on a solid shape X asymptotically yield the affine curve shortening of its boundary.

The leading ideas behind **Moisan's scheme for the affine curve shortening** [Moi98] are given by Corollary 5.1 and Theorem 5.2. **The next section summarizes the main steps of this algorithm, following the description presented in [LMMM03], where a detailed exposition of these steps can be found.**

5.2 A fast invariant curve affine erosion-dilation scheme

Let c_0 be a Jordan curve, which is the boundary of a simply connected set X. As we saw in the previous section, iterating affine erosions and dilations on X gives a numerical scheme that computes the affine shortening c_T of c_0 at scale T. If c_t is the curve represented by the function $s \mapsto \mathbf{C}(s,t)$, then

$$\frac{\partial \mathbf{C}}{\partial t}(s,t) = |\mathrm{Curv}(s,t)|^{\frac{1}{3}}\mathbf{n}(s,t), \tag{5.3}$$

where $\mathrm{Curv}(s,t)$ and $\mathbf{n}(s,t)$ are the curvature and the normal vector at the point with curvilinear abscissa s of the curve $c_t = \mathbf{C}(s,t)$.

A fast algorithm

In general, curves are numerically represented as polygons. Assuming then that c_0 is a polygon, it can be shown that the exact affine erosion of X is made of straight segments and pieces of hyperbolae [Moi98]. However, such a precision is not really needed; numerically, a good approximation by a new polygon is enough. Now the point is that the combination of an affine erosion plus an affine dilation of X can be approximated by computing the affine erosion of each *convex component* of c_0, provided that the erosion/dilation area is small enough. This leads to a fast algorithm proposed by Koepfler and Moisan [KM99], since if X is convex, then it has been shown in [Moi98] that its affine erosion can be exactly computed in linear time.

The algorithm consists in the iteration of a four-steps process:

1. **Break the curve into convex components.** This operation permits to reduce the problem to the computation of affine erosion of convex pieces of curves. This is much faster (the complexity is linear) and can be done simply in a discrete way. Inflexion points are computed as the points in the discrete curve where the sign of the determinant $[P_{i-1}P_i, P_iP_{i+1}]$ given by consecutive points P_{i-1}, P_i and P_{i+1} changes. In order to avoid spurious (small and almost straight) convex components due to numerical artifacts, the computer's finite precision is taken into account in this computation.

2. **Sample each component.** At this stage, in order to ensure the stability of the scheme, points are removed or added in the curve, so that a certain regularity of the curves' discretization step is maintained. The criteria is to have the Euclidean distance between two consecutive points between ε and 2ε (ε being the absolute space precision parameter of the algorithm).

3. **Apply discrete affine erosion to each component.** This step, which is the central core of the algorithm, is detailed below.

4. **Concatenate the pieces of curves obtained at step 3.** The result is then a new closed curve on which the whole process can be applied again.

5.2.1 Discrete affine erosion of convex components

The affine erosion computation based on the breaking of the initial curve into convex components, may give bad estimates of the continuous affine erosion+dilation when the area of one component is less than the erosion parameter. That is why the approximation of the affine erosion of scale σ of the whole curve is not performed in a single step, but as a series of affine erosions of effective area $\sigma_e < \sigma$. This area is computed as follows:

- Compute the area A_j of each convex component $\mathcal{C}^j = P_0^j P_1^j ... P_{n-1}^j$, given by

$$A_j = \frac{1}{2} \sum_{i=1}^{n-2} \left[P_0^j P_i^j, P_0^j P_{i+1}^j \right].$$

- Define the effective area $\sigma_e = \max\left\{ \frac{\sigma}{8}, \min_j A_j \right\}$. The choice of $\sigma/8$ is rather arbitrary and guarantees an upper bound to the number of iterations required to achieve the final scale.

The discrete erosion at scale σ_e of each component is defined as the succession of each middle point of each segment $[AB]$ such that

1. A and B lie on the polygonal curve,
2. A or B is a vertex of the polygonal curve,
3. the area enclosed by $[AB]$ and the polygonal curve is equal to σ_e.

The validity of this strategy is a consequence of the middle point property stated in [Moi98]. This property basically asserts that the boundary of a σ-eroded convex set is included in the envelope of σ-chords, which is given by the middle points of these chords.

5.2.2 Iteration of the process

The numerical iteration scheme follows directly from Theorem 5.2. The affine curve shortening of a Jordan curve c_0 at scale T can be computed as $\left(\tilde{D}_{\left(\frac{T}{n\omega}\right)^{3/2}} \circ \tilde{E}_{\left(\frac{T}{n\omega}\right)^{3/2}}\right)^n (c_0)$, with n large enough. In the fast algorithm described here, curves are broken into their convex components. Hence, in one iteration step only an affine erosion of area $\left(\frac{T}{n\omega}\right)^{3/2}$ has to be applied to each of these convex sets, since for any convex set X, $\tilde{D}_\sigma X \equiv X$ (indeed, the affine distance $\delta(x, X)$ is equal to 0 if $x \in X$ by definition, and its value is $+\infty$ if $x \notin X$ for convex sets); each iteration is then followed by a concatenation of the affine eroded convex components.

5.2.3 Comments

The algorithm involves the following parameters:

- T, the scale to which the input curve must be smoothed;
- ε_r, the relative spatial precision at which the curve must be numerically represented;
- n, the minimum number of iterations required to compute the affine shortening (it seems that $n \simeq 5$ is a good choice). From n, the erosion area σ used in step 3 is computed with the formula $\sigma^{2/3} = \frac{T}{n\omega}$.

  Notice that thanks to the $\sigma/8$ lower bound for σ_e, the effective number of iterations cannot exceed $4n$.

The algorithm complexity is linear in time and in memory. No derivation or curvature computation is necessary, since each new curve is obtained as the set of the middle points of some particular chords of the initial curve, defined themselves by an integration process (an area computation). This ensures the stability of the algorithm.

5.3 Illustration

Figure 5.3 shows that a slight smoothing eliminates the quantization effects. These effects are only due to image representation, and provide therefore no useful information for shape recognition. In a certain manner, discarding it permits to focus on the discriminatory information. Numerically, we set $T = 0.5$ in order to eliminate details of size 1 pixel.

Let us now illustrate the (special) affine invariance property of the affine curve shortening. Figure 5.4 shows two curves c_0 and c_0'; c_0' is the image of c_0 by an affine transform A ($\det A = 1$). Both curves are smoothed using Moisan's geometrical implementation of the affine curve shortening, at scales T

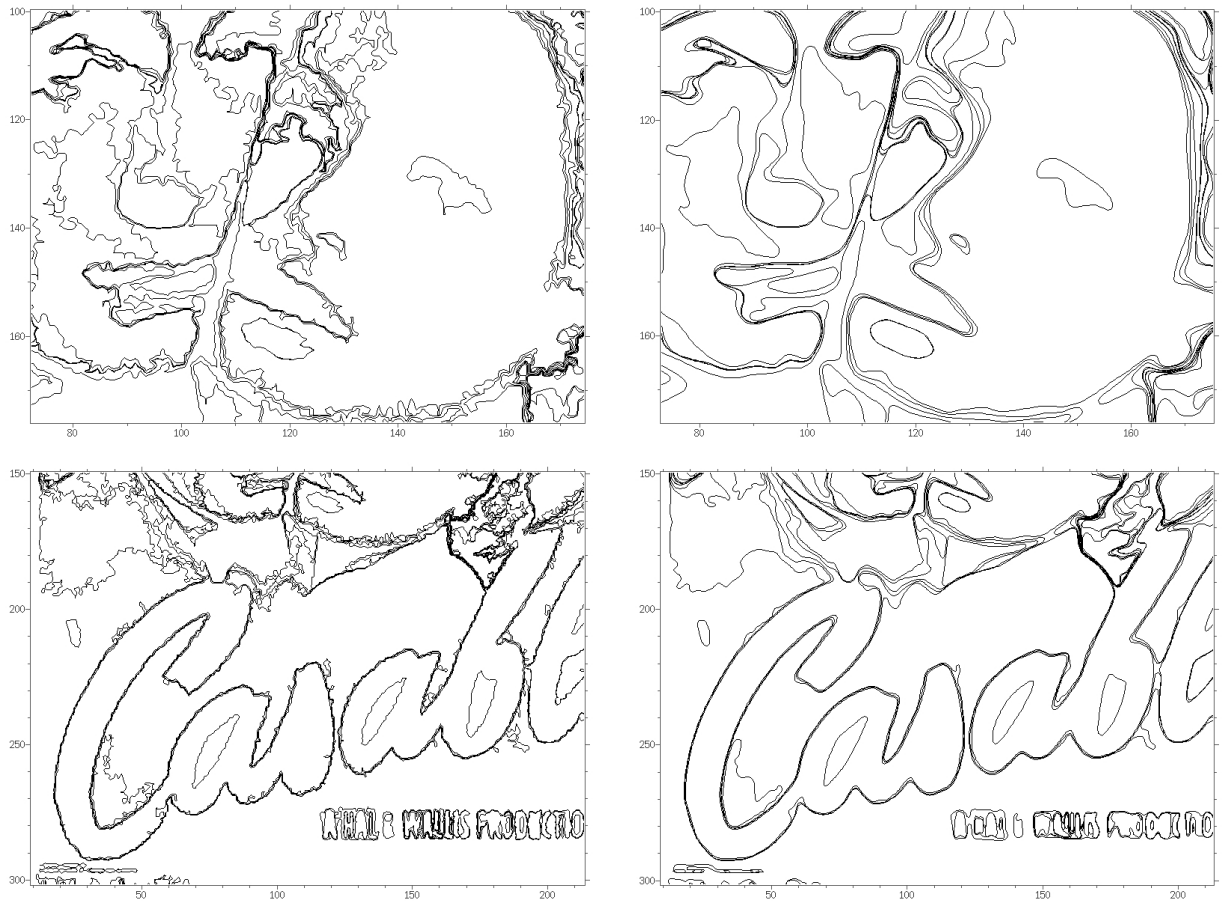


Figure 5.3: *A close-up on meaningful level lines (on the left). Quantization effects can be seen. After a slight smoothing, these effects disappear (on the right).*

equal to 1.0, 2.0, 3.0 and 4.0. Figure 5.5 shows, for each of the scales T, the superimposition of c'_T and the image of c_T by A. Visually, the superimposition appears to be very accurate. The results illustrate then the consistency of the fast affine erosion-dilation scheme with the affine invariant PDE.

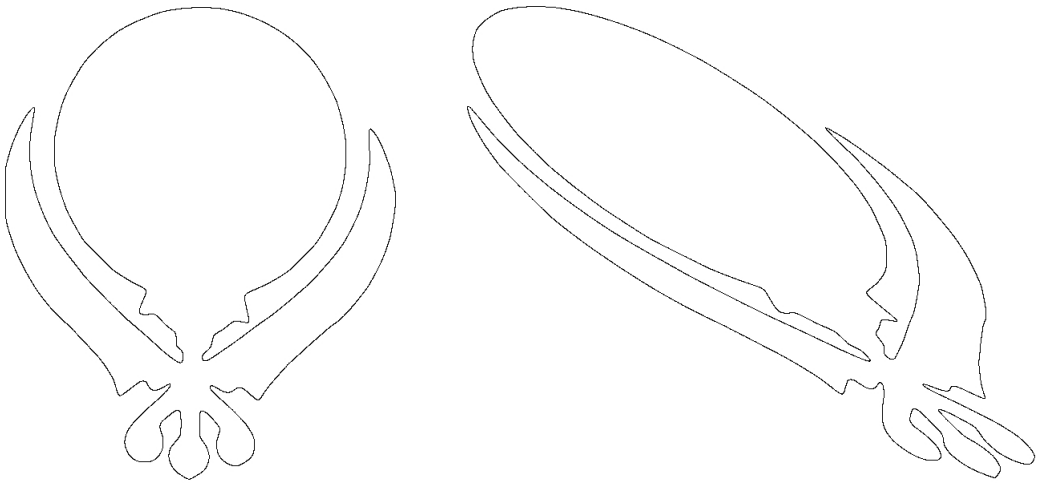


Figure 5.4: *Left: curve c_0. Right: curve c'_0, which is the image of c_0 by an affine transform A.*

In Figure 5.6 we present the same experiment, but using an implementation of the mean curvature motion

$$\frac{\partial \mathbf{C}}{\partial t}(s,t) = |\mathrm{Curv}(s,t)|\mathbf{n}(s,t), \tag{5.4}$$

instead of the affine curve shortening. Like the affine erosion, the implementation used here is based on area computation. Smoothing is also performed at scales T equal to 1.0, 2.0, 3.0 and 4.0, Differences start to be noticeable at $T = 1.0$, in the curvature extrema, but they only become significant at larger scales. Hence, when performing slight smoothing, as we do in our shape recognition framework, using the mean curvature motion would led to similar results. However, in general, if we want to be consistent with the affine invariance requirement for shape recognition (stated in Chapter 1), the geometric affine scale space described in this chapter is called for. This is all the more true when two shapes which are observed at very different scales are to be compared (notice that in that case, a multiscale approach has to be considered: the two shapes would have to be smoothed at different scales, given by the zoom factor between them; since this zoom factor is unknown, this means that we should smooth both shapes at several scales and encode all of them).

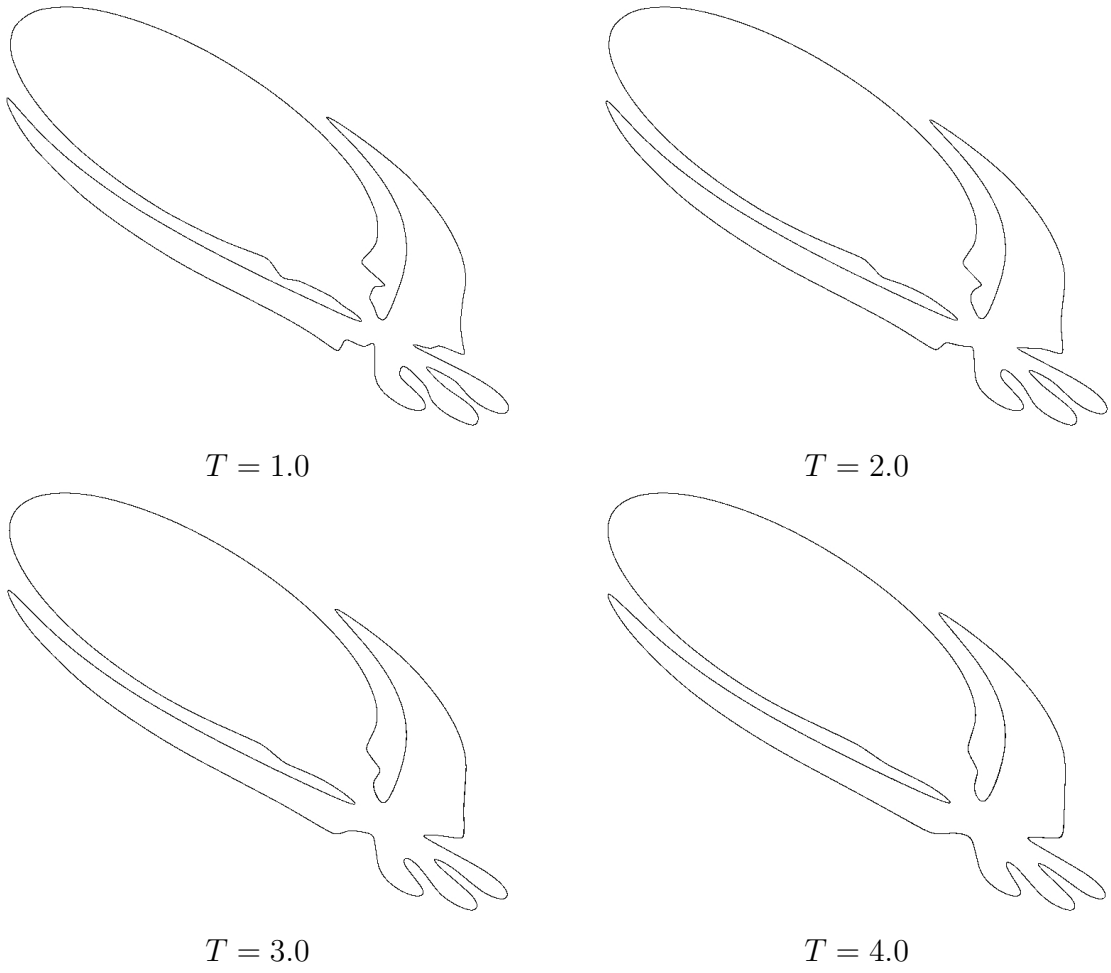


Figure 5.5: *Curves c_0 and c'_0 in Figure 5.4 are smoothed with the affine curve shortening PDE, at scales $T = 1.0, 2.0, 3.0$ and 4.0. Then, for each T in $\{1.0, 2.0, 3.0, 4.0\}$, c_T is transformed by A and superimposed to c'_T. The curves superimposed are very close, and differences cannot be perceived.*

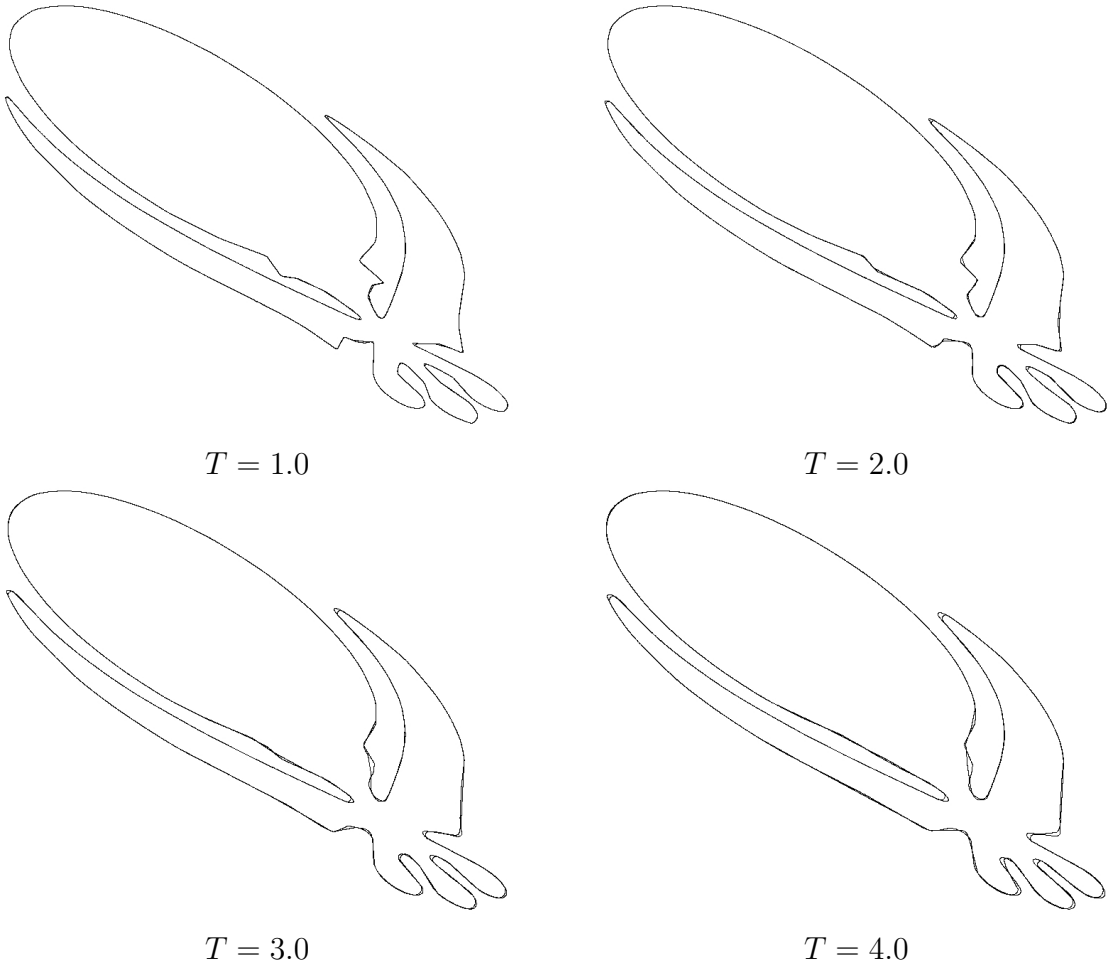


Figure 5.6: *Curves c_0 and c_0' in Figure 5.4 are smoothed with the mean curvature motion, at scales $T = 1.0, 2.0, 3.0$ and 4.0. Then, for each T in $\{1.0, 2.0, 3.0, 4.0\}$, c_T is transformed by A and superimposed to c_T'. Differences start to be noticeable at $T = 1.0$, in the curvature extrema, but they only become significant at larger scales.*

Chapter 6

Flat pieces on curves

Abstract: Since the seminal work of Lamdan *et al.* [LSW88], bitangent lines are well-known to be of high interest to build up semi-local invariant curve descriptions. Nevertheless, these bitangent lines do not enable to encode convex curves (which show none). Moreover, some non-convex curves may show some small oscillations over a straight portion on which numerous bitangent lines are detected, leading to unstable and unreliable invariant descriptors. Another robust direction is given by flat pieces on curves (*i.e.* a piece of curve on which the direction of the tangent lines does not vary too much). For instance, some convex curves such as polygons show many of them. In this chapter, we propose a definition of flat pieces based on an *a contrario* model, that, coupled with bitangent lines, enable to encode nearly all kinds of curves.

Résumé : Les bitangentes sont connues depuis les travaux de Lamdan *et al.* [LSW88] pour leur intérêt dans la définition de descripteurs invariants semi-locaux. Néanmoins, ces bitangentes ne permettent pas de coder les courbes convexes (qui n'en présentent aucune). De plus, des courbes non-convexes peuvent présenter une partie correspondant à des petites oscillations autour d'une droite sur lesquelles de nombreuses bitangentes sont détectées ; les descripteurs invariants qui correspondent sont alors instables et non fiables. Dans ce chapitre, nous proposons une définition des morceaux plats basées sur un modèle *a contrario* qui, couplés aux bitangentes, permettent de coder quasiment tout type de courbe.

6.1 Segment detection in images

Segment or straight line detection is one of the cornerstones of computer vision. Indeed, it is often a preprocessing step of shape recognition, shape tracking [DF90], vanishing point detection [ADV03], convex shape detection [Jac96], *etc.* Most of the time, straight lines in images are conceived as contiguous edges. Many line detection algorithms therefore require a previous local edge extraction step, such as a Canny's filtering [Can86]. Hough Transform [Hou62] and algorithms derived from it [IK88] have been widely studied for that purpose. The goal of these methods is to identify clusters in a particular space (the parameter space of a line, either (ρ, θ) with ρ the distance of the line to the origin, and θ the angle between a vector normal to the line and a fixed direction, or (a, b) where a

is the slope and b the ordinate of the intersection between the straight line and the ordinate axis). Hough Transform method consists in a voting procedure: every pixel votes for the parameters of the line going through it. Another method consists in chaining first the local edges by taking into account connectivity (see for an example [Gir87]), and then in identifying segments among the discrete curves [LB93]. The main drawbacks of these methods are their number of thresholds (edge detection needs at least a gradient threshold, Hough Transform needs a quantization step for the parameter space discretization and a threshold for the voting procedure) and their computational heaviness and instability (due to local edges chaining). They are moreover very sensitive to noise and to the lack of accuracy of edge detectors: they indeed aim at detecting exact discrete straight line, in the sense that no outlier edgel is allowed. A concept of fuzzy segments has been proposed [DRRRD03], but the primary detection is still based on a set of points derived from a local edge detector.

On the other hand, Desolneux *et al.* [DMM00] proposed a parameterless method that detects meaningful alignments in images. A meaningful alignment is conceived as a segment where a certain proportion of points have their gradient orthogonal to the same line, up to a given precision. Let us recall the exact definition of a meaningful alignment. A length l segment is ε-meaningful in a $N \times N$ image if it contains at least $k(l)$ points having their direction aligned with the one of the segment, where:

- $k(l)$ is given by: $k(l) = \min\{k \in \mathbb{N}, P(S_l \geqslant k) \leqslant \varepsilon/N^4\}$, and
- $P(S(l) \geqslant k)$ is the probability that, in at least k points in a straight segment of length l, the gradient of the image is orthogonal to the segment, up to a given precision.

The main drawback of this method for segment detection is that it highlights directions and not segments: while the detected straight lines may correspond to the direction of several disjoint segments, gradient direction is allowed to differ between them from the line direction.

In his PhD thesis [Lis01], Lisani defines "flat points" on curves by using two arbitrary parameters. A "flat point" is the center of a curve segment for which the sum of the angle variations of tangents is small enough (less than 0.2 radian) over a large enough piece of curve (greater than 15 pixels). This algorithm misses many flat points, and does not really detect segments, as we show in Section 6.3.3.

None of the methods we have described is fully satisfying for shape recognition purposes. This chapter presents a new flat pieces detector based on bilinear level lines, since they allow accurate (local) segment detection while avoiding chaining problems intrinsic to edge detection. As we will see, the proposed algorithm extracts accurate straight pieces of level lines. It involves a single parameter, which can be set once for all as discussed in the following section.

Figure 6.1 shows the results for some of the algorithms which we just discussed. As far as flat pieces detection is concerned, Desolneux's alignments are suitable neither for detecting accurate segment directions nor for detecting segment lengths. The naive segment detector based on Hough transform which illustrates the discussion is certainly not the best that can be done using Hough techniques. Nevertheless, even a more clever algorithm would face the same problem as this one: it involves numerous critical parameters (different parameters would drastically change the results). Some isolated

points are detected as segments because they fall "by chance" on the same straight line as another more distant segment and therefore collect its votes. Both algorithms (alignments and the Hough transform-based algorithm) are not local enough: that is why segments over the characters in the test image are not detected. Canny's edge detector is well known to suffer from lack of accuracy at edge junctions (where the gradient is badly estimated). Here it is not a real issue since segment lines are searched between junctions, where edges are more accurately detected. Nevertheless, those edge detectors need several critical thresholds.

6.2 Flat pieces detection

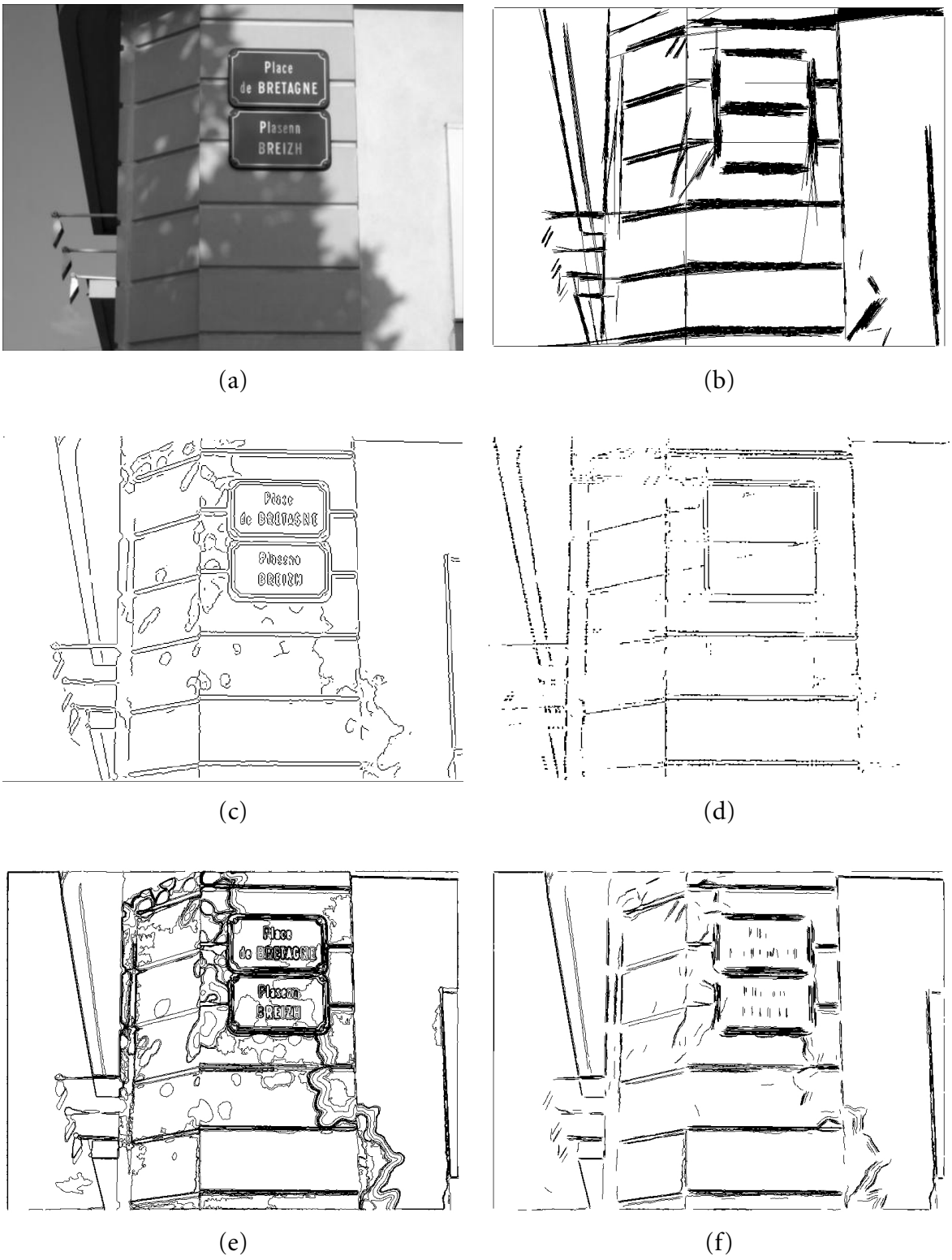
In their founding paper [FB86], M.A. Fischler and R.C. Bowles argue that any curve partitioning technique must satisfy two general principles: stability of the description, and a complete, concise and complexity limited explanation. Smooth sections of curves appear thus to play a major role, because they fit both principles. In other words, Guy and Medioni [GM96] consider segment lines as *salient* features in images.

The concept of "flatness" of a piece of curve is measured in what follows by how much the curve turns on this piece with regard to the direction given by the underlying chord (see figure 6.2). Since we would like to detect flat pieces of arbitrary length, we test chords of various lengths: the proposed algorithm can thus be seen as a multi-scale process. This point of view is not new and has been used for the more general problem of polygonal approximation of digitized curves (see for example [SG80]).

In our opinion, flat pieces detection on a curve should meet the following requirements:

- It should detect not only points around which the curve is flat, but precise pieces on which the curve is pretty straight.
- Long flat pieces should be allowed to move more far away from their underlying chord than short ones.
- The detection should be intrinsic to the curve, and not depend on other curves in the image.
- Detected flat pieces should not overlap.
- Since flat pieces detection is generally the first step of a more general algorithm, it deals with a huge amount of information. Therefore, computational complexity is crucial.

Several methods based on these requirements have been tested. For example, we have tested the method described in Chapter 8, by computing the number of false alarms of a match between a piece of curve and the underlying chord. Results do not appear to be convenient: the NFA of small pieces is too low with regard to overlapping longer pieces, leading to an oversegmentation of straight lines. Moreover, the NFA computation for any "candidate" involves the entire image (through the number of tests and the estimated probabilities), making the detection not intrinsic to the curve but to the



(a) (b)

(c) (d)

(e) (f)

Figure 6.1: *Segment detection. (a) original image; (b) maximal meaningful alignments; (c) Canny's edge detector; (d) Points that correspond to an edge and that lie at the same time on a direction detected by voting in the Hough space; (e) local maximal meaningful level lines; (f) result of the proposed algorithm. See text for discussion.*

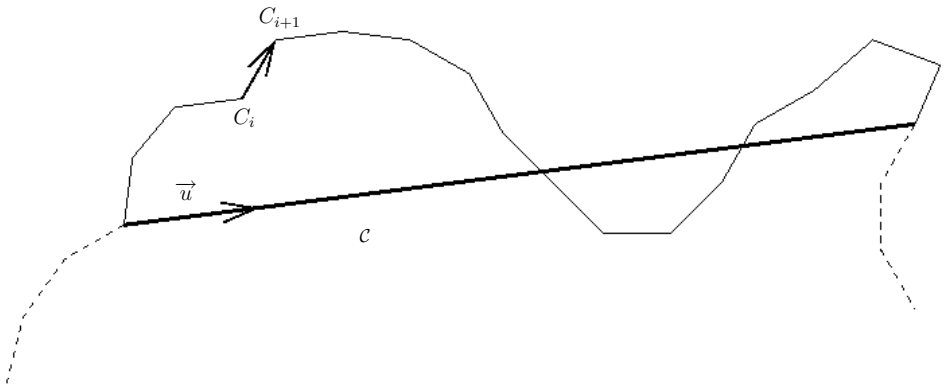


Figure 6.2: *A piece of discrete curve with the underlying chord $\mathcal{C}$ (thick segment line).*

image. Algorithms derived from Desolneux *et al.*'s meaningful alignments were also investigated, and results were unsuccessful. Indeed, estimating the probability that k points among l have a tangent with the same direction as the chord is not relevant to detect flat pieces. In such a model, consecutive alignments are indeed not favored, but here we are particularly interested in them. All these considerations led us to the algorithm proposed in the following section.

6.2.1 Flat pieces detection algorithm

Let us consider a chord from a given curve: its endpoints delimitate a piece of curve of length l (measured in pixels). Since we would like to measure how much the piece turns with respect to the direction $\overrightarrow{u}$ given by the chord, we define:

$$\alpha = \max_{i \in \{1 \dots n-1\}} \left\{ \left| \text{angle}(\overrightarrow{C_i C_{i+1}}, \overrightarrow{u}) \right| \right\},$$

where the discrete piece of curve is made of the n points C_i (there is no direct link between l and n since the discretization step of the curve is unspecified).

Let us suppose that α is below some fixed threshold α^*. Following Desolneux *et al.* [DMM01], we consider that points at a geodesic distance (along the curve) larger than 2 are statistically independent. Thus, we consider $l/2$ statistically independent points along the curve. The probability of the event "$l/2$ statistically independent points on a piece of curve show a tangent line which makes an angle lower than α among all the pieces of curve for which $\alpha < \alpha^*$" is:

$$p(\alpha, l) = \left(\frac{\alpha}{\alpha^*} \right)^{l/2}.$$

Of course, the lower is $p(\alpha, l)$, the more the piece of code is straight.

This straightforward computation is valid under the assumption that among all the pieces of curves such that $\alpha < \alpha^*$, α is uniformly distributed over $[0, \alpha^*]$. This assumption is true under the more general hypothesis that all angles α are uniformly distributed over $[0, \pi]$. Such a model is not completely accurate, since we do not consider random walks but level lines which are constrained not to

self-intersect. However, it is accurate enough to be an *a contrario* model against which an hypothesis is tested ("is this piece of curve likely to be a flat piece?"). This last sentence may sound a bit weird since up to now "flat pieces" have not been precisely defined. In fact, flat pieces are defined as rare events with regard to this model for the distribution of α, with the independence assumption.

For each piece of the curve, the corresponding α is estimated. If it satisfies $\alpha < \alpha^*$, the probability $p(\alpha, l)$ is computed. Only pieces for which $p(\alpha, l)$ is under a predetermined threshold p^* are kept (these pieces are called "candidates"). Since there is no reason that such pieces do not overlap, a selection has to be made among them in order to define the flat pieces of the curves. A greedy algorithm is used: the piece of curve with the lowest p is marked as a "flat piece" , then all candidates that share a common part with this "best" flat piece are eliminated, and the process is iterated with the remaining candidates.

The whole algorithm involves two thresholds. The first one, α^*, is not critical. Indeed, since we are interested in detecting flat pieces, it is natural to *a priori* reject all pieces of curve for which α is upon a large threshold. We choose $\alpha^* = 1$ radian once for all. The second threshold, p^* is not critical either; it is just introduced in order to make the greedy algorithm faster, since it enables to drastically reduce the number of candidates. The choice of p^* is discussed in Section 6.2.2. Increasing α^* and p^* makes the number of candidates greater by accepting less straight pieces of curve. Nevertheless, most of the time the greedy algorithm will eliminate these candidates for the benefit of lower probability candidates which are included in them.

The computation of α clearly depends on the discretization. The curves which the proposed algorithm deals with are level lines of images. Their "natural" discretization is one pixel, that appears to be accurate enough in order to compute α.

Thus, the proposed method involves several parameters (the thresholds α^* and p^*, the discretization step of the curves), but they are set once for all, and do not need to be tuned by the user for each experiment.

The flat pieces detection algorithm is summarized in what follows.

Let us consider a Jordan curve on which flat pieces are searched.
Part I: candidate identification.
For each chord of the curve:

1. Compute the maximum angle α between the chord and the piece of curve delimited by both ends of the chord. If n denotes the number of points C_i on this piece of discrete curve:
$$\alpha = \max_{i \in \{1 \dots n-1\}} \left\{ \left| \text{angle}(\overrightarrow{C_i C_{i+1}}, \overrightarrow{u}) \right| \right\}.$$
2. If $\alpha > 1$ rad, *a priori* reject the piece; otherwise compute $p(\alpha, l) = \left(\frac{\alpha}{\alpha^*}\right)^{l/2} = \alpha^{l/2}$, where l is the length of the considered piece of curve.
3. If $p(\alpha, l) > p^*$, reject the piece.

Part II: greedy algorithm

1. Keep the candidate for which α^n is minimal, mark it as *flat piece*, and discard it from the list of candidates.
2. Reject all candidates that meet this "best" candidate.
3. Iterate until no candidate is available anymore.

More precisely speaking, "all" chords are not tested, but a subsample of them, made of chords of length $10, 20, 30, \ldots, 180, 200$, and then an exponential progression of the tested lengths, so that the algorithm does not waste too much time for long curves. The only consequence of this discretization procedure is that long straight lines (in practice, lines whose length is larger than 100 pixels) could be split into two pieces (see Figure 6.15 for an example).

6.2.2 Probability threshold

As we said earlier, because of the greedy algorithm, there is no need for a very accurate choice of p^*. In most experiments, we set $p^* = 10^{-3}$, or less if we are interested in very accurate flat pieces. Experimental evidence shows that $p^* = 10^{-3}$ is the maximum value for which no detection can be seen in level lines extracted from a white noise image, containing the same amount of level lines than a standard natural image. In this sense, the proposed algorithm satisfies the Helmholtz principle: no detection is found in noise images. Again, what is important here is that the probability threshold is set once for all experiments, and has little influence on the final result.

6.2.3 Some properties of the detected flat pieces

It is true that the condition defining the candidates ($\alpha^{l/2} < p^*$) is not a real constraint for long curves. For example, if $p^* = 10^{-3}$ and $l = 200$, all curve pieces such that $\alpha < 0.97$ are accepted as candidates. Nevertheless, long pieces of curves often show "flatter" subpieces with a lower probability. A case in which this is certainly not true is the case of circles. Let us examine this further, and compute the longer piece of circle which will be marked as a flat piece. Figure 6.3 illustrates the following computations.

PROPOSITION 6.1 *A circle of radius R has flat pieces if and only if $R \geqslant -e \log(p^*)$.*
In such a case, the length of the detected flat pieces is $L = 2R \sin(1/e)$.

Proof: A circle of radius R being given, let us consider a chord of length L defining a maximum angle α with the corresponding piece of curve ($0 \leqslant \alpha \leqslant \pi/2$). The values of α and L are related by $L = 2R \sin(\alpha)$. The probability defined earlier is $\alpha^{R\alpha}$ (or expressed as a function of L: $\arcsin(L/2R)^{R \arcsin(L/2R)}$). The function $\alpha \mapsto \alpha^{R\alpha}$ shows a minimum for $\alpha = 1/e$. Consequently, $\forall\, \alpha,\ \alpha^{R\alpha} \geqslant e^{-R/e}$.

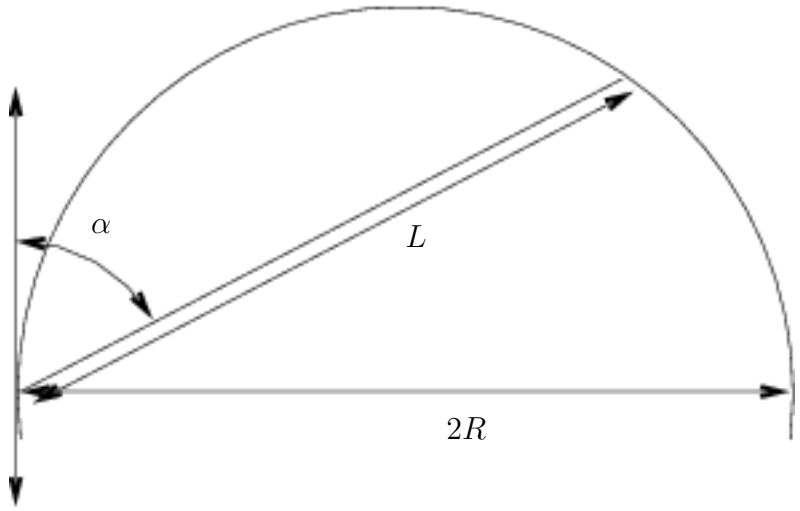


Figure 6.3: *Illustration of the flat pieces computation on a circle.*

Thus, if the probability threshold is set to p^*, and if $R < -e\log(p^*)$, then the circles of length R will show no flat piece. On the contrary, if $R \geqslant -e\log(p^*)$, the detected flat pieces (after the greedy step) in circles of radius R will always show a maximum angle $\alpha = 1/e$ (that is to say 21 degrees, corresponding to an arc of $1/9$ of the total circle), and their length will be $L = 2R\sin(1/e)$ ■

Let us remark that p^* only controls the minimum radius under which no flat piece will be detected: $-e\log(p^*)$. It appears only through its logarithm and small variations of it will not influence the final result. Now, although for symmetry reasons no piece of circle should be favored by the algorithm, the position of the detected flat pieces over a circle strongly depends on the starting point of the discrete curve describing this circle. This makes flat pieces of circular curves not reliable.

6.3 Experiments

6.3.1 Experimental validation of the flat piece algorithm

Experimental results are shown in figures 6.5 to 6.10 (original images can be seen on figure 6.4). For each image, the computation time is less than 10 seconds. When images do not show long level lines, the computation time is less than one second. Such a computation time is clearly too long for real-time processing, but is still lower than the computation time of the other steps of the whole shape recognition method that is described along this thesis.

6.3.2 Flat pieces correspond to salient features

Figure 6.11 shows the result of the proposed flat pieces detector over all level lines in an image ("all" level lines in the sense that the gray level quantization is 1, and therefore permits to exactly reconstruct the original image from the level lines and the corresponding gray level). Some segments are

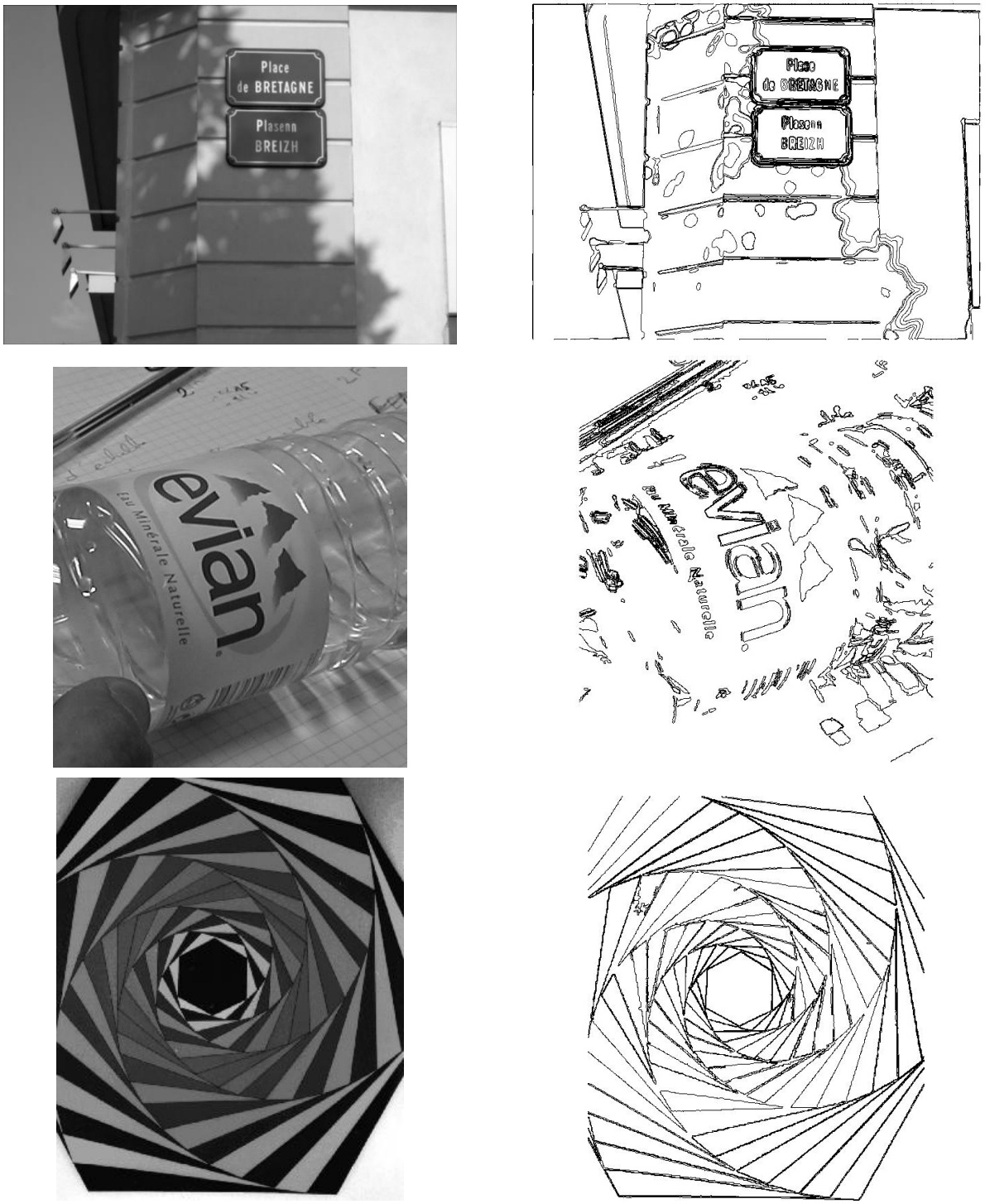


Figure 6.4: *Images (left) and meaningful level lines (right). Top:* bretagne, 413 *level lines. Middle:* evian, 481 *level lines. Bottom:* Vasarely, 172 *level lines.*

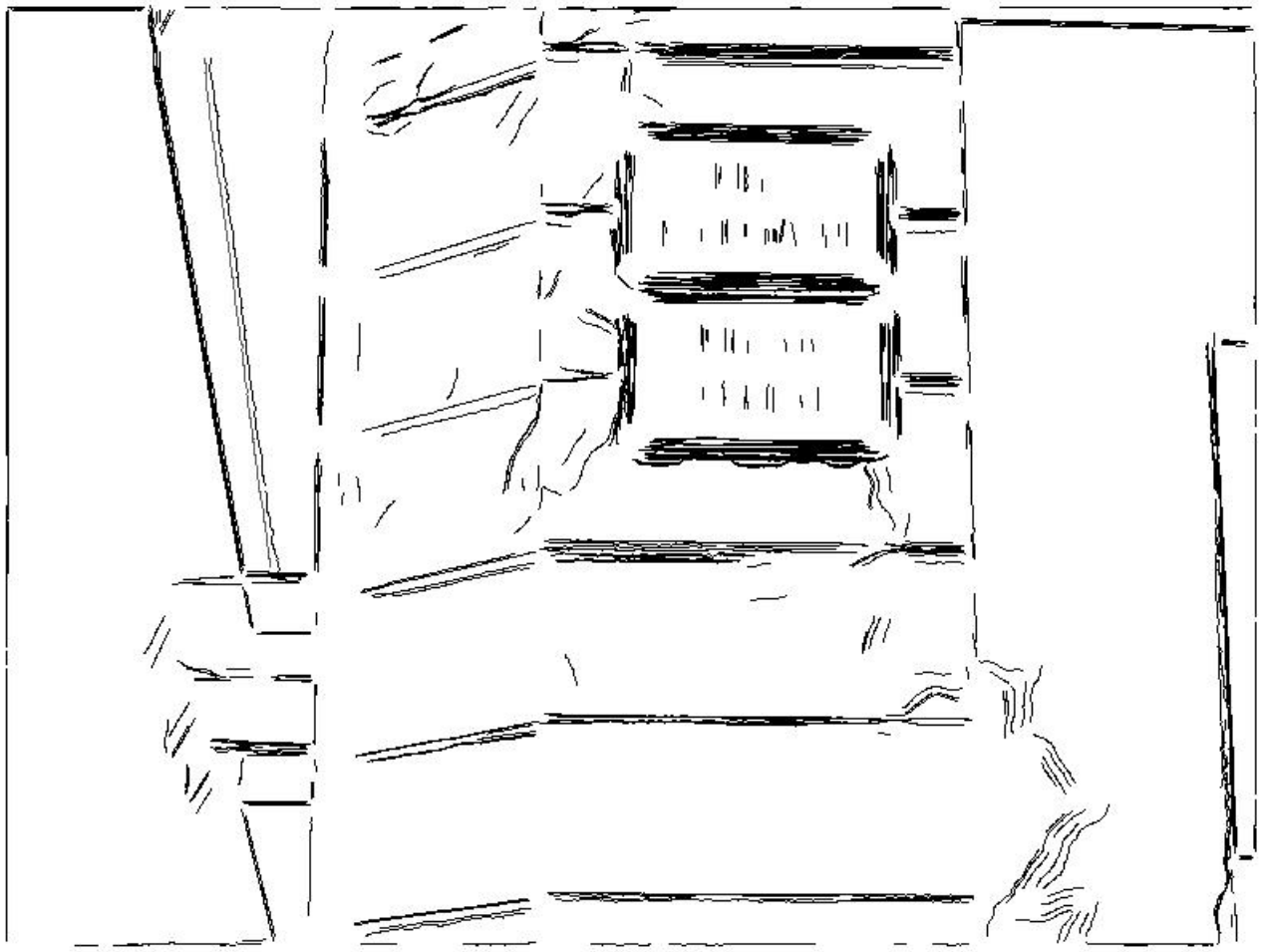


Figure 6.5: *Flat pieces detection:* Bretagne. 1004 *detections. Flat pieces as small as the ones in the letters of the name of the street are detected. Flat pieces in the boundaries of the shadows can be eliminated by dropping down the probability threshold, as can be seen on figure 6.6. Nevertheless, these detections actually correspond to small flat pieces.*

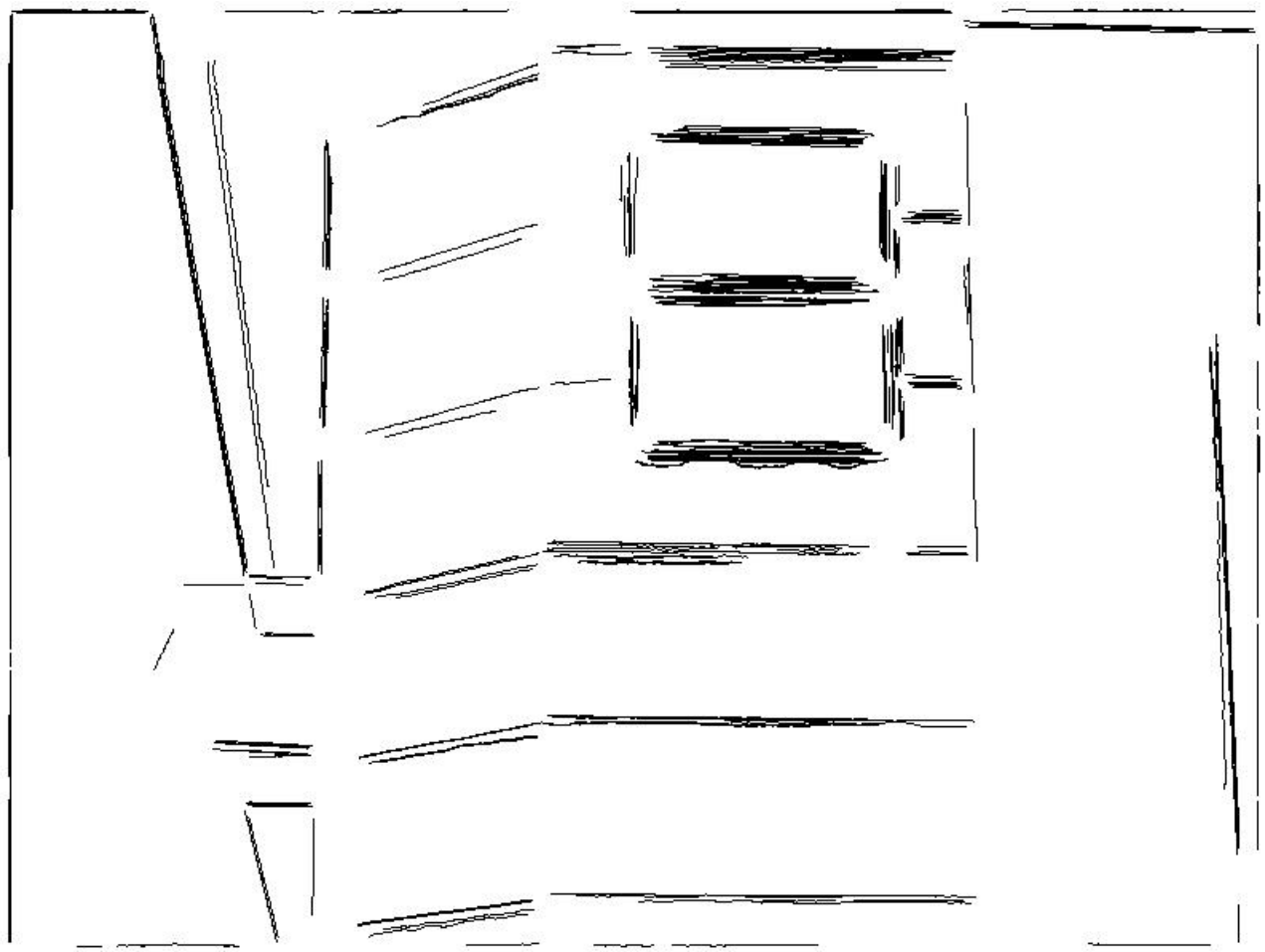


Figure 6.6: *Flat pieces detection:* Bretagne*, with* $p^* = 10^{-10}$*,* 417 *detections. Letters are too small to be detected, but detected flat pieces are very accurate.*

Figure 6.7: *Flat pieces detection:* Evian. 448 *detections.*

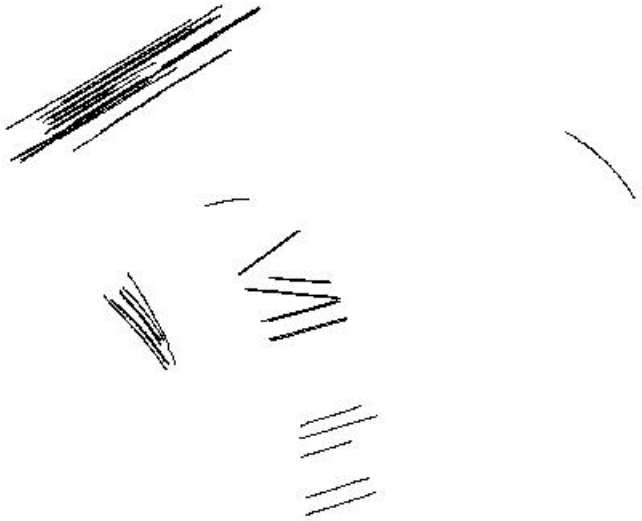


Figure 6.8: *Flat pieces detection:* Evian, *with* $p^* = 10^{-10}$, 64 *detections.*

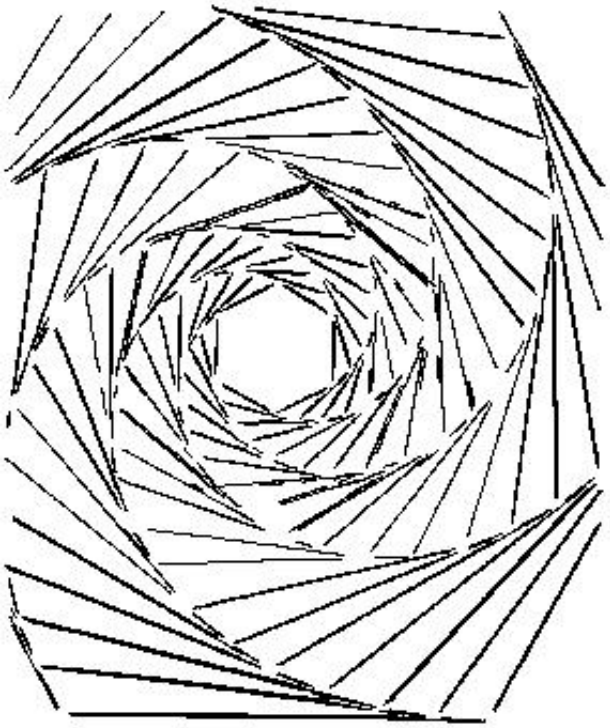


Figure 6.9: *Flat pieces detection:* Vasarely, 774 *detections. Each triangle edge is correctly detected as a single flat piece.*

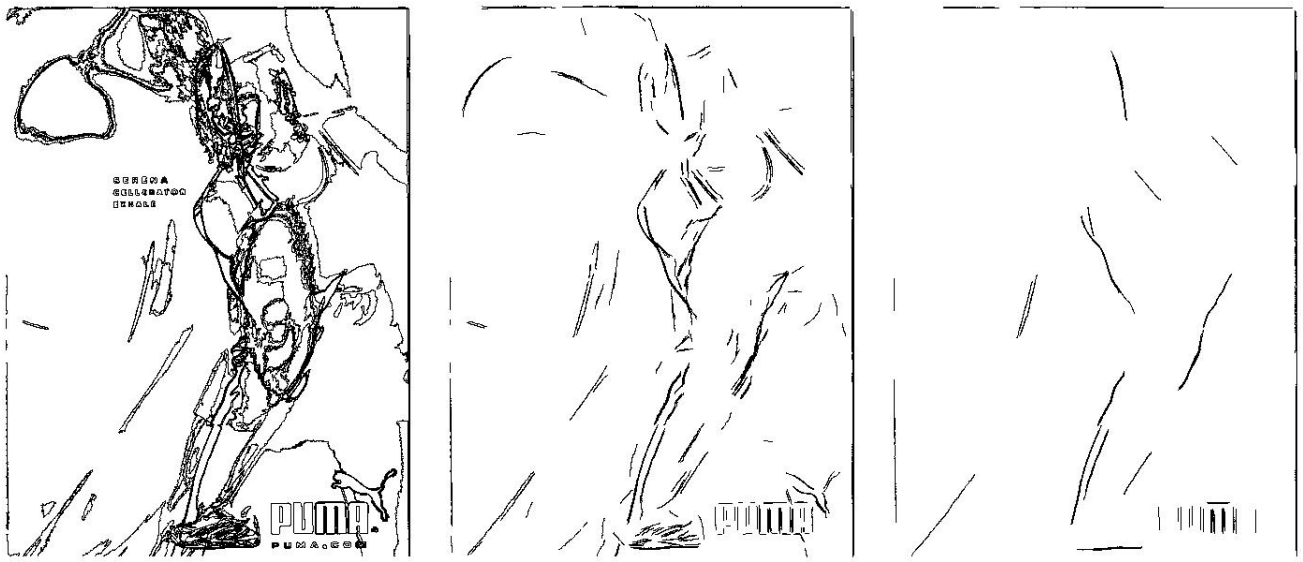


Figure 6.10: *Flat pieces detection:* Serena Williams & Puma. *Left: Original level lines (*425 *lines). Middle:* $p^* = 10^{-3}$ *(*675 *detections). Right:* $p^* = 10^{-10}$ *(*156 *detections). Flat pieces on letters are correctly extracted.*

detected over level lines corresponding to quantization noise (*i.e.* not contrasted level lines over perceptually uniform areas), but these segments actually corresponds to small pieces of straight lines. They are not detected any longer when probability threshold p^* is set to 10^{-10} instead of the standard value (10^{-3}). Flat pieces are concentrated along edges. This experiment can be seen as a confirmation that segment lines are actually salient features of images.

When comparing to figures 6.5 to 6.8, we notice that practically, flat pieces detection can be restricted to maximal meaningful boundaries. Indeed, the other level lines do not provide valuable information.

6.3.3 A comparison between the proposed algorithm and J.L. Lisani's rule

Our aim was to detect flat pieces on curves in various situations. In his PhD thesis, J.L. Lisani uses flat points in order to build robust semi-local normalisations. Figures 6.13 to 6.16 show a comparison between the proposed flat pieces and flat points. See captions for details.

6.4 Conclusion

In this chapter, we presented a method to detect flat pieces in curves. The presented algorithm provides:

- segments and not only points. (the segments showed on the illustrations for flat points are only for visual purpose, flat points do not give any localisation information.)
- a direction given by the endpoints of the detected flat pieces, which is more robust than the tangent to the flat point.

This latter point is all the more important as the proposed encoding procedure is based on these directions. Moreover, many more flat pieces are detected than flat points, increasing the number of codes, and consequently making the representation more complete. As pointed out in Chapter 7, robust inflexion points are most of the time surrounded by a flat piece. Thus, we do not need to estimate and make use of the inflexion points any more.

As far as smoothing is concerned, let us notice that, generally speaking, the assumption "two points at a distance greater than 2 pixels are independent" is no more valid if the curves are smoothed. Nevertheless, since the smoothing which is performed in our shape recognition framework is very slight (its scale corresponds to one pixel), we do not take this bias into account.

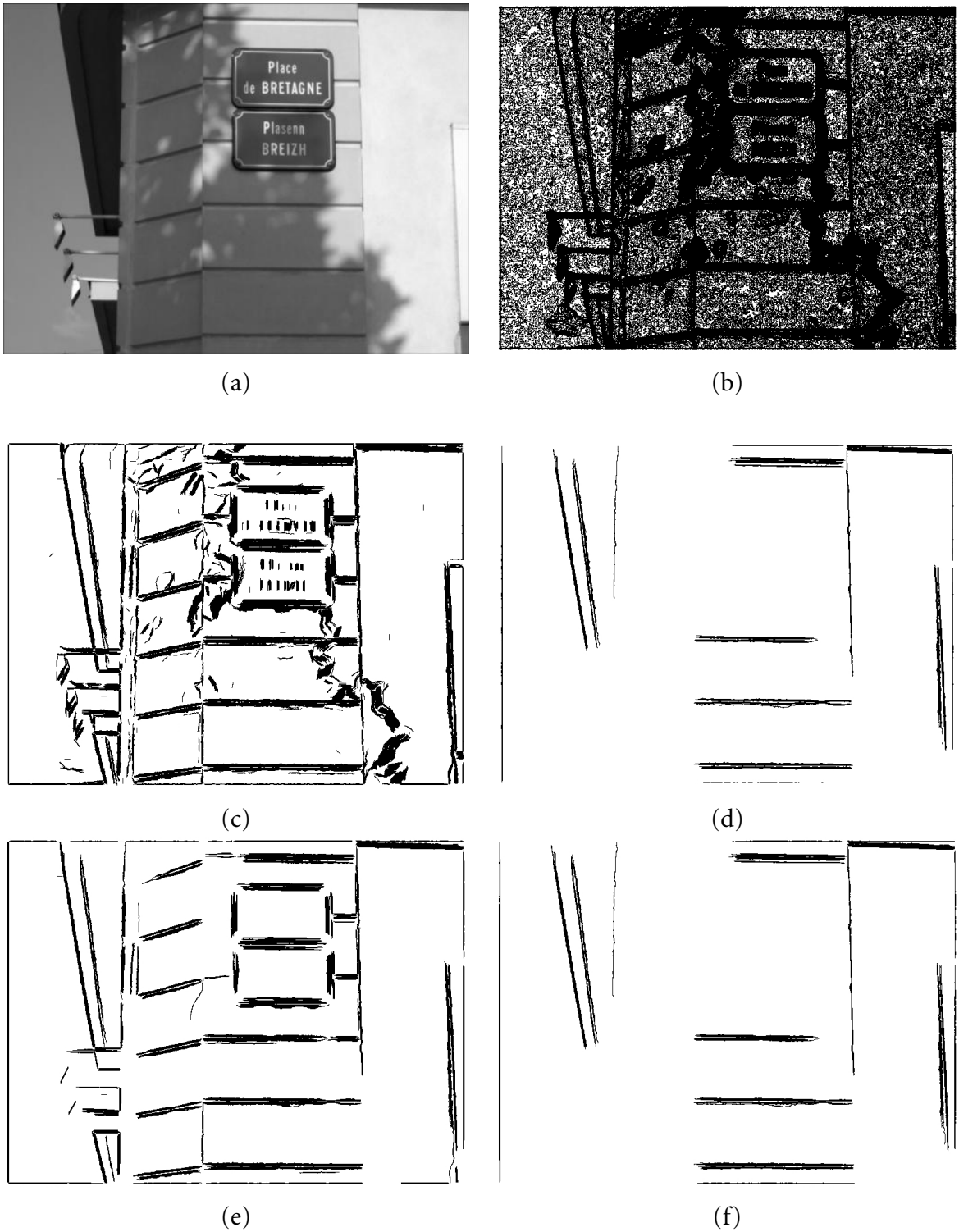


Figure 6.11: *Flat pieces detection. (a) original image (size:* 512×384*); (b)* $25,755$ *level lines (quantization step:* 1 *gray level) (c)* $20,065$ *flat pieces detected over these level lines (probability threshold* p^* *has here its standard value:* 10^{-3}*); (d) flat pieces of length greater than* 100 *pixels among the previous ones; (e)* $6,233$ *flat pieces detected over these level lines, when probability threshold* p^* *is set to* 10^{-10}*; (f) flat pieces of length greater than* 100 *pixels among the previous ones. Flat pieces appear to be concentrated along edges. These edges appear as thick because a strong gradation of grey can be seen at their location, and thus many parallel pieces of level lines.*

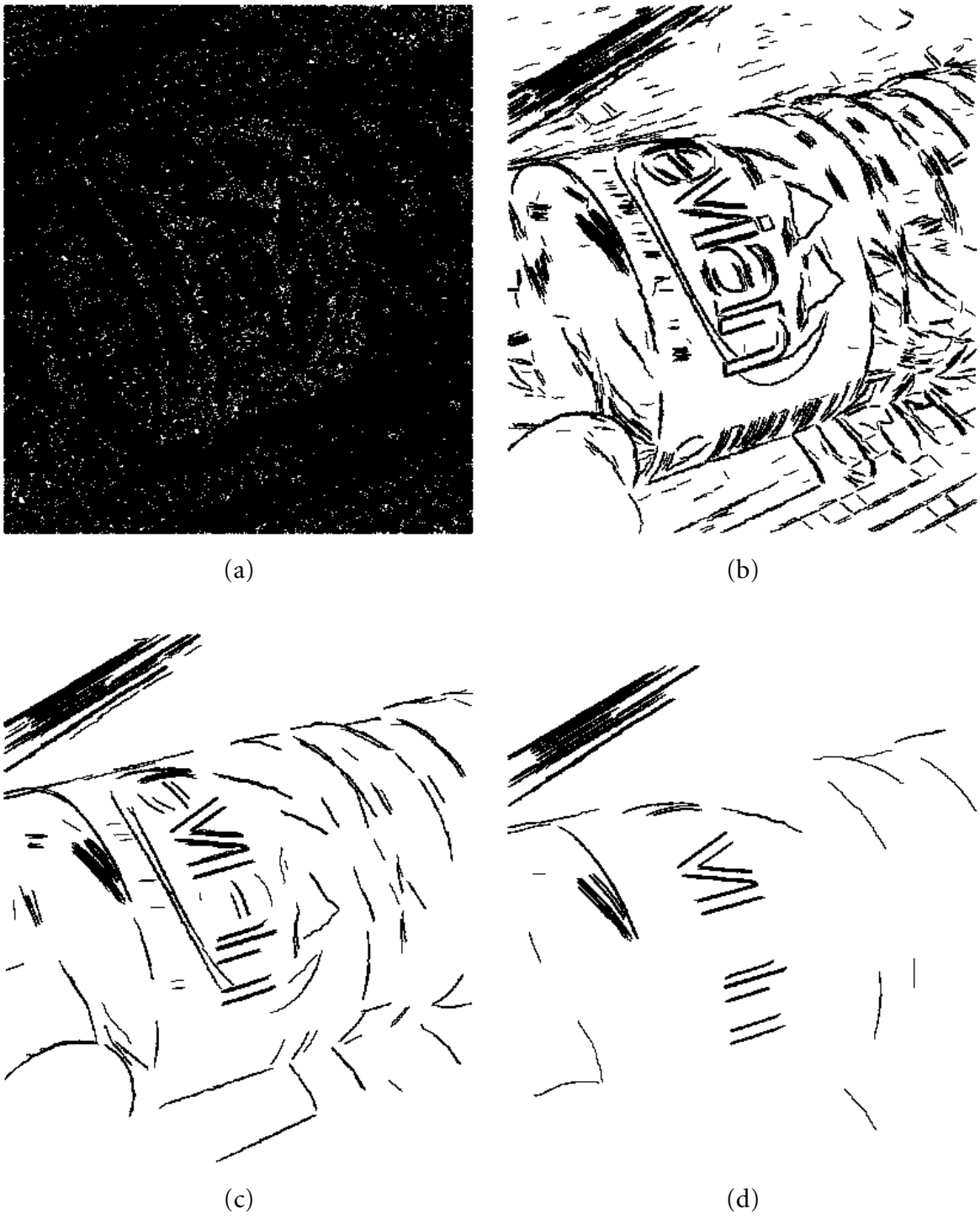


Figure 6.12: *Flat pieces detection. (a)* 90078 *level lines from evian image (quantization step:* 1 *gray level); (b) flat pieces detections over these level lines (*16533 *detections); (c) flat pieces detection with* $p^* = 10^{-6}$ *(*4659 *detections); and (d) flat pieces detection with* $p^* = 10^{-10}$ *(*2041 *detections). Flat pieces are concentrated along edges.*

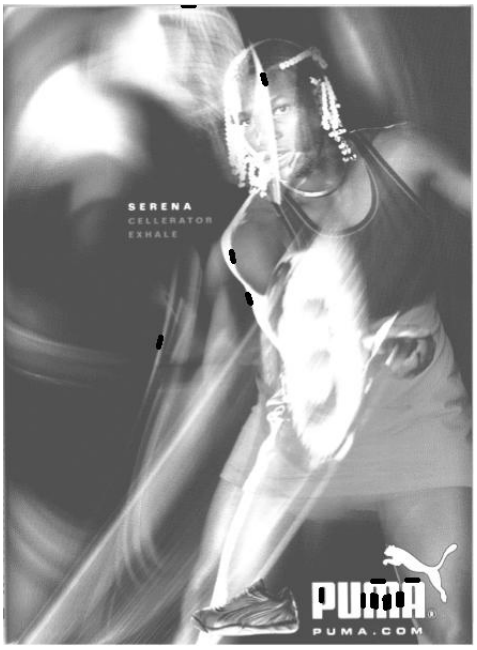


Figure 6.13: *J.L. Lisani's flat points:* Serena Williams & Puma. 15 *flat points are detected. To be compared to the results on figure 6.10.*

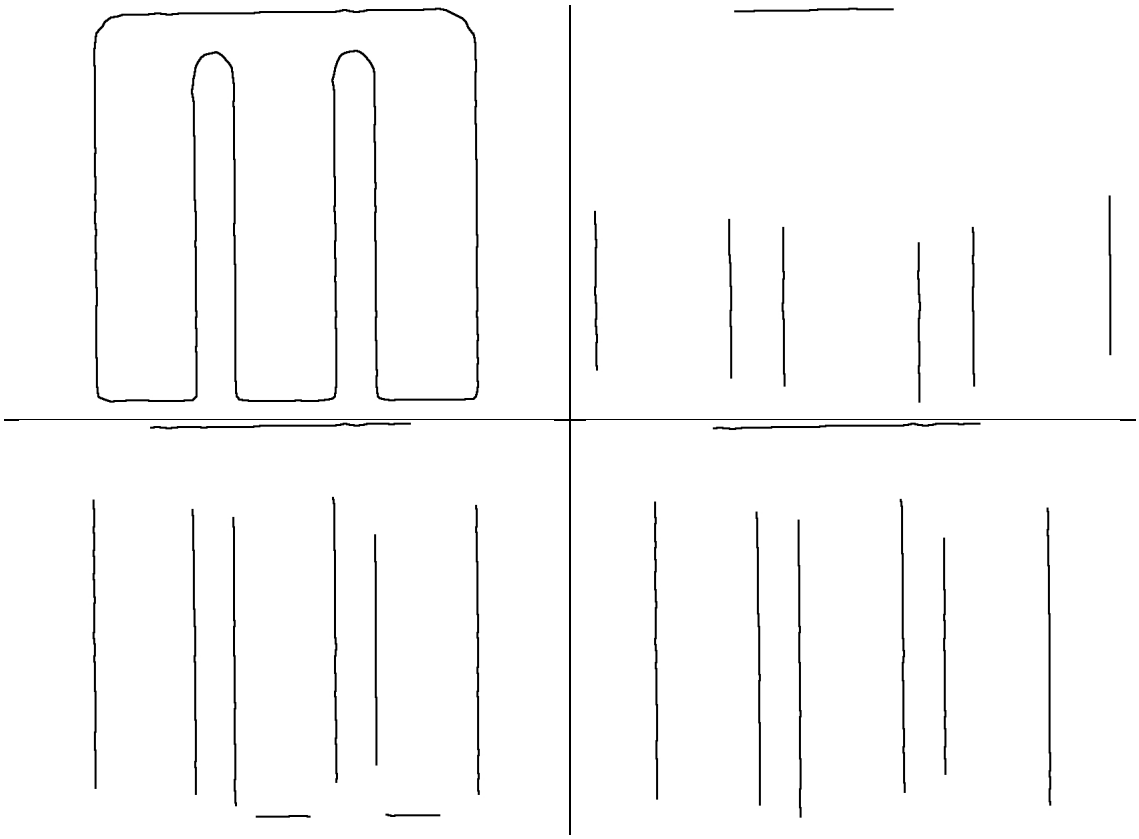


Figure 6.14: *Flat points* vs *flat pieces:* Serena Williams & Puma. *From left to right and from top to bottom: considered shape, flat point (7 detections), flat pieces with $p^* = 10^{-3}$ (9 detections), flat pieces with $p^* = 10^{-10}$ (7 detections). One of the flat pieces in the "legs" of the character M is not detected since these curve pieces are too small and pose a sampling problem. Since not* all *chords are tested but a subset of them, endpoints may sometimes be not conveniently distributed.*

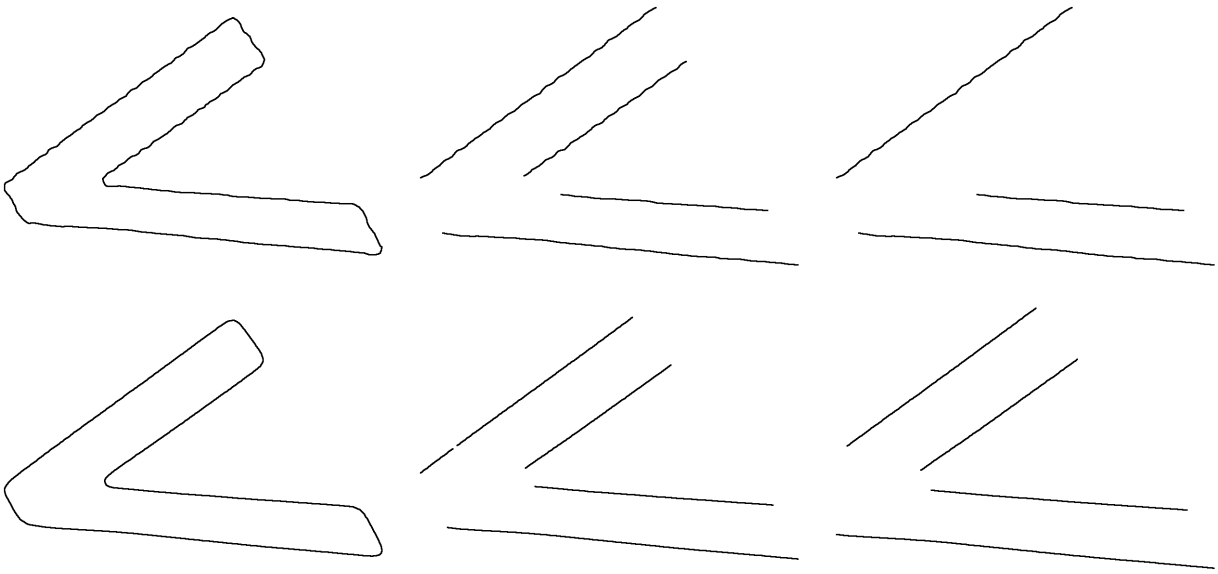


Figure 6.15: *Flat points* vs *flat pieces: character V in* Evian. *Top: no smoothing. From left to right: original shape, flat pieces with $p^* = 10^{-3}$ (4 detections) and with $p^* = 10^{-10}$ (3 detections). Flat points algorithm does not provide any detection. Bottom: after smoothing (see Chapter 5). From left to right: original shape, flat pieces with $p^* = 10^{-3}$ (5 detections) and flat pieces with $p^* = 10^{-10}$ (4 detections). With $p^* = 10^{-3}$, one of the segments is split because of the discretization procedure in the multi-scale test of chords. Flat points algorithm does still not provide any detection.*

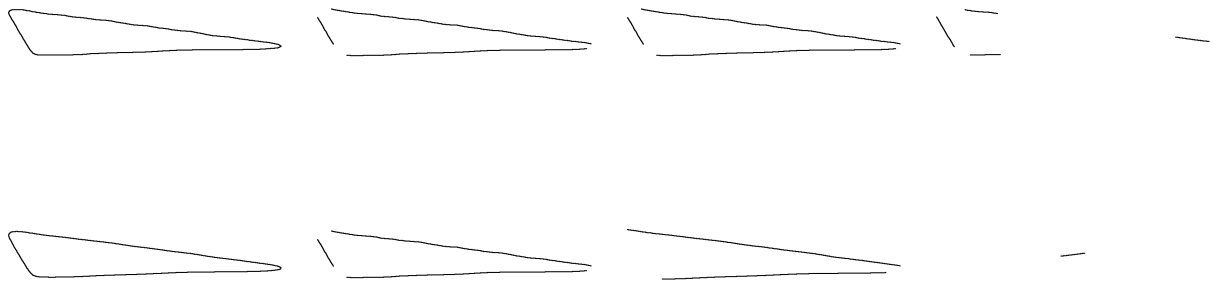


Figure 6.16: *Flat points* vs *flat pieces: a triangle in* Vasarely. *Top: no smoothing. From left to right: original shape, flat pieces with $p^* = 10^{-3}$ (3 detections) and flat pieces with $p^* = 10^{-10}$ (3 detections), and flat points (4 detections). Bottom: after smoothing (see Chapter 5). From left to right: original shape, flat pieces with $p^* = 10^{-3}$ (5 detections) and flat pieces with $p^* = 10^{-10}$ (2 detections), and flat points (1 detection).*

Chapter 7

Local and global invariant encoding of shapes

Abstract: In Chapters 4 and 5 we described the shape extraction and smoothing procedures. Here we present two methods to encode shapes. Both methods build representations invariant up to either similarity or affine transforms. The first one is semi-local, and can deal with occlusions. However, as we will see, it does not allow to encode all of the extracted boundaries. Hence, we introduce a second algorithm to globally encode those boundaries that have not been encoded by the semi-local method.
Both algorithms are based on bitangent lines and flat pieces. This makes encoding very stable, since it relies on robust directions instead of on some points on the considered curve whose localization may not be numerically stable.

Résumé : Dans les chapitres 4 et 5 nous avons décrit les procédures d'extraction et de lissage des formes. Nous présentons ici deux méthodes pour les coder. La première est semi-locale, et est robuste vis-à-vis des occlusions. Néanmoins, comme nous le verrons, elle ne permet pas de coder toutes les frontières extraites. Nous introduisons donc un second algorithme pour coder de manière globale les frontières qui n'ont pas été codées par la méthode précédente.
Les deux algorithmes sont basés sur les bitangentes et les portions plates. Ceci rend le codage très stable, car il repose alors uniquement sur des directions robustes et pas sur des points des courbes dont la localisation pourrait ne pas être très stable numériquement.

7.1 Previous stages: shape extraction and smoothing

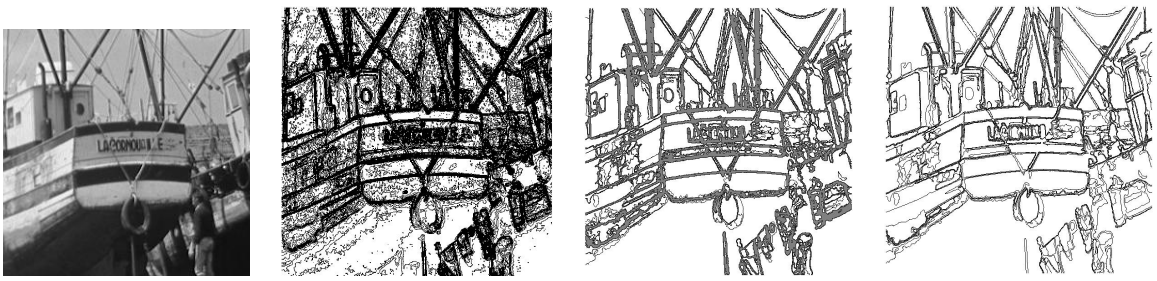
In this chapter we describe a method to build invariant representations of Jordan curves, applied to the sets of maximal meaningful boundaries extracted from images (see Chapter 4). Following invariance requirements presented in Chapter 1, shape representation should be invariant to contrast changes, robust to noise and invariant to a group of geometric transforms (similarity or affine groups). It should also be local, in order to deal with occlusions. An algorithm encoding shapes from an image that almost completely satisfies these requirements can proceed with the following steps:

1. Extraction of maximal meaningful level lines.
2. Affine invariant smoothing of the extracted level lines.
3. Local encoding of pieces of level lines after affine normalization.

Consider the level lines in an image (*i.e.* the boundaries of the connected components of its level sets). This representation has several advantages. Although it is not invariant under *scene illumination* changes (in this case the image in itself is changed and no descriptor remains invariant), it is invariant under *contrast* changes. The mathematical morphology school has claimed that all shape information is contained in level lines, and this is certainly correct, in the sense that we can reconstruct the whole image from its level lines. Moreover, the boundaries of the objects lying in the image are well represented by the union of some pieces of level lines. Thus, level lines can be viewed as concatenations of pieces of boundaries of objects and therefore encode all shape information.

Nevertheless, the representation provided by level lines is highly redundant, and may also contain useless information. That is why, in chapter 4, we described a method to extract meaningful level lines [DMM01, CMS04] from images. The algorithm needs no parameter tuning, since parameters are automatically set based on statistical arguments derived from perceptual principles. Meaningful boundaries are not contrast invariant, since their detection depends on the contrast distribution in the image. However, they are still invariant with respect to affine contrast change.

Figure 7.1 illustrates that the loss of information of maximal meaningful boundaries is negligible compared to the gain of information compactness. This reduction is crucial in order to speed up the shape matching stage following encoding. Otherwise, the presented methodology could not be realistic for applications such as image retrieval from databases.



(a) (b) (c) (d)

Figure 7.1: *Extraction of meaningful level lines. (a) original "La Cornouaille" image, (b) all level lines with grey-level step equal to* 10 *(5479 level lines), (c) all meaningful boundaries (4342 detections), (d) maximal meaningful boundaries (296 detections).*

Once maximal meaningful boundaries are extracted, we need to smooth them in order to eliminate noise and aliasing effects (we fix the smoothing scale in order to remove details of size one pixel). The Geometric Affine Scale Space [ST93, AGLM93], described in Chapter 5, is fully convenient (since

smoothing commutes with affine transforms):

$$\frac{\partial x}{\partial t} = |\text{Curv}(x)|^{\frac{1}{3}}\, \mathbf{n}(x),$$

where x is a point on a level line, $\text{Curv}(x)$ the curvature and $\mathbf{n}(x)$ the normal to the curve, oriented towards concavity. We use a Moisan's fast implementation [Moi98], also described in Chapter 5. This step does not involve any user parameter, since the scale at which the smoothing is applied is fixed and given by the pixel size. Once again, the aim is to reduce the amount of level lines in order to simplify the most pertinent ones, the final goal remaining the same: to make the shape matching stage faster. Indeed, as we will see, smoothing reduces the number of encoded shape elements.

The last stage of the invariant shape encoding algorithm is local normalization and encoding. Roughly speaking, in order to build invariant representations (up to either similarity or affine transforms), we define local frames for each level line, based on robust directions (tangent lines at flat pieces, or bitangent lines). Such a representation is obtained by uniformly sampling a piece of curve in this normalized frame.

The conjunction of these three stages was first introduced by Lisani *et al.* [Lis01, LMMM03]; the third stage is also based on Rothwell's work on invariant indexing [Rot95].

Let us remark that this semi-local normalization/encoding stage only allows to encode non-convex enough curves. We thus need a second algorithm to globally encode those boundaries that have not been encoded by the semi-local method. In the following sections we give a more precise description of both semi-local and global normalization/encoding methods.

7.2 Semi-local normalization and encoding

The semi-local normalization of Jordan curves that we present in this section is based on robust directions, given by tangent lines at flat pieces, or by bitangent lines. While bitangency is an affine invariant property, it is not the case for flat pieces. However, two arguments stand for its consideration. The first one is that, under reasonable zoom factors, flat pieces are preserved. The second argument is that inflexion points, which are conserved by affine transforms, are most of the time surrounded by a flat piece, which is by consequent also conserved by affine transforms. If it is not the case, the tangent at the inflexion point will not be a robust direction. In that sense, tangent at flat pieces can also be considered as robust versions of tangents at inflexion points.

We now give the procedures for semi-local normalization/encoding of level lines, for similarity and affine invariance. In what follows we consider direct Euclidean parameterization for level lines, as usual.

Similarity invariant normalization and encoding

In order to represent a level line $\mathcal{L}$, for each flat piece, and for each couple of points on which the same straight line is tangent to the curve, do the following (this procedure is illustrated in Figure 7.2):

a) Call P_1 the first tangency point and P_2 the other one (for flat pieces P_1 and P_2 are the endpoints of the detected flat segment). Consider the tangent line $\mathcal{D}$ to these points;

b) Call $\mathcal{P}_1$ the previous tangent to $\mathcal{L}$, orthogonal to $\mathcal{D}$, starting from P_1. Call $\mathcal{P}_2$ the next tangent to $\mathcal{L}$, orthogonal to $\mathcal{D}$, starting from P_2;

c) Find the intersection points between $\mathcal{P}_1$ and $\mathcal{D}$, and between $\mathcal{P}_2$ and $\mathcal{D}$. Call them R_1 and R_2, respectively;

d) Store the *normalized* coordinates of N equi-distributed points over an arc on $\mathcal{L}$ of length $F \cdot \|R_1 R_2\|$, centered at C, the intersection point of $\mathcal{L}$ with the perpendicular bisector of $[R_1 R_2]$. By "normalized coordinates" we mean coordinates in the similarity invariant frame defined by points R_1, R_2 mapped to $(-\frac{1}{2}, 0), (\frac{1}{2}, 0)$, respectively.

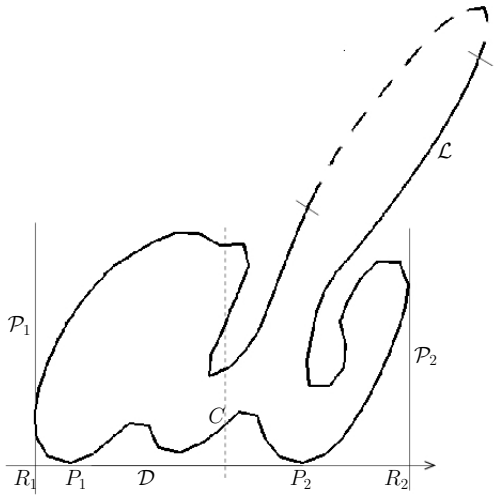


Figure 7.2: *Similarity invariant semi-local encoding based on the bitangent line $\mathcal{D}$.*

Two implementation parameters, F and N, are involved in this normalization procedure. The value of F determines the normalized length of the shape elements, and is to be chosen having in mind the following trade-off: if F is too large, shape elements will not be well adapted to deal with occlusions, while if it is too small, shape elements will not be discriminatory enough. The choice of F faces then a classical dilemma in shape analysis, addressed in Chapter 2: locality *versus* globality of shape representations. The choice of N is less critical from the shape representation viewpoint, since it is just a precision parameter. Its value is to be chosen as a compromise between accuracy of the shape element representation, and computational load.

In Figure 7.3 we show all codes extracted from a single boundary, taking $F = 5$ and $N = 45$. Except for the last five codes, which are based on bitangent lines, all codes correspond to flat pieces. Notice that the representation is quite redundant, and yields shape elements describing the boundary over a wide range of scales. While the representation is certainly not optimal because of redundancy, it

increases the possibility of finding common shape elements when corresponding shapes are present in images, even if they are degraded or subject to partial occlusions.

All the experiments in Chapter 9 concerning matching based on this semi-local encoding (section 9.1) were carried out using $F = 5$ and $N = 45$, since it seems to be a good compromise solution. Hence, in general, these parameters can be fixed once for all, and they are not to be tuned by the user.

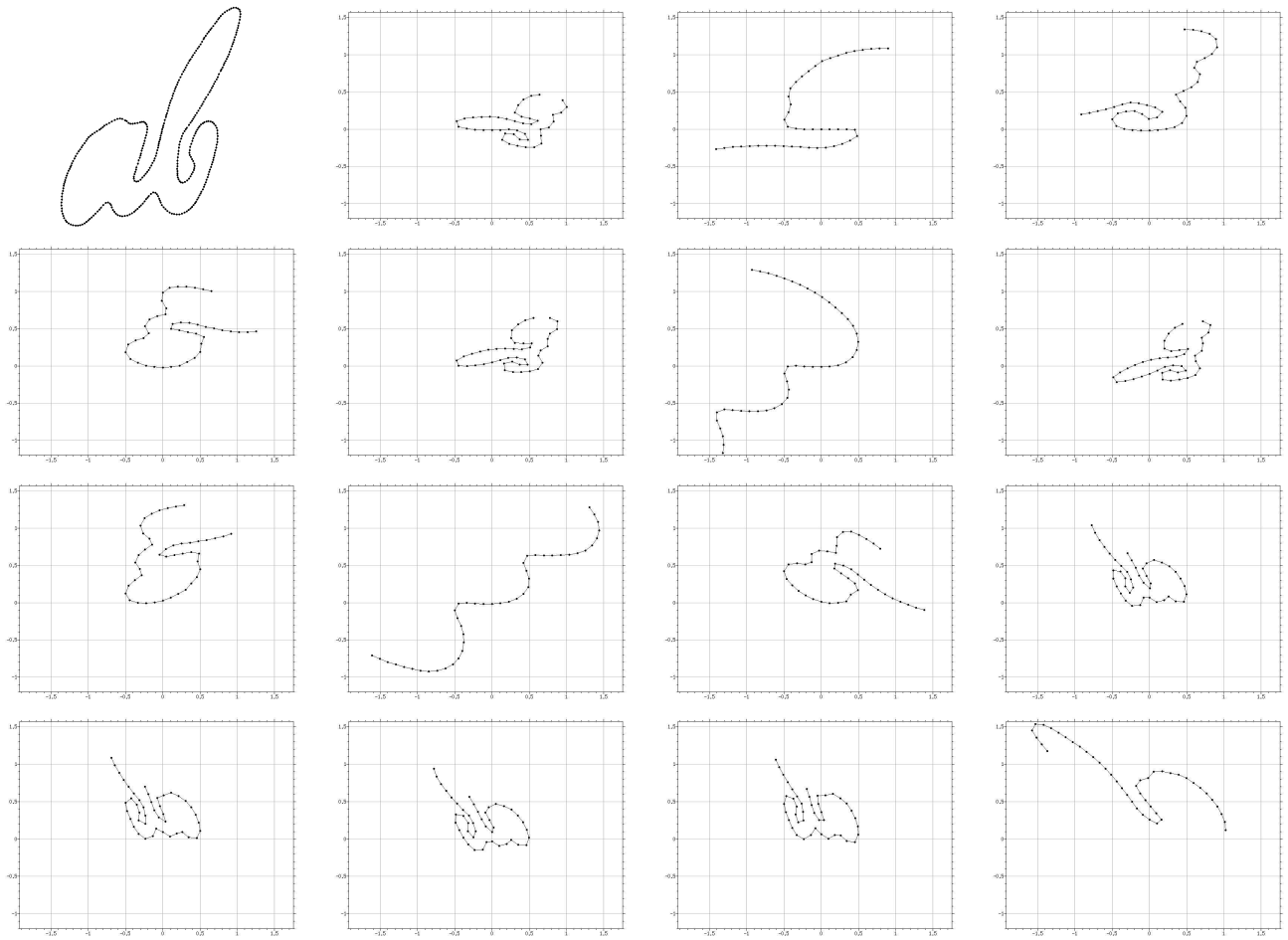


Figure 7.3: *Example of semi-local similarity invariant encoding. The boundary on top left generates 15 codes ($F = 5$, $N = 45$). The five last codes were based on bitangent lines, the other ones were based on flat pieces. The representation is quite redundant, and shape elements describe the boundary over a wide range of scales.*

Affine invariant normalization/encoding

In order to derive an affine invariant representation of a level line $\mathcal{L}$, for each flat piece, and for each couple of points on which the same straight line is tangent to the curve, do the following (this procedure is illustrated in Figure 7.4):

a) Call P_1 the first tangency point and P_2 the other one (for flat pieces P_1 and P_2 are the endpoints of the detected flat segment). Consider the tangent line $\mathcal{D}$ to these point;

b) Starting from P_2, find the next tangent to $\mathcal{L}$ which is parallel to $\mathcal{D}$. Call it $\mathcal{D}'$;

c) Consider the straight lines which are parallel to $\mathcal{D}$ and lay at $1/3$ and $2/3$ of distance from $\mathcal{D}$ to $\mathcal{D}'$. Call them $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively;

d) Starting from P_2, find the next intersection points between $\mathcal{L}$ and $\mathcal{D}_1$, and $\mathcal{L}$ and $\mathcal{D}_2$. Consider the straight line $\mathcal{T}_1$ defined by these two points.

e) Starting from P_1, find the previous tangent to $\mathcal{L}$ parallel to $\mathcal{T}_1$, and call it $\mathcal{T}_2$;

f) Define points R_1, R_2, and R_3 as the intersections between $\mathcal{D}$ and $\mathcal{T}_2$, $\mathcal{D}$ and $\mathcal{T}_1$, and $\mathcal{D}'$ and $\mathcal{T}_2$, respectively;

g) Points R_1, R_2, R_3 define an affine basis. The affine normalization is fixed by mapping $\{R_1, R_2, R_3\}$ into $\{(0,0),(1,0),(0,1)\}$ if $\{R_1, R_2, R_3\}$ is a direct frame, and into $\{(0,0),(1,0),(0,-1)\}$ if not.

h) Encoding: consider the intersection point between $\mathcal{L}$ and the straight line equidistant from $\mathcal{D}$ and $\mathcal{D}'$ (the first one starting from P_2). Call it C. Normalize the portion of $\mathcal{L}$ having normalized length $F/2$ at both sides of C. Store N equi-distributed points over the normalized piece of curve.

As we did for the similarity invariant normalization, implementation parameters were fixed once for all to $F = 5$ and $N = 45$. Figure 7.5 shows all codes extracted from a single boundary for this choice of parameters. Notice the encoding is less redundant than for the similarity encoding procedure. This is due to the fact that the construction of affine invariant local frames imposes more constraints on the curve than the one for similarity invariant frames.

7.3 Global normalization and encoding

The semi-local normalization/encoding procedures we have just described do not allow to encode all kind of Jordan curve, even when they show bitangents or flat pieces. The reason is that these methods impose minimum normalized lengths to shape elements (remember the parameter F was chosen in order to provide discriminatory enough shape elements). For encoding, a good shape element should not be too simple, especially if we are interested in an invariant recognition. For instance, many convex curves can be so alike in affine invariant shape recognition, that decomposing them into smaller parts (shape elements) does not make any sense. That is why such simple curves are not encoded by the semi-local normalization/encoding procedure, and a second procedure to globally encode those boundaries that have not been encoded is needed.

Concerning global shape representations, there are basically two possible approaches: represent shapes by a set of global invariant features (*e.g.* invariant moments), or consider global geometric normalizations. For the first possibility, as we saw in Chapter 2, a critical issue is the definition of a distance for shape comparison. Defining distances between shapes seems to be less difficult for normalization methods, which also have the advantage (contrarily to the representation with a set

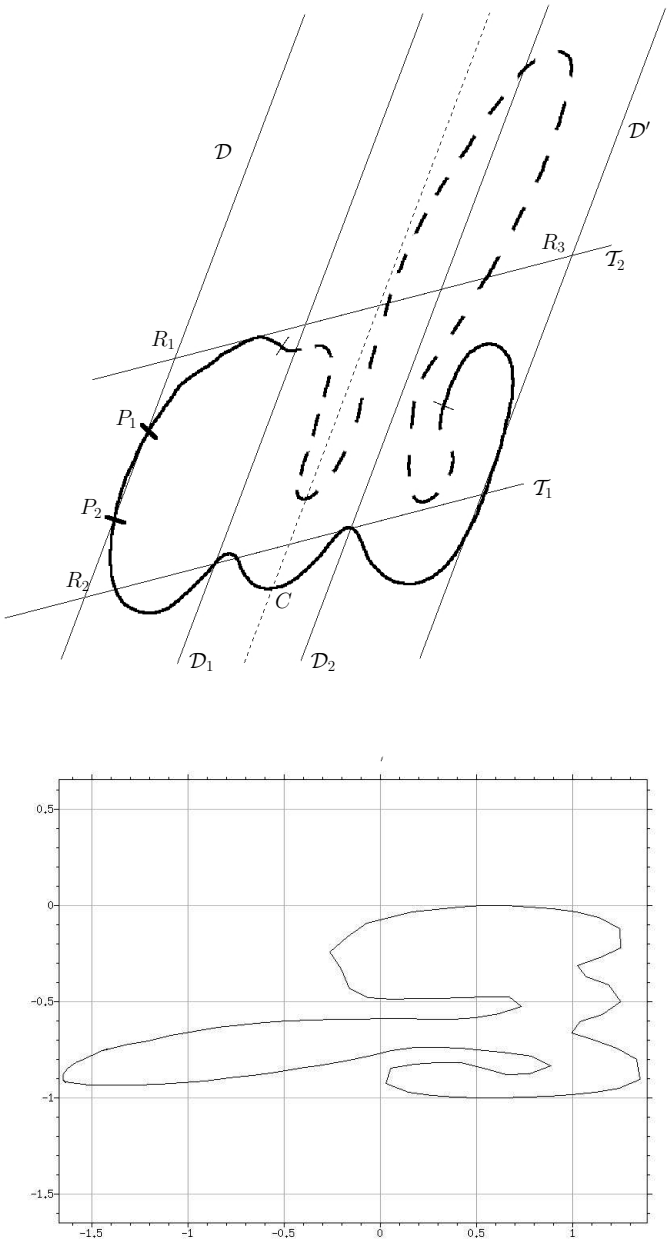


Figure 7.4: *Affine invariant semi-local encoding. The encoded shape element is based on the tangent to the flat piece between marks (top). Bottom: the whole curve normalized in the affine local basis defined on top.*

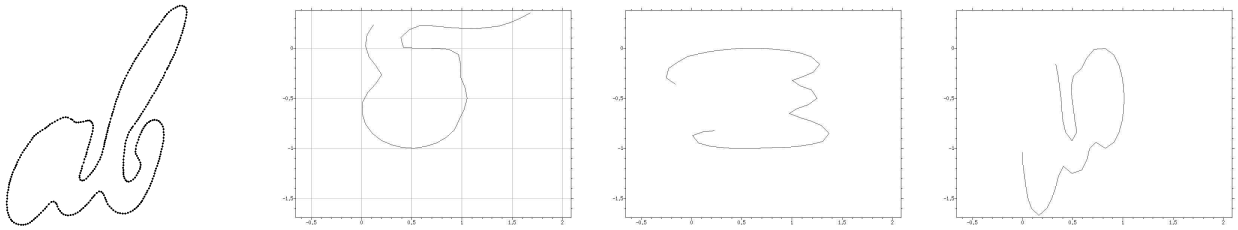


Figure 7.5: *Example of semi-local affine invariant encoding. The boundary on top left generates 3 codes ($F = 5$, $N = 45$). All codes are based on flat pieces.*

of global invariants) to be complete, in the sense the original shape can be reconstructed from its representation.

Classical normalization methods use Hu's invariant moments [Hu62]. In this section we will describe one of these type of methods, proposed by Thierry Cohignac in his PhD thesis [Coh94]. As we will see, this method has some drawbacks common to all moment based normalization methods: they rely in the computation of high order moments, what makes them unstable and very sensitive to noise. This leads us to propose a global normalization technique based on robust directions (shape's bitangent lines and flat pieces).

7.3.1 A global affine invariant normalization method based on moments

In what follows we call "affine invariant normalization" a method to build shape representations that are invariant to any planar affine transform $T(x) = Ax + b$, such that $\det(A) > 0$. In other words, an affine invariant normalization transforms a planar "solid shape" $\mathcal{F}$ (a compact connected subset of $\mathbb{R}^2$) into a normalized shape such that any image of $\mathcal{F}$ by a planar affine transformation will lead to the same normalized shape. Two shapes related by an axial symmetry are not considered to be equivalent in this framework, and will not yield the same normalized shape.

Let us denote by $\mathbb{1}_{\mathcal{F}}$ the indicator of solid shape $\mathcal{F}$. In order to achieve translation invariance of the normalized representation, we may assume $\mathcal{F}$ has been previously translated such that its barycenter is in the origin of the image plane. Hence, the moment of order (p, q) (p and q natural integers) of $\mathcal{F}$ is defined by

$$\mu_{p,q}(\mathcal{F}) = \int_{\mathbb{R}^2} x^p y^q \mathbb{1}_{\mathcal{F}}(x, y) dx dy.$$

Let $S_{\mathcal{F}}$ be the following 2×2 positive-definite, symmetric matrix

$$\frac{1}{\mu_{0,0}} \begin{pmatrix} \mu_{2,0} & \mu_{1,1} \\ \mu_{1,1} & \mu_{0,2} \end{pmatrix},$$

where $\mu_{i,j} = \mu_{i,j}(\mathcal{F})$. By the uniqueness of Cholesky factorization [GL89], $S_{\mathcal{F}}$ may be uniquely decomposed as $S_{\mathcal{F}} = B_{\mathcal{F}} B_{\mathcal{F}}^{\mathrm{T}}$ where $B_{\mathcal{F}}$ is a lower-triangular real matrix with positive diagonal entries.

Definition 7.1 *We call* pre-normalized shape *associated to $\mathcal{F}$ the shape : $\mathcal{F}' = B_{\mathcal{F}}^{-1}(\mathcal{F})$.*

Let us prove that the pre-normalized shape is invariant to affine transforms, up to a rotation. We first state two lemmas.

Lemma 7.1 *Let A be a non-singular 2×2 matrix. Then $S_{A\mathcal{F}} = AS_{\mathcal{F}}A^{\mathrm{T}}$.*

Proof: Let a, b, c and d be real numbers such that:

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

The moment of order $(2,0)$ associated to the shape $A\mathcal{F}$ is given by:

$$\mu_{2,0}(A\mathcal{F}) = \det(A) \int_{\mathbb{R}^2} (ax+by)^2 \mathbb{1}_{\mathcal{F}}(x,y)dxdy = \det(A)(a^2\mu_{2,0} + 2ab\mu_{1,1} + b^2\mu_{0,2}).$$

The same computation for moments of order $(0,2)$ and $(1,1)$ yields

$$\begin{aligned} \mu_{0,2}(A\mathcal{F}) &= \det(A)(c^2\mu_{2,0} + 2cd\mu_{1,1} + d^2\mu_{0,2}), \\ \mu_{1,1}(A\mathcal{F}) &= \det(A)(ac\mu_{2,0} + bd\mu_{0,2} + (ad+bc)\mu_{1,1}). \end{aligned}$$

Since $\mu_{0,0}(A\mathcal{F}) = \det(A)\mu_{0,0}$, one can easily check that $S_{A\mathcal{F}} = AS_{\mathcal{F}}A^{\mathrm{T}}$. ∎

Lemma 7.2 *Let X_0 be a 2×2 invertible matrix. Then, for any 2×2 matrix X: $XX^{\mathrm{T}} = X_0{X_0}^{\mathrm{T}}$ if and only if there exists an orthogonal matrix Q such that $X = X_0Q$.*

Proof: Since X_0 is invertible, $XX^{\mathrm{T}} = X_0{X_0}^{\mathrm{T}}$ iff $X_0^{-1}X(X_0^{-1}X)^{\mathrm{T}} = \mathrm{Id}_2$. Letting $Q = X_0^{-1}X$ yields the result. ∎

Proposition 7.1 *The pre-normalized shape is invariant to any invertible, planar, linear transform $(x,y)^{\mathrm{T}} \mapsto A(x,y)^{\mathrm{T}}$, up to an orthogonal transform. Moreover, if $\det(A) > 0$, the invariance is up to a rotation.*

Proof: Since A is a 2×2 non singular matrix, following Lemma 7.1 we have $S_{A\mathcal{F}} = AS_{\mathcal{F}}A^{\mathrm{T}}$. By letting $B_{\mathcal{F}}$ be the lower-triangular matrix of Cholesky's decomposition of $\mathcal{F}$, it follows that $S_{A\mathcal{F}} = AB_{\mathcal{F}}(AB_{\mathcal{F}})^{\mathrm{T}}$. Now, since $S_{A\mathcal{F}}$ is a 2×2 positive-definite, symmetric matrix, Cholesky factorization yields $S_{A\mathcal{F}} = B_{A\mathcal{F}}{B_{A\mathcal{F}}}^{\mathrm{T}}$, where $B_{A\mathcal{F}}$ is a 2×2 non-singular, lower-triangular real matrix. Then, by Lemma 7.2, we have $B_{A\mathcal{F}} = AB_{\mathcal{F}}Q$, where Q is a 2×2 orthogonal matrix. Hence, $B_{A\mathcal{F}}^{-1}A\mathcal{F} = (AB_{\mathcal{F}}Q)^{-1}A\mathcal{F} = Q^{-1}B_{\mathcal{F}}^{-1}A^{-1}A\mathcal{F} = Q^{-1}B_{\mathcal{F}}^{-1}\mathcal{F}$, what proves the invariance of $\mathcal{F}' = B_{\mathcal{F}}^{-1}\mathcal{F}$ to planar isomorphisms, up to an orthogonal transform. Finally, notice that if $\det(A) > 0$ we have $\det(Q) > 0$. ∎

A closed form for $B_{\mathcal{F}}^{-1}$ in terms of the moments of $\mathcal{F}$ can be computed, by taking the inverse of $B_{\mathcal{F}}$, the lower-triangular matrix given by the Cholesky decomposition of $S_{\mathcal{F}}$:

$$B_{\mathcal{F}}^{-1} = \sqrt{\mu_{0,0}} \begin{pmatrix} \frac{1}{\sqrt{\mu_{2,0}}} & 0 \\ -\frac{\mu_{1,1}}{\mu_{2,0}\sqrt{\mu_{0,2}-\frac{\mu_{1,1}^2}{\mu_{2,0}}}} & \frac{1}{\sqrt{\mu_{0,2}-\frac{\mu_{1,1}^2}{\mu_{2,0}}}} \end{pmatrix}.$$

The pre-normalized shape $\mathcal{F}' = B_{\mathcal{F}}^{-1}\mathcal{F}$ is then an affine invariant representation of $\mathcal{F}$ *modulo* a rotation. In order to obtain a full affine invariant representation, we just need to fix a reference angle. This can be achieved, for instance, by computing

$$\varphi = \operatorname{Arg}\left(\int_0^{2\pi}\int_0^{+\infty} \mathbb{1}_{\mathcal{F}'}(r,\theta)e^{i\theta} r\,dr\,d\theta\right),$$

then rotating $\mathcal{F}'$ by $-\varphi$. Notice this rotation normalization method fails when $\mathcal{F}'$ exhibits central symmetry. However, unlike a classical rotation normalization computing the direction of the principal axis, it has the advantage to assign the same weight to all points in $\mathcal{F}'$, and hence to be more robust to noise affecting its boundary.

Finally, putting all the steps together, the affine invariant normalization of a shape $\mathcal{F}$ is the set of points (x_N, y_N) given by

$$\begin{pmatrix} x_N \\ y_N \end{pmatrix} = \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} B_{\mathcal{F}}^{-1} \begin{pmatrix} x-\mu_{1,0} \\ y-\mu_{0,1} \end{pmatrix},$$

for all $(x, y) \in \mathcal{F}$.

As we can see in Figure 7.6, a classical problem of this kind of normalization is its lack of robustness. Too strong deformations lead to a bad estimation of the moments. The normalization that we propose is based on robust direction, and is redundant, in the sense that a single shape produces several invariant representations.

7.3.2 Global normalization methods based on robust directions

In this section we propose a global normalization framework for shapes. As usual, a "solid shape" $\mathcal{F}$ is a compact connected component of $\mathbb{R}^2$. As we did for the local methods, these global normalizations are based on robust directions given by bitangent lines and flat pieces of shape's boundary $\mathcal{L}$. We describe a variant of this method for translation, translation and rotation, similarity and affine invariant normalizations.

Translation invariant normalization

Robust directions are not used in this particular case.

1. Translate $\mathcal{F}$ so that its barycenter is in the origin of the plane.
2. Define the starting point of $\mathcal{L}$'s parametrization as the intersection of positive ordinate between the vertical axis and the curve. In case of ambiguity, choose the closest one to the origin.

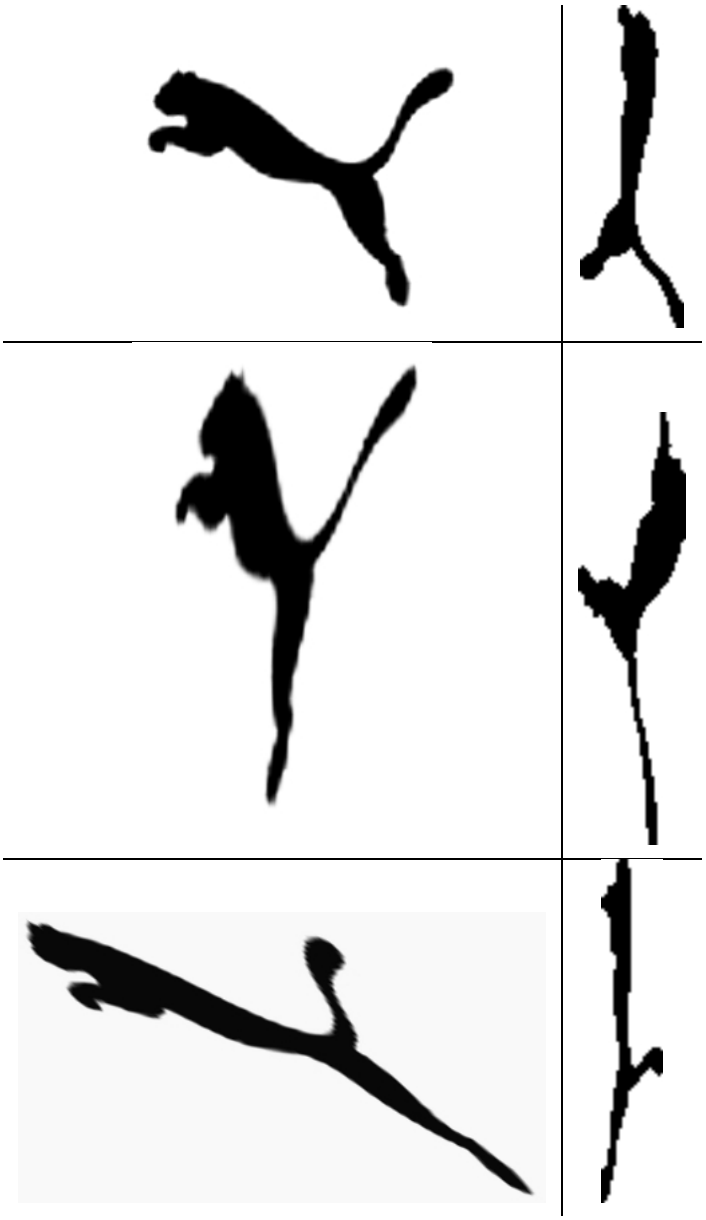


Figure 7.6: *T. Cohignac's normalization. Original images (on the left) and affine normalization using the moments (on the right). The middle and bottom original image were mapped from the top original image up to an affine transform. We can see that, even in this ideal framework, the obtained normalized shapes are not superimposable at all: the moment normalization is not robust. To be compared to the normalization we propose (the middle original image was mapped up to the same transform as on Figure 7.8).*

Translation-rotation invariant normalization

For each robust straight line $\mathcal{D}$ of $\mathcal{L}$:

1. Translate $\mathcal{F}$ so that its barycenter is in the origin of the plane.
2. Rotate $\mathcal{F}$ with respect to the origin so that $\mathcal{D}$ becomes horizontal.
3. Define the starting point of $\mathcal{L}$'s parametrization as the intersection of positive ordinate between the vertical axis and $\mathcal{L}$. In case of ambiguity, choose the closest one to the origin.

Similarity invariant normalization

For each robust straight line $\mathcal{D}$ of the curve:

1. Translate $\mathcal{F}$ so that its barycenter is in the origin of the plane.
2. Scale $\mathcal{F}$ so that its boundary has unit length.
3. Rotate $\mathcal{F}$ with respect to the origin so that the robust direction is horizontal.
4. Define the starting point of $\mathcal{L}$'s parametrization as the intersection of positive ordinate between the vertical axis and the boundary of the shape. In case of ambiguity, choose the closest one to the origin.

Affine invariant normalization (positive determinant)

The procedure is illustrated in Figure 7.7. For each robust straight line $\mathcal{D}$ of $\mathcal{L}$:

1. Consider the straight line passing through the barycenter of $\mathcal{F}$, which is parallel to $\mathcal{D}$. Consider the intersection between $\mathcal{F}$ and the half-plane defined by this straight line which does not contain $\mathcal{D}$; call G_1 its barycenter, and G_3 the barycenter of the complementary part of $\mathcal{F}$.
2. Now consider the straight line passing through G_1 and G_3. It splits the shape into two parts, let G_2 and G_4 be their barycenter, such that $(\overrightarrow{G_3G_1}, \overrightarrow{G_2G_4})$ is directly oriented.
3. Points $\{G_1, G_2, G_3\}$ define an affine basis. Normalize $\mathcal{F}$ by applying to it the affine transform mapping $\{G_1, G_2, G_3\}$ into $\{(0, 1/2), (1/2, 0), (0, -1/2)\}$.
4. Define the starting point of the parametrization as the intersection of positive ordinate between the vertical axis and the boundary of the normalized shape. In case of ambiguity, choose the closest one to the origin.

In Figure 7.8 we show an example of global affine invariant normalization. Notice that shapes represented in the normalized frame are very close.

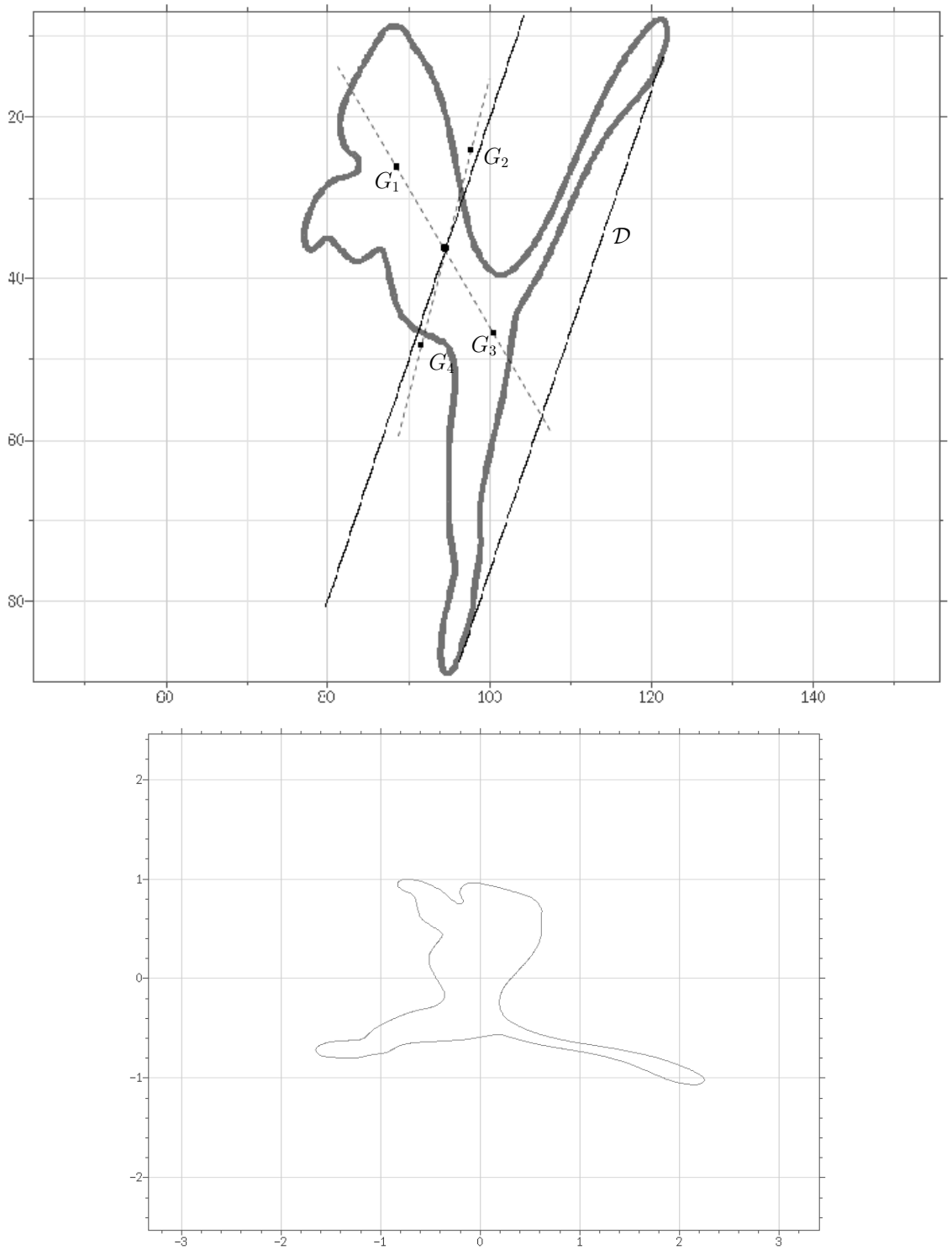


Figure 7.7: *Global affine invariant normalization based on the bitangent line $\mathcal{D}$. Top: definition of points G_1, G_2, G_3 and G_4. Bottom: the normalized shape.*

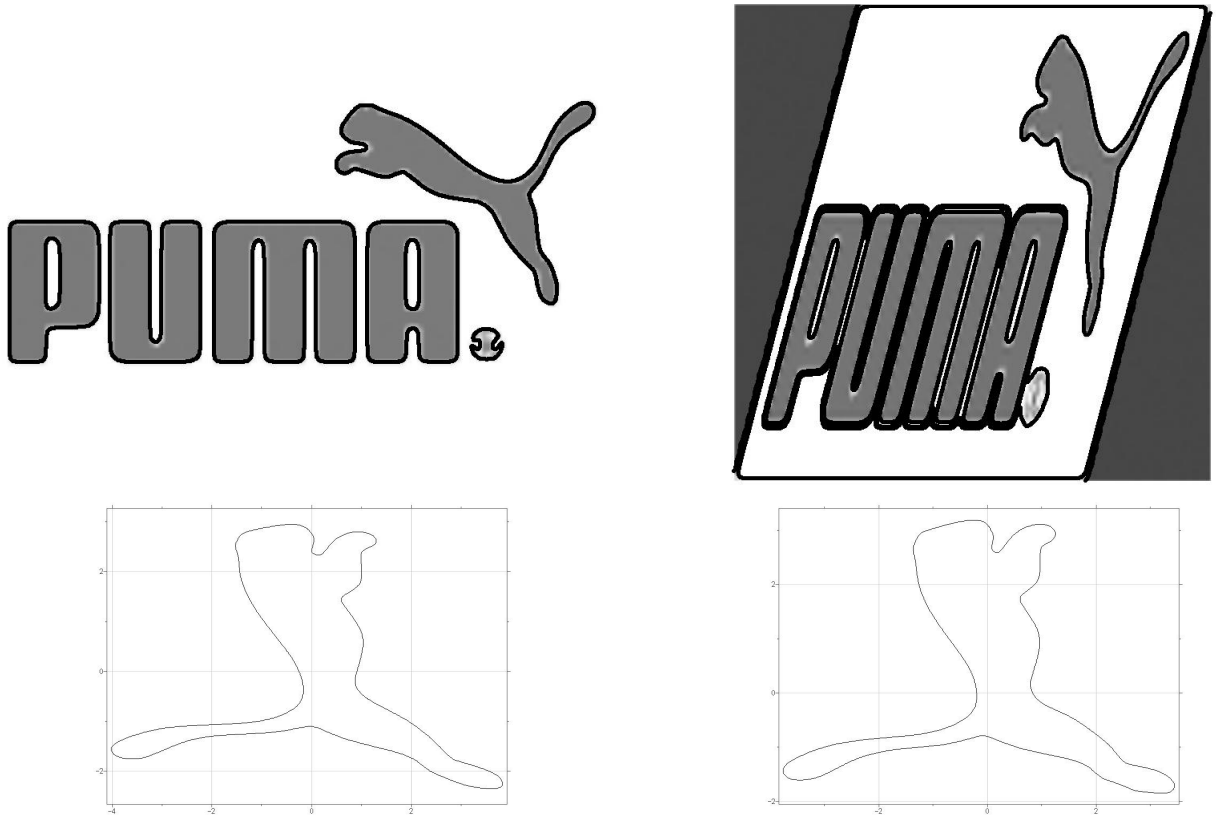


Figure 7.8: *Global affine invariant normalisation based on robust directions. The images on the top are related by an affine transform (the same one as on Figure 7.6. Bottom images: corresponding global affine invariant normalizations of the "puma" shape (both representations were based on flat pieces). The normalized shapes are very close; this is not the case with the invariant moment method.*

7.4 Conclusion

In this chapter we presented and discussed semi-local and global shape normalization methods. The presented semi-local methods, for similarity and affine normalization, are based on bitangent lines and flat pieces, since both provide robust directions to build invariant codes. Each boundary may be represented by several codes, each code representing different parts of the boundary, and often at different scales. Two implementation parameters, F (the normalized length of the shape elements) and N (the number of equi-distributed points over the normalized shape element), are involved in this normalization procedures, but, since they are fixed once for all, they do not need to be tuned by the user. In that sense, these methods can be considered to be parameter free.

The presented semi-local normalization/encoding procedures do not allow to encode all kinds of Jordan curve, even when they show bitangents or flat pieces. Indeed, convex or almost convex curves cannot be encoded by these methods, but this makes sense because the pieces of these curves are too simple, and thus not discriminatory enough to be encoded. In order to encode those curves that have not been represented by local codes, it is sound to consider global normalization/encoding procedures. Classical global normalization methods are based on invariant moments, and are unstable since they use high order moments. We thus propose global normalization methods that, as for the semi-local case, are based on bitangent lines and flat pieces. Normalization is performed by a geometric construction that is very stable, because only area computations are involved. Since one code is built for each bitangent line or flat point, the representation is redundant and permits to avoid ambiguities that may be introduced when curves exhibit some kind of symmetries.

Chapter 8

A CONTRARIO DECISION

Abstract: While shape comparison, shape matching and shape extraction have been the subject of many researches, the decision step has been rarely studied. This chapter presents a framework to answer by *yes* or *no* the question "does that shape look like this one?", and to measure the confidence level in this answer. Since we tackle the general recognition problem, and we do not make use of any *a priori* information, the only possible model is an *a contrario* one. In order to reach high levels of confidence, mutually statistically independent features are extracted from shapes. We check out that our model satisfies the Helmholtz principle: any detection in noise should be considered as not relevant.

Résumé : Alors que la comparaison de formes, leur appariement, et leur extraction, ont été l'objet de nombreuses recherches, l'étape de décision a rarement été étudiée. Le but de ce chapitre est de proposer un cadre pour répondre par *oui* ou par *non* à la question « Cette forme ressemble-t-elle à cette autre forme ? », et de donner un degré de confiance en cette réponse. Comme nous abordons le problème générique de la reconnaissance, le seul modèle possible est un modèle *a contrario*. Afin d'obtenir une confiance élevée dans la reconnaissance, des caractéristiques mutuellement statistiquement indépendantes sont extraites des formes. Nous vérifions que ce modèle satisfait le principe de Helmholtz : toute détection dans le bruit doit être considérée comme non pertinente.

8.1 *A contrario* models

The aim of what follows is to present a method to fix an acceptance/rejection threshold for the recognition of shape elements. The recognition problem is hard since sorting the shapes along a similarity measure to a target shape is not sufficient; we must answer by *yes* or *no* the question "does that shape element look like the target shape element?". Moreover, we would like to estimate the confidence in this answer. We shall first dress up an empirical statistical model of the shape elements database. The relevant matches will be detected *a contrario* as rare events for this *background model*. This detection framework has been recently applied by Desolneux *et al.* to the detection of alignments [DMM00] or contrasted edges [DMM01], by Almansa *et al.* to the detection of vanishing points [ADV03], by

Stival and Moisan to stereo images [MS04] , by Y. Gousseau to the comparison of image "composition" [Gou03] and by F. Cao to the detection of good continuations [Cao04]. The main advantage of this technique is that the only parameter that controls the detection is the Number of False Alarms, a quantity that has a handy meaning.

8.1.1 Shape model *versus* background model

Our aim is to compare a given target shape element $\mathcal{S}$ with the N shape elements of a database $\mathcal{B}$. Since we tackle the general shape matching problem, we suppose that we have no information but the observation of the target shape elements and the database of shape elements, and the value of a distance function between these shape elements. Of course, classifying the matches along this distance is always possible. Nevertheless, we are interested in deciding if either a shape element $\mathcal{S}'$, belonging to the database, looks like the shape element $\mathcal{S}$ or it does not. A straightforward decision is to fix a threshold upon the distances under which the answer is *yes, $\mathcal{S}'$ looks like $\mathcal{S}$*, and *no, $\mathcal{S}'$ does not look like $\mathcal{S}$* otherwise. In that case, the problem consists in automatically setting the threshold δ, and this is precisely the aim of the proposed methodology. To be more precise, let us assume that each shape element is represented by a set of K features $x_1, x_2, \ldots, x_K$, each of them belonging to a metric space (E_i, d_i) $(i \in \{1, \ldots, K\})$. What we mean by "distance between shape elements" is the product distance over $E_1 \times E_2 \times \cdots \times E_K$:

$$d(\mathcal{S}, \mathcal{S}') = \max_{i \in \{1,\ldots,K\}} d_i(x_i(\mathcal{S}), x_i(\mathcal{S}')).$$

The real observation is in fact made of the K distances between features $d_i(x_i(\mathcal{S}), x_i(\mathcal{S}'))$.
We assume no other information but the observed set of features. This means in particular that we have no model for the features, since having such a model would imply an extra knowledge (for instance some "expert" should have first built up the models). We are therefore unable to compute the probability that "a shape element is near $\mathcal{S}$ *because it has been generated by the shape model of* $\mathcal{S}$". We are interested in shape elements which are close to the target shape element $\mathcal{S}$ because their generation shares some common cause with the generation of $\mathcal{S}$. But what is the underlying common cause? We probably do not know, and this is the point. Indeed, directly addressing this problem is not possible, unless we have the exact model of $\mathcal{S}$. We are therefore led to wonder whether a database shape element is near $\mathcal{S}$ just "by chance", and to detect correspondences as unexpected coincidences. In order to address this latest point, we have to build up a *background model*: a model to compute the probability that a shape element is near $\mathcal{S}$ *by chance*.
Here are the assumptions for this model.

(A1) the functions $E_i \to \mathbb{R}$, $y \mapsto d_i(x_i(\mathcal{S}), y)$ $(i \in \{1, \ldots, K\})$, considered as random variables, are mutually statistically independent;

(A2) for each $i \in \{1, \ldots, K\}$, the probability $P_i(\mathcal{S}, \delta) := \Pr(y \in E_i, \text{ s.t. } d_i(y, x_i(\mathcal{S})) \leqslant \delta)$ is empirically estimated over the database (for each i, one computes the distribution function of

$d_i(z, x_i)$ when z spans the i^{th} feature of the shape elements in the database), that is to say:

$$P_i(\mathcal{S}, \delta) = \frac{1}{N} \cdot \#\{\mathcal{S}' \in \mathcal{B}, d_i(x_i(\mathcal{S}'), x_i(\mathcal{S})) \leqslant \delta\},$$

where $\#\cdot$ denotes the cardinality of any finite set (and N is the cardinality of the database $\mathcal{B}$).

If we make the (informal) additional assumption that the target shape element $\mathcal{S}$ has no "deterministic" reason to look like the shape elements in the database, then $P_i(\mathcal{S}, \delta)$ can actually be seen as the probability that a shape feature $x_i(\mathcal{S}')$ is at a distance lower than δ to $x_i(\mathcal{S})$ "just by chance". The independence assumption consequently ensures that the probability that a shape element lies "just by chance" at a distance lower than δ to $\mathcal{S}$ is $\prod_{i=1}^{K} P_i(\mathcal{S}, \delta)$. The following sections make all this rigorous and hopefully very clear.

8.1.2 Decision as hypothesis testing

A distance between shape elements being defined, deciding whether a shape element matches another shape element or not consists in setting a threshold δ over the distances. Ideally, δ should be set automatically, without any user tuning. We propose to use the hypothesis testing framework [DK82, Sil75] in order to replace the distance bound by a probability of false alarms bound, which is much more intuitive and handy.

The hypothesis we would like to test is $\mathcal{H}_0$: "A shape element is close to $\mathcal{S}$ because its generation shares some common cause with the generation of $\mathcal{S}$". However, handling this hypothesis with our assumption (no available model for the target shape element $\mathcal{S}$) is simply impossible. We are therefore led to concentrate on the alternative hypothesis $\mathcal{H}_1$: "A shape element is near $\mathcal{S}$ just by chance" ("just by chance" means that observing a shape at a distance d to $\mathcal{S}$ is predicted by the background model).

Since the only information we have is the distance d between an observed shape element and $\mathcal{S}$, the decision rule will consist in accepting the null hypothesis $\mathcal{H}_0$ if the distance is lower than a predetermined value δ, and rejecting it otherwise. The set of the shape elements that are compared to $\mathcal{S}$ (the whole database) is split into two subsets: $\Omega_0(\delta)$ and $\Omega_1(\delta)$, respectively made of the shape elements whose distance to $\mathcal{S}$ is lower than δ (and for which hypothesis $\mathcal{H}_0$ is accepted), and of those for which the distance to $\mathcal{S}$ is greater than δ (for which hypothesis $\mathcal{H}_0$ is rejected).

The quality of a statistical test is measured by the probability of taking wrong decisions. Two kinds of errors are possible: reject $\mathcal{H}_0$ for an observation $\mathcal{S}$ for which it is actually true (type I error, mis-detection), and accept $\mathcal{H}_0$ for $\mathcal{S}$ although it is false (type II error, false positive). A probability measure can be associated to each type of error. In our framework, each of these probabilities is related to the distance threshold δ.

Let us denote by $\mathcal{L}_0(\mathcal{S})$ and $\mathcal{L}_1(\mathcal{S})$ the likelihoods of an observation $\mathcal{S}$ under hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$, respectively. Then, we define the values

$$\alpha = \int_{\Omega_0} \mathcal{L}_1(\mathcal{S}) d\mathcal{S}\,,$$

$$\alpha_2 = \int_{\Omega_1} \mathcal{L}_0(\mathcal{S}) d\mathcal{S} \,,$$

and:

$$\beta = 1 - \alpha_2 = \int_{\Omega_0} \mathcal{L}_0(\mathcal{S}) d\mathcal{S} \,;$$

α (associated with type I error) is the *probability of false alarm*, and β is the *power function* of the test (α_2 is the *probability of non-detection* or *probability of a miss*, associated with type II error).
It is clear that the lower is α and the larger is β, the better is the test, but it is also clear that α and β cannot be optimized independently. Classically, *ROC* curves (Receiver Operating Characteristic curves) representing $\beta = f(\alpha)$ are associated with a statistical test $\mathcal{T}$. Robust tests show characteristic *ROC* curves looking like a Heaviside step: if α is close to 0, β should be close to 1.
The problem is to fit a trade-off between α and β (or equivalently α_2). Let us analyze two widely used techniques for doing that.

1. **Likelihood ratio test**. If we look for powerful tests (*i.e.* tests with the lowest rate of non-detection) among the tests whose probability of false alarm α is bounded (by a user defined threshold α^*), Neyman-Pearson lemma ensures that the most powerful test is the following likelihood ratio test:

   Classify the observation $\mathcal{S}$ in Ω_0 if $\dfrac{\mathcal{L}_1(\mathcal{S})}{\mathcal{L}_0(\mathcal{S})} < h$ and in Ω_1 otherwise, where the positive real number h is solution of :

   $$\int_{\left\{\mathcal{S}\in\Omega, \frac{\mathcal{L}_1(\mathcal{S})}{\mathcal{L}_0(\mathcal{S})} \geqslant h\right\}} \mathcal{L}_0(\mathcal{S}) d\mathcal{S} = \alpha^*.$$

   In order to achieve this test, the two likelihood functions are needed. Moreover, the value of α^* has to be fixed by the user, and its value has a strong influence on the results.

2. **Bayesian test**. A test $\mathcal{T}$ being given, it is also possible to model the trade-off between α and α_2 by a weighted sum (Bayes cost): $J(\mathcal{T}) = p_0\alpha + p_1\alpha_2$, where p_0 (resp. p_1) is the prior probability of hypothesis $\mathcal{H}_0$ (resp. of counter-hypothesis $\mathcal{H}_1$) (p_0 and p_1 verify $p_0 + p_1 = 1$). It can be shown that the classification test that minimizes J is:

   Classify the observation $\mathcal{S}$ in Ω_0 if $\mathcal{L}_0(\mathcal{S}) \cdot p_0 > \mathcal{L}_1(\mathcal{S}) \cdot p_1$ and in Ω_1 otherwise.

   Hence, the Bayesian test requires not only knowing the likelihoods but also the prior probabilities of the hypotheses. Compared to the likelihood ratio test, there is no need for an "arbitrary" threshold over the false alarm rate.

In fact, if we write the Bayesian inequality as:

$$\frac{\mathcal{L}_1(\mathcal{S})}{\mathcal{L}_0(\mathcal{S})} < \frac{p_0}{p_1} \,,$$

then it is clear that the ratio $\frac{p_0}{p_1}$ essentially plays the same role as the parameter $h(\alpha^*)$ in the Neyman-Pearson theory. "Bayesians" such as Jaynes [Jay03] argue that each test can be explained as a Bayesian

test somehow or other. Let us quote Grenander [Gre93]: "*Suffice is to say that when the notion of a prior makes sense and when there is sufficient knowledge about this prior we cannot afford to throw away this subject matter information: a Bayesian treatment is called for.*"

However, the practical limits of this theoretical framework are obvious. Assuming the knowledge of the likelihood of both the hypothesis ($\mathcal{L}_0(\mathcal{S})$) and the counter-hypothesis ($\mathcal{L}_1(\mathcal{S})$) is in general unrealistic in detection problems. Indeed, in order to compute the likelihood of hypothesis $\mathcal{H}_0$, a generative model is needed. Moreover, the Bayesian approach needs for prior information. Nevertheless, while choosing the ratio $\frac{p_0}{p_1}$ seems more satisfying than fixing α^*, priors remain spoilt by arbitrariness, or are strongly related to a specific problem for which supplementary information is provided.

Let us summarize the situation. Since we are not in position to compute the probability of non-detection $\Pr(\mathcal{S}' \in \Omega_1(\delta)|\mathcal{H}_0)$ (recall that this is one of the two classification errors: reject $\mathcal{H}_0$ although this hypothesis is true), neither the likelihood ratio test nor the Bayesian test can be performed. On the other hand, a straightforward computation provides the value of the probability of false alarms denoted by

$$\text{PFA}(\mathcal{S}, \delta) := \Pr(\mathcal{S}' \in \Omega_0(\delta)|\mathcal{H}_1)$$

(this is the second kind of error of the test: accept $\mathcal{H}_0$ although it is false). Since $\mathcal{S}' \in \Omega_0(\delta)$ if and only if $d(\mathcal{S}, \mathcal{S}') \leqslant \delta$, it follows that

$$\begin{aligned}\text{PFA}(\mathcal{S}, \delta) &= \Pr\left(d(\mathcal{S}, \mathcal{S}') \leqslant \delta|\mathcal{H}_1\right) \\ &= \Pr\left(\max_{i\in\{1,\dots,K\}} d_i(x_i(\mathcal{S}), x_i(\mathcal{S}')) \leqslant \delta|\mathcal{H}_1\right).\end{aligned}$$

Then, since hypothesis $\mathcal{H}_1$ is governed by the background model, assumptions **(A1)** and **(A2)** yield:

$$\begin{aligned}\text{PFA}(\mathcal{S}, \delta) &= \prod_{i\in\{1,\dots,K\}} \Pr\left(y \in E_i, \text{ s.t. } d_i(x_i(\mathcal{S}), y) \leqslant \delta\right) \\ &= \prod_{i\in\{1,\dots,K\}} P_i(\mathcal{S}, \delta).\end{aligned} \tag{8.1}$$

Let us write the likelihood of $\mathcal{H}_1$ in the classical hypothesis testing framework. Under the background model hypothesis, the likelihood of $\mathcal{H}_1$ over the set of observed distances is:

$$\mathcal{L}_1(d_1, d_2, \dots, d_K) = \prod_{i=1}^{K} f_i(\mathcal{S}, d_i),$$

where the density laws $d_i \mapsto f_i(\mathcal{S}, d_i)$ are indirectly estimated over the database (assumption **(A2)**). Indeed, with the preceding notations we have: $P_i(\mathcal{S}, \delta) = \int_0^\delta f_i(\mathcal{S}, x)dx$ for each $\delta > 0$, and these probabilities are estimated over the database. The probability of false alarms of the statistical test can be written as:

$$\begin{aligned}\text{PFA}(\mathcal{S},\delta) &= P(\mathcal{S}' \text{ s.t. } d(\mathcal{S},\mathcal{S}') \leqslant \delta | \mathcal{H}_1) \\ &= \int_{\Omega_0(\delta)} \mathcal{L}_1(d_1, d_2, \ldots, d_K) \mathrm{d}d_1 \mathrm{d}d_2 \ldots \mathrm{d}d_K \\ &= \prod_{i=1}^{K} \int_0^{\delta} f_i(\mathcal{S}, d_i) \mathrm{d}d_i \\ &= \prod_{i=1}^{K} P_i(\mathcal{S}, \delta).\end{aligned}$$

The second to last equation stands because of the expression of the likelihood $\mathcal{L}_1$, and because of the fact that $\Omega_0(\delta)$ is the hypercube $[0,\delta]^K$.
Classical hypothesis testing theory consists in minimizing both probability of false alarms and probability of non detection. Since we have no model for $\mathcal{H}_0$, we cannot evaluate here the probability of non detection. Nevertheless, evaluating the probability of false alarms is enough to make decisions. Indeed, the probability of false alarms $P(d < \delta | \mathcal{H}_1)$ being non-decreasing with δ, an upper bound p on this quantity provides immediately an upper bound on the distances:

$$\delta^*(p) = \max\{\delta > 0, P(d \leqslant \delta | \mathcal{H}_1) < p\}.$$

Consequently, if the test is to accept $\mathcal{H}_0$ if the observed distance is below $\delta^*(p)$, and to reject this hypothesis otherwise, then the associated probability of false alarms is bounded by p. This rule is said to be an *a contrario* decision since we accept the null hypothesis as soon as the alternative hypothesis is not likely to be valid (*i.e.* the probability of false alarms of the statistical test is low). Applied here to the shape recognition problem, we accept the hypothesis "a database shape element $\mathcal{S}'$ matches the target shape element $\mathcal{S}$" as soon as it is not likely that $\mathcal{S}'$ is near $\mathcal{S}$ "by chance". Notice that, according to this decision, all we are saying is that, under the background model, such a coincidence is so astonishing that there must be a better explanation than randomness. We are by no means asserting that this better explanation is "matched shape elements correspond to instances of the same object", though this might be the cause, among other possibilities. Experiments (see Chapter 9) indeed show matched shape elements that are actually alike, but do not correspond to the same object.

8.1.3 Number of false alarms and meaningful matches

The *a contrario* decision consists in fixing a threshold over the probability of false alarms rather than over the distance between shape elements. Since a probability has little meaning *per se*, we now introduce the *number of false alarms*. Let us recall that the database is made of N shape elements to which the target shape element $\mathcal{S}$ is compared.

DEFINITION 8.1 *The* Number of False Alarms *of the shape element $\mathcal{S}$ at a distance d is:*

$$NFA(\mathcal{S}, d) := N \cdot \prod_{i \in \{1,\ldots,K\}} P_i(\mathcal{S}, d).$$

Since the latest product of probabilities is the probability of false alarms when testing if the database shape elements are at a distance lower than d to $\mathcal{S}$, the number of false alarms can be seen as the average number of false alarms that are expected when we test if the distance from each shape element in the database to $\mathcal{S}$ is below d. Instead of bounding the probability of false alarms in order to deduce a distance threshold, we bound the number of false alarms.

DEFINITION 8.2 *The number of false alarms of the target shape element $\mathcal{S}$ and a database shape element $\mathcal{S}'$ is the number of false alarms of $\mathcal{S}$ at a distance $d(\mathcal{S}, \mathcal{S}')$:*

$$NFA(\mathcal{S}, \mathcal{S}') := NFA(\mathcal{S}, d(\mathcal{S}, \mathcal{S}')).$$

For the sake of simplicity, the same notation is used for both preceding definitions of the number of false alarms. Let us remark that the arguments of the NFA seen as a two variables function do not play a symmetric role.

DEFINITION 8.3 *A shape element $\mathcal{S}'$ is an* ε-meaningful match *of the target shape element $\mathcal{S}$ if their number of false alarm is bounded by ε:*

$$NFA(\mathcal{S}, \mathcal{S}') \leqslant \varepsilon.$$

Notice that since the functions $P_i(\mathcal{S}, d) : d \mapsto \Pr(y \in E_i \text{ s.t. } d_i(x_i(\mathcal{S}), y) \leqslant d)$ are non-decreasing, the function $NFA(\mathcal{S}, d) := N \cdot \prod_{i \in \{1,\dots,K\}} P_i(\mathcal{S}, d)$ is pseudo-invertible with respect to d. That is, there exist a unique positive real number $d^*(\varepsilon/N)$ (depending also on $\mathcal{S}$) such that

$$d^*(\varepsilon/N) := \max\{d > 0, \text{PFA}(\mathcal{S}, d) \leqslant \varepsilon/N\}.$$

The proposition that follows is then straightforward.

PROPOSITION 8.1 *A shape element $\mathcal{S}'$ is an ε-meaningful match of a shape element $\mathcal{S}$ if and only if $d(\mathcal{S}, \mathcal{S}') \leqslant d^*(\varepsilon/N)$.*

The ε-meaningful matches of $\mathcal{S}$ are then those shape elements for which the distance to $\mathcal{S}$ is below $d^*(\varepsilon/N)$ (consequently, the probability of false alarms of the associated test is less than ε/N). We therefore expect on the average less than ε false alarms among all ε-meaningful matches over all the tested shape elements. In other words, if all shape elements in the database were generated by the background model, then the hypothesis $\mathcal{H}_0$ should never be accepted; all ε-meaningful detections should thus be considered as false alarms. The following proposition makes this claim more formal.

PROPOSITION 8.2 *Under the assumption that the shape elements are generated by the background model, the expectation of the number of ε-meaningful matches over the set of all shape elements in the database is less than ε.*

Proof: Let $\mathcal{S}'_j$ $(1 \leqslant j \leqslant N)$ denote the shape elements in the database, assumed to be generated according to the background model, and let χ_j be the indicator function of the event e_j: "$\mathcal{S}'_j$ is an ε-meaningful match of $\mathcal{S}$." Let $R = \sum_{j=1}^{N} \chi_j$ be the random variable representing the number of shape elements matching ε-meaningfully $\mathcal{S}$. The expectation of R is $E(R) = \sum_{j=1}^{N} E(\chi_j)$. Using Proposition 8.1, it follows that $E(R) = \sum_{j=1}^{N} \text{PFA}(\mathcal{S}, d^*(\varepsilon/N))$, so $E(R) \leqslant \sum_{j=1}^{N} \varepsilon \cdot N^{-1}$, yielding $E(R) \leqslant \varepsilon$. ∎

The key point is that we control the expectation of R. Since dependencies between events e_j are unknown, we are not able to estimate the probability law of R. Nevertheless, the linearity still allows to compute the expectation.

Notice that the empirical probabilities take into account the 'rareness' or 'commonness' of a possible match; indeed the threshold d^* is less restrictive in the first case and stricter in the other one. This point is discussed and illustrated in Chapter 9, section 9.3.

The advantages of the *a contrario* decision based on the NFA compared to directly setting a distance threshold between shape elements are obvious. On one hand, thresholding the NFA is much more handy than thresholding the distance. We indeed simply put $\varepsilon = 1$ (we simply refer to 1-meaningful matches as "meaningful matches"), or $\varepsilon = 10^{-1}$ if we want to impose a higher confidence in the obtained matches. The detection threshold ε is set uniformly whatever may be the target shape element and the database: the resulting distance threshold adapts according to them. On the other hand, the lower ε, the "surer" the ε-meaningful detections are. Of course, the same claim is true when considering distances: the lower the distance threshold δ, the surer are the matches distance lower than δ to $\mathcal{S}$, but considering the NFA quantifies this confidence. In practice, the number of false alarms between two similar shape elements can be as low as 10^{-10}. This means that we need to observe a database 10^{10} times larger in order that a meaningful match at the same distance ought to be a false alarm. In other words, the corresponding match would still be a meaningful match in a database 10^{10} times larger. What we have proposed is an automatic decision rule that ensures the number of false alarms to be very low.

Let us end up with the definition of the number of false alarms when comparing all shape elements in a database to all shape elements in another database, and not only a single shape element to a database. When searching the shape elements belonging to a database $\mathcal{B}_1$, made of N_1 shape elements, among the N_2 shape elements belonging to a database $\mathcal{B}_2$, we define:

DEFINITION 8.4 *The* Number of False Alarms *of a shape $\mathcal{S}$ (belonging to $\mathcal{B}_1$) at a distance d is:*

$$NFA(\mathcal{S}, d) = N_1 \cdot N_2 \cdot \Pr(\mathcal{S}', \max_{i \in \{1 \ldots K\}} d_i(x_i(\mathcal{S}), x_i(\mathcal{S}')) \leqslant d).$$

The probabilities (depending on the searched shape element $\mathcal{S}$) are estimated as before, as a product of K empirical estimates. For each shape element in $\mathcal{B}_1$ we also define ε-meaningful matches. The claim up to which we shall expect on the average ε false alarms among the ε-meaningful matches over all $N_1 \cdot N_2$ tested pairs of shape elements still holds.

8.1.4 Building statistically independent features

Now, why considering independent features is so important? The reason is the following one: using independent features is a way to beat the *curse of dimensionality* [HTF01]. By combining a few independent features, we can easily reach very low numbers of false alarms without needing huge databases to estimate the probability of false alarms. In his pioneer work, D. Lowe [Low85] presents this same viewpoint for visual recognition: "*Due to limits in the accuracy of image measurements (and possibly also the lack of precise relations in the natural world) the simple relations that have been described often fail to generate the very low probabilities of accidental occurrence that would make them strong sources of evidence for recognition. However, these useful unambiguous results can often arise as a result of combining tentatively-formed relations to create new compound relations that have much lower probabilities of accidental occurrence*".
Let us give a numerical example. If the considered database is made of N shape elements, the lowest value reachable by each empirical probability

$$P_i(\mathcal{S}, d) = \frac{1}{N} \cdot \#\{\mathcal{S}' \in \mathcal{B}, d_i(x_i(\mathcal{S}'), x_i(\mathcal{S})) \leqslant d\}$$

is $1/N$. Consequently, if the background model is built on $K = 1$ feature, and the database is made of $N = 1000$ shape elements, then the lowest reachable number of false alarms would be $1000 \cdot 1/1000 = 1$. This means that even if two shape elements $\mathcal{S}$ and $\mathcal{S}'$ are almost identical, based on the NFA we cannot ensure that this match is not casual. Indeed, an NFA equal to 1 means that, on the average, one of the shape elements in the database can match $\mathcal{S}$ by chance. Assume now that the background model is built on $K = 6$ features (and still $N = 1000$), then the lowest reachable number of false alarms would be $1000 \cdot 1/1000^6 = 10^{-15}$. This means that we are able to observe ε-meaningful matches with ε as low as 10^{-15}.

To sum up, shape features have to meet the three following requirements:

1) Features provide a complete description: two shape elements with the same features are alike.

2) Features are mutually statistically independent (more precisely speaking, distances between features are independent).

3) Their number is as large as possible.

The first requirement means that the features describe shape elements well, the second one is imposed in order to design the background model, and the third requirement is needed in order to be able to reach low number of false alarms. Finding features that meet these three requirements together is a hard problem. Indeed, there must be enough features in order that the first requirement is valid, but not too many otherwise the second requirement falls.

The decision framework we have been describing up to here is actually completely general, in the sense it can be applied to find correspondences between any kind of structures for which K independent features can be extracted. In what follows, we will concentrate on the problem of extracting

independent features from *shape elements*. Shape elements are normalized before comparison in order to meet the geometric invariance requirement of recognition (see Chapters 1 and 7); therefore, we will more specifically deal with *normalized shape elements*.

Since in order to meet the geometric invariance requirement of recognition, shape elements are normalized before comparison, we will deal more specifically with *normalized shape elements*.

Semi-local encoding

We consider here the semi-local encoding algorithm described in Chapter 7. Recall that a normalized shape element is in fact a piece of Jordan curve, normalized in a local frame built on a bitangent or on a flat piece. We empirically found that the best trade-off achieving simultaneously the three feature requirements is the following (see figure 8.1 for an illustration). Each normalized representation C is split into five pieces of equal length. Each one of these pieces is normalized by mapping the chord between its first and last points on the horizontal axis, the first point being at the origin: the resulting 'normalized small pieces of curve' are five features $C_1, C_2, \ldots, C_5$. These features ought to be independent; nevertheless, $C_1, \ldots, C_5$ being given, it is impossible to reconstruct the shape element they come from. For the sake of completeness a sixth, global, feature C_6 is therefore made of the endpoints of the five previous pieces, in the normalized frame. For each piece of level line, the shape features introduced in section 8.1.1 are made of these six 'generic' shape codes $C_1, \ldots, C_6$. Using the notations introduced in the previous sections, we have $x_i(\mathcal{S}) = C_i$, $i \in \{1, \ldots, 6\}$; the distances d_i between them are L^∞-distances.

Another possibility that we have investigated is to use the principal component analysis (PCA) [MSM03]. Although PCA does not provide independent features but uncorrelated ones, the approximation does not seem to be critical. We show experiments in Chapter 10.

Global encoding

We have also proposed in Chapter 7 a global curve normalization. The *a contrario* decision is still valid, considering these normalized curves as shape elements, and building the features in a similar way as for the semi-local encoding. Precisely speaking, each normalized piece of curve is split into six pieces. The starting point was defined in Chapter 7 as the nearest point to the barycenter intersecting the vertical line to the bottom, with a positive ordinate. In the same way as for semi-local encoding, each of these pieces is normalized by mapping the chord between its first and last points on the horizontal axis, the first point being at the origin: these resulting 'normalized pieces of curve' are six features $C_1, C_2, \ldots, C_6$. For the sake of completeness, a seventh global feature C_7 is made of the endpoints of the six previous pieces. The features are made of the $C_1, \ldots, C_7$. The distances d_i between them are still L^∞-distances.

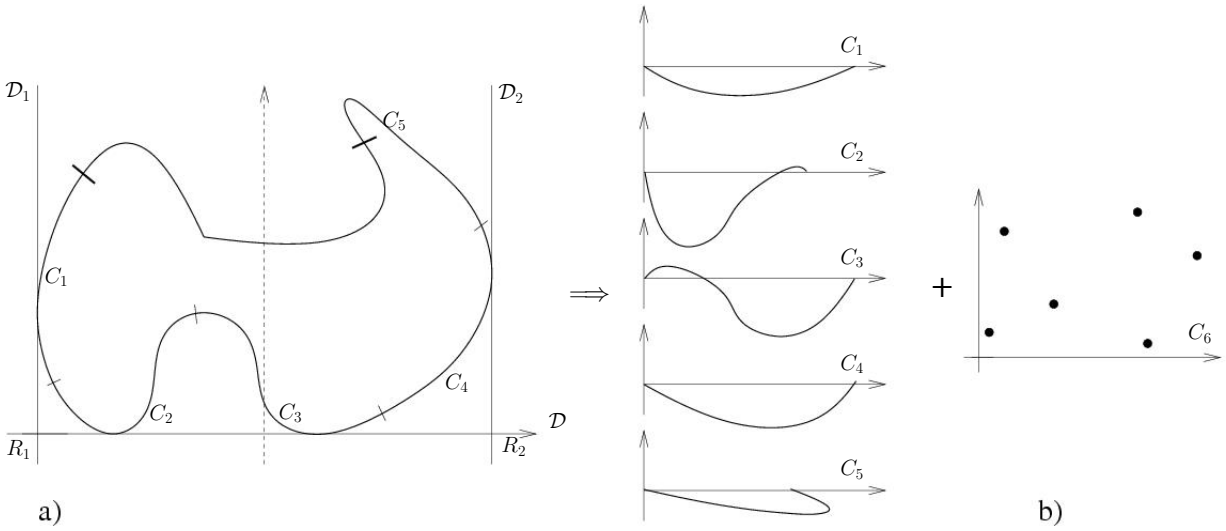


Figure 8.1: *Semi-local encoding procedure. Example of a similarity-invariant encoding. Sketch (a): original shape element in a normalized frame based on a bitangent line. Both ends of the piece of the shape element, of length $F \cdot \|R_1R_2\|$, are marked with bold lines: this representation is split into 5 pieces C_1, C_2, C_3, C_4, and C_5. Sketch (b): each of them is normalized, and a sixth feature C_6 made of the endpoints of these pieces is also built.*

8.2 Testing the background model

The computation of the probability $\text{PFA}(\mathcal{S}, \delta)$ that a shape element could be just by chance at a distance lower than δ to $\mathcal{S}$ is correct under the independence assumption on the pieces of codes (formula 8.1). Of course, the degree of trust that we are able to give to the associated Number of False Alarms $NFA(\mathcal{S}, \delta)$ (Definitions 8.1 and 8.4) strongly depends on the validity of this independence assumption. Before applying this methodology to realistic applications, we have to test the independence of the pieces of codes, in order to ensure the correctness of the methodology. This is the aim of what follows. Although a decision rule is proposed for both global and semi-local shape elements matching, we only give the results of the tests for semi-local encoding. We show here that the pieces of codes obtained by the proposed normalization (section 8.1.4) are *not* independent (section 8.2.1), and that some dependence is introduced by the non-self intersection constraint of level lines and (mainly) by the normalization procedure (section 8.2.2). Nevertheless, experiments point out that detection under Helmholtz principle (*i.e.* a meaningful match is a match that is not likely to occur in a noise image) is fully satisfying (section 8.2.2).

8.2.1 Independence testing

In order to compute the probability $\text{PFA}(\mathcal{S}, \delta)$, the mutual independence of the 'pieces of codes' is needed. More precisely speaking, a code made of pieces x_i being given, the binary random variables $y \mapsto d_i(x_i, y) \leqslant \delta$ are supposed mutually independent.

We cannot estimate the joint probability

$$\Pr\left((y_1,\dots,y_n)\in E_1\times\cdots\times E_n \text{ s.t. } d_1(x_1,y_1)\leqslant\delta,\dots,d_n(x_n,y_n)\leqslant\delta\right)$$

(estimating the law of this random vector would indeed require too many samples) and compare it to the product of the probabilities $\prod_{i=1}^{n}\Pr(y_i\in E_i \text{ s.t. } d_i(x_i,y)\leqslant\delta)$. On the other hand, it turns out that the joint probability associated to two pieces of codes can be accurately enough estimated. Thus, instead of testing the mutual independence of the pieces of codes, we merely test the independence pairwise.

Let us explain the Chi-square test framework, applied to independence testing [RH79]. Two binary random variables X and Y being given, let us denote $p_{ij}=P((X,Y)=(i,j))$ for i and j in $\{0,1\}$: these probabilities are empirically estimated over samples following the laws of X and Y. Thus, $P(X=i)=p_{i0}+p_{i1}$ and $P(Y=j)=p_{0j}+p_{1j}$. If independence assumption holds, we have: $p_{ij}=P(X=i)\cdot P(Y=j)$. The Chi-square statistical test consists in evaluating the difference between the expected number of samples such that $(X,Y)=(i,j)$ if this assumption were true, and the observed number of samples. If N is the number of samples following the law of (X,Y), and if O_{ij} is the observed number of samples (i,j), we compute:

$$\begin{aligned}\chi^2=&\frac{(N(p_{00}+p_{01})(p_{00}+p_{10})-O_{00})^2}{N(p_{00}+p_{01})(p_{00}+p_{10})}+\frac{(N(p_{10}+p_{11})(p_{10}+p_{00})-O_{10})^2}{N(p_{10}+p_{11})(p_{10}+p_{00})}\\&+\frac{(N(p_{01}+p_{00})(p_{01}+p_{11})-O_{01})^2}{N(p_{01}+p_{00})(p_{01}+p_{11})}+\frac{(N(p_{11}+p_{10})(p_{11}+p_{01})-O_{11})^2}{N(p_{11}+p_{10})(p_{11}+p_{01})}.\end{aligned}$$

This quantity can be assumed to follow a Chi-square distribution with one degree of freedom, if enough samples are provided in order to estimate accurately the probabilities p_{ij}, and the O_{ij}.

Of course, the lower is χ^2, the likelier we are to accept the hypothesis, and vice-versa. By comparing the obtained value with the quantiles of the Chi-square law, we are able to accept or reject the hypothesis (independence between the random variables), with a certain significance level.

We have led this experiment with the binary random variables associated to the codes that we introduced in what precedes, with different target codes and database codes. The results are clear: in all cases, we are able to reject the independence assumption with a high significance level. Nevertheless, the rejection is strong because the tested databases are very large: Chi-square test is all the more accurate (and so is the rejection confidence) as the number of samples is large. In other terms, a “slight" dependence with a large number of samples leads to a very significant rejection; this means that the Chi-square test does not yield an absolute measurement of how dependent or how independent variables are.

The next section shows that the independence assumption is true enough to keep the Helmholtz detection principle true, in a sense that will be made clear.

8.2.2 Checking the Helmholtz principle

We test here the main property of the proposed method, namely the control of the expected number of 'random' detections (the Number of False Alarms, Proposition 8.2). Number of False Alarms computation holds under the independence assumption. If this assumption is true, and if the database contains no copy of a sought code (*i.e.* the target shape element is not "generic" among the shape elements in the database), the expected number of false alarms among all ε-meaningful matches with the target shape element should be lower than ε. Nevertheless, we are not able to separate false alarms and real matches: we only observe detections. The Chi-square test proved that, strictly speaking, the independence assumption is not valid. Now, Helmholtz principle states that no detection in "noise" (which has to be precised) should be considered as relevant. All ε-meaningful matches in the noise should thus be considered as false alarms: in such a noise situation there should be on the average about ε many of them. The following experiments test this claim. We show that the NFA is a pretty good prediction of the number of detections. The independence assumption is enough valid, so that the claim according to which there is on average at most ε false alarms among ε-meaningful matches still holds.

As a first experiment we check the detection thresholds on a very simple model: we consider as code database and code query some random walks with independent increments. In this case the background model is ensured to be true, in the sense that the considered codes fit perfectly the independence assumption.

Table 8.1 shows that the Number of False Alarms is very accurately predicted for various database sizes: the number of detections with a NFA lower than ε is about ε indeed.

$100,000$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$
Numb. of det. with $NFA < \varepsilon$:	0	0	2.3	15.2	122.2	$1,075.5$	$9,872.2$
$50,000$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$
Numb. of det. with $NFA < \varepsilon$:	0.2	0.3	1.5	11.9	106.1	$1,001.1$	$9,789.5$
$10,000$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	
Numb. of det. with $NFA < \varepsilon$:	0	0	1.2	12.5	108.4	985.0	

Table 8.1: *Random walks. Average (over* 10 *samples) number of detections* vs ε. *Tabular 1: database of* $100,000$ *codes. Tabular 2: database of* $50,000$ *codes. Tabular 3: database of* $10,000$ *codes.*

Of course, modeling codes with random walks is not realistic. As proved in what precedes, distances between codes are actually *not* independent. In our opinion, the lack of independence comes from two points. On one hand, codes correspond to pieces of level lines, and consequently they are constrained not to self-intersect. On the other hand, codes are normalized, and show therefore structural similarities (for example, codes coming from bitangent points show mostly common structures). In order to quantify the 'amount of dependence' due to these two aspects, we have led the two following experiments.

Table 8.2 shows the number of detections *versus* the number of false alarms for databases made of pieces of level lines (*not* normalized, the codes are just made out of 45 consecutive points on pieces of level lines). Consequently, the obtained codes are constrained not to self-intersect. In this experiment, the independence can only spoiled by this property, not by the normalization. Although the Chi-square test shows that the codes are not independent, once again the number of detections is accurately predicted: the number of matches with a NFA less than ε is indeed about ε.

$101,438$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$
Numb. of det. with $NFA < \varepsilon$:	0.1	0.1	1.7	13.8	95.3	942.5	$9,789.4$
$50,681$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$
Numb. of det. with $NFA < \varepsilon$:	0	0	1.2	10.3	90.5	955.1	$9,859.3$

$9,853$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$
Numb. of det. with $NFA < \varepsilon$	0	0.1	0.9	9.5	94.3	973.1

Table 8.2: *Pieces of white noise level lines. Average (over* 10 *samples) number of detections* vs ε. *Tabular 1: database of* $101,438$ *codes. Tabular 2: database of* $50,681$ *codes. Tabular 3: database of* $9,853$ *codes.*

Let us consider databases made of normalized codes extracted from pieces of level lines in white noise images. Table 8.3 shows that the number of detections is still of the same magnitude as the number of false alarms ε, but is not as precisely predicted as in the latest experiments. Roughly speaking, it means that 'most of the dependence' comes from the normalization procedure, and not from the non-self-intersection constraint. Nevertheless, the order of magnitude is still correct, and does not depend on the size of the database. These properties are sufficient for setting the Number of False Alarms threshold under the Helmholtz principle. Following this method, a match is supposed to be highly relevant if it cannot happen in white noise images. According to table 8.3, matches with a NFA lower than 0.1 are ensured to be impossible in white noise images. If we want to ensure a strong confidence in the detected matches, we are thus led to consider 0.1-meaningful matches in realistic experiments (see Chapter 9, section 9.1).

$104,722$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$	$100,000$
Numb. of det. with $NFA < \varepsilon$:	0.3	1.5	6.5	31.5	173.9	$1,264.4$	$9,803.1$	$99,899.5$
$47,033$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$	
Numb. of det. with $NFA < \varepsilon$:	0.1	0.3	3.7	20.2	125.4	976.3	$9,854.2$	
$10,784$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$		
Numb. of det. with $NFA < \varepsilon$:	0	0.2	2.6	14.8	107.6	973.3		

Table 8.3: *Normalized pieces of white noise level lines. Average (over* 10 *samples) number of detections* vs ε. *Tabular 1: database of* $104,722$ *codes. Tabular 2: database of* $47,033$ *codes. Tabular 3: database of* $10,784$ *codes.*

As a last experiment, table 8.4 shows the number of detections *versus* number of false alarms for a database made of normalized long (length greater than 135 pixels) pieces of level lines from white

noise images. The results are not better than in the preceding experiment, we cannot assert that the independence violation is due to short pieces of level lines.

$101,743$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$	$100,000$
Numb. of det. with $NFA < \varepsilon$:	0	0.4	2.8	18.5	124.3	$1,123.2$	$9,693.8$	$99,921.0$

$51,785$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$
Numb. of det. with $NFA < \varepsilon$:	0	0.3	2.9	16.0	118.6	983.4	$9,800.4$

$11,837$ codes, value of ε:	0.01	0.1	1	10	100	$1,000$	$10,000$
Numb. of det. with $NFA < \varepsilon$:	0	0.2	1.4	12.3	105.9	975.2	$9,974.7$

Table 8.4: *Normalized long pieces of white noise level lines. Average (over* 10 *samples) number of detections* vs ε. *Tabular 1: database of* $101,743$ *codes. Tabular 2: database of* $51,785$ *codes. Tabular 3: database of* $11,387$ *codes.*

Chapter 9

Experiments on meaningful matches detection

Abstract: In this chapter, we present several experiments that illustrate and validate all the stages of the recognition methods that were presented in the previous chapters. Section 9.1 deals with the semi-local invariant recognition method. Both similarity and affine methods are considered, and a comparative study based on some examples is presented. Section 9.2 presents some examples of the recognition method based on the global comparison of meaningful boundaries. The similarity and the affine versions are also compared. Finally, section 9.3 illustrates a general property of the *a contrario* detection framework we propose: the recognition of shape elements is relative to the context.

Résumé : Dans ce chapitre, nous présentons plusieurs expériences qui illustrent et valident toutes les étapes des méthodes de reconnaissance, exposées dans les chapitres précédents. La section 9.1 traite de la méthode de reconnaissance semi-locale. Les méthodes invariantes par similitudes et par transformations affines sont considérées dans cette section, et comparées en se basant sur quelques exemples. La section 9.2 présente quelques exemples concernant la méthode de reconnaissance globale basée sur les frontières maximales significatives. Les versions similitude et affine sont également comparées. Finalement, la section 9.3 illustre une propriété générale du cadre de détection *a contrario* que nous proposons : la reconnaissance d'éléments de forme est relative au contexte.

9.1 Local meaningful matches

In this section we present several experiments that illustrate all the stages of the semi-local invariant recognition method, in particular the semi-local normalization procedures (Chapter 7) and the decision method (Chapter 8). Both similarity and affine versions are considered, and compared.

9.1.1 Toy example

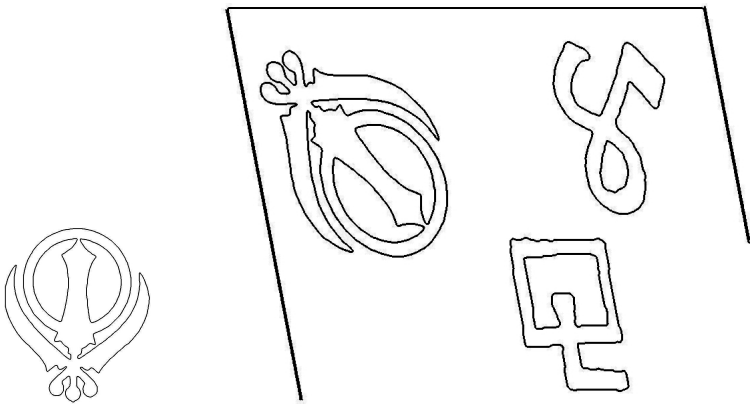
This first experiment compares the performance of the affine invariant and the similarity invariant recognition methods, on simple, synthetic images. A toy example was chosen here in order to illustrate all the stages in the considered recognition methods. Figure 9.1 shows the two synthetic images involved in the experiment. Shape elements from the image on the left (the "target" image) are searched in the right image (the "scene" image).

In the scene image we included an affine distorted version of the sketch in the target image. Shape elements were extracted by computing the maximal meaningful boundaries in the images (Figure 9.1(b)) using the algorithm described in Chapter 4, and smoothed using the affine curve shortening described in Chapter 5. Then the affine and the similarity semi-local invariant encoding algorithms, described in Chapter 7, were applied to the smoothed extracted boundaries. Finally, meaningful matches were detected in both cases, based on the *a contrario* method presented in Chapter 8.

Let us start by giving some precisions on the semi-local affine invariant recognition method, and by describing its results. In the target image, 44 shape elements were extracted from its meaningful boundaries. These shape elements were represented as affine normalized codes of 45 points, as explained in Chapter 7. The same encoding procedure, applied to the scene image, led to 105 normalized codes. Meaningful matches between these two sets of normalized codes were detected. Following the rationale for the meaningfulness computation presented in Chapter 8, a perfect match between codes would have reached a NFA of $44 \times 105/105^6 = 3.45\,10^{-9}$ (when the empirical distributions of distances to target codes are learned using only the considered scene image, as we do here). But perfect matching is impossible even in this condition, where we deal with synthetic images. This is due to the fact that the interpolation involved in the affine transformation of the image leads to boundaries that are not exactly the transformed boundaries of the original image. Another reason for that is that, as pointed out in Chapter 7, flat pieces are not affine invariant (they are not even similarity invariant), and their position may vary, particularly when dealing with curves showing relatively high curvature. This is exactly what we observe in this experiment. All 42 detected meaningful matches between shape elements for the affine invariant framework ($NFA < 1$) are shown (superimposed) in Figure 9.2(a). No false match was detected. The best match attains $NFA = 5.4\ 10^{-7}$, and the worst one has an NFA of $9.6\ 10^{-1}$. These two matches are displayed in Figure 9.3(a); the left most and middle images correspond respectively to the target and the scene shape elements, and the right most image shows their normalized codes in the normalized frame, superimposed. The shapes elements matching at $NFA = 9.6\ 10^{-1}$ do not correspond exactly to the same piece of curve, but they are still detected since they are relatively close. This kind of instability is not really a problem, since in general the encoding is redundant enough to capture better matches involving the same portions of the curve. This is illustrated in Figure 9.2(b), where almost all the same pieces of boundary showed in Figure 9.2(a) are still present for meaningfulness $\varepsilon < 10^{-2}$.

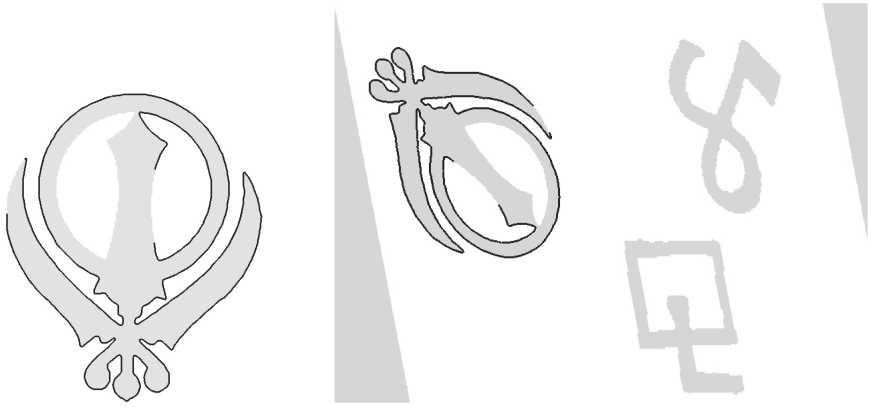
Finally, notice that one of the nested boundaries of the sketch does not present any matched shape

(a)

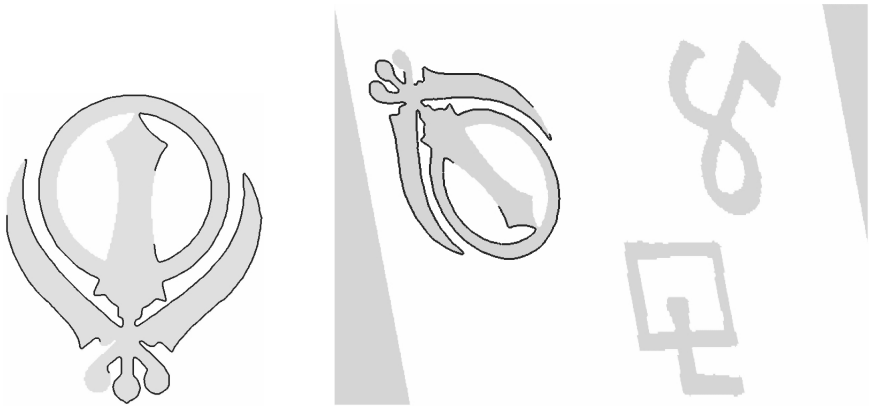


(b)

Figure 9.1: *Toy example. (a) Original images; the image on the right contains an affine distorted version of the sketch in the left image. (b) Corresponding maximal meaningful boundaries.*



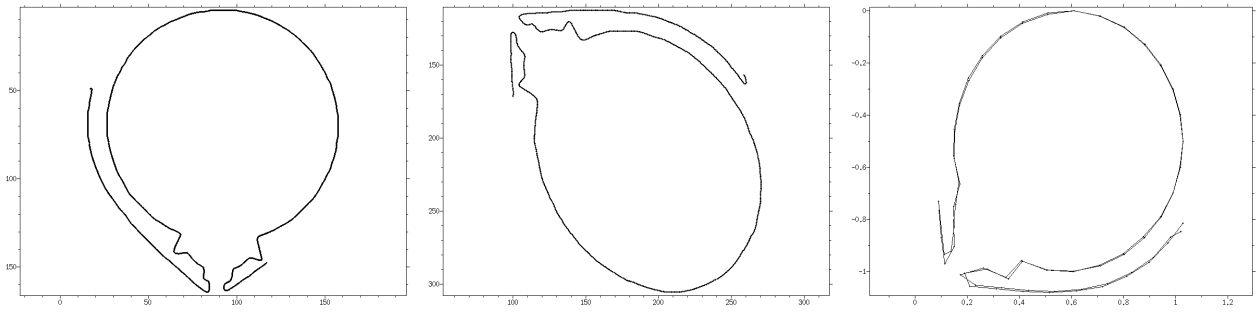
(a) $NFA < 1$



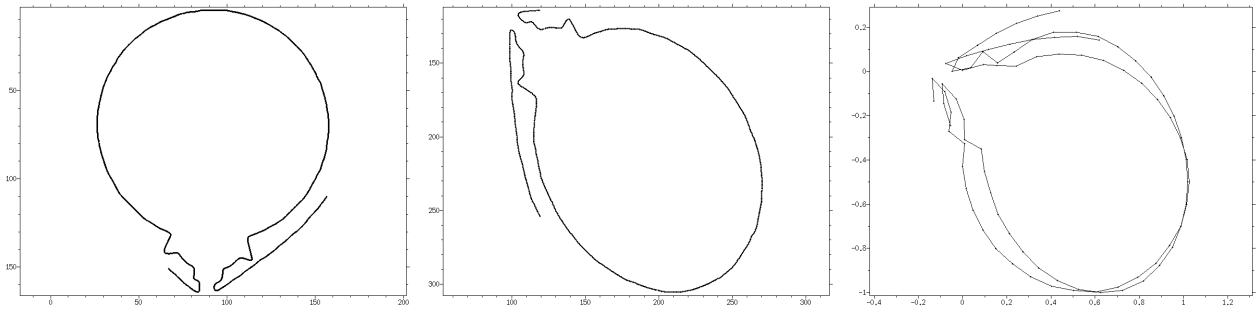
(b) $NFA < 10^{-2}$

Figure 9.2: *Affine invariant semi-local recognition: meaningful matches ($NFA < 1$) between shape elements. No false match was detected.*

element, while the other one (which is almost symmetric to it) does. This is due to the fact that, in the scene image, one of this nested boundaries presents a single flat piece leading to an "encodable" shape element (all the others bitangent lines or flat pieces do not because the curve is not long enough), while in the other one this flat piece is not detected. Once again, this is not really a problem, since these quasi-convex curves are encoded by the global method presented in Chapter 7, section 7.3.



(a) *Best match,* $NFA = 5.4\ 10^{-7}$



(b) *Worst match,* $NFA = 9.6\ 10^{-1}$

Figure 9.3: *Affine invariant semi-local recognition: the matches showing the lowest and the largest* NFA.

Let us now describe the second part of this experiment, which consists in applying the semi-local similarity invariant recognition method to the same target and scene images that were used in the fist part. We do not expect this method to do better than the previous one, since the common shape elements in the target and the scene images are related by an affine transform. However, we are interested in finding out if the semi-local similarity invariant method is able to retrieve some matches. In this second part of the experiment we follow the same stages than in the previous one, except for the normalization/encoding procedure, where the semi-local similarity invariant encoding method described in Chapter 7 was used. In the target image, 80 shape elements were extracted from its meaningful boundaries, and 127 for the scene image. Notice that the similarity invariant encoding is more redundant than the affine invariant encoding, since more shape elements are extracted in the latter case. The reason for that is simple: as we pointed out in Chapter 7 (section 7.2), the construction of our affine invariant semi-local frames imposes more constraints on the curve than the one for

similarity invariant frames. (These affine semi-local frames are also more global than similarity semi-local frames, what makes them less robust to occlusion). Perfect matches in this second part of the experiment should reach numbers of false alarms as low as $88 \times 127/127^6 = 2.66\,10^{-9}$. Here perfect matches cannot happen, mainly because boundaries are not related by similarity transforms. All 44 detected meaningful matches between shape elements ($NFA < 1$) for the similarity semi-local invariant recognition method are shown, superimposed, in Figure 9.4. In Figure 9.5 we display the shape elements and the normalized codes for the largest and the lowest NFA ($2.5\ 10^{-5}$ and $7.1\ 10^{-1}$), as well as another example of matched shape elements.

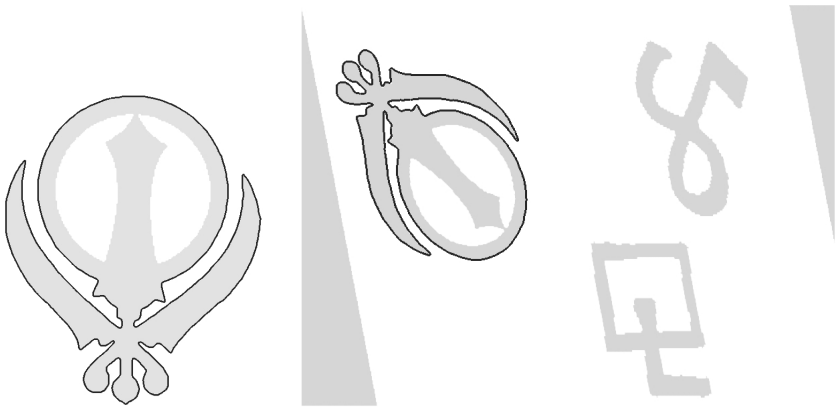
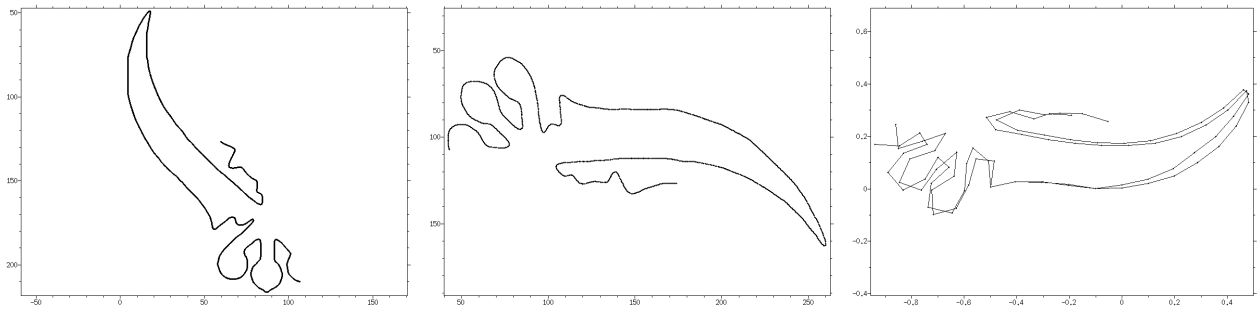


Figure 9.4: *Similarity invariant semi-local recognition: meaningful matches ($NFA < 1$) between shape elements. No false match was detected.*

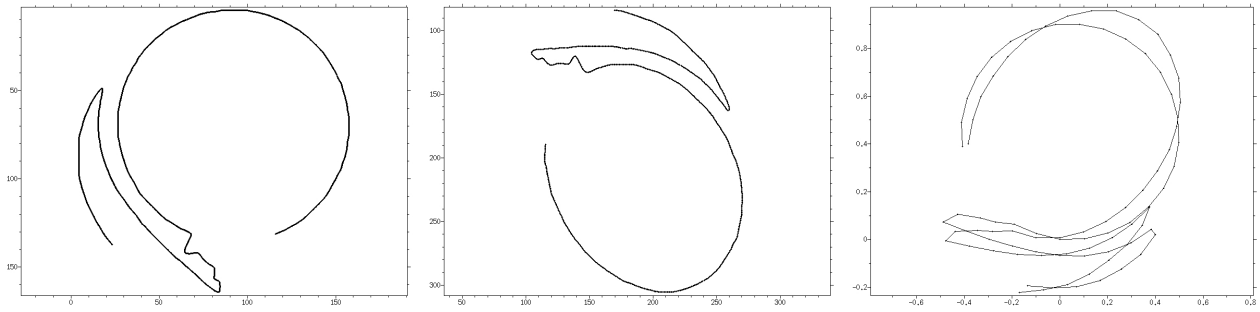
As we can see from the superimposed normalized codes, these codes are not as close as for the affine encoding. However, just looking at shape elements in Figures 9.5(a) and 9.5(c) is enough to see that, even if the target and the scene images are related by an affine transform presenting non isotropic zooms and a considerable shear, almost the entire shape (except for the nested boundaries, which are "too convex" to be encoded by the semi-local method) can be recognized with a relatively high degree of confidence.

Part of the discussion presented in this section can be summarized in Figure 9.6. The list of meaningful matches is ordered from best (lowest NFA) to worst (largest NFA), and the index i of this sorted list is plotted *versus* $-\log_{10}(NFA_i)$, where NFA_i is the NFA of the i-th best match. Such a function is plotted for the similarity and for the affine matches. The affine semi-local invariant matches reach lower NFA. Notice that in both affine and similarity invariant recognition methods, there are several matches that show small NFA, leading to sure detections of common shapes.

(a) *Best match,* $NFA = 2.5\ 10^{-5}$



(b) *Worst match,* $NFA = 7.1\ 10^{-1}$



(c) *Another example,* $NFA = 2.5\ 10^{-4}$

Figure 9.5: *Similarity invariant semi-local recognition: the matches showing the lowest and the largest* NFA.

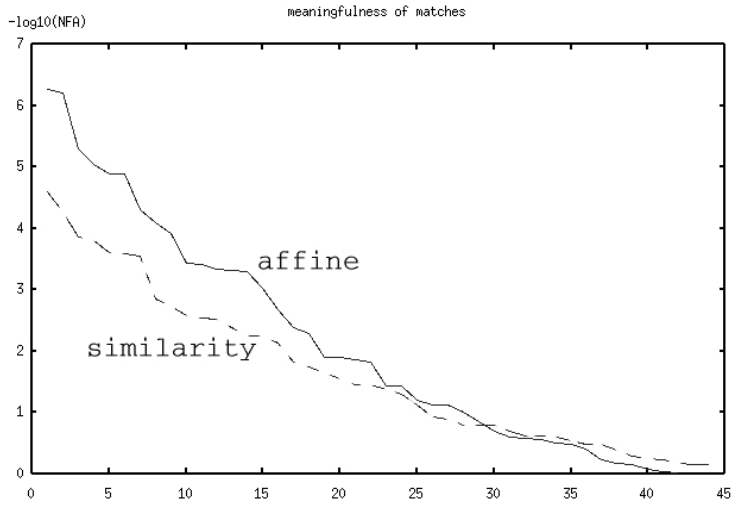


Figure 9.6: *NFA of affine and similarity semi-local invariant matches for the toy example. Both lists of meaningful matches are ordered from best (lowest NFA) to worst (largest NFA), and for each list, the index i of the sorted list is plotted* versus $-\log_{10}(NFA_i)$*, where NFA_i is the NFA of the i-th best match.*

9.1.2 Perspective distortion

It is not surprising that the affine method performs better than the similarity method, when dealing with images related through an affine transform, which do not suffer from occlusion. In this second experiment, we show that, as expected, the affine method also performs better than the similarity method, when applied to real images related through moderately weak perspective transformations. The two images considered in this experiment (which we call "Hitchcock experiment") are shown in Figure 9.7, with their corresponding level lines. The resolution of these images is 640×480, which is enough to ensure good accuracy in the extracted level lines.

Figure 9.7: *Hitchcock experiment: original images and their corresponding level lines. The image on top is considered as "target" image. In the target image,* 307 *maximal boundaries were detected, and* 266 *maximal boundaries were detected in the scene image.*

For the affine semi-local invariant method, 1150 and 853 shape elements were extracted from the target image and from the scene image, respectively. The number of 1-meaningful matches detected was 517. In order to reduce the redundancy of the output, we use a greedy algorithm that eliminates matched shape elements which share a large piece of curve with other shape elements presenting lower NFA. More precisely speaking, if a pair of shape elements $(\mathcal{S}_1, \mathcal{S}'_1)$ is an ε_1-meaningful match, and there exists another pair $(\mathcal{S}_2, \mathcal{S}'_2)$ matching ε_2-meaningfully, with $\varepsilon_2 < \varepsilon_1$, such that $\mathcal{S}_1$ shares at least half of its length with $\mathcal{S}_2$, and the same for $\mathcal{S}'_1$ and $\mathcal{S}'_2$, the pair $(\mathcal{S}_1, \mathcal{S}'_1)$ is eliminated from the output list of matches. Hence, the list of meaningful matches is drastically reduced from 517 to 16

elements (this shows how redundant is the encoding). These 16 matched shape elements are shown, superimposed, in Figure 9.8. No false matches were detected, and all matches have their NFA below 0.1. The best match, shown in Figure 9.9, reaches $NFA = 6.5\,10^{-11}$. This value is remarkably low, considering that ideal perfect matches in this experiment would have a number of false alarms of $1150 \times 853/853^6 = 2.5\,10^{-12}$ (when the empirical distributions of distances to target codes are learned using only the considered scene image, as we do here).

Figure 9.8: *Affine invariant semi-local recognition method: meaningful matches between shape elements. No false matches were detected, and all detections show an NFA below* 0.1. *The lowest NFA is* $6.5\,10^{-11}$.

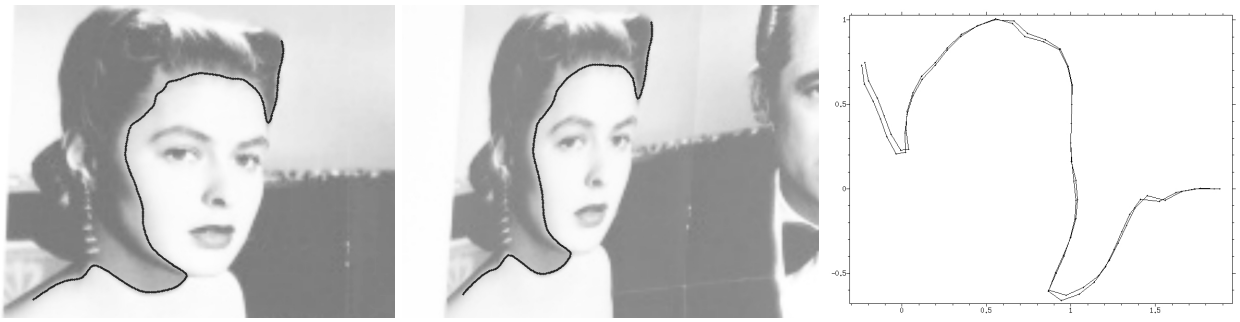


Figure 9.9: *Affine invariant semi-local recognition method: the match showing the lowest NFA* ($6.5\,10^{-11}$).

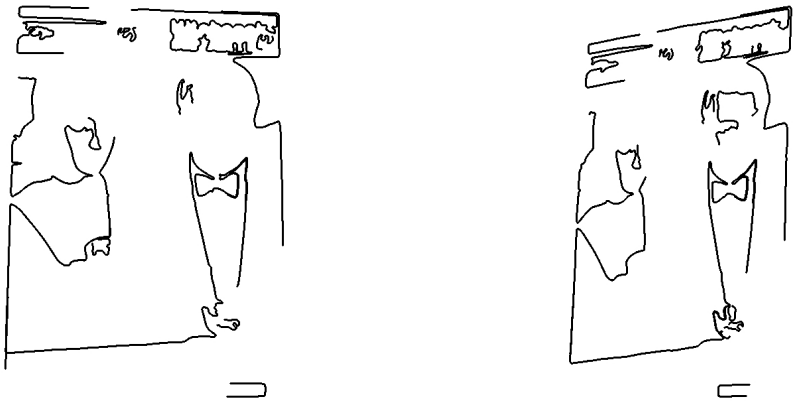
In Figure 9.10 we display the meaningful matches detected using the similarity semi-local invariant recognition method. In this case, 2033 and 1463 shape elements were extracted from the target image and from the scene image, respectively. As we noticed for the toy example, the similarity method allows to extract more shape elements than the affine method. A total number of 244 meaningful matches ($NFA < 1$) were detected, and 26 matches were left after applying the greedy algorithm. The meaningful matches for the similarity method are shown in Figure 9.10. The lowest NFA reached with the similarity method is $3.8\,10^{-8}$, and corresponds to the shape elements and the normalized codes presented in Figure 9.11. In Figures 9.10(b) and 9.10(c), we present, respectively, the shape elements matching at $\varepsilon < 0.1$, and those for which the NFA is between 0.1 and 1. Notice

that none of the 10^{-1}-meaningful matches are false matches, and that the corresponding shape elements are in general much more local than the shape elements matching in Figure 9.10(c). Indeed, the more global are the shape elements, the less accurate is the similarity approximation of the underlying transformation, which is in fact a projective transform. Two false matches, for which the NFA is larger than 0.1, can be seen in Figure 9.10(c). In Figure 9.12 we show the shape elements of these false matches, as well as the superimposed normalized codes represented in the normalized frame.

We end up the discussion on the "Hitchcock experiment" with a comparison between the NFA of the meaningful matches for the affine and the similarity semi-local invariant methods. Such a comparison is illustrated in Figure 9.13. The principle is the same as for the toy example from section 9.1.1. The list of meaningful matches is ordered from best (lowest NFA) to worst (largest NFA), and the index i of this sorted list is plotted *versus* $-\log_{10}(NFA_i)$, where NFA_i is the NFA if the i-th best match. Such a function is plotted for the similarity and for the affine matches. The affine semi-local invariant matches reach lower NFA. Notice that in both affine and similarity invariant recognition methods, there are several matches that show small NFA, leading to sure detections of common shapes.

(a) *All* 26 *matches having an* NFA *below* 1.

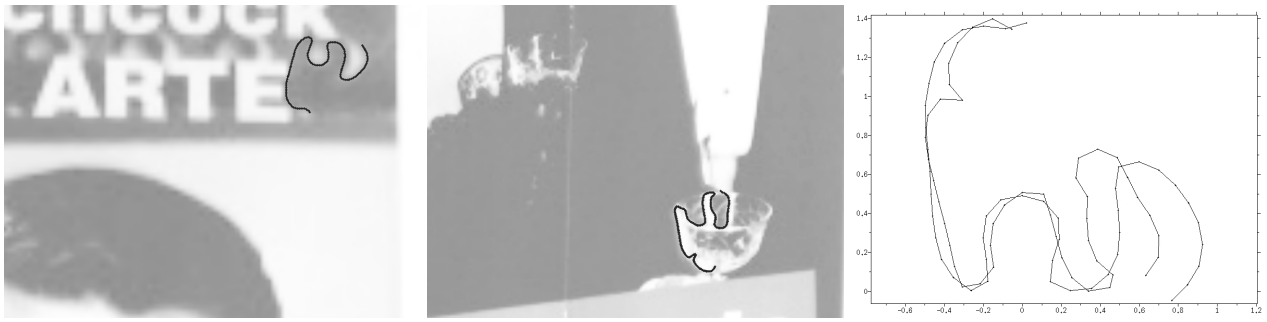
(b) 12 *matches show an* NFA *below* 0.1.



(c) 14 *matches show an* NFA *between* 0.1 *and* 1.

Figure 9.10: *Similarity invariant semi-local recognition method: meaningful matches between shape elements. Among the* 26 *matches having an* NFA *below* 1, 12 *are* 10^{-1}*-meaningful. False matches (two) can only be seen in (c), and their* NFA *is above* 0.1.

Figure 9.11: *Similarity invariant semi-local recognition method: the match showing the lowest NFA ($3.8\,10^{-8}$).*



(a) *False match,* $NFA = 0.64$

(b) *False match,* $NFA = 0.68$

Figure 9.12: *Similarity semi-local invariant method: the two false matches. Their NFA are larger than* 0.1.

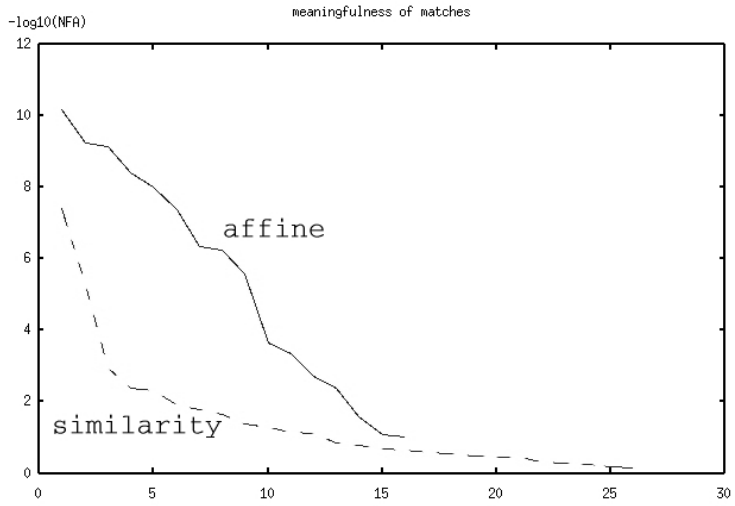


Figure 9.13: *Hitchcock experiment: NFA of affine and similarity semi-local invariant matches. Both lists of meaningful matches are ordered from best (lowest NFA) to worst (largest NFA), and for each list, the index i of the sorted list is plotted* versus $-\log_{10}(NFA_i)$*, where NFA_i is the NFA of the i-th best match.*

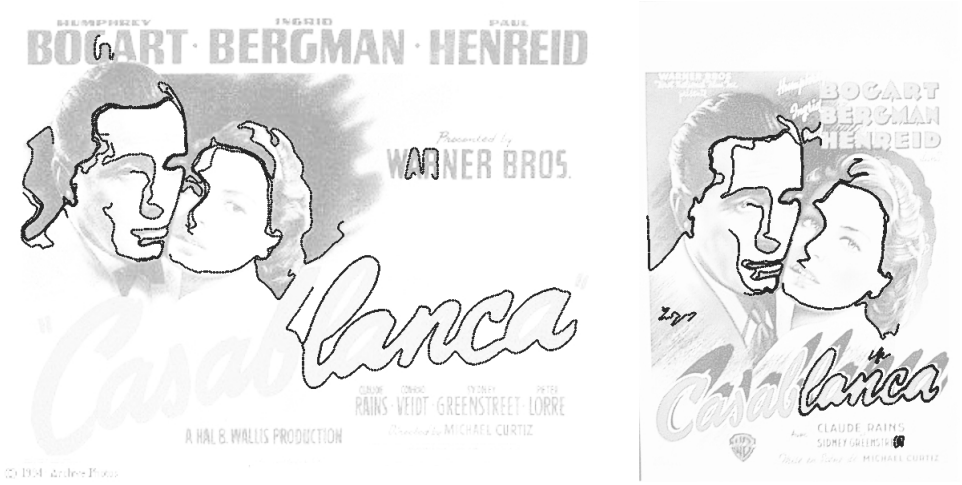
9.1.3 A more difficult problem

Both in the toy example and the in "Hitchcock experiment", target and scene images represented different views of the same planar "objects" or elements. Corresponding shapes were accurately described by the meaningful boundaries, leading to the detection of several matching shape elements, with high detection confidence. In this subsection we consider a more difficult example, which consists in finding common shape elements between the pair of images in Figure 9.14. Although at first sight these two different posters of the movie *Casablanca* are very similar, they present many slight differences that considerably affect the topographic map, and consequently the set of maximal meaningful boundaries. For instance, the actors' faces in the target image (the one on top in Figure 9.14) come from a snapshot, while in the scene image they are a drawing.

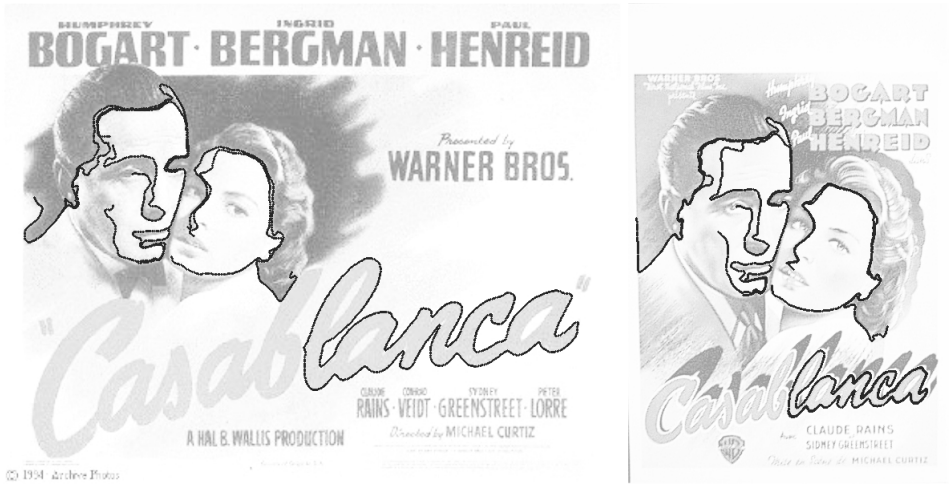
Figure 9.14: *Casablanca experiment: original images (on the left) and level lines (on the right). The image on top was considered as target image.*

In this example, we only consider the similarity semi-local invariant method. The number of shape

elements that were extracted from the target and the scene images were 3540 and 8554, respectively. Figure 9.15 shows the 1-meaningful matches (*i.e.* matches for which $NFA < 1$) in the top row, and the 10^{-1}-meaningful matches in the bottom row. The number of 1-meaningful matches detected was 211, which was reduced to 17 after applying the greedy algorithm. It seems that the majority of the relevant shape information that the two images have in common has been detected. No meaningful match was found for characters 'Casab', which are indeed quite different (up to similarity invariance) from one image to the other one.



(a) $NFA < 1$



(b) $NFA < 0.1$

Figure 9.15: *Casablanca experiment: meaningful matches between shape elements. Top:* $NFA < 1$. *Bottom:* $NFA < 10^{-1}$, *no false match can be seen.*

Figure 9.16 shows the shape elements corresponding to the most meaningful match, for which $NFA = 5.0\,10^{-13}$. Such a low NFA is a consequence of the fact that the target shape elements is so rare

among all the extracted shape elements, that it is almost impossible that just by chance another shape element lies so close to it. At this point, we should stress the following remark, which is obvious from the definition of the NFA given in Chapter 8. Suppose we are given two target shape elements $\mathcal{S}_1$ and $\mathcal{S}_2$, and two scene shape elements $\mathcal{S}'_1$ and $\mathcal{S}'_2$, such that $d(\mathcal{S}_1, \mathcal{S}'_1) = d(\mathcal{S}_2, \mathcal{S}'_2) = \delta$. If

$$\#\{\mathcal{S}' \in \mathcal{B} \text{ s.t. } d(\mathcal{S}_1, \mathcal{S}') \leqslant \delta\} < \#\{\mathcal{S}' \in \mathcal{B} \text{ s.t. } d(\mathcal{S}_2, \mathcal{S}') \leqslant \delta\},$$

it follows that $NFA(\mathcal{S}_1, \mathcal{S}'_1) < NFA(\mathcal{S}_2, \mathcal{S}'_2)$. Hence, for a given distance d, the "rarer" is a target shape element $\mathcal{S}$ (with respect to $\mathcal{B}$), the lower is $NFA(\mathcal{S}, d)$. This makes sense, since a rare shape element is more discriminatory than a banal one.

$$NFA = 5 \cdot 10^{-13}$$

Figure 9.16: *The match with the lowest NFA. The query shape element (left image) matches the database shape (right image).*

Figure 9.17 shows all the false matches detected at $NFA < 1$. They all have an NFA between 0.1 and 1. The matches at $NFA = 0.34$ and $NFA = 0.84$, for example, illustrate the previous remark; the distance between the normalized codes for which $NFA = 0.34$ is 0.26, while the one for that whose $NFA = 0.84$ is 0.14. The former corresponds to a target shape element which seems to be rare, while target shape elements like the one matching at $NFA = 0.84$ are more "banal" (less discriminatory) among extracted shape elements.

Finally, notice that all matches which semantically correspond to the same shape elements (the "correct" matches) show NFAs below 0.1, except for the one presented in Figure 9.18 (parts of two characters 'c' with different font type). However, shape elements like the target shape element in Figure 9.18 are not so rare, and even if the distance between codes is not that large ($d = 0.15$), the match is almost not meaningful.

Figure 9.17: *The six false matches for $NFA < 1$, with their normalized codes. The left most and middle images correspond to the target and the scene shape elements, respectively. The right most image shows their normalized codes, superimposed. All false matches show NFAs between* 0.1 *and* 1.

Figure 9.18: *Casablanca experiment: match between shape elements which semantically correspond, but shows an* NFA *close to* 1 *(*$NFA = 0.13$*). Notice however that the fonts of the characters are not the same, and that the shape element is not very discriminatory. This explains why the* NFA *is high.*

9.1.4 Meaningful matches between unrelated images

The experiment we present in this subsection consists in finding common shape elements between two unrelated images. We consider two examples. "Target" and "scene" images for the first experiment are shown in Figure 9.19. We also display all the matches for which NFA is below 1, superimposed to the original images. 4731 and 4946 shape elements were extracted from target and scene images, respectively. Among all $4731 \times 4946 \approx 23\,10^6$ pairs of target-scene shape elements, only 6 matches having $NFA < 1$ were detected. Their NFAs range from 0.21 to 0.97. The matched shape elements, as well as their corresponding normalized codes, are shown in Figure 9.20. Numbers 1), 4) and 5), are "simple" (they are relatively short and do not present much oscillations), and match at pretty small distances. However, because of their "banality", they do not show lower NFAs. Matches number 2) and 6), while locally different, are quite similar at coarse scale, as it can be seen from their superimposed normalized codes. For such long codes, a representation in 45 points may not be accurate enough; a finer sampling would have probably led to larger NFAs for that kind of matches.

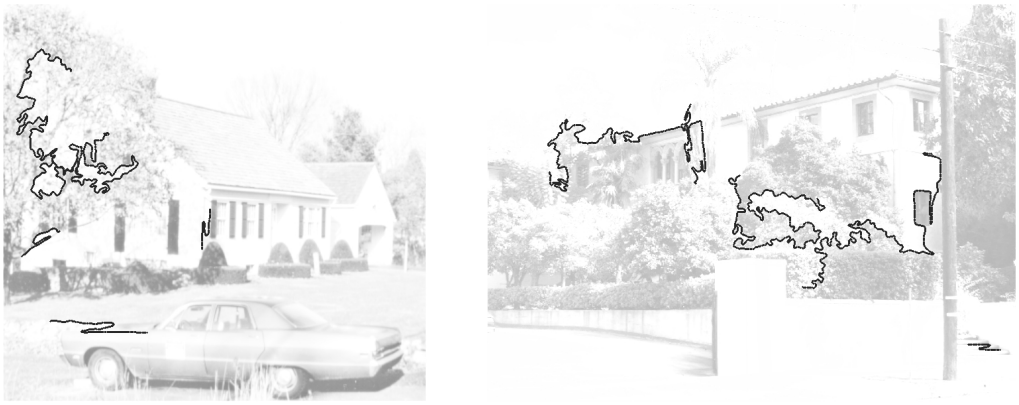


Figure 9.19: *Left: target image;* 4731 *shape elements were extracted from this image. Right: scene image; the number of shape elements extracted from it was* 4946*. Among the* $23\,10^6$ *pairs of target-scene shape elements, only six match at* $NFA < 1$*. The* NFA *of these matches range from* 0.21 *to* 0.97*.*

The second example of common shape elements between two unrelated images involves the images in Figure 9.21. The 22 shape elements extracted from the target image are searched in the 546 shape elements from the scene image on the left. Superimposed to the original images, we show the two shape elements that match at $NFA < 1$. Unlike the previous example, here these matches show NFAs lower than 0.1. The matched shape elements and their normalized codes are shown in Figure 9.22. Notice that, according to what was presented in Chapter 8, matches showing NFAs lower than 0.1 are not supposed to happen "by chance" (as matches between shape elements extracted from random level lines), and some common cause should be behind such an unexpected coincidence. This is what happens here. Indeed, many shapes in images derive from natural or man-made objects having a common structure. For instance, many objects are built of parallel or equal-length parts.

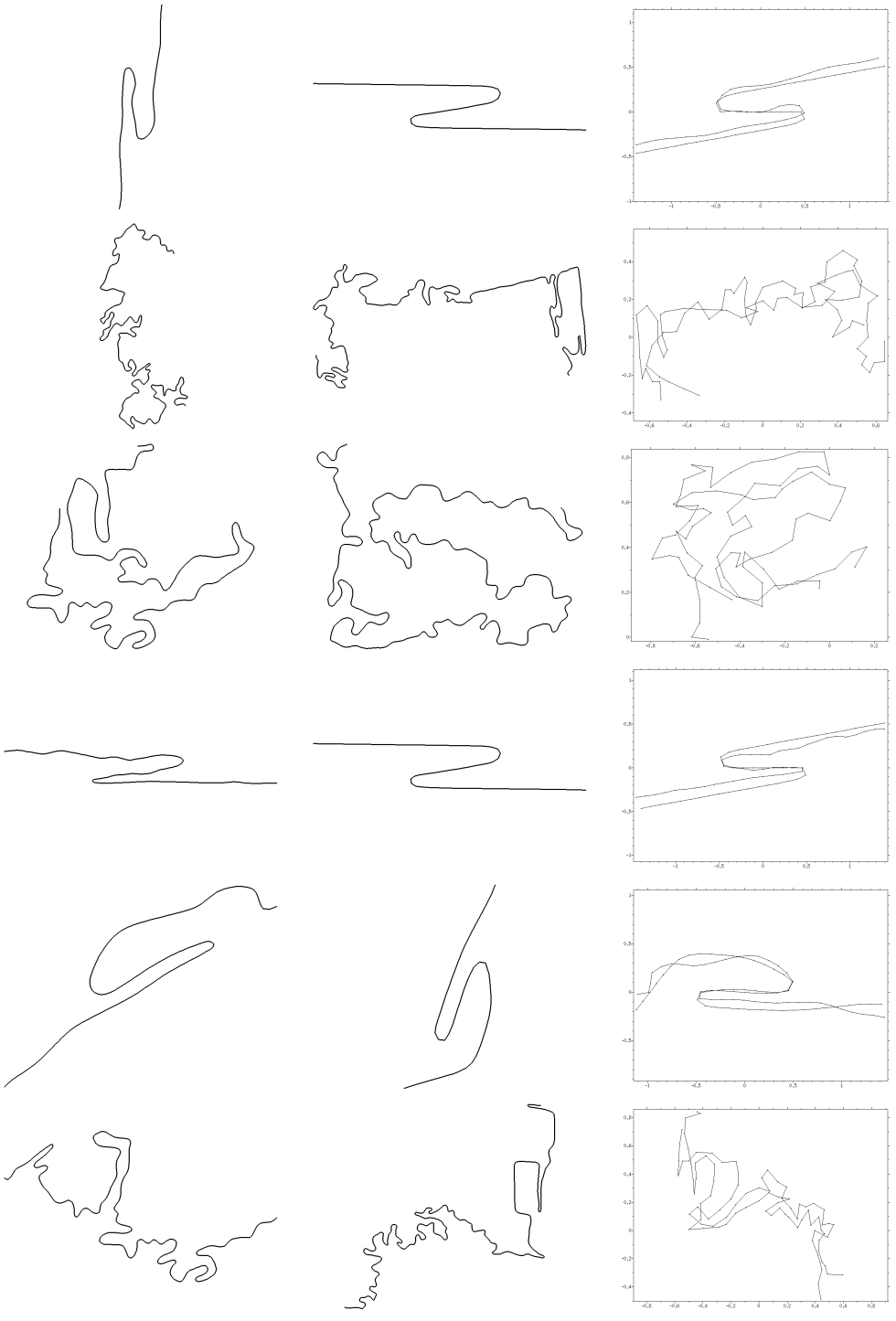


Figure 9.20: *The six false matches detected for $NFA < 1$, with their normalized codes. The left most and middle images correspond to the target and the scene shape elements, respectively. The right most image shows their normalized codes, superimposed.All false matches show NFAs between 0.1 and 1.*

Figure 9.21: *Puma experiment. Left: target image, from which* 22 *shape elements were extracted. Right: scene image;* 546 *shape elements were extracted from it. The two matches detected at* $NFA < 1$ *are superimposed to the original images.*

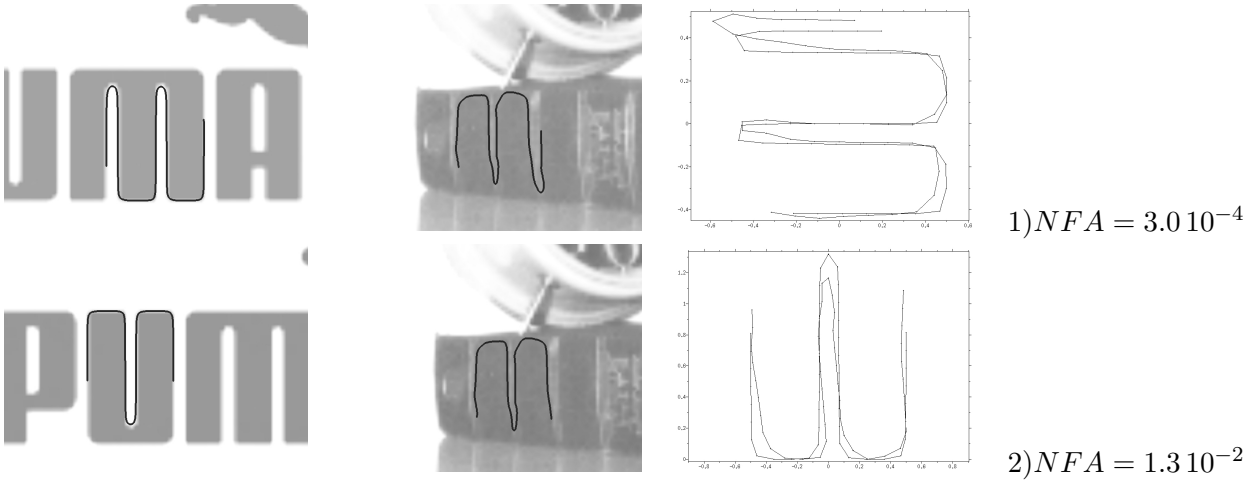


Figure 9.22: *Puma experiment: the two matches detected for* $NFA < 1$*, with their normalized codes. The left most and middle images correspond to the target and the scene shape elements, respectively. The right most image shows their normalized codes, superimposed. Such a conspicuous coincidence admits a better explanation than randomness: many shapes in images derive from natural or man-made objects having a common structure. For instance, many objects are built of parallel or equal-length parts.*

9.1.5 Blur introduced by long distances to the camera

In this subsection we present the last experiment dealing with the semi-local invariant method for shape recognition. This experiment just aims at illustrating how small objects' meaningful boundaries are affected by the blur introduced when objects are far from the camera, and how this problem can be solved by representing the target image at multiple scales.

The target and scene images for this example are shown in Figure 9.23. On the left column of this image, we display the extracted maximal meaningful boundaries. Images are presented at the same scale. Figure 9.24 illustrates a detail of the maximal meaningful boundaries of the scene image, corresponding to the region of interest for this experiment. Compare now these boundaries with those ones extracted from the target image, in Figure 9.23 (on top right). The characters in the scene image have been almost completely destroyed, and not many similar shape elements can be observed.

Figure 9.23: *Top row: target image and its maximal meaningful boundaries;* 312 *shape elements were extracted from this image. Bottom row: scene image and corresponding maximal meaningful boundaries.* 1859 *shape elements were extracted from it.*

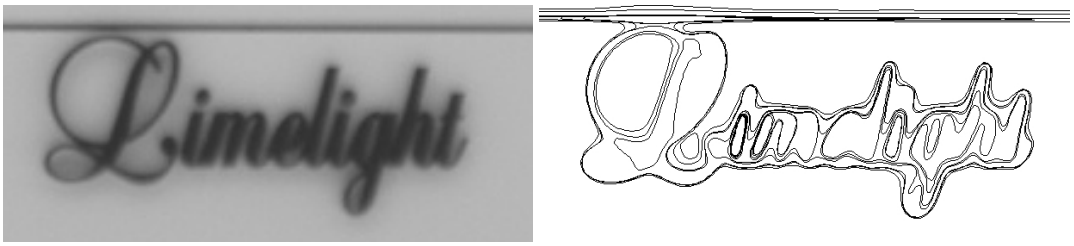


Figure 9.24: *detail of the maximal meaningful boundaries of the scene image, corresponding to the region of interest for this experiment. The boundaries of characters have been very degraded by the blur and the smoothing.*

In Figure 9.25 we display, on the top row, the original image and two image reductions, by factors 4

and 8. On the bottom row we illustrate their corresponding maximal meaningful boundaries (followed by an affine shortening at scale $T = 0.5$, see Chapter 5). Image reductions were performed using a prolate kernel.

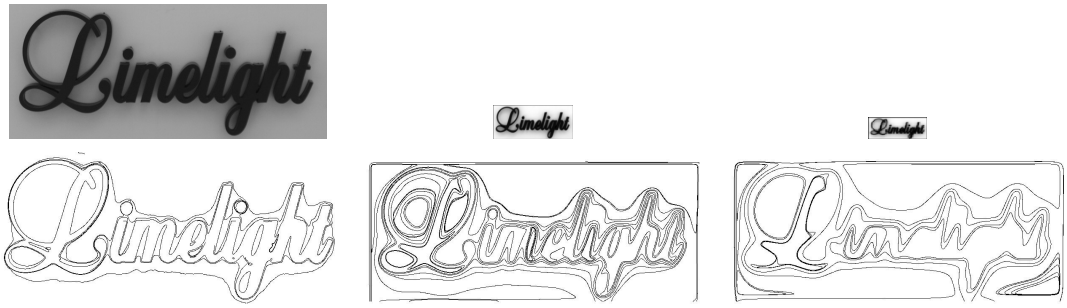
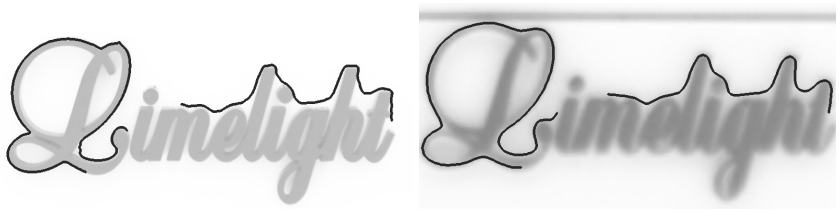
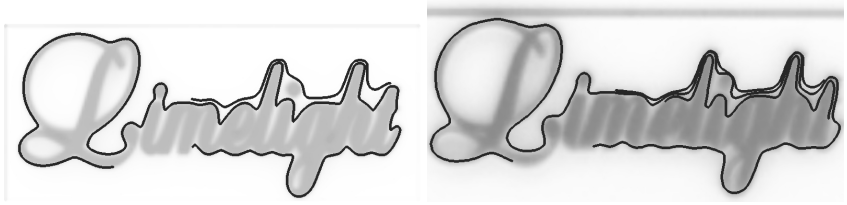


Figure 9.25: *Original target image and two image reductions. Left column: original image and corresponding maximal meaningful boundaries. Middle: image reduction by a factor 4 (324 shape elements were extracted from this image). Right: image reduction by a factor 8 (73 shape elements were extracted from this image). Reductions were performed using a prolate kernel.*

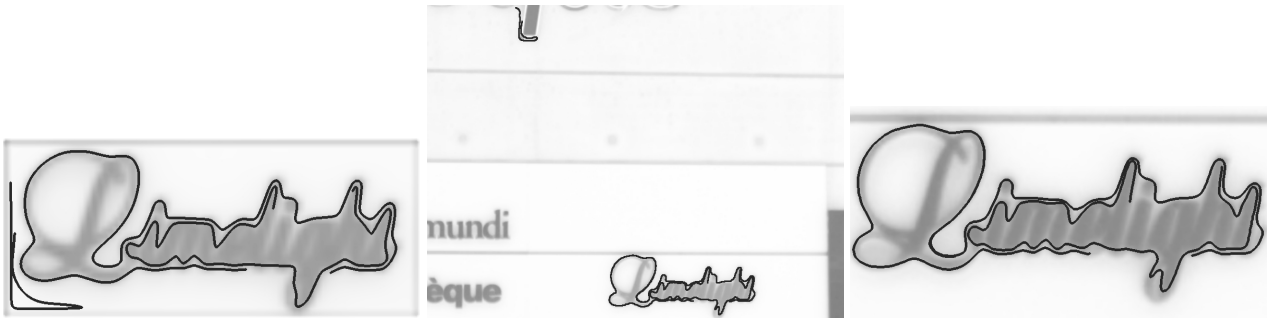
Figure 9.26 shows the detected matches at $NFA < 1$, for each target image (the original image and the two reductions) with the scene image. When the shape elements of the original target image are searched, only two matches having $NFA < 1$ are found (Figure 9.26(a)). Both matches are correct, and their NFAs are $8.8\,10^{-6}$ and $1.9\,10^{-4}$. Using as target image the image reduced by a factor 4, more meaningful matches are found, and the best one reaches an NFA of $2.3\,10^{-10}$. In this case all matches are correct also (Figure 9.26(b)). Finally, using the target image reduced by a factor 8, even more meaningful matches are detected, reaching still lower NFAs. In this case, the NFA of correct matches ranges from $2.1\,10^{-3}$ to $3.6\,10^{-12}$. A false match of $NFA = 7.6\,10^{-1}$ was detected, but it corresponds to an artifact (a border effect) of the image reduction, as can be seen in Figure 9.26(c).



(a) *Using the original target image.* 2 *matches have their* NFA *below* 1 *(*$8.8\,10^{-6}$ *and* $1.9\,10^{-4}$*).*



(b) *Using an image reduction by* 4 *of the target image:* 4 *meaningful matches, at* NFA*s* $2.3\,10^{-10}$, $1.3\,10^{-6}$, $1.5\,10^{-5}$ *and* $5.7\,10^{-1}$.



(c) *Using an image reduction by* 8 *of the target image:* 5 *meaningful matches, at* NFA*s* $3.6\,10^{-12}$ $7.4\,10^{-5}$, $4.6\,10^{-4}$, $2.1\,10^{-3}$ *and* $7.6\,10^{-1}$. *The last one corresponds to a false match, but was introduced by an artifact in the image reduction procedure.*

Figure 9.26: *Shape elements matched with the scene images, using three different scales of a target image. The number of meaningful matches, as we their meaningfulness, increases when we consider image reductions. These image reductions try to simulate the effect of distance to the camera.*

9.2 Global meaningful matches

In chapter 7 normalizations invariant up to translation, rotation, similarity, or affine transforms were presented. In this section, we show several experiments on global matching of shapes that validate the normalization and the distance threshold derived from the number of false alarms (see chapter 8). Before going to the experiments, let us recall that a curve leads to as many descriptions (or "codes") as bitangent or flat pieces are present in the curve.

9.2.1 Global affine invariant recognition: toy example

We start this section on the global recognition method with a simple example. This toy example involves the two images presented in Figure 9.27. The image on the left, which was considered as target image, represents the "Puma" logo, and the image on the right is an affine distorted version of the left image. The extracted meaningful boundaries, followed by an affine curve shortening, are shown superimposed to the images. Global shape elements are extracted from both images, using the global affine invariant normalization method described in Chapter 7, section 7.3.2.

The detection of meaningful matches between all global shape elements extracted from both images was performed using the detection method presented in Chapter 8. No false match was detected. Figure 9.28(a) shows the matches with all global shape elements extracted from the puma boundary in the target image, superimposed. Figures 9.28(b) and 9.28(c) illustrate the normalized curves of the "target" and "scene" global shape element for which the match presents the lowest NFA ($2.0\,10^{-8}$).

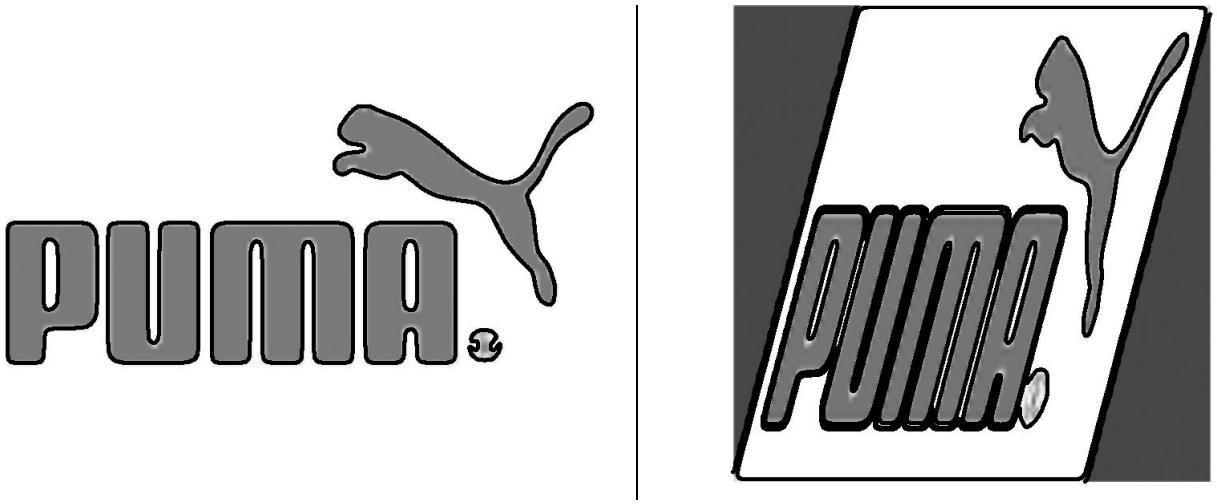


Figure 9.27: *Puma. Global shape elements extracted from the superimposed level lines on the left image are sought among global shape elements from the level lines on the right image.*

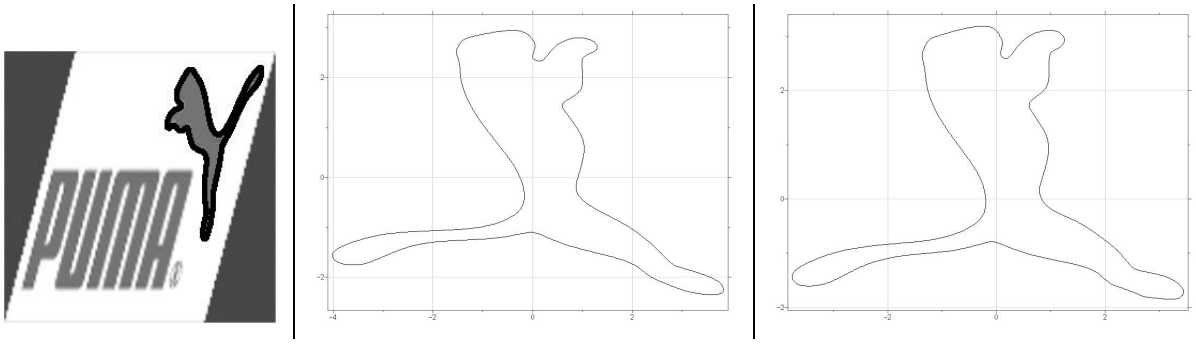


Figure 9.28: *Puma, global affine invariant encoding. Left:* 1*-meaningful matches with the puma boundary (superimposed). No false detection can be seen. Middle and right images: normalized global shape elements from the target and the scene images, that match at the lowest* NFA *(*$2\,10^{-8}$*).*

9.2.2 Comparing similarity and affine invariant global recognition methods

In this experiment we compare the performance of the affine and the similarity global recognition methods, on two images whose maximal meaningful boundaries are shown in Figure 9.29. The boundaries on the left were considered as target. The shapes in the scene images are distorted not only by projective transforms but also from projection on a cylinder (the bottle).

The example we consider here consists in finding the character 'n' from the target image in the scene image.

Figure 9.29: *Evian: maximal meaningful boundaries. Left: target. Right: scene.*

Figure 9.30 shows the detected 1-meaningful matches with global shape elements extracted from the 'n' in the target image. The target 'n' was represented with 10 global shape elements. 36 matches were found in the scene image. The lowest NFA was 10^{-11}. Some false matches can be seen, but they all show NFAs between 0.7 and 1.

Figure 9.31 shows the matched global shape elements when considering the global affine method. The target 'n' is still represented with 10 codes, since it is the same 'n' that was considered for the global similarity matching (and there is always one global shape element extracted for each bitangent line or flat piece on the curve). 35 matches showed an NFA below 1. The matches that actually correspond to the 'n' on the bottle, show NFAs that range from 10^{-15} to 10^{-8}. The NFA of matches which do not correspond to global shape elements in the 'n' on the bottle, are between 10^{-3} and 1. However, some "false matches" are not really "false" but "casual" matches, since they correspond to other characters 'n' or 'u' that appear on the bottle ("Minérale" and "Naturelle").

In Figure 9.32 we display, for both methods, the matches showing the lowest NFA. The top row shows the normalized global shape element for the global similarity invariant method, and the bottom row shows the normalized global shape element for the affine method. Notice that the pair of

Figure 9.30: *Evian: global similarity invariant matching. All* 1*-meaningful matches with character 'n' from the target image. The target 'n' is represented with* 10 *global shape elements, that match with* 36 *global shape elements from the scene image. The lowest* NFA *is* 10^{-11}*. False detections show* NFA*s between* 0.7 *and* 1.

Figure 9.31: *Evian: affine invariant global matching. Meaningful matches with character 'n' from the target image, represented with* 10 *global shape elements. Left:* 1*-meaningful matches,* 35 *matches. False matches show an* NFA *between* 10^{-3} *and* 1*, but some of them are not really "false" but "casual" matches, since they correspond to other characters 'n' and 'u' which are present in the scene. Good matches show* NFA *ranging from* 10^{-15} *to* 10^{-8}*. Right image: the* 23 *meaningful matches showing* NFA*s below* 10^{-2}*.*

affine normalized shape elements are much more close to one another than the pair of similarity normalized shape elements. It seems then reasonable that the NFA reached with the global affine invariant method (10^{-15}) is lower than the one reached with the similarity method (10^{-8}).

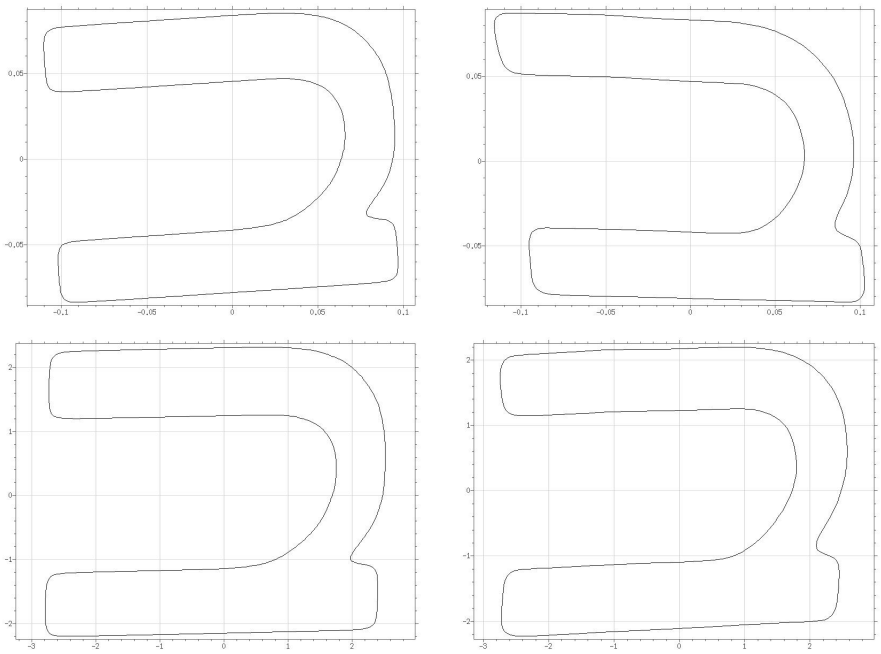


Figure 9.32: *Evian experiment: matches for the 'n' showing the lowest NFA for the global similarity (top row) and affine (bottom row) invariant recognition methods. In each of the rows, the curve on the left is the normalized global shape element extracted from the target 'n', and the one on the right is the corresponding normalized global shape element extracted from the scene image. The NFA for the similarity method was 10^{-11}, and for the affine method was 10^{-15}. In spite of the projection on the bottle, the normalized shapes elements are very alike.*

In Figure 9.33 we display, for both methods, the false matches that show the lowest NFA for both methods. The top row shows the normalized global shape element for the global similarity invariant method, and the bottom row shows the normalized global shape element for the affine method. The NFA for the similarity invariant match was 0.7, and for the affine method was $4.0\,10^{-3}$.

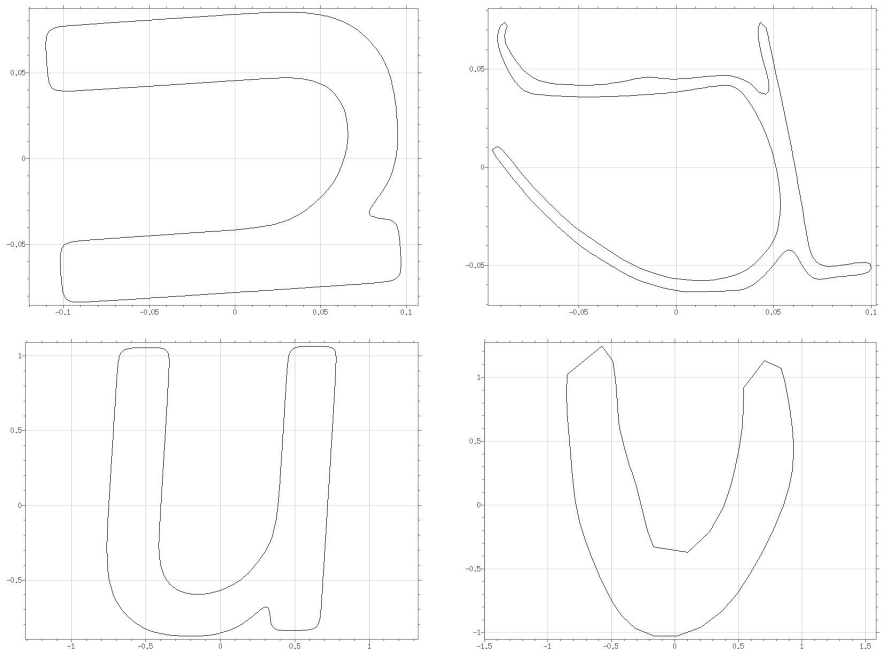


Figure 9.33: *Evian experiment: the false matches for the 'n' that show the lowest NFA, for the global similarity (top row) and affine (bottom row) invariant recognition methods. In each of the rows, the curve on the left is the normalized global shape element extracted from the target 'n', and the one on the right is the corresponding normalized global shape element extracted from the scene image. The NFA for the similarity invariant match was* 0.7*, and for the affine method was* $4.0\,10^{-3}$ *(it can be seen in the handwritings on the top of the right image from Figure 9.31).*

9.2.3 Global matches of non-locally encoded shapes elements

The main drawback of global shape matching is its sensitivity to occlusions, whereas local matching is especially designed to deal with them. Nevertheless, the semi-local encoding we presented in Chapter 7 is unable to encode curves which are convex or "quasi-convex". While in general (as we will show with some experiments) these "quasi-convex" boundaries are not very discriminatory, some of them may provide useful information we would not like to miss. We therefore make the global and semi-local methods work together: the non-locally encoded boundaries are globally encoded, and then globally compared.

First example: a book cover

Figure 9.34 shows two different views of a book cover, and its corresponding maximal meaningful boundaries. The "target" image (on top) consists of a partial view.

Figure 9.34: *Book cover. Top row: "target" image, and its corresponding* 208 *maximal meaningful boundaries. Bottom row: "scene" image, and its* 1185 *maximal meaningful boundaries.*

The two images are related by a strong perspective transformation. Perspective transforms can be locally approximated by affine transforms; since many boundaries in images are quite local, it is

sound to try to find correspondences between the considered pair of images using our semi-local and global affine invariant recognition methods.

Figure 9.35 shows the 1-meaningful matches between shape elements detected by the semi-local affine recognition method. Among the 16 matches that were found, a single false match having an NFA equal to 0.6 was detected (it can be seen on the right part of the scene image), and the lowest NFA was 10^{-10}.

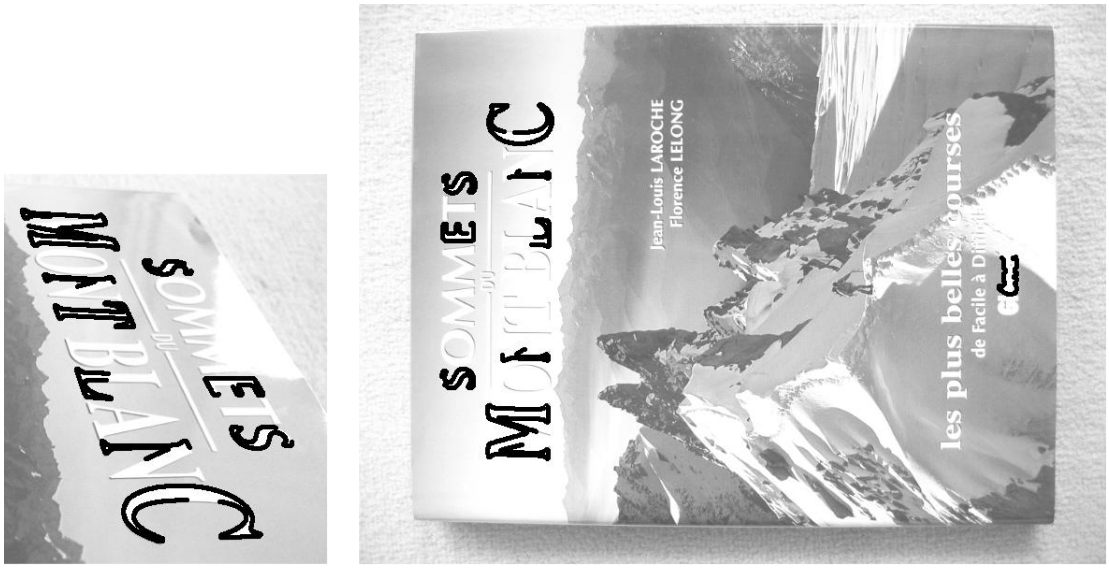
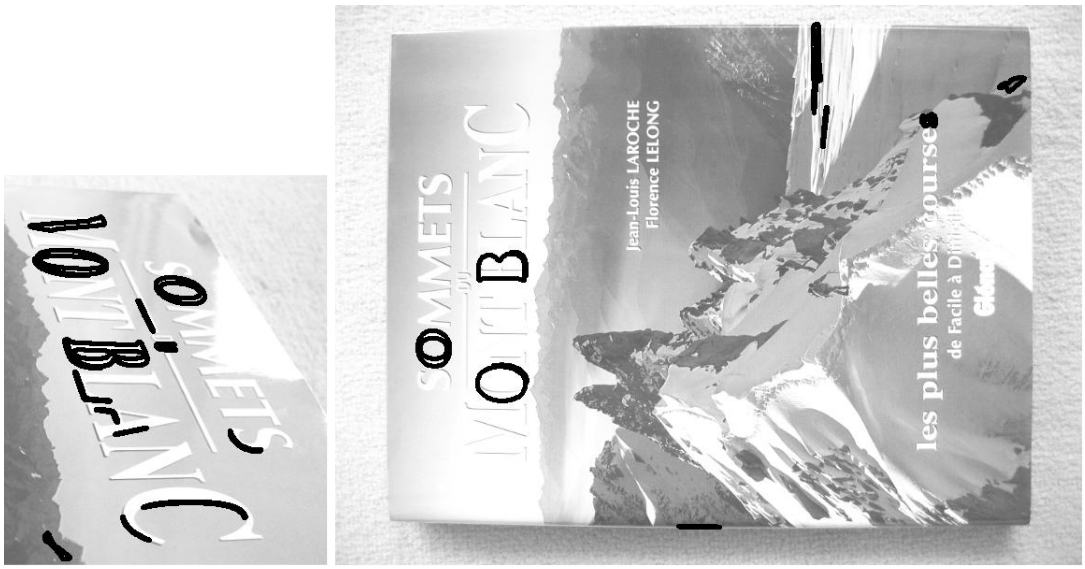


Figure 9.35: *Book cover: the* 16 *semi-local affine invariant matches. The best match has an* NFA *of* 10^{-10}*. The scene shape element of the only false match* ($NFA = 0.6$) *that was detected can be seen in the right part of the scene image.*

The next stage of our matching procedure consists in finding matches between global shape elements, extracted from those maximal meaningful boundaries that were not described by any semi-local shape element. All not semi-locally encoded maximal meaningful boundaries are shown in Figure 9.36(a). These two sets of curves are used as the input of the global affine invariant recognition method. Figure 9.36(b) shows the detected 1-meaningful matches between global shape elements. Good matches reach NFAs as low as 10^{-10}. Some false matches were detected, but we can only say they are false because, semantically, they do not correspond to the same shapes. However, these "false matches" correspond to global shapes elements that look actually alike. Such "false" correspondences can often occur: convex or "quasi-convex" shapes are indeed not very discriminatory, and higher level information (such as spatial coherence between matches) is needed in order to assess their semantic validity.

Notice that if we combine the matches that were obtained from both the semi-local and affine invariant methods, almost all shapes in common have been detected. Compare now the combination of these matches with the matches detected when using the global method over all the maximal meaningful boundaries (Figure 9.37). We can observe that using first the semi-local method, and then the

(a)



(b)

Figure 9.36: *Book cover. (a) all not locally encoded maximal meaningful boundaries. Too small or too convex level lines are not encoded. (b) the* 160 *matches between global shape elements, using the global affine invariant recognition method. The search is only performed upon the maximal meaningful boundaries which were not locally encoded. The NFA of some matches reach values as low as* 10^{-10}*. Since spatial coherence between matches is not taken into account, "false" matches (false from a semantic viewpoint) are unavoidable (these matches correspond to global shapes elements that look actually alike).*

global method over the non semi-locally encoded boundaries, produces less false matches than using the global method over the original sets of maximal meaningful boundaries. Indeed, even when we do not deal with occlusions, considering semi-local descriptions for "complex" boundaries is more sound than describing them globally, since it allows to increase the discriminatory power.

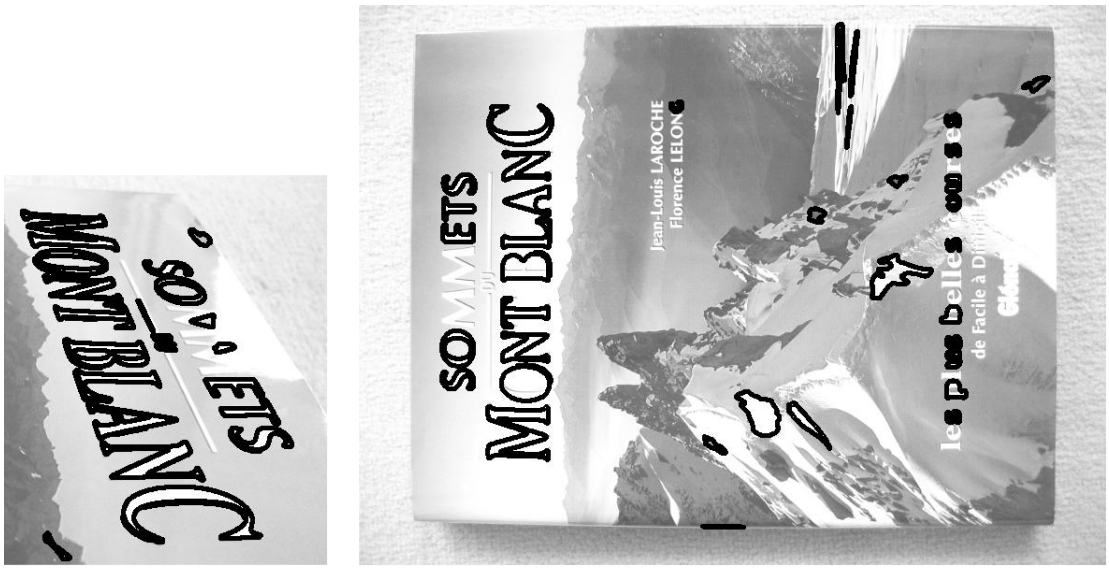


Figure 9.37: *Book cover: the* 857 *global shape elements detected as* 1*-meaningful matches, among all maximal meaningful boundaries. The lowest* NFA *reaches* 10^{-14}*. The majority of the "false" matches are unavoidable, since globally, the matched shape elements are very alike.*

Two frames of a sequence

Figure 9.38 shows the semi-locally matched shape elements between two frames of a sequence, using the semi-local similarity invariant method. The non semi-locally encoded maximal meaningful boundaries are displayed in Figure 9.39. The majority of the non semi-locally encoded boundaries are oval shaped, and not discriminatory enough to decide if a match is "semantically correct". Nevertheless, while pairing two of them may not provide much information, looking for spatial coherence between all pairs of matches can lead to high confidence detections.

Figure 9.40 shows some global matches (those for which the NFA is below 10^{-2}). Among the represented shape elements, almost all of them seem to be discriminatory enough, and no "oval" shaped (not discriminatory) boundary is present. This fact is consistent with one of the features of our detection methodology: good matches between discriminatory shape elements show the lowest NFAs.

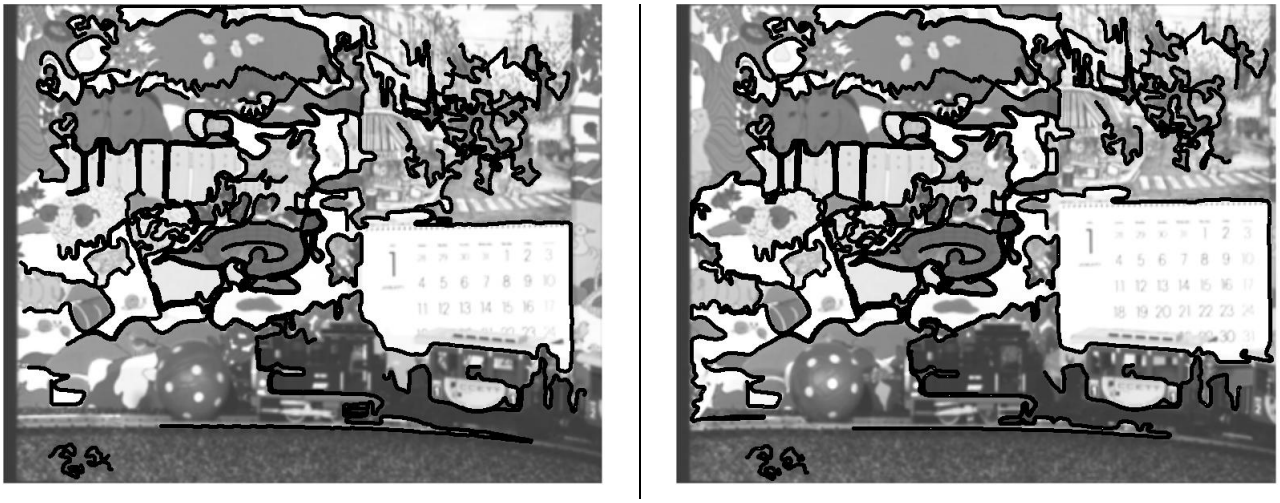


Figure 9.38: *Movie frames. The* 75 *semi-local similarity invariant* 1*-meaningful matches. The lowest* NFA *is about* $2.0, 10^{-16}$.

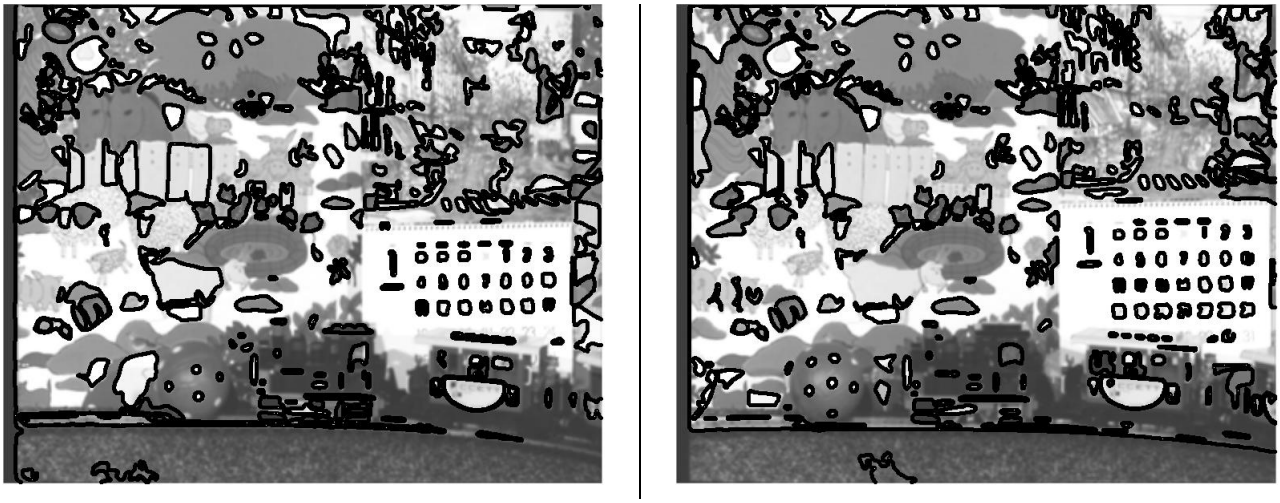


Figure 9.39: *Movie frames. Non semi-locally encoded maximal meaningful boundaries. There are* 356 *lines in the target image (left) and* 373 *in the scene image (right).*

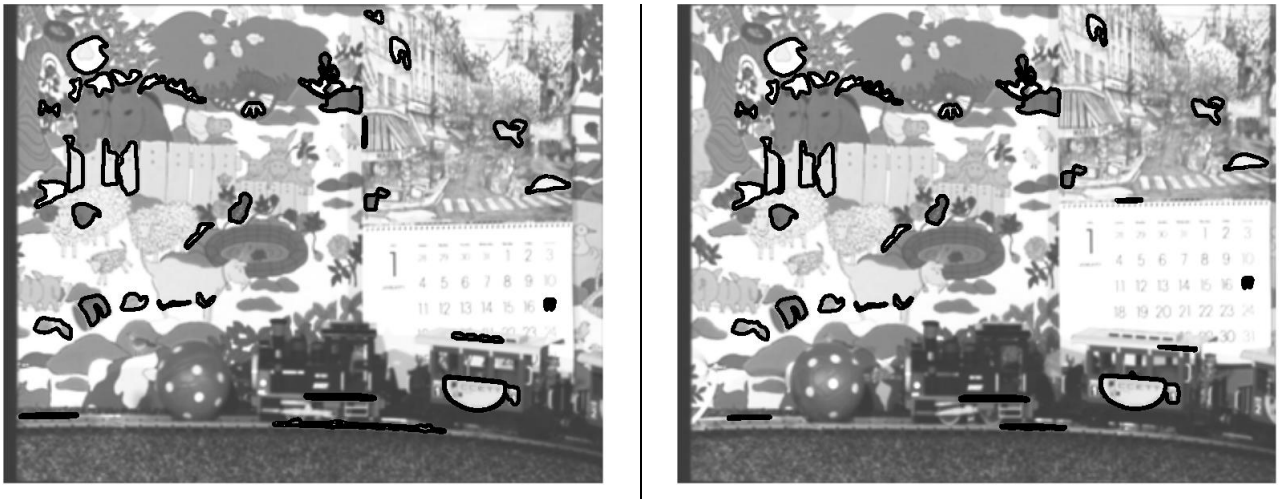


Figure 9.40: *Movie frame. The* 120 *global* 10^{-2}*-meaningful matches among the non semi-locally encoded level lines. The lowest* NFA *is about* $5.0\,10^{-13}$.

9.3 Recognition is relative to the database

In this section, we illustrate a property of the distance threshold derived from the number of false alarms: the distance threshold for recognition depends on the "rareness" or on the "banality" of the target shape element relative to a database of shape elements, and therefore on the context.

9.3.1 The recognition threshold depends on the database

Recall that the Number of False Alarms of a shape element $\mathcal{S}$ at a distance d is given by:

$$\text{NFA}(\mathcal{S}, d) = N \cdot \prod_{i=1}^{K} \Pr(\mathcal{S}' \text{ s.t. } d_i(x_i(\mathcal{S}), x_i(\mathcal{S}')) \leqslant d),$$

where each shape element is described by K features $x_1, \ldots, x_K$.
Let us consider the 1-meaningful matches with $\mathcal{S}$. We can then derive the distance threshold for recognition:

$$\delta^*(\mathcal{S}) = \max\{d > 0, \text{NFA}(\mathcal{S}, d) < 1\}.$$

If a shape element $\mathcal{S}'$ satisfies

$$\max_{i \in \{1,\ldots,K\}} d_i(x_i(\mathcal{S}'), x_i(\mathcal{S})) \leqslant \delta^*(\mathcal{S}),$$

then it matches $\mathcal{S}$ 1-meaningfully.

Notice that this recognition threshold depends on the "rareness" or on the "banality" of the target shape element with respect to the considered database of shape elements. If a target shape element $\mathcal{S}_1$ is "rarer" than another one $\mathcal{S}_2$, then the database contains more shapes element close to $\mathcal{S}_2$ than shapes elements close to $\mathcal{S}_1$, below a given distance. Now, since the probabilities are in fact empirical frequencies estimated over the database of shape elements, it follows that if a target shape element $\mathcal{S}_1$ is rarer than another one $\mathcal{S}_2$, then we should have, for $i \in \{1, \ldots, K\}$:

$$\Pr\left(\mathcal{S}' \text{ s.t. } d_i(x_i(\mathcal{S}_1), x_i(\mathcal{S}')) \leqslant d\right) \leqslant \Pr\left(\mathcal{S}' \text{ s.t. } d_i(x_i(\mathcal{S}_2), x_i(\mathcal{S}')) \leqslant d\right),$$

at least for d small enough. This yields: $\delta^*(\mathcal{S}_2) \leqslant \delta^*(\mathcal{S}_1)$, *i.e.* the rarer the sought shape element is, the largest is the distance threshold for recognition.

We can give another formulation of the same property. Consider two databases of shape elements, $\mathcal{B}_1$ and $\mathcal{B}_2$. If a given target shape element $\mathcal{S}$ is rarer among the shapes elements in $\mathcal{B}_1$ than among those in $\mathcal{B}_2$, then we should have for all $i \in \{1, \ldots, K\}$:

$$\Pr\left(\mathcal{S}' \in \mathcal{B}_1 \text{ s.t. } d_i(x_i(\mathcal{S}), x_i(\mathcal{S}')) \leqslant d\right) \leqslant \Pr\left(\mathcal{S}' \in \mathcal{B}_2 \text{ s.t. } d_i(x_i(\mathcal{S}), x_i(\mathcal{S}')) \leqslant d\right).$$

This still yields: $\delta_2^*(\mathcal{S}) \leqslant \delta_1^*(\mathcal{S})$.
The conclusion is that the distance threshold proposed by our algorithm auto-adapts to the relative "rareness" of the target shape element with respect to the database shape elements. The "rarer" the target shape element is, the more permissive the corresponding distance threshold, and conversely.

9.3.2 An experimental verification

The aim of this experiment is to validate the previous claim. We search for the four shape elements extracted from the character 'm' (Figure 9.41) into 14 scanned pages, by using the semi-local similarity invariant recognition method.
We led two experiments: in the first one the database that was used to learn probabilities consisted of these 14 scanned pages (79, 376 shape elements), whereas in the second one the database was made of shape elements extracted from 21 'natural' images (98, 857 codes).
Figure 9.42 shows the shape elements from one of the 14 scanned pages that matched with the target shape elements, when probabilities are estimated over the scanned text database (notice that all 'm' are recognized).

Figure 9.43 shows the recognition result when the scanned text database is replaced by the natural image database. We can see that the recognition thresholds are more permissive in the second case (Figure 9.44). This result is fully coherent with the theory: in the first case, the focus is put on recognition of shapes elements that share some common structure with 'm' *among other characters*, that is to say other 'm', whereas in the second case, we are interested in recognizing shapes elements that share a common structure with 'm' *relative to a large universe of shape elements extracted from natural images*, that is to say other 'similar' characters (that is why we get italic 'm' and other "bad" matches).

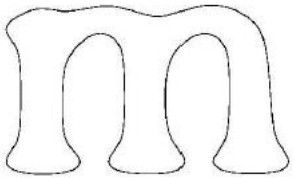
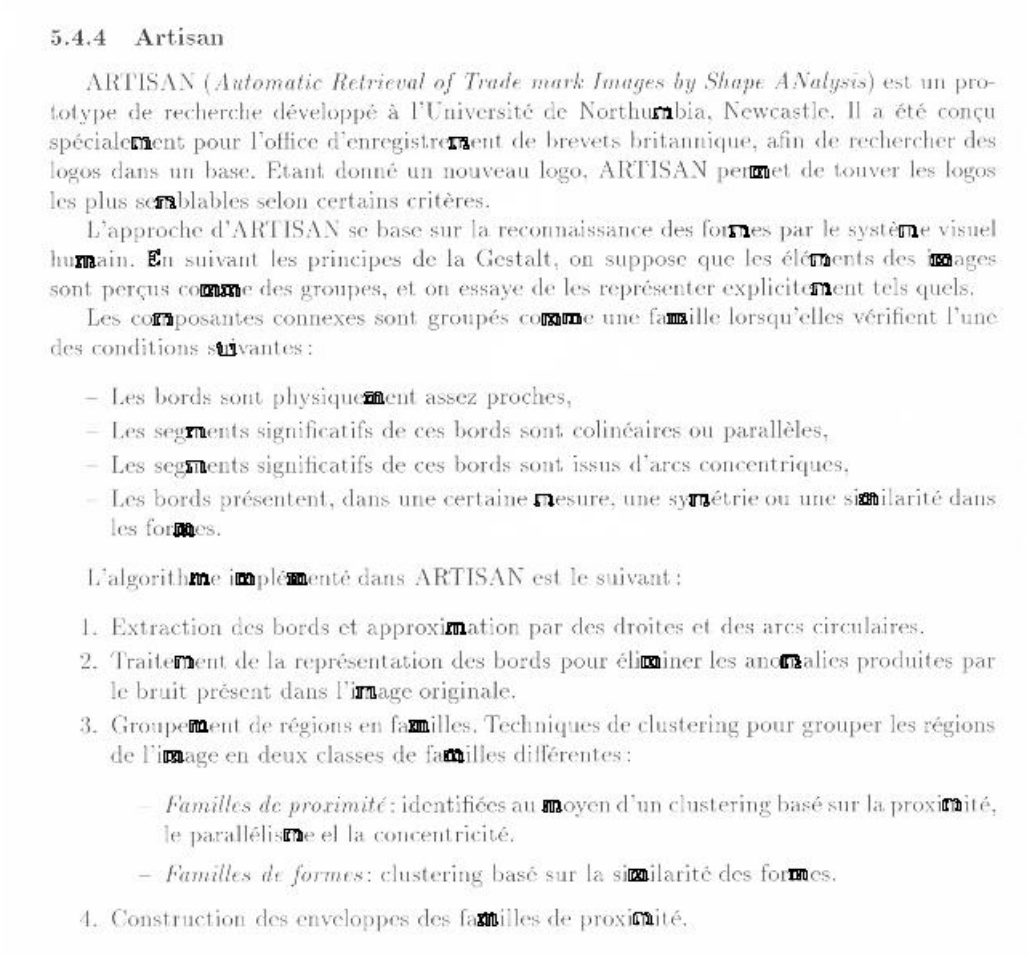


Figure 9.41: *Characters - the query curve.*



5.4.4 Artisan

ARTISAN (*Automatic Retrieval of Trade mark Images by Shape ANalysis*) est un prototype de recherche développé à l'Université de Northumbia, Newcastle. Il a été conçu spécialement pour l'office d'enregistrement de brevets britannique, afin de rechercher des logos dans un base. Etant donné un nouveau logo, ARTISAN permet de touver les logos les plus semblables selon certains critères.

L'approche d'ARTISAN se base sur la reconnaissance des formes par le système visuel humain. En suivant les principes de la Gestalt, on suppose que les éléments des images sont perçus comme des groupes, et on essaye de les représenter explicitement tels quels.

Les composantes connexes sont groupés comme une famille lorsqu'elles vérifient l'une des conditions suivantes :

- Les bords sont physiquement assez proches,
- Les segments significatifs de ces bords sont colinéaires ou parallèles,
- Les segments significatifs de ces bords sont issus d'arcs concentriques,
- Les bords présentent, dans une certaine mesure, une symétrie ou une similarité dans les formes.

L'algorithme implémenté dans ARTISAN est le suivant :

1. Extraction des bords et approximation par des droites et des arcs circulaires.
2. Traitement de la représentation des bords pour éliminer les anomalies produites par le bruit présent dans l'image originale.
3. Groupement de régions en familles. Techniques de clustering pour grouper les régions de l'image en deux classes de familles différentes :
   - *Familles de proximité* : identifiées au moyen d'un clustering basé sur la proximité, le parallélisme el la concentricité.
   - *Familles de formes* : clustering basé sur la similarité des formes.
4. Construction des enveloppes des familles de proximité.

Figure 9.42: *Characters - Recognition when probabilities are estimated over the database of scanned text pages.* 111 *matches were detected.*

5.4.4 Artisan

ARTISAN (*Automatic Retrieval of Trade mark Images by Shape ANalysis*) est un prototype de recherche développé à l'Université de Northumbia, Newcastle. Il a été conçu spécialement pour l'office d'enregistrement de brevets britannique, afin de rechercher des logos dans un base. Etant donné un nouveau logo, ARTISAN permet de touver les logos les plus semblables selon certains critères.

L'approche d'ARTISAN se base sur la reconnaissance des formes par le système visuel humain. En suivant les principes de la Gestalt, on suppose que les éléments des images sont perçus comme des groupes, et on essaye de les représenter explicitement tels quels.

Les composantes connexes sont groupés comme une famille lorsqu'elles vérifient l'une des conditions suivantes :

- Les bords sont physiquement assez proches,
- Les segments significatifs de ces bords sont colinéaires ou parallèles,
- Les segments significatifs de ces bords sont issus d'arcs concentriques,
- Les bords présentent, dans une certaine mesure, une symétrie ou une similarité dans les formes.

L'algorithme implémenté dans ARTISAN est le suivant :

1. Extraction des bords et approximation par des droites et des arcs circulaires.
2. Traitement de la représentation des bords pour éliminer les anomalies produites par le bruit présent dans l'image originale.
3. Groupement de régions en familles. Techniques de clustering pour grouper les régions de l'image en deux classes de familles différentes :
   - *Familles de proximité* : identifiées au moyen d'un clustering basé sur la proximité, le parallélisme el la concentricité.
   - *Familles de formes* : clustering basé sur la similarité des formes.
4. Construction des enveloppes des familles de proximité.

Figure 9.43: *Characters - Recognition when probabilities are estimated over the database of natural images database.* 154 *matches were detected.*

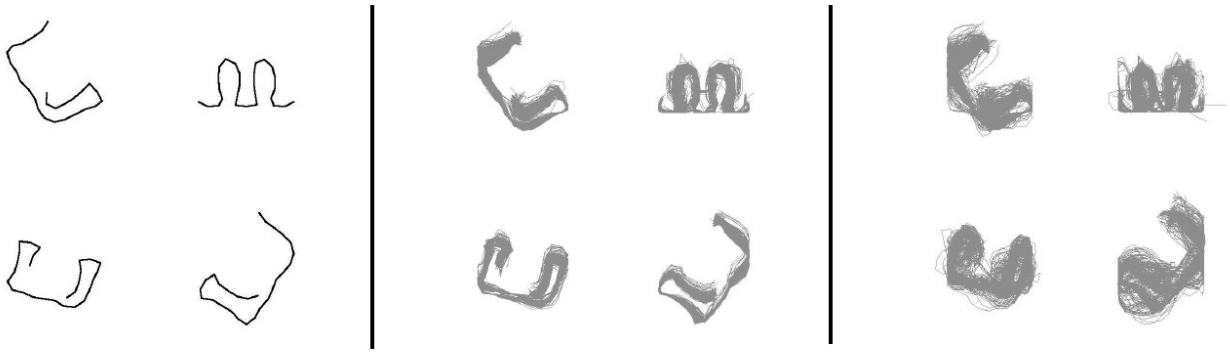


Figure 9.44: *Characters - Superimposition of the matched normalized codes. Left: the four target codes. Middle: all codes from the scanned text that match the corresponding target code, superimposed; probabilities were estimated using the* 14 *scanned pages. Right: superimposed matched codes when probabilities were estimated over the database of natural images. Notice that the matching threshold is larger in the latter case.*

Intermezzo

Meaningful matches based on alternative descriptions

Chapter 10

Shape elements comparison based on Principal Components Analysis

Abstract: The method presented in Chapter 8 can be adapted to shapes described by other features, whenever these features are (almost) statistically independent. When such a set of features is available, low numbers of false alarms can be reached. In this chapter we address the shape matching problem in this framework, by representing shape elements with sets of features provided by principal components analysis (PCA). Although these features are not necessarily independent, they are at least uncorrelated. We perform some experiments under the assumption that principal components analysis features are independent, making the usual *a contrario* decision still valid. These experiments show that a PCA method based on shape elements is not as well adapted as the strategy proposed in Chapter 8. The significance of these results is discussed.

Résumé : La méthode présentée dans le chapitre 8 peut être adaptée pour des formes décrites par d'autres types de caractéristiques, dès que celles-ci sont (suffisamment) indépendantes. De telles caractéristiques permettent d'atteindre des nombres de fausses alarmes bas. Dans ce chapitre nous abordons le problème de l'appariement de formes dans ce cadre, en représentant les éléments de forme par des caractéristiques issues d'une analyse en composantes principales (ACP). Bien que ces caractéristiques ne soient pas nécessairement indépendantes, elles sont au moins décorrélées. Nous menons quelques expériences sous l'hypothèse que les caractéristiques fournies par l'ACP sont indépendantes, ce qui rend valide la théorie usuelle de la décision *a contrario*. Ces expériences montrent qu'une méthode basée sur l'ACP n'est pas aussi bien adaptée que la stratégie proposée dans le chapitre 8. La signification des résultats obtenus est dicutée.

10.1 Facing the independence problem

As explained in Chapter 8, if the shapes are described through independent features, then we are able to compute a Number of False Alarms for shape matching that reaches very small values, and hence to derive an acceptation / rejection threshold for this problem. In order to correctly estimate this threshold, the statistical model that describes the shape database has to be as accurate as possible.

Although principal components analysis (PCA) provides uncorrelated variables, and not independent ones, a naïve first experiment could be to test if the PCA would not be suitable for computing a Number of False Alarms. In what follows we describe this setting, and related issues. The contents of this chapter is extracted from the article [MSM03].

10.1.1 Definition of a Number of False Alarms

The definition proposed here is slightly different from the one in Chapter 8.

Suppose that the problem is to decide whether a query shape element matches some shape elements in a database of cardinality N_B. We suppose that each normalized shape element is well described by n features $x_1, x_2, \ldots, x_n$, each of them belonging to a metric space (E_i, d_i), $1 \leqslant i \leqslant n$. Instead of referring to a shape element $\mathcal{S}$, we will refer to the correponding code $(x_1(\mathcal{S}), \ldots, x_n(\mathcal{S}))$ as well, which is an equivalent formulation.

Let $X = (x_1(\mathcal{S}), \ldots, x_n(\mathcal{S}))$ be a query code, and $Y = (y_1, \ldots, y_n)$ an element of the cartesian product $E_1 \times \cdots \times E_n$. Let $\delta_1, \ldots, \delta_n$ be n positive real numbers. We say that X and Y are $(\delta_1, \ldots, \delta_n)$-close if:

$$\forall i \in \{1, \ldots, n\},\ d_i(x_i, y_i) \leqslant \delta_i.$$

Two codes match if they are $(\delta_1, \ldots, \delta_n)$-close, with $\delta_1, \ldots, \delta_n$ small enough. As we can see, we must fix a threshold upon the δ_i.

Suppose moreover that, crucial hypothesis, the shape features are statistically independent. Then the probability that a code Y is $(\delta_1, \ldots, \delta_n)$-close to the query shape X can be expressed in the following way:

$$\Pr(Y \text{ s.t. } Y \text{ is } (\delta_1, \ldots, \delta_n)\text{-close to} X) = \prod_{i=1}^{n} \Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leqslant \delta_i). \tag{10.1}$$

We will denote this probability by $\mathcal{P}(X,\ \delta_1, \ldots, \delta_n)$.

Each term of the product is estimated over the database: for each i, one computes the distribution function of $d_i(z, x_i)$ when z spans the i^{th} feature of the shapes in the database, that is to say:

$$\Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leqslant \delta_i) = \frac{1}{N_B} \cdot \#\{\mathcal{S}' \in \mathcal{B}, d_i(x_i(\mathcal{S}'), x_i(\mathcal{S})) \leqslant \delta_i\},$$

where $\#\cdot$ denotes the cardinal of a finite set.
We then define an ε-meaningful match of a query code.

DEFINITION 10.1 *We say that a code $Y = (y_1, \ldots, y_n)$ is an ε-meaningful match of a query code $X = (x_1, \ldots, x_n)$ if one has:*

$$N_B \cdot \left(\max_{1 \leqslant i \leqslant n} \Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leqslant \delta_i) \right)^n \leqslant \varepsilon,$$

where $\forall i \in \{1 \ldots n\}, \delta_i = d_i(y_i, x_i)$.

Let us remark that this condition is equivalent to:

$$\forall i \in \{1 \dots n\}, \Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leqslant \delta_i) \leqslant \left(\frac{\varepsilon}{N_B}\right)^{\frac{1}{n}}. \tag{10.2}$$

We impose a uniform bound on the probabilities corresponding to each feature since there is no reason to differentiate between them.

As the functions $d \mapsto \Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leqslant d)$ are non-decreasing, they are pseudo-invertible. Thus there exist some maximum real numbers δ_i^* (depending on X and ε) such that

$$\delta_i^* = \max\{d > 0, \Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leqslant d) \leqslant (\varepsilon/N_B)^{1/n}.$$

As a consequence, if $\delta_i < \delta_i^*$, then inequality 10.2 holds. The proposition that follows is then straightforward.

PROPOSITION 10.1 *A code $Y = (y_1, \dots, y_n)$ is a ε-meaningful match with $X = (x_1, \dots, x_n)$ if: Y is $(\delta_1, \dots, \delta_n)$-close to X, with the real number δ_i satisfying:*

$$\forall i \in \{1 \dots n\}, \delta_i < \delta_i^*,$$

where $\forall i \in \{1 \dots n\}, \delta_i^* = \max\left\{d > 0, \Pr\left(y \in E_i \text{ s.t. } d_i(y, x_i) \leqslant d\right) \leqslant \left(\frac{\varepsilon}{N_B}\right)^{\frac{1}{n}}\right\}.$

The following proposition makes all these definition consistent, and shows that, if the NFA computation holds (namely the features are statistically independent), the number of detections should be bounded by ε. This is a much more handy way to control detections than tuning the distance thresholds δ_i^* by hand for each target code.

PROPOSITION 10.2 *If the NFA computation is valid, then the expectation of the number of ε-meaningful matches over the set of all shapes in the database is less than ε.*

Proof: Let us recall the proof given in Chapter 8.
Let X be the target shape, $Y_j = (y_1^j, \dots, y_n^j)$ $(1 \leqslant j \leqslant N_B)$ denote the codes from the database, and χ_j be the indicator function of the event e_j: "Y_j is an ε-meaningful match of X".
Let $R = \sum_{j=1}^{N_B} \chi_j$ the random variable which represents the number of ε-meaningful matches with X. The expectation of R is given by $E(R) = \sum_{j=1}^{N_B} E(\chi_j)$. Then, it follows from Proposition 10.1 that $E(R) = \sum_{j=1}^{N_B} \mathcal{P}(X,\ \delta_1^*, \dots, \delta_n^*)$. As a consequence: $E(R) \leqslant \sum_{j=1}^{N_B} \varepsilon \cdot N_B^{-1}$, leading to: $E(R) \leqslant \varepsilon$.
■

We can also give a quality measure for a match between two codes.

DEFINITION 10.2 *A code X and distances $\delta_1, \dots, \delta_n > 0$ being given, we call Number of False Alarms of X at a distance $\delta_1, \dots, \delta_n$:*

$$NFA(X,\ \delta_1, \dots, \delta_n) = N_B \cdot \mathcal{P}(X,\ \delta_1, \dots, \delta_n).$$

The number of false alarms of X at distances $\delta_1, \ldots, \delta_n$ should be an estimate of the number of codes that are $(\delta_1, \ldots, \delta_n)$-close to X among the database, *if the background model is true.*

In the framework we have just presented, we control the number of casual matches. If we want that these casual matches appear on the average at most once, we simply fix $\varepsilon = 1$. If the query is made of N_Q shape codes of equal importance, and if we want to detect on the average at most one casual match *over all possible matches*, we still set $\varepsilon = 1$ after replacing N_B with $N_B \cdot N_Q$ in Definition 10.1 (in this case, Proposition 10.2 still holds).

10.1.2 Modeling

We now describe how we extract the independent features out of shapes, in order to correctly estimate formula (10.1).

Normalized codes are extracted with the algorithm described in Chapter 7. They are made of 45 equally sampled points. These points are of course not independent (if the 20 first points match the corresponding points in another shape piece, then the following points also ought to match). Here is the process to build what we consider as independent features. Considering the whole database as a finite subset of $\mathbb{R}^{90}$, we compute its barycenter, and we center the database at this point. Then, the principal components analysis of this cloud of normalized shape elements is computed, leading to an orthonormal basis $\{e_1, \ldots, e_{90}\}$, where the e_i are sorted according to the decreasing variances (*cf* Figure 10.1). Precisely speaking, the basis $\{e_1, \ldots, e_{90}\}$ is made of the unitary eigenvectors (sorted up to the corresponding eigenvalues) of the positive symmetric matrix equal to $\sum_{i=1}^{N_B} (Y_i - \overline{Y}) \cdot (Y_i - \overline{Y})^{\mathrm{T}}$, where Y_i corresponds to the component row of a code from the database, and $\overline{Y}$ to the average of the Y_i over the database. Each normalized shape element in the database is projected on the subspace of dimension M spanned by $\{e_1, \ldots, e_M\}$: the list of coordinates in this frame constitutes the list of features $(x_1, \ldots, x_M)$. In the same way, features are computed out of the target code. With the notation of section 10.1, here we have: $(E_i, d_i) = (\mathbb{R}, |\cdot|)$.

Now, how to choose M? The number of components that are taken into account should be as large as possible, in order that the description is as complete as possible. Nevertheless, many components show a very small variance, so the corresponding coordinates does not provide much information, and may spoil the recognition task since they appear as a "noise" upon the real shape. Figure 10.1 shows that, while choosing $M = 10$ seems a bit arbitrary, it is quite reasonable.

10.2 Experiments

In all of the presented experiments, the semi-local similarity invariant encoding is used. Shape codes have a normalized length equal to 5, and are made of 45 points, issued from a regular sampling (geodesic distance) of the normalized shape element. We should mention that the shape encoding used in these experiments corresponds to an early version of the method, where flat points were not used. Consequently, some shapes were not retrieved simply because they were not encoded.

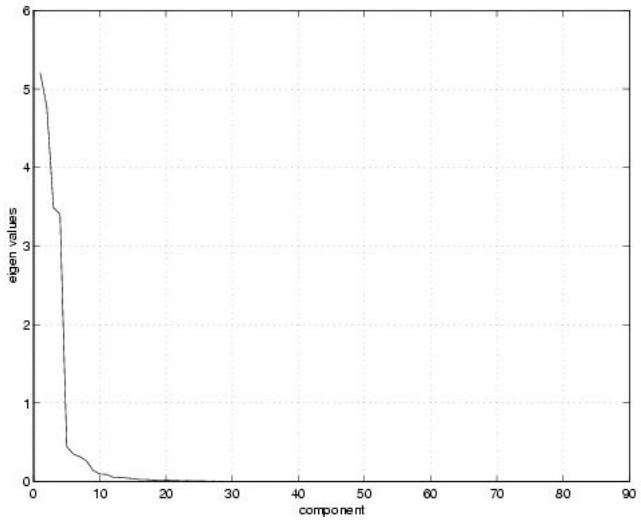


Figure 10.1: *The* 90 *eigenvalues from the principal components analysis (corresponding to the experiment of section 10.2.2), sorted in decreasing order.*

Nevertheless, this fact is not relevant here, since what we aim at showing with these experiments is that the PCA model is not well adapted for an accurate detection of meaningful matches between shape elements.

10.2.1 Checking the model

In order to check the estimate of the number of false detections, we have led the following experiment (see Chapter 8 for a description of explanation about the methodology). The shape database is made of $100,000$ random walks with independent increments (constant distance between two consecutive points, uniform assumption over the angle distribution). These codes are ensured to satisfy the independence assumption, what is not the case for codes extracted from level lines. Such codes are indeed constrained not to self-intersect, and can share common causality (such as parallelism): these properties introduce a bias in the estimates of the number of false alarms. The following table shows the estimated number of false alarms *versus* several values of the meaningfulness ε and of the number of components kept in the PCA. The number of detections and ε have the same order of magnitude, as predicted by the theory.

ε	1	10	100	1000
20 components	0.7	9.7	85	847
10 components	0.3	8.7	90	933
5 components	1	10	105	975

Principal components analysis therefore provides features that are independent enough to make the number of false alarms computation valid.

10.2.2 Shape matching

As an experiment, two (different) images of the same painting are compared (see Figure 10.2). Level lines are extracted, corresponding local codes are computed, and the PCA-based decision rule of this chapter is then applied. Results can be seen on Figure 10.3. Among the 975 codes of the query image, 26 are matched with at least one code of the database image (made of $38,700$ codes), leading to a total number of 53 matches. This means that some query codes match several database codes, which are in fact slight variations of the same code. We can see five "false alarms", showed on Figure 10.4.

Figure 10.2: *Top: query image and extracted level lines. Bottom: database image, and extracted level lines. These images are two different views of the same painting (*Saint-Georges and the dragon *by Paolo Uccello). In particular, contrasts of both images are not alike.*

Figure 10.3: *The* 53 1*-meaningful matches in target image. Some false alarm can be seen on the left.*

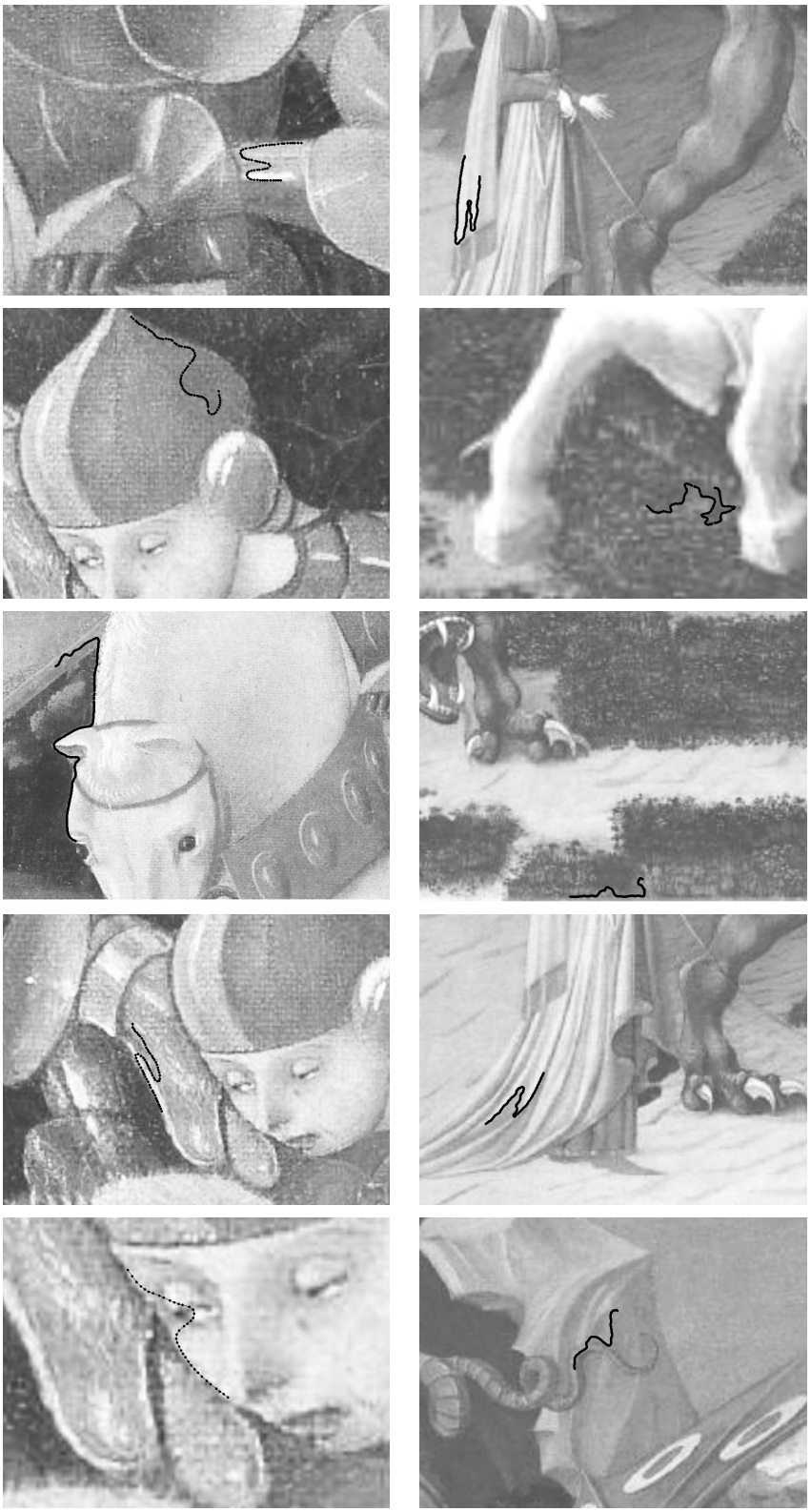


Figure 10.4: *The five* 1*-meaningful false alarms in the target image (on the left) and in the database image (on the right). As far as only geometry and not semantic is concerned, only the second and perhaps the fifth can be considered as true false alarms.*

10.2.3 Toy example: application to logo recognition

We show here the results of a logo recognition experiment. The target image is made of a logo (*cf* Figure 10.5). It is searched among the 21 images of a database, represented by $90,000$ codes. Figures 10.7 to 10.9 show the images of the database for which at least one 1-meaningful detection was found (left: target logo codes; right: corresponding codes in the database images). Some false alarms can still be seen. Nevertheless, none of these detections is 10^{-1}-meaningful, as proved by Figure 10.9. On the other hand, the number of false alarms of the "good" detections reaches values as low as $\varepsilon = 10^{-8}$.

Figure 10.5: *Target: "puma" logo. On the right: extracted level lines.*

10.3 Conclusion for the PCA-based model

The principal components analysis provides independent enough features to make the number of false alarms computation valid. However, results are not as good as they should be. The description is indeed not complete, as we cannot take into account all of the PCA components. We could perhaps have improved the algorithm by investigating Independent Components Analysis (ICA, see [HO00, Hyv99]), in order to make features more local. Nevertheless, PCA as ICA suffer from the same inherent drawback: they are correct under the strong assumption that the feature space is linear. This is clearly not true for the space of shapes. All these considerations illustrate how difficult it can be to find a set of features meeting the three requirements presented in Chapter 8:

1) Complete description: two shapes with the same features are alike.

2) Mutual statistical independence (more precisely speaking, independence for the distances between features).

3) Enough features in order to be able to reach low numbers of false alarms.

Looking for such a set of features led us to consider the strategy presented in Chapter 8, section 8.1.4, which gave, among several explored methods based on shape elements, the best trade-off in achieving these three requirements.

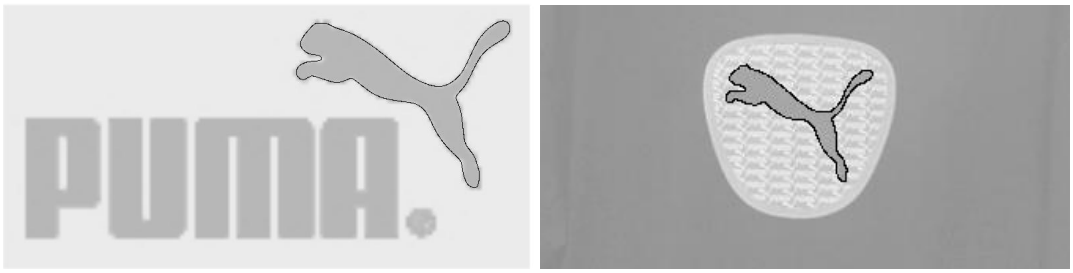
Figure 10.6: *The database.* 90, 000 *codes are extracted from these images.*

1 *detection*

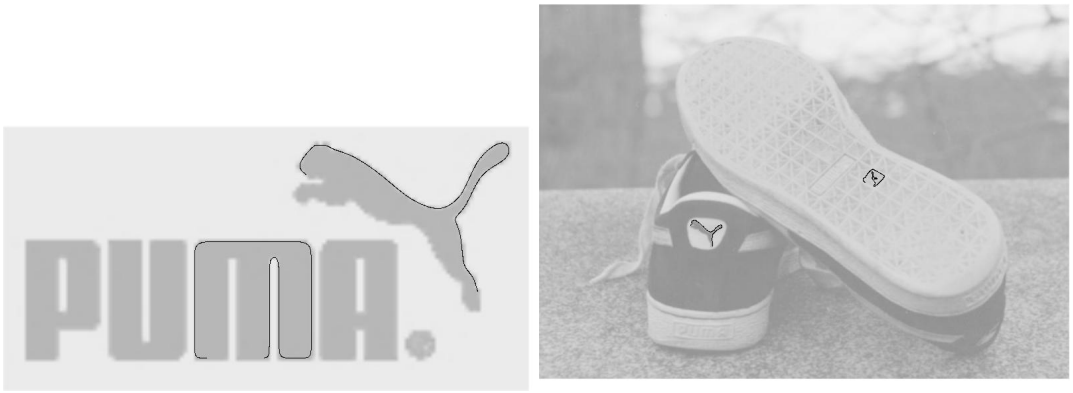
3 *detections*

20 *detections*

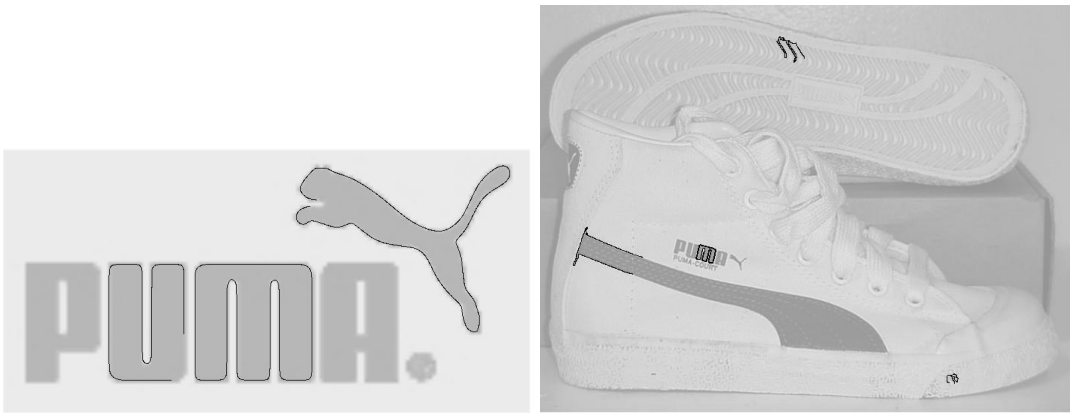
Figure 10.7: *Logo experiment:* 1*-meaningful matches.*



7 *detection*

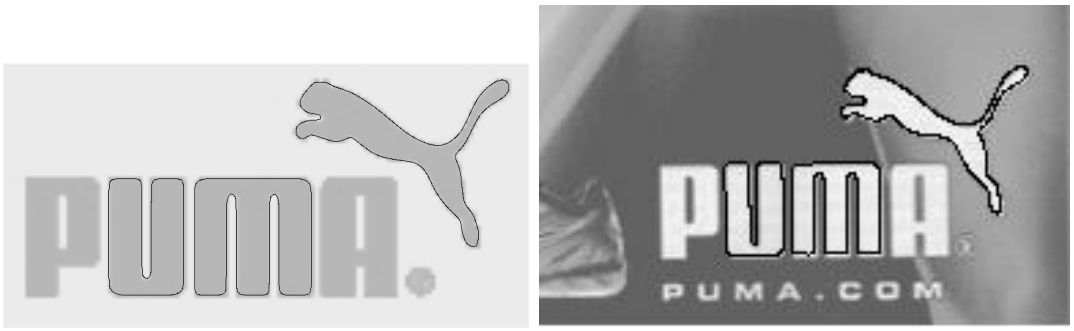


3 *detections*

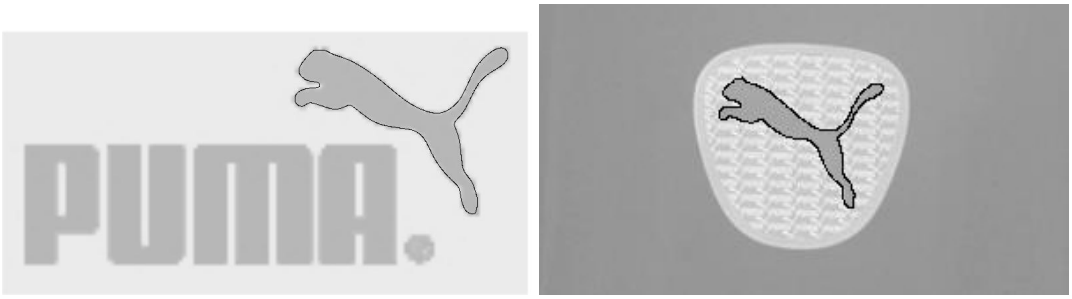


14 *detections*

Figure 10.8: *Logo experiment:* 1*-meaningful matches.*



18 *detection*



6 *detections*

1 *detection*

7 *detections*

Figure 10.9: *Logo experiment:* 10^{-1}*-meaningful matches. Relevant detections reach NFA values as low as (from top to bottom):* $\varepsilon = 10^{-8}$, $\varepsilon = 10^{-4}$, $\varepsilon = 10^{-2}$ *and* $\varepsilon = 10^{-2}$. *The two latest images show a strong difference on the scale with the target image. The corresponding level lines are in fact quite different, that is why so few local codes are matched.*

Chapter 11

Number of false alarms for size functions

Abstract: The proposed *a contrario* decision rule for shape matching is valid as soon as statistically independent features are provided. The theory is general enough in order that the decision rule does not depend on the kind of shape features that are matched. The shape features consist here in size function, and the model proposed in Chapter 8 is adapted in order to use uncorrelated random variables. The results seem to be still valid in spite of this adaptation and of the uncompleteness of the shape description. This chapter presents some preliminary results of a work in progress (Galileo project with P. Frosini's group in the University of Bologna).

Résumé : La règle de décision *a contrario* pour la reconnaissance de formes que nous avons proposée est valide dès qu'on dispose de caractéristiques statistiquement indépendantes. La théorie est suffisamment générale pour que la règle de décision ne dépende pas des caractéristiques en elles-mêmes. Les caractéristiques de forme sont ici des fonctions de taille, et le modèle proposé au Chapitre 8 est adapté pour l'utilisation de variables aléatoires décorrélées. Les résultats semble bien encore valables, malgré cette adaptation et l'incomplétude de la description des formes. Dans ce chapitre, nous présentons les résultats préliminaires d'un travail en cours (projet Galileo avec le groupe de P. Frosini à l'Université de Bologne).

Figures 11.1 to 11.4 are reproduced here by courtesy of Patrizio Frosini.

11.1 Size function theory in short

11.1.1 Size functions and their representation

P. Frosini's group proposes an original approach for describing and comparing shapes of topological spaces using size functions. A simplified description of the size function theory is given here, directed towards shape recognition. Articles [FL99], [FL01], and references therein present an overview of the general theory. Size functions can be seen (in particular) as tools providing information about the topology of any graph. Let G be a planar graph (given by the coordinates of its vertices in the plane,

and edges between them), and F be a measuring function on G (a function which associates to each vertex of G a non-negative real number). A size function is a mapping L from $\mathbb{R}^+ \times \mathbb{R}^+$ into $\mathbb{N}$ that associates to each couple (x, y) the number of connected components C of the graph G satisfying both following conditions:

- for each vertex v of G, one has $F(v) \leqslant y$;
- at least one vertex v_0 of G satisfies $F(v_0) \leqslant x$.

Of course, the size function L provides information only if $x \leqslant y$.

Let us illustrate this definition with a running example. Figure 11.1 presents an example of size function computation, and Figure 11.2 shows a handy way to represent size function. From Figure 11.3 one can argue that this representation is convenient for shape comparison. In that sense, a distance between size functions corresponds to some similarity or dissimilarity for the quality captured by the corresponding measuring function between two shapes.

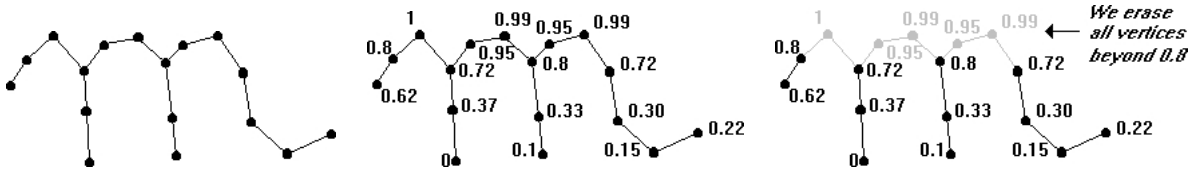


Figure 11.1: *An example of size function computation. From left to right: 1) a discretization of the character "m", seen as a graph; 2) Values of a measuring function over the graph vertices (it consists here in a kind of normalized ordinate); 3) There are* 3 *connected components in this graph containing at least one vertex whose measuring function value is not larger than* 0.5 *and upon which the measuring function values are under* 0.8*. Therefore we have:* $L(0.5, 0.8) = 3$.

To make the representation of size functions handier and more compact, *cornerpoints* are defined as any point $p = (x, y)$ such that:

$$\mu(p) := \min\{(L(x+\alpha, y-\beta) - L(x+\alpha, y+\beta)) - (L(x-\alpha, y-\beta) - L(x-\alpha, y+\beta)) \text{ s.t. } \alpha > 0, \beta > 0, x+\alpha < y+\beta\}$$

is positive.

The integer number $\mu(p)$ is called *multiplicity* of the corner point p. A size function representation is proved to be equivalent to giving the set of the corresponding cornerpoints with their respective multiplicity. Figure 11.4 illustrates this property.

11.1.2 Size functions and shape recognition

In the framework developed all along this thesis, shapes are represented by the boundaries of the connected components of level sets in an image (level lines, which are Jordan curves). To each level

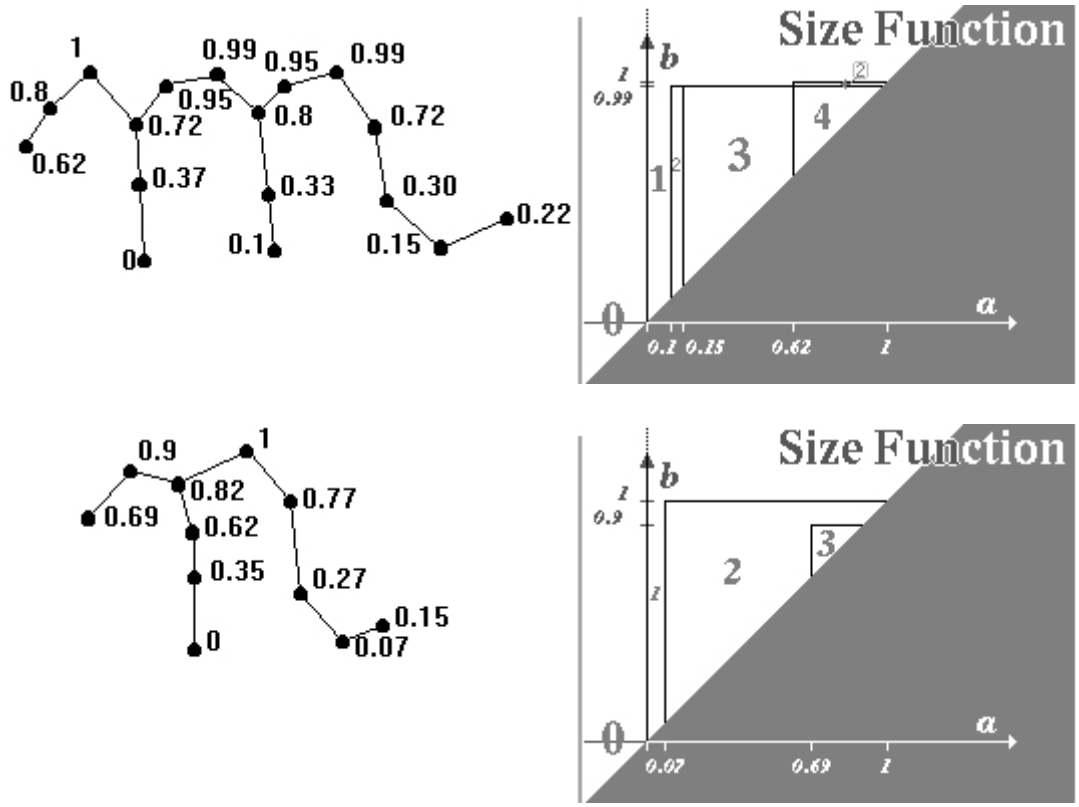


Figure 11.2: *Size function representation. The size function values are represented on the subset $x < y$. Left: measuring function values for the graph of figure 11.1 and for the graph corresponding to the discretization of a character "n". Right: associated size function representation.*

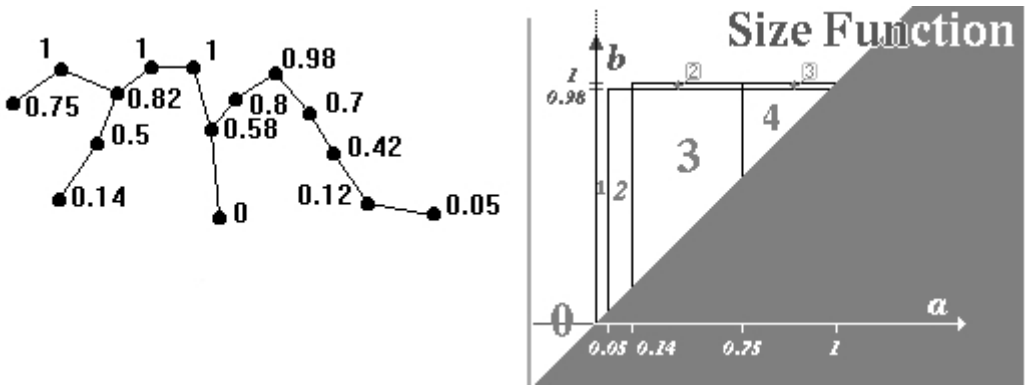


Figure 11.3: *Size function is robust with respect to variations of the measuring function. Here is shown the size function representation of another instance of the character "m". To be compared with figure 11.2 upper sketch.*

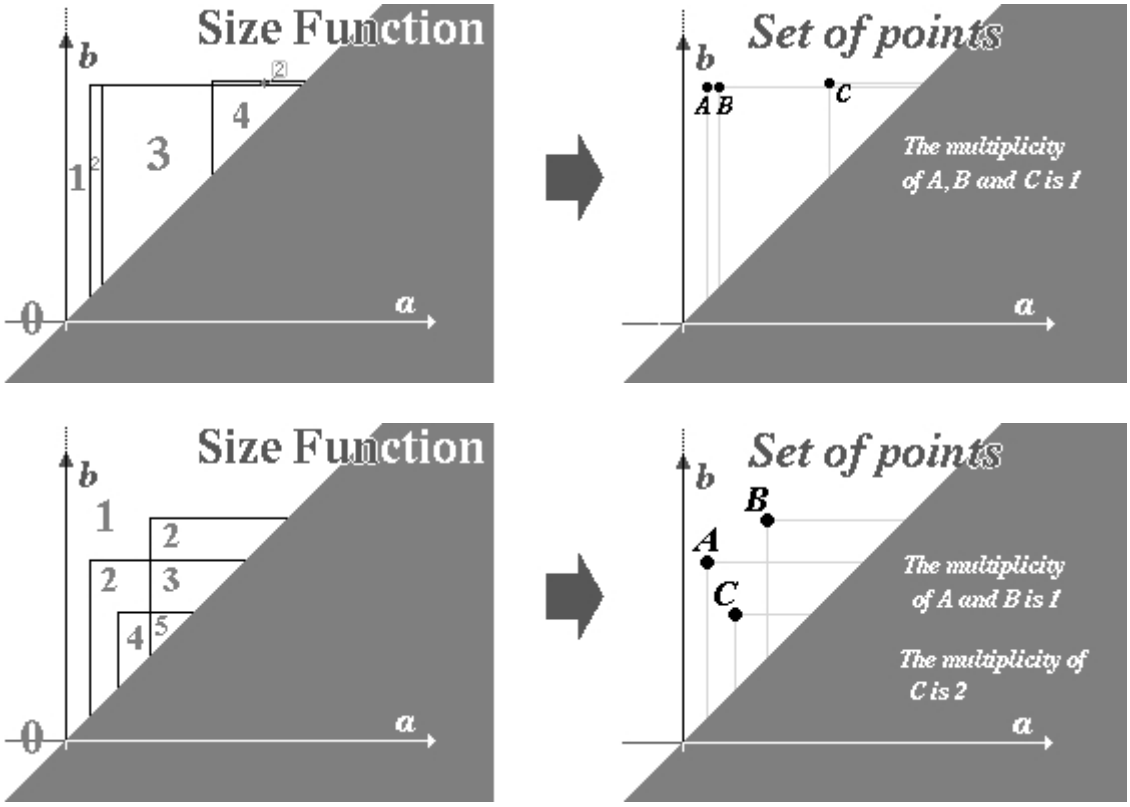


Figure 11.4: *Definition of corner points for size function, with their multiplicity.*

line is associated a "circular" graph whose vertices correspond to discretization points of the level line, and whose edges connect adjacent discretization points. It is thus possible to represent shapes (in the sense of this thesis) in the size function theory framework.

When applied to shape recognition, size function theory leads to (quasi) invariant descriptions, which are well adapted for perceptual matching (and not exact matching). Measuring functions are indeed devoted to measure topological properties of shapes, rather than purely geometric ones. See for example [HZW99] where sign language recognition using size function is addressed.

Size functions appear to be robust to occlusions, provided the occlusion prunes the considered graph without changing its connectivity. Let us notice that the underlying invariance is determined by the measuring function. For instance, the "normalized ordinate" used in Figure 11.1 leads to an invariance with respect to a change of position and scale along the y-axis. The provided description is not complete, in the sense that, a measuring fonction being given, two different graphs can be represented by the same size function (think of the measuring function of Figure 11.1: any horizontal translation of the vertices would lead to the same size function). Thus, representations based on several size functions (depending on different measuring functions) are needed, in order to increase the discriminatory power.

11.2 Proposing a number of false alarms for size functions

11.2.1 Three families of size functions for shape comparison

As said before, shapes are conceived as Jordan curves, and therefore represented by cyclic graphs. Three groups of measuring functions are designed:

1. The first group G_1 is made up of eight measuring functions d_k, $k = 1, \ldots, 8$, each of them representing the distance of the considered point on the curve from a fixed point P_k which lies on a spiral in the plane of the shape:

$$P_k = (\lambda_k \cos\theta_k, \lambda_k \sin\theta_k)$$

where: $\theta_k = \dfrac{k \cdot 2\pi}{n}$ and $\lambda_k = \dfrac{4 \cdot k \cdot d}{3 \cdot n}$ where $n = 8$, and d is the mean distance to the barycenter, divided by the number of points of the curve.

2. The second group G_2 consists of eight measuring functions r_k, $k = 1, \ldots, 8$, corresponding to distances to eight lines passing through the barycenter of the curve, defined by eight increasing angles formed with the canonical reference frame of the plane.

3. The measuring functions belonging to the third group G_3 are the derivatives of the eight distances from points defined in the first group (that is to say derivatives of the function $t \mapsto d(\mathcal{C}(t), P)$, where $\mathcal{C}(t)$ is an Euclidean curve parameterization, computed using a two steps formula).

These $3 \cdot 8 = 24$ measuring functions are expected to provide a complete enough description of shapes. Since shapes are related to their barycenter, the description is translation invariant.

11.2.2 Deriving a number of false alarms

Two shapes $\mathcal{S}$ and $\mathcal{S}'$ being given, the three distances between groups of size functions are computed: $d_1 = d(G_1(\mathcal{S}), G_1(\mathcal{S}'))$, $d_2 = d(G_2(\mathcal{S}), G_2(\mathcal{S}'))$, and $d_3 = d(G_3(\mathcal{S}), G_3(\mathcal{S}'))$. Each d_i is the average matching distance between size functions belonging to the group i ($i \in \{1, 2, 3\}$). The problem we are coming up against is to mix the information provided by d_1, d_2, and d_3. Since these distances do not share the same distribution at all, we cannot simply define the distance between two shapes as the average, or as the maximum of the three distances d_1, d_2, and d_3. A statistical model is called for.

Let us make the proposed approach more precise. Suppose that the problem is to search a query shape $\mathcal{S}$ among shapes in a database $\mathcal{B}$ (cardinal N). The three distances $d_i(\mathcal{S}, \mathcal{S}')$ between the query shape and a shape $\mathcal{S}'$ from the database have no reason to be mutually independent. In a similar manner as in chapter 10, the principal components of the set $\{(d_1(\mathcal{S}, \mathcal{S}'), d_2(\mathcal{S}, \mathcal{S}'), d_3(\mathcal{S}, \mathcal{S}')), \mathcal{S}' \in \mathcal{B}\}$ (which is a subset of $\mathbb{R}^3$) are computed. To each triplet $(d_1(\mathcal{S}, \mathcal{S}'), d_2(\mathcal{S}, \mathcal{S}'), d_3(\mathcal{S}, \mathcal{S}'))$ we thus associate the triplet $(D_1(\mathcal{S}, \mathcal{S}'), D_2(\mathcal{S}, \mathcal{S}'), D_3(\mathcal{S}, \mathcal{S}'))$ made of its coordinates in the principal components analysis basis. Although these components are not stricly speaking statistically independent, there are at least mutually uncorrelated.

Let us note, for each $i \in \{1, 2, 3\}$ and $d > 0$:

$$H_i(d) = 1/N \cdot \#\{\mathcal{S}' \text{ s.t. } D_i(\mathcal{S}, \mathcal{S}') \leqslant d\},$$

where $\#\cdot$ denotes the cardinality of a finite set.
Each H_i is an empirical estimate of the probability that a shape lies at a distance D_i less than d from the query shape $\mathcal{S}$. Assuming D_1, D_2, and D_3 are mutually statistically independent, the probability that there exists a shape at a distance D_1 less than d_1 according to the first group of size function, at a distance D_2 less than d_2 according to the second one, and at a distance D_3 less that d_3 according to the third one is simply given by the product:

$$H_1(d_1) \cdot H_2(d_1) \cdot H_3(d_3).$$

Thus, the number of false alarms associated to two shapes $\mathcal{S}$ and $\mathcal{S}'$ is derived as:

$$\text{NFA}(\mathcal{S}, \mathcal{S}') = N \cdot H_1(D_1(\mathcal{S}, \mathcal{S}') \cdot H_2(D_2(\mathcal{S}, \mathcal{S}')) \cdot H_3(D_3(\mathcal{S}, \mathcal{S}')). \tag{11.1}$$

We also define ε-meaningful matches of a query $\mathcal{S}$ as shapes $\mathcal{S}'$ belonging to the database such that $NFA(\mathcal{S}, \mathcal{S}') < \varepsilon$. As usual this yields that the expectation of the number of false ε-meaningful matches of a query shape $\mathcal{S}$ is less than ε.

11.2.3 Preliminary experimental results

The following experiment aims at testing the proposed methodology. It consists in searching characters in a scanned text. Level lines (a total number of 8907) are extracted from a scanned text. Although size functions are designed to deal with perceptual matching, we choose to lead this experiment in an exact matching framework so that the provided results can be more easily interpreted. In this case, the list of the characters that are expected to match with a query character is indeed not ambiguous. This is not so easy in a perceptual matching framework.

Figures 11.5, 11.6, and 11.7: searching character `t`.
Figures 11.8 and 11.9: searching an opening bracket `(` .
Figure 11.10: searching character `E`.
Figure 11.11: searching character `m`.
Figure 11.12: searching character `s`.
See captions for details.

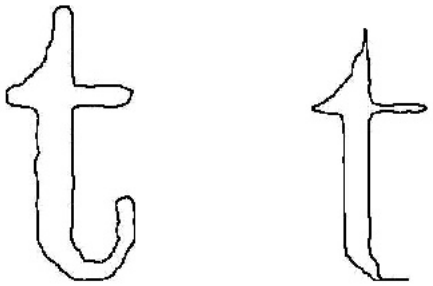


Figure 11.5: *Searching a character* `t`. *On the left: query shape, and on the right the misdetected* `t` *(see caption of figure 11.6). Not detecting the right curve when searching the left curve is not surprising.*

11.3 Tentative conclusion

The *a contrario* decision rule based on a background model estimated over the database seems to give qualitatively interesting results. The experiments we have just led show that, whatever the shape features are, provided they are independent enough, the Number of False Alarms computation still holds. These preliminary results are encouraging, since most of corresponding characters are correctly retrieved. The false matches show a higher Number of False Alarm than the correct ones. Now, the characters generally show a strong inter-cluster variability and a weak intra-cluster one. The lack of independence between PCA components and the lack of accuracy of size function description of shapes is possibly offset by the robustness of the clusters of characters.

Concerning size functions themselves, we have to work further in order to define several measuring functions capturing more shape information. It would improve the discriminatory power of the

method, by making the estimated NFA reach lower values. On the other hand, since increasing the number of size functions means possibly increasing the information redundancy, we should also quantify the independence between the provided features. Efforts will also be made in order to define classes of size functions which are invariant up to similarities.

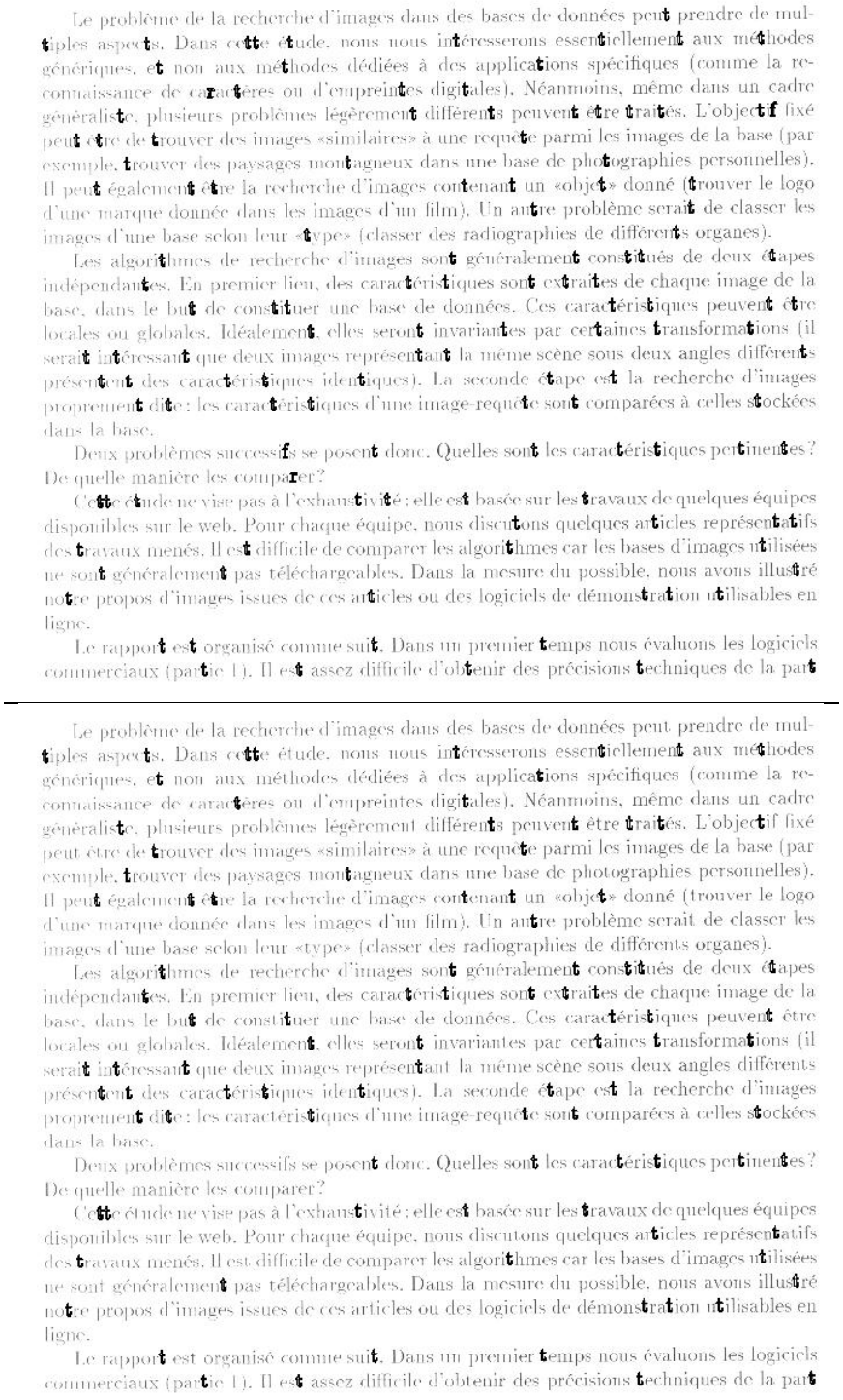


Figure 11.6: *Searching a character* `t`. *Top: the* 129 1*-meaningful matches. Bottom: the* 97 10^{-1}*-meaningful matches. Four false matches can be seen in the upper image: « caractère » (line 4, NFA:* 7 10^{-1}*), « objectif » (line 5, NFA:* 3 10^{-1}*), « successifs » (line 19, NFA:* 4 10^{-1}*), « comparer » (line 20, NFA:* 8 10^{-1}*). Only one* `t` *is missed, in the word: « téléchargeables » (row nb.* 24*). This character* `t` *is missed since the corresponding level line was actually wrongly extracted (see figure 11.5). In the lower image, no false match can be seen, but some true matches disappear. "Best" matches show a NFA as low as* 2 10^{-4}.

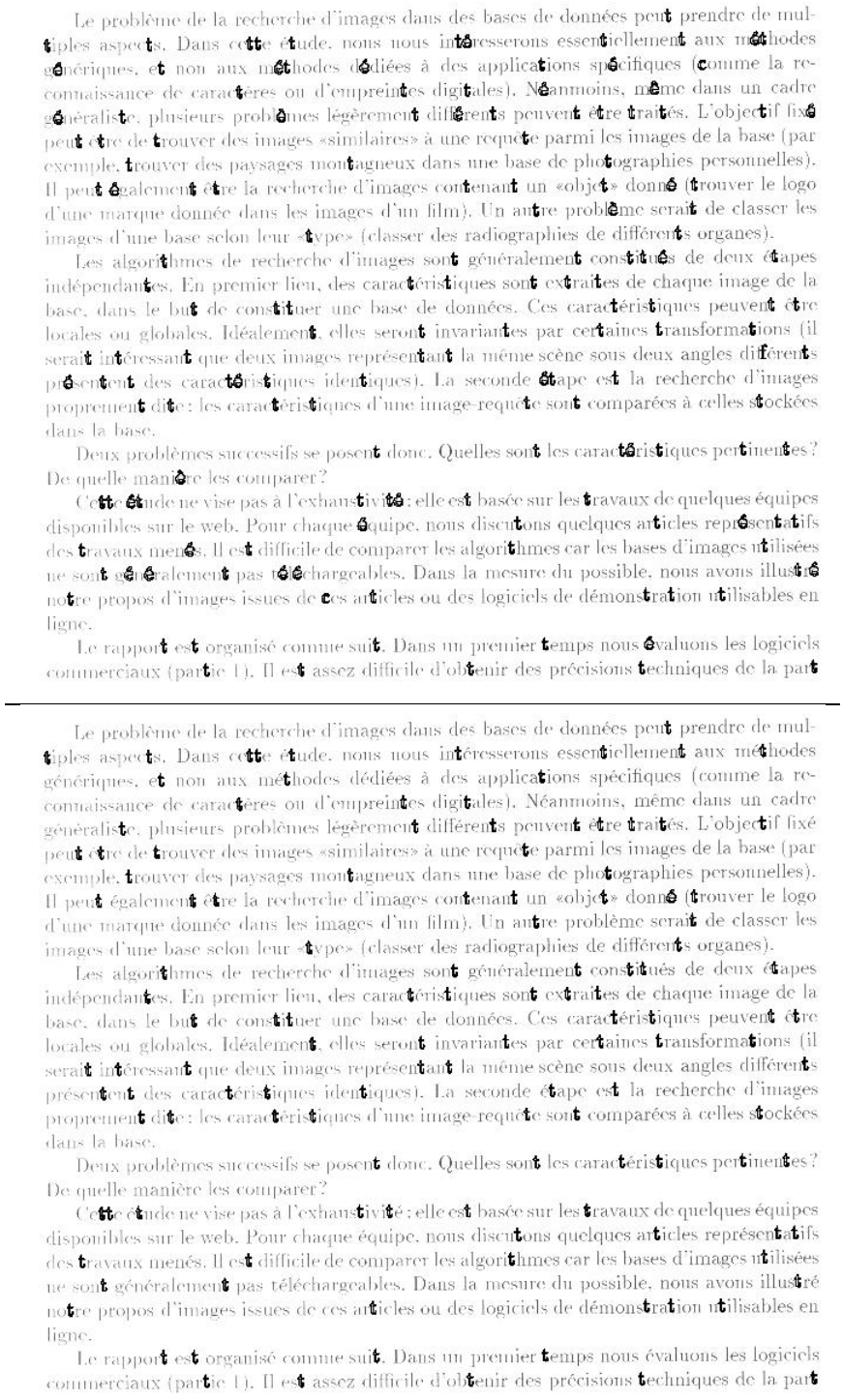
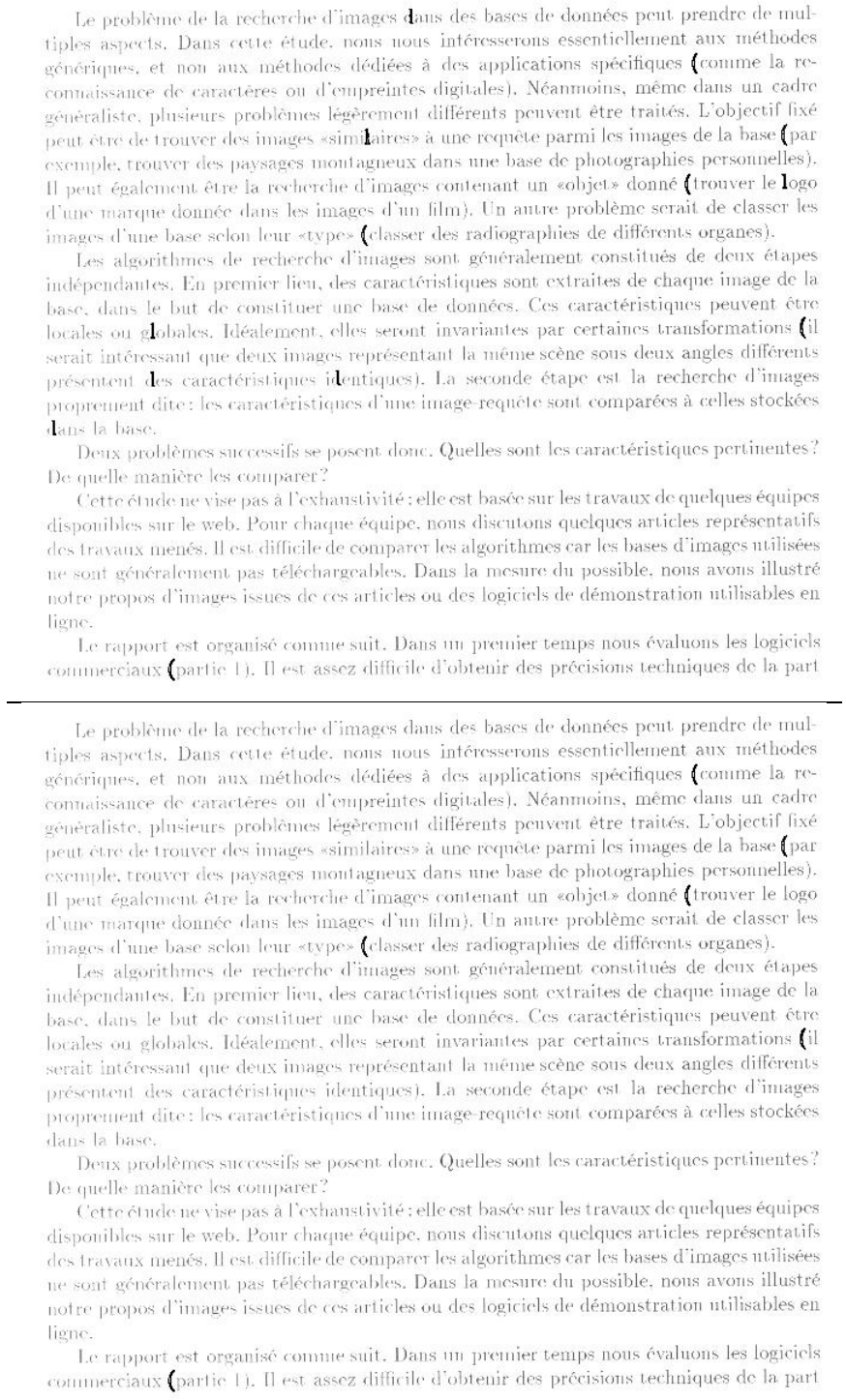


Figure 11.7: *Searching a character* t without Principal Component Analysis *(that is to say the D_i of section 11.2.2 are replaced by the d_i in the formula 11.1). Top: the* 162 1*-meaningful matches. Bottom: the* 127 10^{-1}*-meaningful matches. Many false matches can be seen among the* 1*-meaningful matches, and one false match can be still be seen among the* 10^{-1}*-meaningful matches: « donné » (row nb.* 8*, NFA:* 6 10^{-2}*). Best matches show a NFA equal to* 4 10^{-4}*. Principal Components Analysis actually improves the Number of False Alarms estimate.*



Le problème de la recherche d'images dans des bases de données peut prendre de multiples aspects. Dans cette étude, nous nous intéresserons essentiellement aux méthodes génériques, et non aux méthodes dédiées à des applications spécifiques (comme la reconnaissance de caractères ou d'empreintes digitales). Néanmoins, même dans un cadre généraliste, plusieurs problèmes légèrement différents peuvent être traités. L'objectif fixé peut être de trouver des images «similaires» à une requête parmi les images de la base (par exemple, trouver des paysages montagneux dans une base de photographies personnelles). Il peut également être la recherche d'images contenant un «objet» donné (trouver le logo d'une marque donnée dans les images d'un film). Un autre problème serait de classer les images d'une base selon leur «type» (classer des radiographies de différents organes).

Les algorithmes de recherche d'images sont généralement constitués de deux étapes indépendantes. En premier lieu, des caractéristiques sont extraites de chaque image de la base, dans le but de constituer une base de données. Ces caractéristiques peuvent être locales ou globales. Idéalement, elles seront invariantes par certaines transformations (il serait intéressant que deux images représentant la même scène sous deux angles différents présentent des caractéristiques identiques). La seconde étape est la recherche d'images proprement dite : les caractéristiques d'une image-requête sont comparées à celles stockées dans la base.

Deux problèmes successifs se posent donc. Quelles sont les caractéristiques pertinentes? De quelle manière les comparer?

Cette étude ne vise pas à l'exhaustivité ; elle est basée sur les travaux de quelques équipes disponibles sur le web. Pour chaque équipe, nous discutons quelques articles représentatifs des travaux menés. Il est difficile de comparer les algorithmes car les bases d'images utilisées ne sont généralement pas téléchargeables. Dans la mesure du possible, nous avons illustré notre propos d'images issues de ces articles ou des logiciels de démonstration utilisables en ligne.

Le rapport est organisé comme suit. Dans un premier temps nous évaluons les logiciels commerciaux (partie 1). Il est assez difficile d'obtenir des précisions techniques de la part

Le problème de la recherche d'images dans des bases de données peut prendre de multiples aspects. Dans cette étude, nous nous intéresserons essentiellement aux méthodes génériques, et non aux méthodes dédiées à des applications spécifiques (comme la reconnaissance de caractères ou d'empreintes digitales). Néanmoins, même dans un cadre généraliste, plusieurs problèmes légèrement différents peuvent être traités. L'objectif fixé peut être de trouver des images «similaires» à une requête parmi les images de la base (par exemple, trouver des paysages montagneux dans une base de photographies personnelles). Il peut également être la recherche d'images contenant un «objet» donné (trouver le logo d'une marque donnée dans les images d'un film). Un autre problème serait de classer les images d'une base selon leur «type» (classer des radiographies de différents organes).

Les algorithmes de recherche d'images sont généralement constitués de deux étapes indépendantes. En premier lieu, des caractéristiques sont extraites de chaque image de la base, dans le but de constituer une base de données. Ces caractéristiques peuvent être locales ou globales. Idéalement, elles seront invariantes par certaines transformations (il serait intéressant que deux images représentant la même scène sous deux angles différents présentent des caractéristiques identiques). La seconde étape est la recherche d'images proprement dite : les caractéristiques d'une image-requête sont comparées à celles stockées dans la base.

Deux problèmes successifs se posent donc. Quelles sont les caractéristiques pertinentes? De quelle manière les comparer?

Cette étude ne vise pas à l'exhaustivité ; elle est basée sur les travaux de quelques équipes disponibles sur le web. Pour chaque équipe, nous discutons quelques articles représentatifs des travaux menés. Il est difficile de comparer les algorithmes car les bases d'images utilisées ne sont généralement pas téléchargeables. Dans la mesure du possible, nous avons illustré notre propos d'images issues de ces articles ou des logiciels de démonstration utilisables en ligne.

Le rapport est organisé comme suit. Dans un premier temps nous évaluons les logiciels commerciaux (partie 1). Il est assez difficile d'obtenir des précisions techniques de la part

Figure 11.8: *Searching an opening bracket* **(**. *There are* 97 1*-meaningful matches (not shown). Top: the* 13 10^{-1} *meaningful matches. Bottom: the* 6 *detections which NFA is less than* 10^{-2} *(their NFA is in fact between* 5 10^{-5} *and* 2 10^{-4}*). It seems that the level line corresponding to the contour of character* **(** *is not precisely described by size functions. Many* 10^{-1}*-meaningful matches are false. Nevertheless, there is a "gap" between true matches (NFA about* 10^{-4}*) and false matches (NFA greater than* 10^{-2}*).*

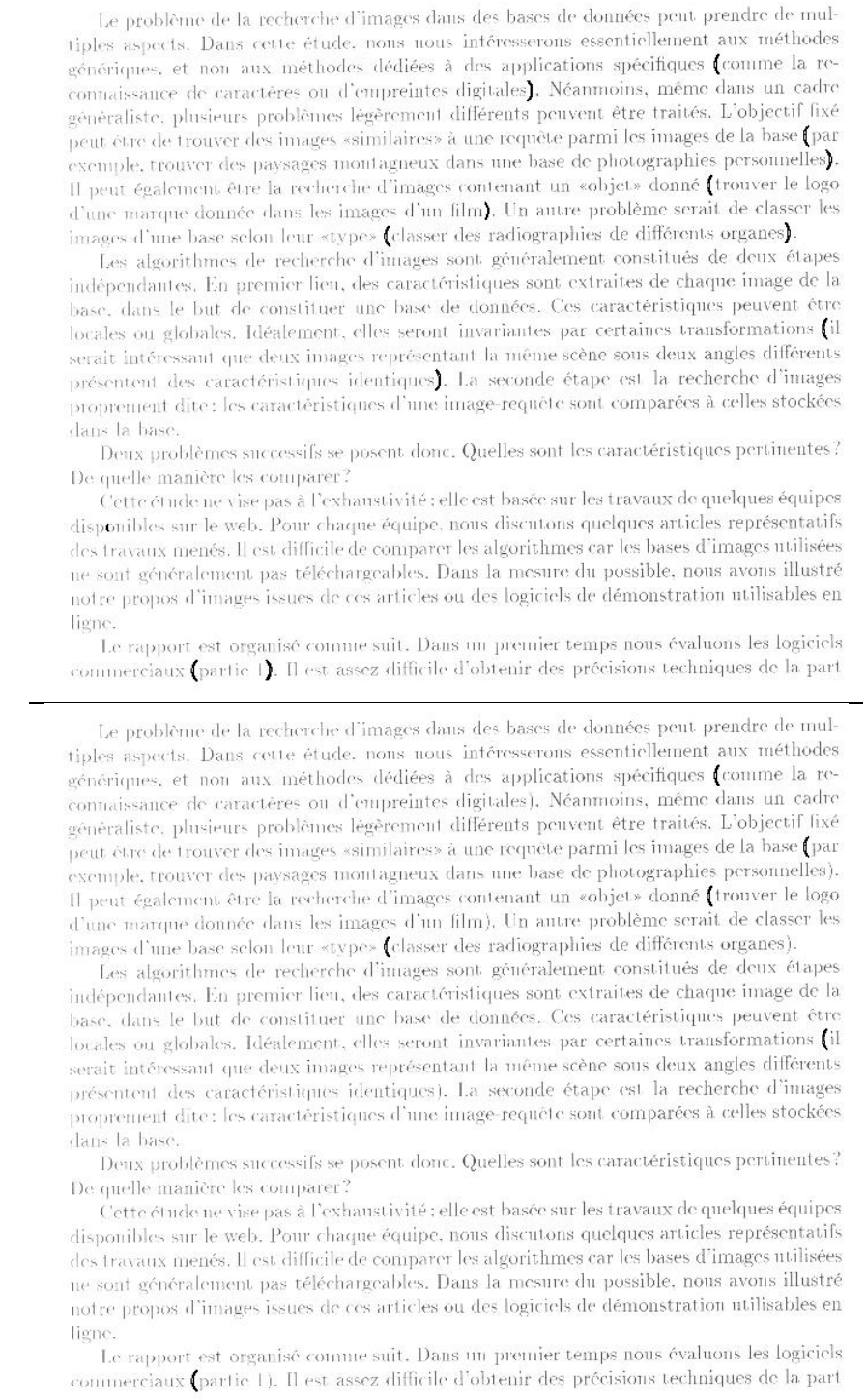


Figure 11.9: *Searching an opening bracket* (without the principal components analysis step. *Top: the* 10^{-2} *meaningful matches. Bottom: the detections which NFA is less than* 10^{-5}*. The NFA estimate is bad, since there are many false matches with a very low NFA (let us remark that these false matches surprisingly correspond to closing brackets). Nevertheless, the NFA is still a pertinent ranking criterion: the matches that show the lowest NFA are indeed opening brackets. Once again, we can notice a gap between false matches NFA* (10^{-2}) *and true matches NFA* (10^{-5})*.*

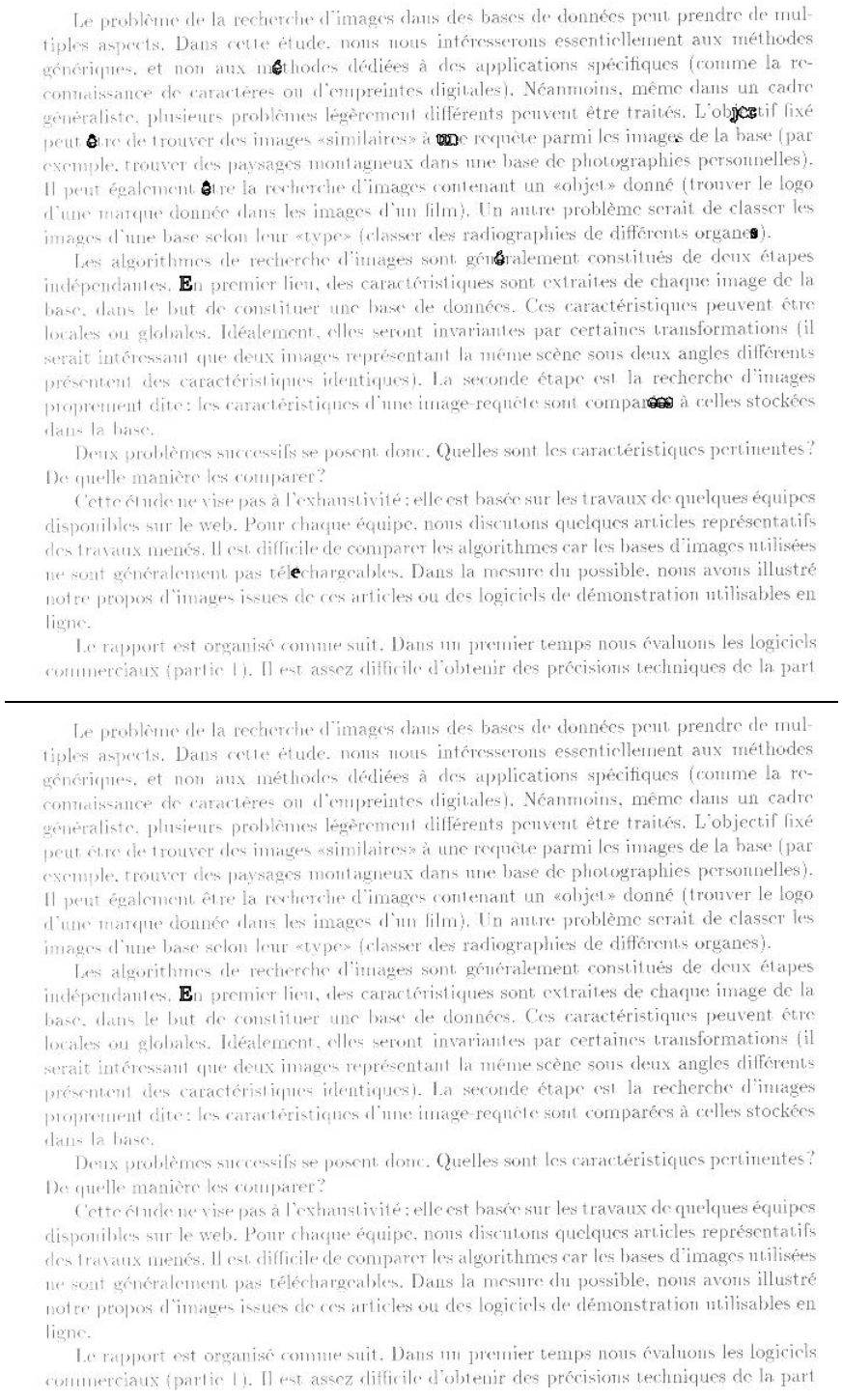


Figure 11.10: *Searching character* E. *Top: the* 11 1*-meaningful matches. Bottom: the single* 10^{-1}*-meaningful match.*

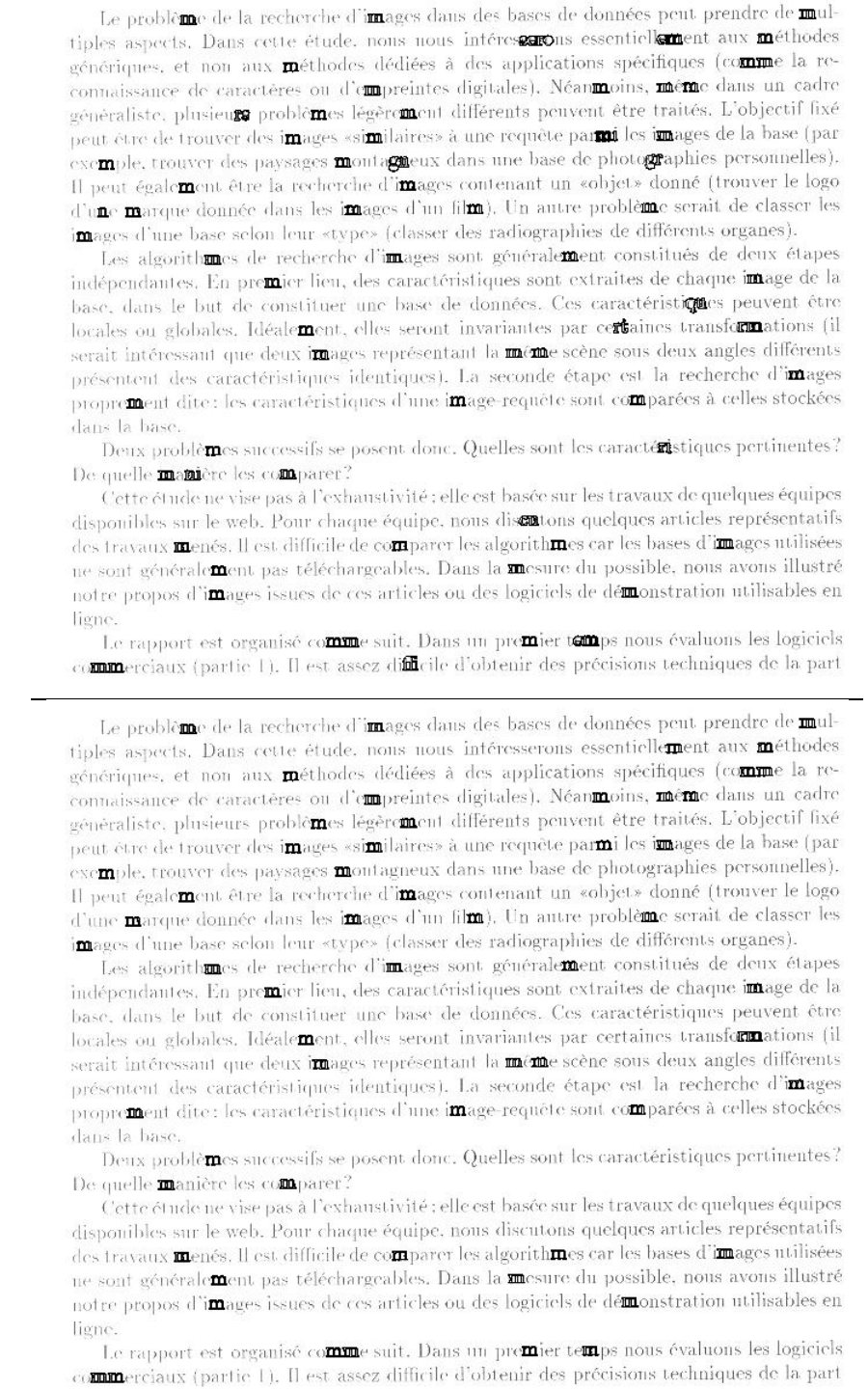


Figure 11.11: *Searching character* m*. Top: the* 74 1*-meaningful matches. Some false matches can be seen among them. Bottom: the* 59 10^{-1}*-meaningful matches. All false matches disappear, except for « transformations » (NFA:* 3 10^{-2}*). The explanation of such a false match with a not so low NFA is that the corresponding level lines are actually similar. Moreover, all of the* m *correspond to* 10^{-1}*-meaningful matches. Best matches NFA is about* 10^{-5}*.*

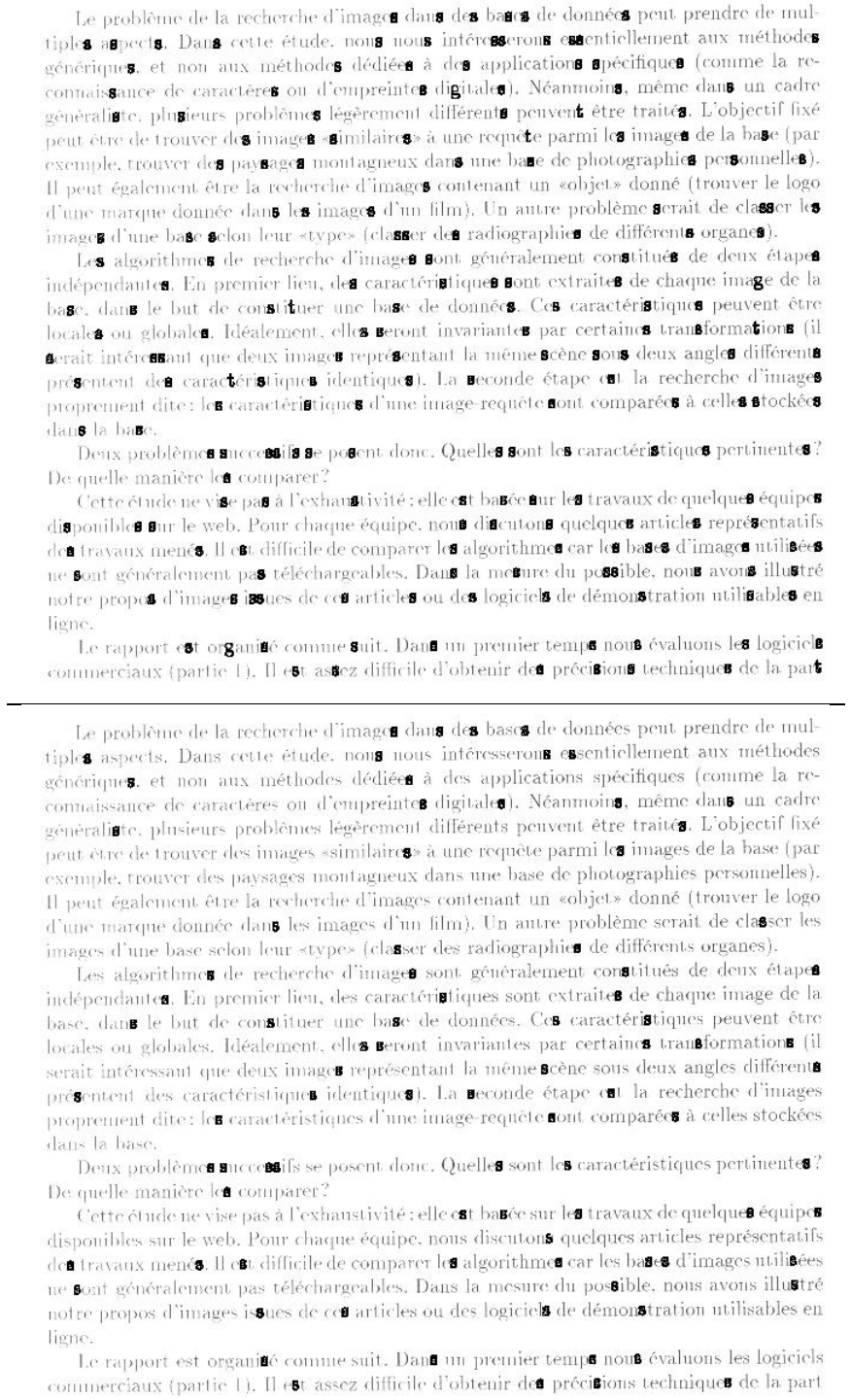


Figure 11.12: *Searching character* s. *Top: the* 232 1*-meaningful matches. All* s *are detected, but some false matches can be seen among them. Bottom: the* $88\ 10^{-1}$*-meaningful matches. No more false matches can be seen, but some true matches are missed. Best matches NFA is equal to* $4\ 10^{-4}$.

Part II

SHAPE RECOGNITION AS A GROUPING PROCESS

Chapter **12**

Hierarchical clustering and validity assessment

Abstract: The unsupervised classification of patterns into groups is commonly referred as *clustering* or *grouping*. Clustering aims at discovering structure in a data set, by dividing it into its "natural" groups. Most of the clustering methods are either partitional, either hierarchical methods. While partitional methods produce a single partition of the data, hierarchical methods produce a nested series of partitions. Agglomerative hierarchical methods build these nested partitions by recursively merging two groups. Thus, "stopping rules" have to be defined in order to extract, among the nested structure, the partition providing the best data representation. Since in general, clustering algorithms always produce clusters, whether they do exist or not, an assessment of the detected groups is needed. This is the aim of *cluster validity* analysis.

In this chapter we present a method to detect natural groups in a data set, based on hierarchical clustering. A measure of the meaningfulness of clusters, derived from a *background model* assuming no structure in the data, provides a way to compare clusters, and leads to a cluster validity criterion. This criterion is applied to every cluster in the nested structure. While all clusters passing the validity test are meaningful in themselves, the set of all of them does not necessarily reveals the structure of the data set. However, by selecting a subset of the meaningful clusters, a good data representation can be achieved. We propose a method combining a new merging criterion (also derived from the *background model*) with a selection of local maxima of the meaningfulness with respect to inclusion in the nested hierarchical structure.

Résumé : La classification non supervisée de motifs est communément appelée *clustering*, ou *grouping*. Le but du *clustering* est la découverte de structures dans des ensembles de données, en la divisant en groupes « naturels ». La plupart des méthodes de *clustering* construisent soit une partition, soit une structure hiérarchique. Alors que les premières produisent une unique partition des données, les secondes construisent une suite de partitions emboîtées par des fusions successives de deux groupes. Ainsi, des « rêgles d'arrêt » doivent être définies afin d'extraire, dans cette structure emboîtée, la partition donnant la meilleure représentation des données.

Comme, en général, les algorithmes de *clustering* produisent toujours des groupes, qu'ils existent effectivement ou pas, il est nécessaire d'évaluer la validité des groupes détectés.

Dans ce chapitre, nous présentons une méthode pour détecter les groupes naturels dans un ensemble de don-

nées, qui est basée sur un *clustering* hiérarchique. Une mesure de la significativité des groupes est déduite d'un *modèle de fond* construit en supposant que les données ne présentent aucune structure. Elle permet de les comparer entre eux, et conduit à un critère de validité. Tous les groupes satisfaisant le critère sont significatifs, néanmoins leur réunion ne révèle pas nécessairement la structure de l'ensemble de données. Cependant, en sélectionnant un sous-ensemble des groupes significatifs, une bonne représentation des données peut être obtenue. Nous proposons une méthode combinant un nouveau critère de fusion (également déduit du modèle de fond) et une sélection des maxima locaux de la significativité par rapport à l'inclusion.

12.1 Clustering analysis

Most detection or recognition problems can be posed as classification or categorization tasks. A data set being given, two different situations may occur: 1) classes have already been defined, and one has to identify each data item ("patterns") as a member of one of these classes; 2) each pattern is assigned to a hitherto unknown class. The first classification task, known as supervised classification, assumes a certain knowledge of the data leading to the definition of classes (see [DHS00, HTF01, JDJ00] for an overview and a state of the art in supervised classification). In this chapter we will concentrate on the second situation, since it is the one that corresponds to the detection problems we are concerned with. In such situations, there is little if any prior information available about the data, and one is led to classify patterns (the data items) making as few assumptions as possible. This task is known as unsupervised classification or data clustering. The goal is to find "natural" groupings in a set of data, so that patterns within each cluster are more closely related to one another than to patterns assigned to different clusters.
Typical clustering methods consist of the following steps [JMF99]:

1. pattern representation (preceded by feature extraction and selection, if needed),
2. definition of a similarity measure between patterns,
3. clustering or grouping,
4. data abstraction (if needed),
5. assessment of output (if needed).

The first step deals with pre-processing the rough data in order to extract an appropriate set of features, and to build the patterns (usually feature vectors) that will be used in clustering. *Feature extraction* techniques compute features from the original data, and *feature selection* consists in identifying a subset of these features for subsequent use. A good feature selection method should select the subset of features leading to the smallest classification error. Feature selection has been widely studied in the statistical pattern recognition field [JZ97].

The second and the third steps are the central core of all clustering methods. As we just said, the goal is to find natural groupings in data; therefore, we need to specify in what sense patterns in one cluster are more similar to one another, than to patterns in other clusters. The first issue in this specification is the definition of a *notion of similarity or dissimilarity* between patterns. In order to find natural groupings, this definition should be specially adapted to the particular problem. The most common approach to measure pattern dissimilarity is to consider a distance function defined on the feature space, but in its general form, a dissimilarity measure does not need to be a metric [DHS00, JMF99, KR90a]. Defining a metric between patterns is not trivial. Minkowski metrics (the ℓ_p-norms) are among the most popular dissimilarity measures. Theoretically, these metrics do not perform well unless the feature space is close to isotropic and features are spread roughly evenly along all directions. Linear correlation between features can also distort distance measures. A commonly used approach to solve these problems is to normalize the data and to decorrelate it prior to clustering (a whitening transformation), or, equivalently, to directly compute the Mahalanobis distance on the original data [Sma96, DHS00]. However, this procedure is just appropriate for normally distributed data, and can lead to particularly bad results when applied to multimodal distributions (see Figure 12.1). Often practitioners use this metrics in an abusive manner. A frequent mistake relative to dissimilarity measures consists in defining metrics as norms on feature spaces that do not exhibit a vector space structure (e.g., the 4-D parameter space of similarity transforms we will consider in Chapter 13).

We will not discuss dissimilarity measures further in this chapter, since this aspect strongly depends on domain knowledge specifics. Concerning the *grouping* step, hundreds of algorithms have been reported in the literature, but most of these algorithms can be classified as one of these two clustering techniques [KR90a, JMF99, DHS00]:

- *Partitional algorithms* identify the partition that optimizes a clustering criterion (e.g. minimum variance partition),
- *Hierarchical algorithms* produce a hierarchical representation, in which each level of the hierarchy is itself a partition on the data, whose clusters were obtained by merging clusters at the next lower level.

We will describe both techniques more precisely in the following subsection. Let us close this subsection with a few words on the last two steps of a general clustering method.

In the cluster analysis context, *data abstraction* consists in extracting a compact description of each cluster, usually a representative pattern, like its barycenter or centroid (when the notion of barycenter makes sense), or its medoid (the pattern of the cluster for which the average dissimilarity to all the patterns of the cluster is minimal). A set of representative patterns not only provides a characterization of the data, but it can often be used for further work (*e.g.* in Chapter 13, we will be led to characterize groups of spatially coherent meaningful matches, by representative transformations aligning these groups).

Finally, the last step of a generic clustering method is *cluster validity assessment*. All clustering algorithms produce clusters, whether they do exist or not. Another critical issue is the selection of the number of clusters in the final solution. In some applications, assuming a known number of clusters makes sense, but in general this is not the case, specially if we are exploring data whose properties are unknown. Cluster validity assessment deals with this kind of problems. In section 12.3 we will discuss some cluster validity techniques. As we will see, the majority of them are *ad hoc* procedures, and the statistical problem of testing cluster validity is still essentially unsolved [DHS00, JMF99].

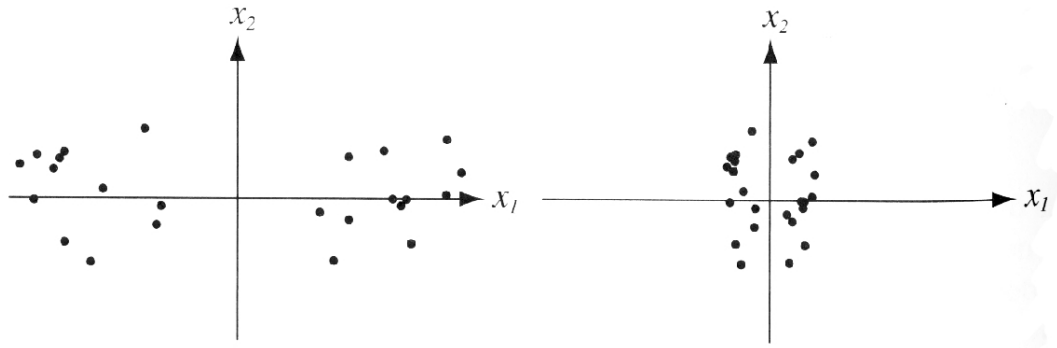


Figure 12.1: *(From [DHS00]) In the original data (left), points fall into two well separated clusters. Data normalization to zero mean and unit variance (right) reduces the separation, and clustering methods may no longer be able to give the appropriate data representation.*

12.2 Clustering techniques

As we just said, most of the clustering algorithms are either partitional, either hierarchical methods. While partitional methods produce a single partition, hierarchical methods produce a nested series of partitions. In this sense, they provide a totally different data description and should not be considered as two competiting techniques. However, as we will see, because of their different nature, the corresponding strategies for cluster validity assessment may be quite different.

Cluster methods have been and are still the object of applied and theoretical research in many different fields, such as statistical pattern recognition, data mining, image processing, biomedical sciences, etc. It is not the aim of this section to present a complete overview of clustering techniques, but just to provide enough information to justify why we are led to choose a particular technique (we should keep in mind that there is no universal "best" clustering algorithm, and choices and compromises have to be made). A good review of clustering techniques by Jain *et al.*, from a statistical pattern recognition viewpoint, can be found in [JMF99]. The main concepts can also be found in Duda and Hart [DHS00], Hastie *et al.* [HTF01] and Kaufman and Rousseeuw [KR90a] textbooks.

12.2.1 Partitional clustering methods

Let us denote by T_k a pattern (a D-dimensional feature vector), by $\mathcal{T} = \{T_k,\ k \in \{1,\dots,M\}\}$ the data set, and by $d_T : \mathcal{T} \times \mathcal{T} \to \mathbb{R}^+$ the dissimilarity measure. Assuming for the moment that the partition size c is given, the goal of a partitional clustering algorithm is to identify the partition $\mathcal{P}(\mathcal{T}) = \{\mathcal{T}_1,\dots,\mathcal{T}_c\}$ on $\mathcal{T}$ that optimizes a criterion function. We will not address here the family of methods based on mixture decomposition, since we assume we do not have any knowledge on the underlying probability distribution. (In these methods, the data set is assumed to be drawn from a mixture of c underlying parametric distributions, and the goal is to determine the involved parameters; the standard algorithm is the Expectation-Maximization algorithm [DLR77].) Hence, since there are approximately $c^M/c!$ ways of partitioning a set of M elements into c subsets (a Stirling number of the second kind), optimizing the criterion function by exhaustive search is intractable and iterative optimization procedures are needed.

The simplest and most widely used family of criteria function is the one of related minimum variance criteria [DHS00, KR90a]. The energy to be minimized here is

$$E = \frac{1}{2}\sum_{m=1}^{c} n_m \langle d_m \rangle,$$

where n_m is the number of points in the m-th cluster, and

$$\langle d_m \rangle = \frac{1}{n_m^2}\sum_{T_i \in \mathcal{T}_m}\sum_{T_j \in \mathcal{T}_m} d_T(T_i, T_j)$$

is the average dissimilarity measure between points in the m-th cluster. If $\mathcal{T}$ was a subset of a vector space, and d_T was the squared Euclidean distance, the resulting criteria would be the sum of variances of each clusters,

$$\sum_{m=1}^{c}\sum_{T \in \mathcal{T}_m} \|T - \langle T_m \rangle\|_2^2, \quad \text{where } \langle T_m \rangle = \frac{1}{n_m}\sum_{T \in \mathcal{T}_m} T.$$

Strictly speaking, this criterion only makes sense when clusters are isotropic, multivariate normally distributed. Moreover, the solution is not invariant to linear transformations of the data. Many variations on this method exists, taking any Minkowski metric or the squared Mahalanobis distance instead of the squared Euclidean distance [JMF99]. Notice however that all these methods are based on the notions of medoid or centroid (barycenter) of a set of points and, as we said earlier, this does not make sense unless patterns live in a vector space.

Related minimum variance criteria suffer from the problem that partitions that split large clusters may be favored over ones that maintain the integrity of natural clusters [DHS00]. When natural clusters have very different number of points, the partition minimizing this criteria may not reveal the intrinsic structure of the data (see Figure 12.2). Another weakness of these methods is the lack of ability to extract a very dense cluster embedded in the center of a diffuse cluster. Besides, the partition solution has to be found by iterative optimization procedures. These iterative procedures, which are nothing but c-means or c-medoids like procedures [DHS00, HTF01] (also referred in the literature

as k-means or k-medoids), are to be initialized by a reasonable initial partition and solution can be trapped in local minima [JMF99].

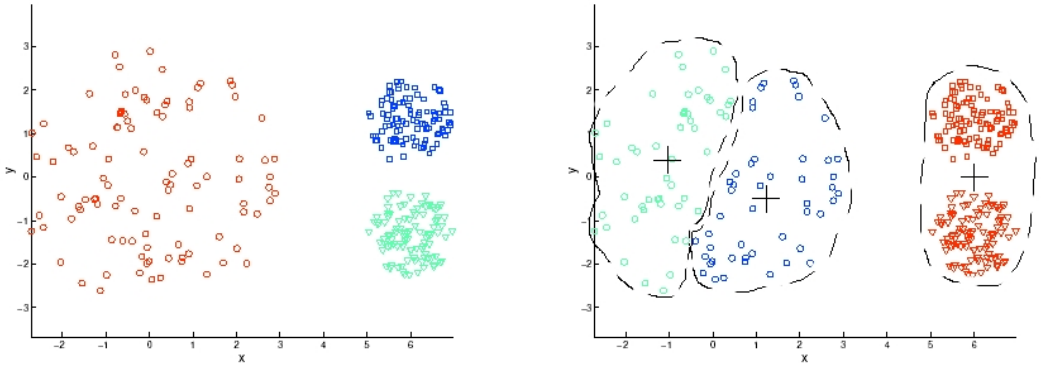


Figure 12.2: *(From [TSK03]) On the left, the original data : patterns are 2D feature vectors. On the right: the partition determined by a "minimum-variance" partitional algorithm (c-means). The dashed lines indicate the groups. Even if the user specifies to the algorithm the correct number of clusters, the algorithm is not able to detect the natural clusters. Related minimum-variance partitional methods perform particularly bad, when the natural clusters present very different numbers of points, or very diffent densities.*

Other popular criterion functions, also defined only when patterns live in euclidean (or hermitian) spaces, and closely related to the ones we have just described, can be derived based on the "within cluster" scatter matrix $W(\mathcal{P}(\mathcal{T}))$, and the "between cluster" scatter matrix $B(\mathcal{P}(\mathcal{T}))$ [DHS00],

$$W(\mathcal{P}(\mathcal{T})) = \sum_{m=1}^{c} \sum_{T \in \mathcal{T}_m} (T - \langle T_m \rangle) \cdot (T - \langle T_m \rangle)^{\mathrm{T}},$$

$$B(\mathcal{P}(\mathcal{T})) = \sum_{m=1}^{c} n_m (\langle T_m \rangle - \langle T \rangle) \cdot (\langle T_m \rangle - \langle T \rangle)^{\mathrm{T}},$$

$$S = \sum_{m=1}^{c} (T - \langle T \rangle) \cdot (T - \langle T \rangle)^{\mathrm{T}} = W(\mathcal{P}(\mathcal{T})) + B(\mathcal{P}(\mathcal{T})),$$

where $\langle T \rangle$ is the barycenter of all patterns in the data set, and S is the "total" scatter matrix, which is a constant given the data, independent on the partition. One can define optimal partitions as minimizers of $\mathrm{tr}[W(\mathcal{P}(\mathcal{T}))]$ (or equivalently maximizers of $\mathrm{tr}[B(\mathcal{P}(\mathcal{T}))]$); this turns out to be a minimum variance criterion. Another possiblity is to minimize $\det[W(\mathcal{P}(\mathcal{T}))]$, whose solution is invariant to linear transformations of the data. In any case, combinatorial optimization is intractable and one has to consider iterative procedures, with the subsequent limitations.

12.2.2 Hierarchical clustering methods

While partitional clustering algorithms construct a single partition with c clusters (a "flat" description), hierarchical methods obtain a clustering structure. Since they represent data in different ways, they do not really compete one with the other. Indeed, when data is to be described in terms of classes,

subclasses, subsubclasses (e.g. a biological taxonomy), flat representations do not make sense, and hierarchical methods are needed. There are, of course, many applications in which data is not inherently hierarchical, and one has to make a choice among clustering methods from both types. As we will see in what follows, hierachical methods are more versatile than partitional methods, and can deal with many differently shaped clusters, but they are more time consuming (their complexity is typically $O(M^2 \log M)$, while c-means complexity is $O(M)$ [BB95]).

Depending on the direction they build the hierarchy, these clustering methods can be agglomerative (bottom-up) or divisive (top-down). The former, which are usually computationally simpler, start with each single point as a cluster, and iteratively merge the closest pair of clusters in the sense of a chosen dissimilarity measure. The generic algorithm is as follows [JMF99]:

1. *Initialization*: compute the proximity matrix (the matrix containing the dissimilarity between each pair of patterns).
2. Find the most similar pair of clusters using the proximity matrix. Merge these two clusters.
3. Update the proximity matrix according to this merging.
4. Repeat steps 2 and 3 until all patterns are in one cluster.

At each iteration step, two clusters are merged. The procedure builds up a tree or dendrogram, where leaves are the M elements of $\mathcal{T}$ (step 1). At level l, there are $M-l$ nodes, each node being a cluster. At level $l+1$, the closest clusters from level l are merged (step 2). By "closest" we intend the pair $\mathcal{T}_i$ and $\mathcal{T}_j$ minimizing a given distance or proximity measure $\delta(\mathcal{T}_i, \mathcal{T}_j)$ between clusters. Different strategies for updating the proximity matrix lead to different hierarchical clustering methods. (Moreover, since all these algorithms are merging methods, they admit a variational formulation and can be solved as an energy minimization problem; see [MS95], chapter 3.) Lance and Williams [LW67] define a class of methods by specifying a generalized recurrence formula for updating the proximity matrix:

$$\delta(\mathcal{T}_i \cup \mathcal{T}_j, \mathcal{T}_k) = \alpha_i\, \delta(\mathcal{T}_i, \mathcal{T}_k) + \alpha_j\, \delta(\mathcal{T}_j, \mathcal{T}_k) + \beta\, \delta(\mathcal{T}_i, \mathcal{T}_j) + \gamma\, |\delta(\mathcal{T}_i, \mathcal{T}_k) - \delta(\mathcal{T}_j, \mathcal{T}_k)|,$$

where parameter values α_i, α_j, β and γ characterize the particular clustering method. Let us describe the most popular ones:

- Choosing $\alpha_i = \alpha_j = 1/2$, $\beta = 0$ and $\gamma = -1/2$, leads to the following distance between clusters:

  $$\delta_{min}(\mathcal{T}_p, \mathcal{T}_q) = \min_{T_i \in \mathcal{T}_p, T_j \in \mathcal{T}_q} d_T(T_i, T_j).$$

  The corresponding algorithm is known as *single-linkage algorithm* [JMF99, DHS00]. Here the nearest-neighbor points determine the nearest subsets. If we think of elements in $\mathcal{T}$ as nodes of a graph, merging $\mathcal{T}_p$ and $\mathcal{T}_q$ corresponds to adding an edge between the nearest points in $\mathcal{T}_p$ and $\mathcal{T}_q$. This procedure generates a tree, and if one lets the procedure evolve up to having a single cluster containing all points, we get a *minimal spanning tree*.

- Taking $\alpha_i = \alpha_j = \gamma = 1/2$, $\beta = 0$, yields

$$\delta_{max}(\mathcal{T}_p, \mathcal{T}_q) = \max_{T_i \in \mathcal{T}_p, T_j \in \mathcal{T}_q} d_T(T_i, T_j).$$

  The resulting algorithm is called *complete-linkage algorithm* [JMF99, DHS00]. Here distance between two clusters is given by the farthest pair of points in the two clusters. This procedure produces a graph in which edges connect all of the nodes in a cluster. When the nearest clusters are merged, edges between every pair of nodes in the two clusters are added. If we define the diameter of a partition as the largest diameter for clusters in the partition, then each iteration of the complete-linkage algorithm increases the diameter of the partition as little as possible.

- Taking $\alpha_i = n_i/(n_i + n_j)$, $\alpha_j = n_j/(n_i + n_j)$, and $\beta = \gamma = 0$, leads to a group average method, where

$$\delta_{avg}(\mathcal{T}_p, \mathcal{T}_q) = \frac{1}{n_p n_q} \sum_{T_i \in \mathcal{T}_p} \sum_{T_j \in \mathcal{T}_q} d_T(T_i, T_j).$$

- Some clustering methods based on barycenters, like Ward's minimum variance method [War63], can also be represented in terms of Lance and Williams formula. For Ward's method, $\alpha_i = (n_i + n_k)/(n_i + n_j + n_k)$, $\alpha_j = (n_j + n_k)/(n_i + n_j + n_k)$, $\beta = -n_k/(n_i + n_j + n_k)$, $\gamma = 0$, and the corresponding cluster proximity measure is

$$\delta_{ward}(\mathcal{T}_p, \mathcal{T}_q) = \frac{n_p n_q}{n_p + n_q} \|\langle T_p \rangle - \langle T_q \rangle\|_2^2,$$

  where $\langle T_p \rangle$ and $\langle T_q \rangle$ denote the barycenters of $\mathcal{T}_p$ and $\mathcal{T}_q$, respectively.

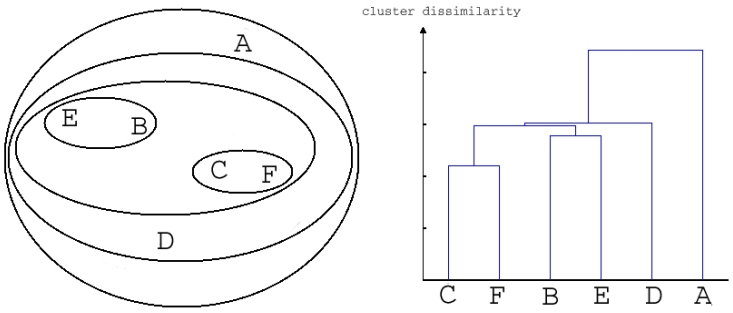
Figure 12.3 illustrates the results of applying the single-linkage, the complete linkage and the group average methods, to a small data set.

Time and space complexity of algorithms given by Lance and Williams formula are studied in [DE84]. Overall, the time required for hierarchical clustering is $O(M^2 \log M)$, and the space complexity is $O(M^2)$.

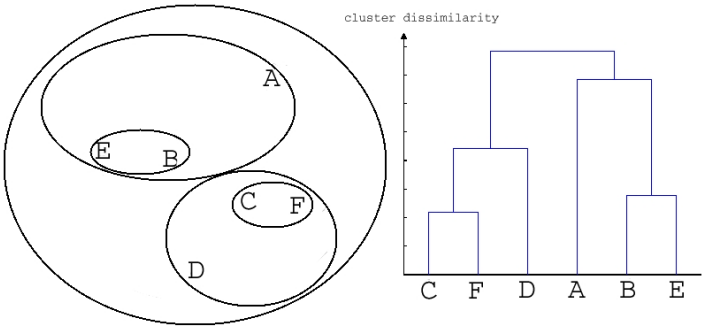
In practice, if clusters are compact and well separated, all methods yield the same results. However, when this is not the case, the resulting partitions may be quite different. Depending on the cluster proximity measure, different methods of clustering can be more or less successful with different types of clusters. The single-linkage algorithm suffers from the "chaining effect": a single corrupted point somewhere in between two compact clusters may lead to an unwanted merging between them [JMF99, DHS00]. However, this property is very useful if one wants to detect elongated clusters (see Figure 12.4 (a)).

The complete-linkage algorithm tends to produce compact clusters with small diameters. However, patterns assigned to a cluster can be much closer to patterns in other clusters [HTF01, DHS00]. This method is not adapted to extract concentric clusters, like the ones in figure 12.4 (b).

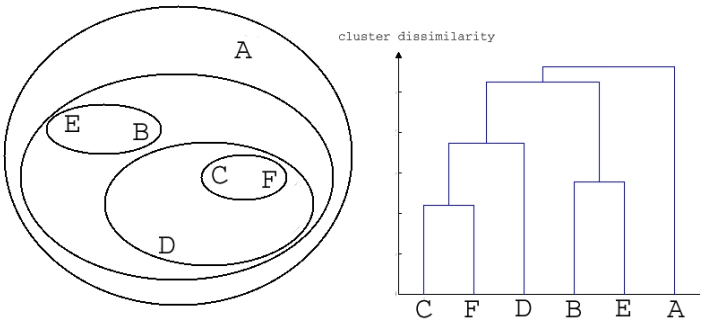
The single-linkage and the complete-linkage algorithms are both sensitive to outliers, since they rely on extremal measures. One way to reduce the influence of outliers is using δ_{avg} as cluster proximity



(a) *Single-linkage algorithm.*



(b) *Complete-linkage algorithm.*

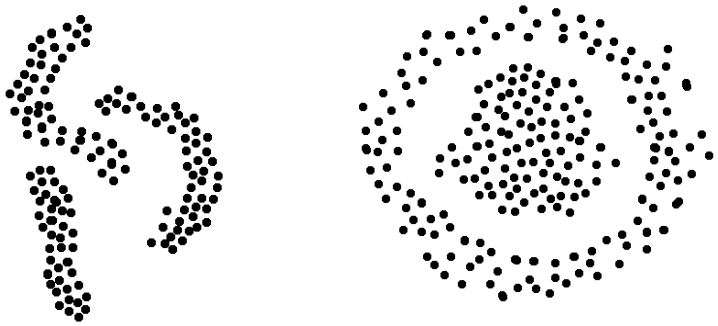


(c) *Group average algorithm.*

Figure 12.3: *Three agglomerative hierarchical clustering methods.*

measure, though the improvement is often not good enough. Besides, average methods have another drawback compared to single or complete linkage methods: they are not invariant under monotone transformations on the dissimilarity measure d_T (invariance of the former ones is a consequence of being based on extremal values) [HTF01].

To end with this section, let us make a few general remarks. In section 12.2.1 we assumed, for par-



(a) (b)

Figure 12.4: *Examples of (a) elongated clusters, (b) concentric clusters.*

titional clustering algorithms, that the number of clusters c was given. Then, the goal was to find the c-partition on the data optimizing a global criterion (in practice iterative methods are used, and the convergence to a global minimum is not ensured). Agglomerative hierachical clustering methods perform good in making local decisions about cluster merging, since they make use of the proximity matrix. As hierarchy is built by means of local optimization, the level corresponding to a c-partition will not correspond in general to a global optimum (unless clusters are compact and well separated). For instance, Ward's method will not lead to the same c-partition than a c-means method, despite the fact that both attempt to minimize variance. In this sense, one would rather say that partitional methods are better than hierarchical methods. But how can we be sure that there are exactly c groups of patterns in the data? Is the criterion function well adapted to the shape of clusters that are present in the data? From this viewpoint, hierarchical clustering may be more appealing than partitional ones. Another argument in favor for hierarchical clustering methods is their versatility and their ability to cope with differently shaped clusters. For instance, the single linkage algorithm can deal with non-isotropic, elongated or concentric clusters, while partitional methods like c-means can only deal with isotropic clusters (see Figure 12.5). Since their outputs are nested series of partitions, ranging from M clusters to one single cluster, one can imagine methods to determine the number of clusters, as stopping rules of the merging process. If stopping rules are correctly designed, hierarchical methods would also be able to detect clusters having different densities or different number of points (this was another important drawback of c-means methods, see Figure 12.2). We will discuss related issues in the following section.

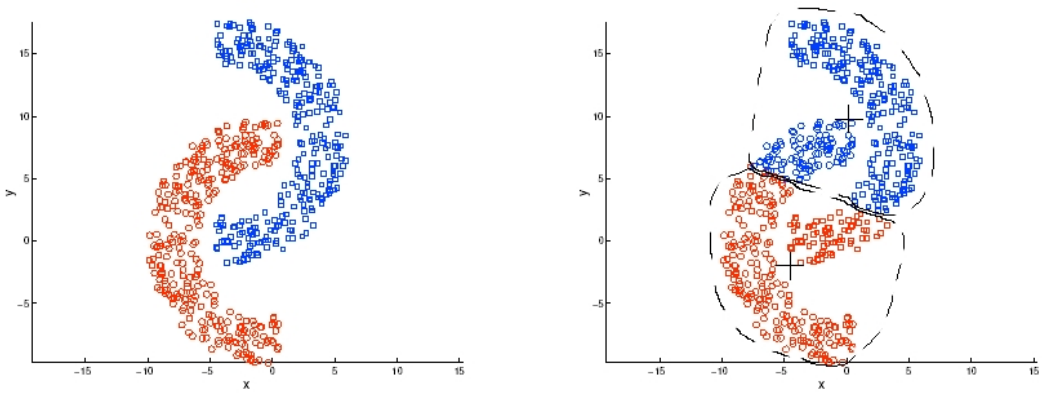


Figure 12.5: *(From [TSK03]) On the left, the original data : patterns are 2D feature vectors. On the right: the partition determined by a "minimum-variance" partitional algorithm (c-means). The dashed lines indicate the groups. These methods cannot deal with non-isotropic clusters, even if the user specifies to the algorithm the correct number of clusters.*

12.3 Cluster validity analysis and stopping rules

The great variety of clustering methods that have been proposed in the recent past has been followed by an increasing interest in clustering validation methods. In [Gor99], a comprehensive study of these techniques is presented.

Cluster validity analysis deals with assessing the validity of classifications obtained from the application of clustering procedures. There are different validation approaches [Dub87, Gor99], depending on the amount of prior information on the data. In this section we will deal with *internal validation tests*, which consist in determining if the structure is intrinsically adapted to the data. In other words, internal tests are derived from some *internal criteria* measuring the suitability of the clustering structure for the original data set, with no other information than the data themselves.

Classical issues in cluster validity analysis are the assessment of individual cluster validity, and the assessment of a whole partition. (In some applications it can also be required to assess the validity of a dendrogram; we will not address this problem here.) In what follows we briefly summarize these two issues.

Partition validity assessment

A relevant question to address in order to assess the validity of a partition, is deriving the number of clusters [Dub87], that we denoted by c. Notice that by solving this problem, it cannot be ensured that the c clusters are valid clusters. Figures 12.2 and 12.5 clearly illustrate this situation: even when the correct numbers of clusters is specified, the c-means method does not extract the natural clusters. The most common approach to decide how many clusters are best consists in finding partitions for $c = 1, \ldots, c_{max}$ and optimizing a measure $G(c)$ of partition adequacy, which is usually based on the within-cluster and between-cluster variability. When applied to hierarchical clustering methods,

these cluster validity assessment techniques are known as *global stopping rules*, because the choice of c can be seen as stopping the merging process (in the agglomerative case) at a certain level of the dendrogram.

When dealing with hierarchical classifications, another approach to determine the most appropriate number of clusters are *local stopping rules*. In the agglomerative case, these rules are *merging criteria* for deciding whether two clusters should be merged. Usually, the merging process is continued until it is decided, for the first time, that two clusters should not be aggregated.

Milligan and Cooper [MC85], and Dubes [Dub87], present comparative studies of some stopping rules. Milligan and Cooper's paper provides a particularly comprehensive Monte-Carlo evaluation of these rules, by comparing 30 local and global stopping rules. In their simulation experiment, only strongly clustered data sets (internally cohesive and well separated clusters) were considered. Hence, since clustering this kind of data should not be a challenging problem, techniques that do not perform well on it are also expected to be ineficient when dealing with any data set. The main conclusion of this experiment is that only five or maybe six of the compared rules perform quite well on strongly clustered data. One can also observe that the majority of the stopping rules described in the study are based on heuristics and lack of theoretical foundation. Those derived from rigorous statistical techniques, assume in general hypotheses on the data which are unrealistic in most real applications (*e.g.* multivariate normal distribution for the patterns). In order to briefly illustrate the considered stopping rules, it is worth to describe Calinski and Harabasz's index [CH74] and Duda and Hart's rule [DHS00], since these methods provided the best results.

- Calinski and Harabasz propose a *global stopping rule* for assessing partitions, by choosing the partition size c that minimizes the index

$$G(c) = \frac{\frac{1}{c-1}\text{tr}\left[B\left(\mathcal{P}(\mathcal{T})\right)\right]}{\frac{1}{M-c}\text{tr}\left[W\left(\mathcal{P}(\mathcal{T})\right)\right]},$$

  where $B\left(\mathcal{P}(\mathcal{T})\right)$ and $W\left(\mathcal{P}(\mathcal{T})\right)$ are, respectively, the between-cluster and within-cluster scatter matrices of a c-partition $\mathcal{P}$, defined in section 12.2.1. The index $G(c)$ is the ratio between the total within-cluster sum of squared distances about the centroids, and the total between-cluster sum of squared distances. This index is only defined for sets of patterns living in an Euclidean space. Moreover, since the index is based on the sum of squares criterion, it has a tendency to partition the data into hyperspherical shaped clusters, having roughly equal numbers of patterns [Gor99] (this is probably the main reason for its first position in Milligan and Cooper's ranking, since their data was strongly clustered, and clusters contained almost the same numbers of points and were pretty isotropic).

- Duda and Hart proposed the "$Je(2)/Je(1)$" *local stopping rule* for deciding whether or not a cluster should be splitted into two subclusters. The rule consists in computing the ratio between the total within sum of squared distances about the centroids of the two clusters for the two-cluster solution ($Je(2)$), and the within sum of squared distances about the centroid

when only one cluster is present ($Je(1)$). The method is based in considering a null hypothesis, that assumes all patterns come from a normal distribution, whose mean and variances are empirically estimated over all the data set. The null hypothesis of one single cluster is rejected if $Je(2)/Je(1)$ is smaller than a specified critical value, fixed by a significance level for the hypothesis testing. While considering a normal distribution as a null hypothesis and using the sum of squared distances may not be well adapted to real clustering problems (particularly when the number of patterns in the data set is not as large to be well represented by an asymptotic distribution), the proposed *a contrario* formulation is appealing from our point of view.

Let us finish the discussion on partition validity assessment by quoting one of Bock's conclusions from its work on significance tests in cluster analysis [Boc85], where a comparison between global and local methods is made: *"Some care is needed when applying any test for clustering, bearing in mind that different types of clusters may be present simultaneously in the data, and that the number of clusters is, in some sense, dependent on the intended level of information compression. Thus, a global application of a cluster test to a large or high-dimensional data set will not be advisable in most cases. However, a "local" application (...) to a specific part of the data will often be useful for providing evidence for or against a prospective clustering tendency".*

Validity assessment of individual clusters

Now we are concerned with the problem of deciding, among the candidate clusters furnished by the clustering procedure, which are the ones that correspond to "natural" clusters. But what does a "natural" cluster look like? As pointed out by Gordon [Gor99], it may be difficult to specify a relevant definition of ideal cluster for a particular data set. However, we can think of clusters as some structure in the data. Clustered data then reveals structure, that is perceived as opposite to a complete absence of structure. Thus, in order to decide whether the clusters we have found are significant, we can proceed by comparing our actual data with some appropriate random distribution. This leads to a general methodology for cluster validity analysis, based on the statistical approach of hypothesis testing [Boc85, Gor96, Gor99]. Following Bock [Boc85], this framework consists in:

1. Design a null hypothesis $\mathcal{H}$ for the absence of class structure in the data (a *background model*, or *null model*), meaning that patterns are sampled from a "homogeneous" population. Then, "heterogeneity" or "clustering structure" are involved in the alternative hypothesis $\mathcal{A}$.

2. Define a test statistic, which will be used as validity index to discriminate between $\mathcal{H}$ and $\mathcal{A}$.

3. If, for a given significance level (error probability) α, the test statistic of the observed data exceeds the corresponding critical value c_α, the null hypothesis $\mathcal{H}$ is rejected, in favor of $\mathcal{A}$.

This general framework can be adapted for assessing the validity of individual clusters. A general approach within this framework is Monte-Carlo validation, which is described in [Gor99]. Assume

one wants to assess the validity of an observed cluster $\mathcal{T}_i$ having n patterns, in a data set having M patterns. In the Monte-Carlo validation method, data sets of M patterns are simulated under the background model, and classified using the same clustering procedure that was used to classify the original data. The test statistic is computed for those clusters having n patterns, and the distribution of the test statistic is estimated. Then, using the value of the test statistic of $\mathcal{T}_i$, one can compute the significance level of rejecting $\mathcal{H}$. Two popular test statistics are the maximum F test and the U statistic (see Bock [Boc85] and Gordon [Gor99]).

We have not addressed the choice of the null model yet. The specification of appropriate null models for data is the subject of the study presented in [Gor96]. These models, which specify the distribution of patterns in the absence of structure in the data, can be of two types:

- *Standard (data-independent) null models*. Two well known standard null models are the *Poisson model* and the *Unimodal model* [Boc85]. The main problem with the Poisson model is the choice of the region R within which patterns are uniformly distributed (standard choices for normalized data are the unit hypercube and the unit hypersphere). The Unimodal model assumes that the joint distribution of the variables describing the patterns is unimodal, but the choice of the distribution may not be easy.

- *Data-influenced null models*. Here the data is used to influence the specification of the null model. Examples of these null models are the Poisson model where R is chosen to be the convex hull of the data set, or the *Ellipsoidal model*, which is a multivariate normal distribution, whose mean and covariance matrix are given by the data set.

In [Gor96], Gordon concludes that the results of the tests considerably depend on the choice of the null model, and that, in general, the results based on data-influenced null models are more relevant than those obtained using a standard null model.

In the following section we propose a method to detect valid clusters from an agglomerative hierarchical classification, that combines an individual cluster validity method and a local merging criterion. The first step consists in deciding, *a contrario* to a data-influenced background model, whether a cluster is valid or not. All clusters in the hierarchical structure are examined. While all clusters passing the validity test are meaningful in themselves, the set of all of them does not necessarily reveals the structure of the data set. However, by selecting a subset of the meaningful clusters, a good data representation can be achieved. Hence, in the second step such a selection is performed, by means of a new merging criterion, also derived from the *background model*. Unlike the classical hypothesis testing methods presented in this section, the proposed method does not require to fix a significance level for deciding the validity of clusters.

12.4 Meaningful clusters

12.4.1 *A contrario* definition of meaningful groups

The background model

Helmholtz principle states that if an observed arrangement of objects in an image is highly unlikely, the occurrence of such arrangement is significant and the objects should be grouped together into a single structure. This perceptual organization principle, also known as the principle of common cause or of the coincidental explanation, was first stated in computer vision by Lowe [Low85]. Helmholtz principle applies, for instance, to clusters of points under uniform distribution assumption. If the density of points in a given space location exceeds a certain threshold, then the "proximity" gestalt leads to the perceptual grouping of individual dots, and a better interpretation of this set of points is the cluster as a whole [Wer23]. This cluster is significant if its density is so high that such an arrangement is unlikely to be due to randomness. In other words, there must be a better explanation for the observed cluster than randomness: the formation of causal relations. This gives us a qualitative definition of meaningful clusters, but in order to detect them, we need a more precise definition. This can be done based on the hypothesis testing ideas presented in the former section. *Randomness* can be modeled by means of a *background process*, governed by the following assumptions:

(A) Patterns T_j, $j \in \{1, \ldots, M\}$, are mutually independent random variables, identically distributed on the feature space according to an *a contrario* law p defined on it. We will denote by $p(R)$ the *a contrario* "region" probability of a region R in the feature space.

The definition of the *a contrario* law is problem specific. In Chapter 13 we will derive it for the detection of spatially coherent groups of meaningful matches. In general, the *a contrario* law is not known *a priori* but it can be empirically estimated over the data.

Meaningful groups

Having a background model, we are in position to evaluate the probability of a given cluster of patterns as a realization of the background process. Hence, we are able to detect relevant clusters by Helmholtz principle: those clusters being unlikely to be observed by chance will be considered as meaningful groups. Let us give an example to illustrate this idea. In Figure 12.6, we display the six 2-D projections of 4-dimensional patterns T_k (these patterns correspond to the kind of problem we will study in Chapter 13). The "high density" cluster we observe reveals a conspicuous coincidence. Indeed, the probability of its being a realization of the background process should be very low, and one would expect it to be an exception to randomness.

Let us make things more precise. For any region R in the feature space, we know how to compute the "region probability" $p(R)$, the probability that a pattern generated by the background process falls in

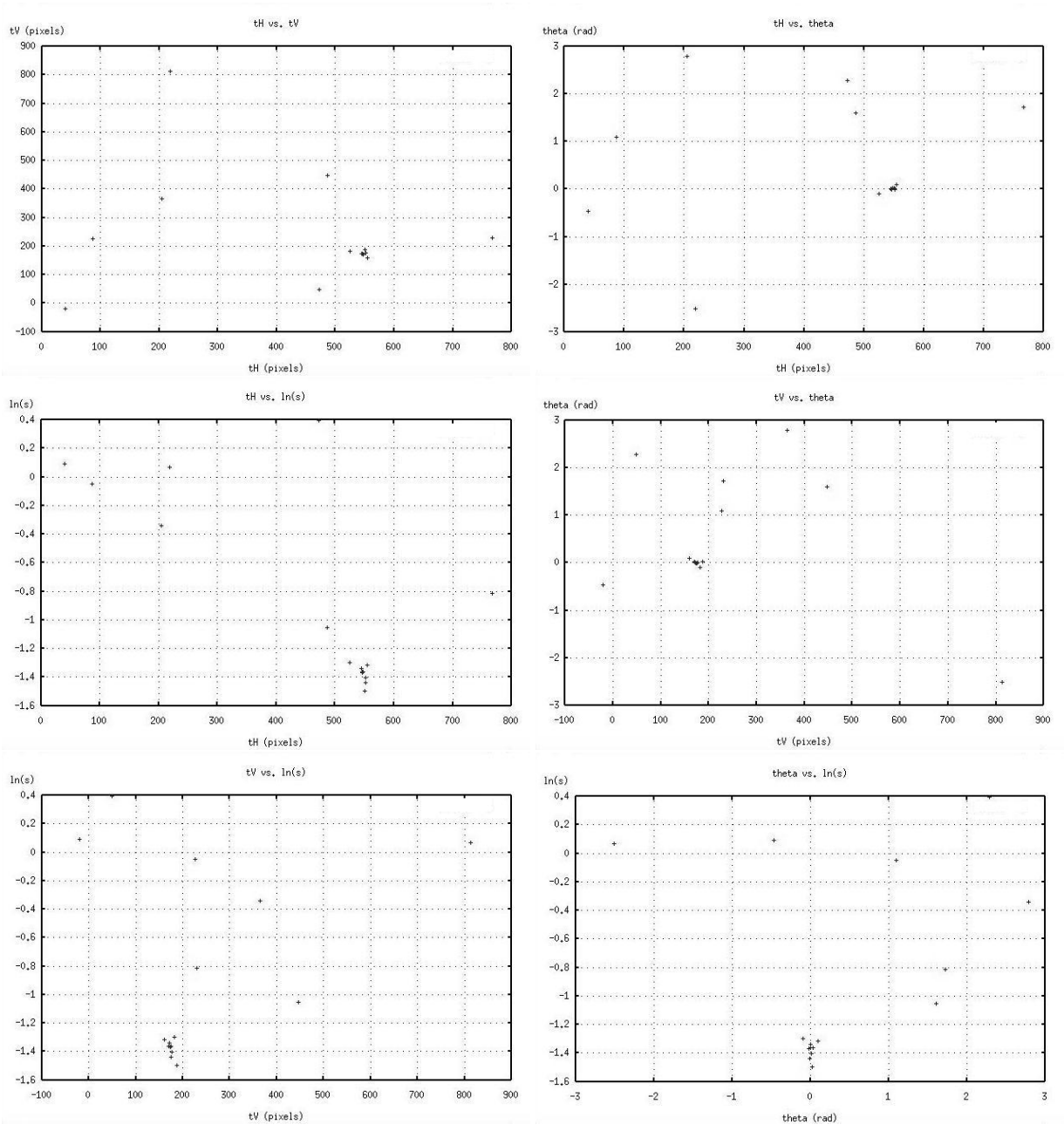


Figure 12.6: *All six projections of 4-dimensional patterns corresponding to a problem studied in Chapter 13.*

R. Then, since patterns are mutually independent, the probability that R contains at least k patterns out of M under the *a contrario* model is given by the tail of the binomial probability distribution

$$\mathcal{B}(M, k, p(R)) = \sum_{i=k}^{M} b(M, i, p(R)),$$

where

$$\forall\, i \in \{0, \ldots, M\}, \quad b(M, i, p(R)) = \binom{M}{i} p(R)^i (1 - p(R))^{M-i}.$$

The next step prior to detection of relevant clusters, is to define a set of reasonable clusters candidates. One possibility could be the set of all hyper-rectangles of different sizes within a given quantization grid of the feature space. Let us denote by $\#\mathcal{R}$ the cardinality of $\mathcal{R}$. If each dimension is divided into L bins, then $\#\mathcal{R} = (L(L+1)/2)^D$. It can be argued that this set of regions is not well adapted to the case of sparse data, since the majority of them will not contain any pattern. A more adapted $\mathcal{R}$ set for the sparse data case may be defined as the set of feature space hyper-rectangles made up with all possible hyper-rectangles of edge sizes

$$a, a\sqrt{2}, a(\sqrt{2})^2, \ldots, a(\sqrt{2})^n,$$

centered in each point T_i $(1 \leqslant i \leqslant M)$ corresponding to a pattern. The cardinality of this set is $\#\mathcal{R} = M(n+1)^D$. Notice that such a choice implies that every region R in $\mathcal{R}$ contains at least one pattern (the central point). Consequently, in that case we are no longer concerned with finding at least k patterns out of M, but at least $k-1$ patterns out of $M-1$. The choice of the minimal size a is not relevant for what immediately follows (it is based on precision arguments and will be addressed later). Concerning the number of considered nested hyper-rectangles $(n+1)$, once a is fixed n is chosen such that $a(\sqrt{2})^n$ does not exceed the feature space dimensions. Now, why is $\mathcal{R}$ a reasonable set of candidates? On the one hand, since we center hyper-rectangles of different scales at each point T_i, we are sure we will not miss any cluster. On the other hand, $\mathcal{R}$ does not contain (irrelevant) empty hyper-rectangles.

DEFINITION 12.1 (ε-MEANINGFUL GROUP) *We say that a group of patterns is ε-meaningful if*

$$\#\mathcal{R} \cdot \mathcal{B}(M, k, p(R)) \leqslant \varepsilon.$$

PROPOSITION 12.1 *The expected number of ε-meaningful groups in $\mathcal{R}$ is less than ε.*

Proof: Let us denote by χ_R the binary random variable equal to 1 if the hyper-rectangle R in $\mathcal{R}$ is ε-meaningful and 0 else. Let $S = \sum_{R \in \mathcal{R}} \chi_R$ be the random variable representing the number of ε-meaningful hyper-rectangles in $\mathcal{R}$.

By linearity, the expectation of S is $\mathbb{E}[S] = \sum_{R \in \mathcal{R}} \mathbb{E}[\chi_R]$. Hence, since χ_R is a Bernoulli variable,

$$\mathbb{E}[S] = \sum_{R \in \mathcal{R}} \Pr(\chi_R = 1).$$

Let us denote by $k^*(\varepsilon)$ the minimum number of points in R such that R is ε-meaningful:

$$k^*(\varepsilon) = \min\left\{k \in \mathbb{N},\ \mathcal{B}(M,k,p(R)) \leqslant \frac{\varepsilon}{\#\mathcal{R}}\right\}.$$

(This number is well defined because $\mathcal{B}(M,k,p(R))$ is a decreasing function of k.) It follows that

$$\Pr(\chi_R = 1) = \Pr\left(k \geqslant k^*(\varepsilon)\right) = \mathcal{B}(M,k^*(\varepsilon),p(R)) \leqslant \frac{\varepsilon}{\#\mathcal{R}}.$$

Thus,

$$\mathbb{E}[S] \leqslant \sum_{R\in\mathcal{R}} \frac{\varepsilon}{\#\mathcal{R}},$$

yielding $\mathbb{E}[S] \leqslant \varepsilon$. ■

Remark: The key point is that we control the expectation of S. Since dependencies between random variables χ_R are unknown, we are not able to compute the probability law of S. Nevertheless, linearity still allows to compute the expectation.

The following definition provides a quality measure for a cluster or group of patterns.

DEFINITION 12.2 (NUMBER OF FALSE ALARMS) *Given a group G of k meaningful patterns among M, we call number of false alarms of G the number*

$$NFA_g(G) = \#\mathcal{R} \cdot \mathcal{B}(M,k,p(R)),$$

where $p(R)$ is the probability that a pattern falls in R, the smallest hyper-rectangle in $\mathcal{R}$ containing all k patterns.

The number of false alarms of G is a measure of how likely it is that a group having at least k meaningful patterns and a region probability $p(R)$, was generated "by chance", as a realization of the background process. The lower is $NFA_g(G)$, the more unlikely G is generated by the background, and hence, the more meaningful is G. Indeed, if $NFA_g(G)$ is very small, elements in G certainly violate assumptions in (A), leading to an *a contrario* detection. Notice also, from Proposition 12.1, that the only parameter that controls detections is ε. This provides a handy way to control false detections: if we want to detect on the average at most one "non relevant cluster" we just set $\varepsilon = 1$. From now on we refer to 1-meaningful groups as "meaningful groups".

Testing strategy and numerical issues

Testing all $(L(L+1)/2)^D$ or $M(n+1)^D$ (depending on the choice) hyper-rectangles in $\mathcal{R}$ can result in a heavy computational burden. Since we know that relevant clusters are nodes of the hierarchical clustering dendrogram, we can restrict the search to the smallest hyper-rectangles in $\mathcal{R}$ containing the clusters given by the nodes of the complete hierarchical structure. This greatly reduces the actual

number of tests from $M(n+1)^D$ to $2M-1$.
We have not addressed yet the problem of fixing a, the minimal edge size of a cell. We can proceed as follows:

1. Normalize the transformation space into $[0,1]^D$.

2. Discretize each direction into $l = 1000$ bins, for instance. Thus, a is equal to a bin size, namely $a = 1/l = 0.001$. Then we have $n = \lfloor -2\log_2(a) \rfloor = 19$ (here $\lfloor x \rfloor$ denotes the integer part of x).

Although this discretization choice may seem arbitrary, it is not crucial since it has almost no influence on the general setting. First of all, the clusters that are effectively tested are issued from the hierarchical clustering procedure. Thus, as long as the discretization is fine enough to ensure an accurate localization of clusters (so that the bounding hyper-rectangles fit well the clusters), it plays a secondary role in this decision framework (unlike for voting schemes over cells issued from space discretization, were the size of cells plays a major role). Secondly, let us see how parameters a and n affect the detection of clusters. Since parameter n is fixed by parameter a, the point is the dependence of the method on a. A straightforward adaptation of results reported in [DMM00] (in the framework of alignment detection in digital images) yields

$$k^*(\varepsilon) \approx Mp(R) + \sqrt{CM\,(D\ln(1-2\log_2 a) - \ln(\varepsilon))},\ \ C \in [2p(R)(1-p(R)), 1/2]\,,$$

where $k^*(\varepsilon)$ is the minimum number of points in R such that R is ε-meaningful. This approximation shows that the dependence on a is very weak. Also the dependence on ε is weak. This is a nice property, since it means that detection is not very sensitive to the only user defined parameter.

12.4.2 Cluster validity and maximality criterion

In section 12.4.1 we have defined *meaningful groups of patterns*, and proposed to restrict the space of tests to the smallest hyper-rectangles containing clusters from the dendrogram. While each meaningful group we detect will be relevant by itself, the whole set of meaningful groups will probably exhibit high redundancy in the sense that we will get many nested meaningful groups. In this section we describe a strategy to reduce this redundancy by combining the inclusion tree given by the hierarchical clustering procedure, and the measure of meaningfulness given by NFA_g.
Let us start this discussion by the following issue. At each step of the hierarchical clustering procedure, two clusters are merged. This merging is not necessarily a better data representation than the two separate clusters. By using the complete dendrogram (that we denote by $\mathcal{D}$) of $2M-1$ clusters, we can decide *a posteriori* whether pairs of clusters should be merged or not. Let us denote by G, G_1 and G_2 the groups of patterns corresponding respectively to a node and its two children nodes in $\mathcal{D}$. Roughly speaking, we will accept merging if, under the *a contrario* model, the expected number of groups like G we would observe is smaller than the one of observing groups like G_1, G_2, or the pair

G_1 and G_2. Hence, we will say that G is valid if it verifies this criterion. Before giving the definition of a valid group, let us define $NFA_{gg}(G_1, G_2)$, the number of false alarms of the pair (G_1, G_2):

$$NFA_{gg}(G_1, G_2) = \frac{\#\mathcal{R}\,(\#\mathcal{R}-1)}{2} \sum_{i=k_1}^{M} \sum_{j=k_2}^{M-i} \binom{M}{i,j} {p_1}^i {p_2}^j (1-p_1-p_2)^{M-i-j}, \tag{12.1}$$

where k_1 and k_2 are the number of elements in G_1 and G_2, and p_1 and p_2 their associated region probabilities. $\binom{M}{i,j}$ denotes the trinomial coefficient. $NFA_{gg}(G_1, G_2)$ is an estimate of the number of occurrences, under the *a contrario* model, of the event $\mathcal{E}$: "there are two non overlapping groups A and B, with region probabilities p_1 and p_2 (resp.), containing at least k_1 and k_2 patterns (resp.) among M". Indeed, $\#\mathcal{R}\,(\#\mathcal{R}-1)\,/2$ is the number of pairs of clusters of $\mathcal{R}$, and the probability of event $\mathcal{E}$ is given by the joint tail of the trinomial probability distribution.

DEFINITION 12.3 (VALID GROUP) *Let G, G_1 and G_2 be the groups of patterns corresponding respectively to a node and its two children nodes in $\mathcal{D}$. We say that G is a valid group if both following inequalities hold:*

$$NFA_g(G) < \min\left\{NFA_g(G_1), NFA_g(G_2)\right\}, \tag{12.2}$$

$$NFA_g(G) \leqslant NFA_{gg}(G_1, G_2). \tag{12.3}$$

Eq. (12.2) corresponds to the condition that merging cannot be suitable if one of the child nodes is more meaningful than the father. Eq. (12.3) means that for G to be valid, it is necessary that its number of false alarms is lower than the number of false alarms of the pair (G_1, G_2). The following lemma leads to a necessary condition of cluster validity.

LEMMA 12.1

$$\sum_{i=k_1}^{M} \sum_{j=k_2}^{M-i} \binom{M}{i,j} {p_1}^i {p_2}^j (1-p_1-p_2)^{M-i-j} \leqslant \mathcal{B}(M, k_1, p_1) \cdot \mathcal{B}(M, k_2, p_2). \tag{12.4}$$

Proving lemma 12.1 directly by calculation is not trivial. Inequality (12.4) is a consequence of the *negative dependence* amongst random variables $\#A$ and $\#B$, the number of patterns in two random clusters A and B. Intuitively, this dependence (which is obvious because of the condition $\#A + \#B \leqslant M$) is negative in the sense that, if $\#A$ is "large", $\#B$ is less likely to be "large". In Appendix 12.7, we introduce the notion of *negative association* (a strong notion of negative dependence) and some relevant consequences, first reported by Joag-Dev and Proschan in [JDP83]. These results lead to a simple proof of lemma 12.1, also presented in in appendix 12.7.

PROPOSITION 12.2 *If G is a valid group, then $NFA_g(G) < \frac{1}{2} \cdot NFA_g(G_1) \cdot NFA_g(G_2)$.*

Proof: The result follows immediately from (12.1), definition 12.3 and lemma 12.1. ∎

Notice that the necessary condition for merging given by Proposition 12.2, is equivalent to

$$\log(\mathcal{B}(M, k_1 + k_2, p)) < \log(\mathcal{B}(M, k_1, p_1)) + \log(\mathcal{B}(M, k_2, p_2)) + \log\left(\frac{\#\mathcal{R}}{2}\right),$$

where p ($p \geqslant p_1 + p_2$) is the region probability of G. This shows that merging depends on a natural trade-off between goodness of fit and model complexity.

Remark: Proposition 12.2 can be useful from the computational viewpoint, since in many cases one can avoid computing the tail of the trinomial distribution, by "filtering" those clusters that do not pass the necessary condition.

Definition 12.4 (Maximal ε-meaningful group) *We say that a group of patterns G is a maximal ε-meaningful group if and only if:*

1. $NFA_g(G) \leqslant \varepsilon$,
2. *G is valid,*
3. *for all valid descendant F, $NFA_g(F) > NFA_g(G)$,*
4. *for all valid ancestor F, $NFA_g(F) \geqslant NFA_g(G)$.*

Remark: Imposing items 3 and 4 ensures that two different maximal meaningful groups are disjoint. Hence, maximal meaningful groups define a set of groups on the data, which is optimal in the sense that these groups are maxima of meaningfulness with respect to inclusion, and where outliers have been automatically rejected.

12.5 An alternative definition of meaningful groups

Up to now we have considered that every pattern T_k was equally relevant, as the family of random variables $\{T_k, 1 \leqslant k \leqslant M\}$ was assumed to be mutually independent, identically distributed. In this section we propose a more general definition of group meaningfulness, which associates a measure of confidence to each pattern. Assume, for instance, that patterns correspond to observations having different relevance. What we want to evaluate here is the probability that, just "by chance", several relevant patterns fall into a feature space region R. It is sound to expect that, for a cluster to be meaningful, the more relevant are its patterns, the lesser the minimum number of required patterns should be. Let us make things more concrete by defining the corresponding background model.

12.5.1 The background model

We define the background process by means of the following assumptions:

(A1) Patterns T_i, $i \in \{1, \ldots, M\}$, are mutually independent random variables, and for any hyper-rectangle R in the feature space,

$$\Pr(T_i \in R) = p(R),$$

p is an *a contrario* probability defined on the feature space.

(A2) *Independent saliency measure.* Patterns T_i are observations detected with a measurement system, which assigns to each observation T a confidence index c_T. The confidence index decreases with the relevance of the observation; an infinitely relevant pattern T will have a confidence index $c_T = 0$. All patterns T_i in the data set, come from observations whose confidence index (denoted by c_{T_i}) is lower than γ, a predetermined threshold. Let us define, for all non-negative real number x,

$$p_c(x) := \Pr(T \text{ s.t. } c_T \leqslant x),$$

a probability measure on the confidence index, which is given by the measurement system. We call saliency measure of a pattern T_i, $i \in \{1, \ldots, M\}$, the number

$$\eta_i := \Pr(T \text{ s.t. } c_T \leqslant c_{T_i} \mid c_T \leqslant \gamma) = \frac{p_c(c_{T_i})}{p_c(\gamma)}$$

(the last equality follows from $c_{T_i} \leqslant \gamma, \ \forall\ i \in \{1, \ldots, M\}$). Finally, we assume that, under the *a contrario* model, the saliency measure of a pattern is independent from its location in the feature space.

Now we can summarize assumptions (A1) and (A2) as follows:

(A) Patterns T_i, $i \in \{1, \ldots, M\}$, are mutually independent random variables, and for any hyper-rectangle R in the feature space, the probability that a pattern has at the same time a saliency measure below c_{T_i} and falls in R is:

$$\Pr(T \text{ s.t. } c_T \leqslant c_{T_i}, T \in R) = \eta_i \times p(R),$$

where:

- p is an *a contrario* probability defined on the feature space.
- $\eta_i = \frac{p_c(c_{T_i})}{p_c(\gamma)}$ is the saliency measure of pattern T_i ($c_{T_i} \leqslant \gamma$), which is assumed to be independent of pattern location in the feature space.

The definition of the saliency measure is maybe too general, making its sense a bit confuse. In Chapter 13 we will apply the theory presented in this chapter to the detection of spatially coherent meaningful matches. There, the saliency measure will have a concrete meaning, and its definition should become clearer.

12.5.2 Meaningful groups taking into account the relevance of patterns

Let us denote by χ_i, $1 \leqslant i \leqslant M$, the indicator function of event $\mathcal{E}_i$: "T_i has saliency measure η_i, and falls in R". By using assumption (A), we can compute the expectation of such an event under the background model:

$$\mathbb{E}[\chi_i] = \Pr(\chi_i = 1) = p(R)\eta_i.$$

Hence, given the transformation region R and M real numbers $\eta_1, \ldots, \eta_M$ in $(0, 1]$, the expected number of events $\mathcal{E}_i$ we can observe simultaneously in a trial is

$$\mathbb{E}\left[\sum_{i=1}^{M} \chi_i\right] = \sum_{i=1}^{M} \Pr(\chi_i = 1) = p(R) \sum_{i=1}^{M} \eta_i. \tag{12.5}$$

According to the former definition of meaningful group, we will consider we have a meaningful group in R if its number of patterns is unexpectedly high under the background process.

DEFINITION 12.5 (ε-MEANINGFUL GROUP) *Let R be a feature space hyper-rectangle in $\mathcal{R}$, containing a group of k among M patterns. We say the group is ε-meaningful if*

$$k \geqslant k^*(R, \eta_1, \ldots, \eta_M) := \min\left\{n \in \mathbb{N} : \Pr\left(\sum_{i=1}^{M} \chi_i \geqslant n\right) \leqslant \frac{\varepsilon}{\#\mathcal{R}}\right\}.$$

The following proposition follows immediately from definition 12.5.

PROPOSITION 12.3 *The expected number of ε-meaningful groups in $\mathcal{R}$ is less than ε.*

Computing the meaningfulness of a group using definition 12.5 is not practical, and too much expensive in terms of computation. Indeed, computing

$$\Pr\left(\sum_{i=1}^{M} \chi_i \geqslant n\right) = \sum_{l=n}^{M} \Pr\left(\sum_{i=1}^{M} \chi_i = l\right)$$

requires to evaluate, for each l, all $\binom{M}{l}$ probabilities

$$\Pr\left(\chi_i = 1 \; \forall \, i \in I, \chi_j = 0 \; \forall \, j \in \{1, \ldots, M\} \backslash I \text{ s.t. } I \subset \{1, \ldots, M\}, \#I = l\right).$$

Nevertheless, one can make the meaningfulness computation possible by using the first Hoeffding inequality, which gives a good upper bound for $\Pr\left(\sum_{i=1}^{M} \chi_i \geqslant n\right)$.

LEMMA 12.2 (HOEFFDING INEQUALITIES [HOE63]) *Let $Z_1, \ldots, Z_M$ be independent random variables with $0 \leqslant Z_i \leqslant 1$ for $i = 1, \ldots, M$. Let $S = \sum_{i=1}^{M} Z_i$. Then,*

$$\Pr(S \geqslant k) \leqslant \left(\frac{\mathbb{E}[S]/M}{k/M}\right)^k \left(\frac{1 - \mathbb{E}[S]/M}{1 - k/M}\right)^{M-k} \leqslant \exp\left(-h\left(\frac{\mathbb{E}[S]}{M}\right) \frac{(k - \mathbb{E}[S])^2}{M}\right), \tag{12.6}$$

where

$$h(\mu) = \begin{cases} \frac{1}{1-2\mu} \ln\left(\frac{1-\mu}{\mu}\right) & \text{if } 0 < \mu < \frac{1}{2} \\ \frac{1}{2\mu(1-\mu)} & \text{if } \frac{1}{2} \leqslant \mu < 1. \end{cases}$$

The second inequality in (12.6) leads to the following condition.

PROPOSITION 12.4 (SUFFICIENT CONDITION OF ε-MEANINGFULNESS) *Let R be a feature space hyper-rectangle from $\mathcal{R}$, containing a group of k among M patterns. Then, if*

$$k \geqslant p(R)\sum_{i=1}^{M}\eta_i + \sqrt{\frac{M\left(\ln(\#\mathcal{R}) - \ln\varepsilon\right)}{h\left(p(R)\sum_{i=1}^{M}\eta_i/M\right)}}, \tag{12.7}$$

the group is ε-meaningful.

Notice the right hand side of (12.7) is an increasing function of $\sum_{i=1}^{M}\eta_i$. Applying Hoeffding inequality to the former case (where the relevance of patterns was not taken into account, section 12.4), comes to take $\eta_i = 1$ for all $1 \leqslant i \leqslant M$, that is $\sum_{i=1}^{M}\eta_i = M$. Hence, when we take into account the relevance of patterns, meaningful group need containing less points than previously.

Proof: From (12.7) we have

$$\exp\left(-h\left(\frac{p(R)\sum_{i=1}^{M}\eta_i}{M}\right)\frac{\left(k - p(R)\sum_{i=1}^{M}\eta_i\right)^2}{M}\right) \leqslant \frac{\varepsilon}{\#\mathcal{R}}.$$

Then, since $\mathbb{E}\left[\sum_{i=1}^{M}\chi_i\right] = p(R)\sum_{i=1}^{M}\eta_i$, applying the second inequality in lemma 12.2 with $Z_i = \chi_i$ for $i = 1,\ldots,M$ yields $\Pr\left(\sum_{i=1}^{M}\chi_i \geqslant k\right) \leqslant \varepsilon/\#\mathcal{R}$. The result follows from definition 12.5. ∎

The more accurate first inequality in (12.6) motivates the following definition.

DEFINITION 12.6 (NUMBER OF FALSE ALARMS) *Given a group G of k patterns among M, we call number of false alarms of G the number*

$$NFA_g(G) = \#\mathcal{R} \times \left(\frac{p(R)\sum_{i=1}^{M}\eta_i}{k}\right)^k \left(\frac{M - p(R)\sum_{i=1}^{M}\eta_i}{M-k}\right)^{M-k},$$

where $p(R)$ is the probability that a pattern falls in R, the smallest hyper-rectangle in $\mathcal{R}$ containing all k patterns, and $\eta_1,\ldots,\eta_M$ are the saliency measures of the M patterns.

Remark: If $NFA_g(G) \leqslant \varepsilon$, then G is ε-meaningful (this is straightforward from the first inequality in (12.6)).

Remark: Introducing the saliency measure of patterns makes good detections even surer, since many patterns in the corresponding group may have low saliency measures, diminishing its number of false alarms. But, concerning this approach, maximality issues are not completely solved. Indeed, the NFA_{gg} of a pair of groups used for the validity criterion, defined in (12.1), only holds when patterns T_i are independent, identically distributed, so new definitions have to be explored. We have

not addressed this problem yet. A first attempt could consist in replacing the validity criterion by the necessary condition of cluster validity, given by Proposition 12.2 (section 12.4.2), but this will certainly fail in giving the "good" maximal groups. Indeed, this necessary condition may be too strong because estimate (12.6) is not sharp enough, specially when the clusters to be merged concentrate the majority of the patterns.

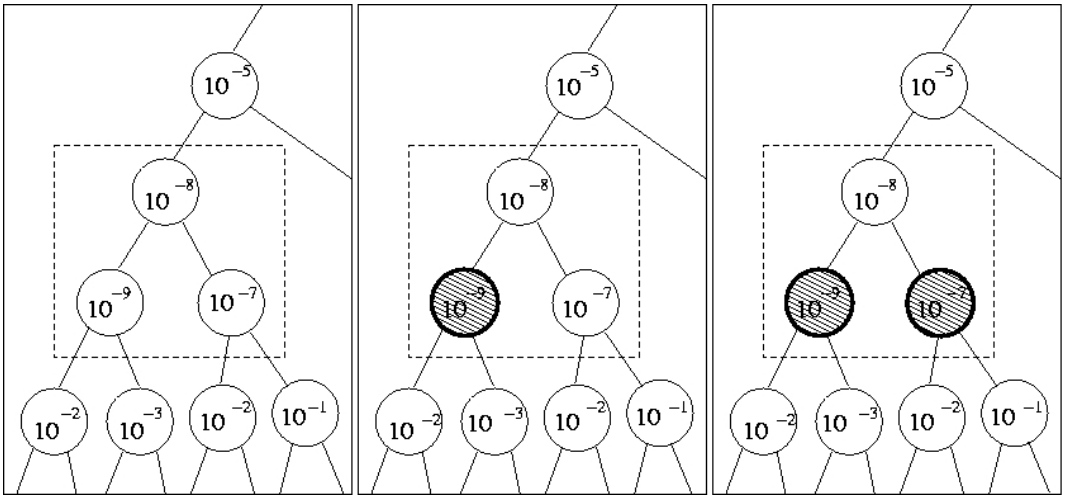
12.6 Conclusion

Finding groups in data sets is a major problem in many fields of knowledge such as statistical pattern recognition, image processing, or data mining. Grouping phenomena are essential in human perception, since they are responsible for the organization of information. In vision, grouping has been especially studied by Gestalt psychologists like Wertheimer [Wer23]. In computer vision, the first attempts in perceptual organization understanding certainly date back to Marr [Mar82]. In his pioneer work on perceptual organization and visual recognition [Low85], D. Lowe proposes a detection framework based on the computation of accidental occurrence. He writes: *"In other words, we can shift our attention from finding properties with high prior expectations to those that are sufficiently constrained to be detectable among a realistic distribution of accidentals.[...] Even when we do not know the ultimate interpretation for some grouping and therefore its particular* a priori *expectation, we can judge it to be significant based on the non-accidentalness criteria.*" In this chapter we proposed a method to find "natural" clusters in a data set, based on this principle of non-accidentalness. This method is inspired by Desolneux *et al.*'s method for detecting dots in an image [DMM03a]. In this method, a hierarchical classification of the set of dots is considered, and meaningful clusters are detected *a contrario* to a standard Poisson null model. Then, maximal meaningful clusters are selected as local maxima of the meaningfulness with respect to inclusion, in the nested hierarchical structure. We considered an extension of this method to multidimensional data, introducing two main improvements:

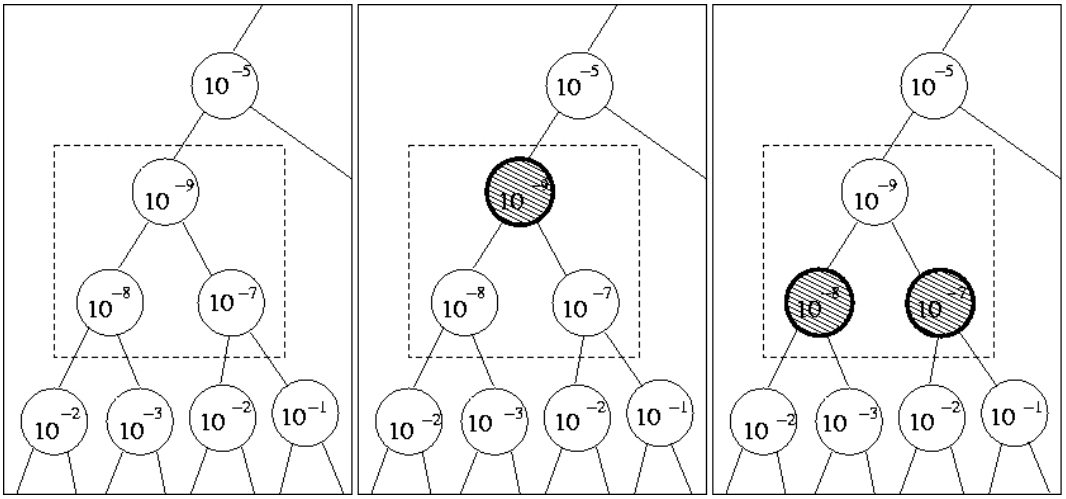
- Data-dependent null models for "a realistic distribution of accidentals" are considered, instead of standard null models, since in general they lead to better results [Gor96].

- The merging criterion described in section12.4.2 is added. This merging criterion solves an important drawback of Desolneux *et al.*'s method related to the selection of maximal clusters, as we illustrate with some examples in Figure 12.7 (see caption for details).

In the next chapter we will apply this general cluster detection framework to the shape matching problem.



(a) *Left: original configuration. Middle: the node selected by Desolneux* et al.*'s method; this maximality criterion yields some relevant misses, such as the node having* $NFA_g = 10^{-7}$*. Right: by combining merging and maximality criteria, both clusters are selected.*



(b) *Left: original configuration. Middle: the node selected by Desolneux* et al.*'s method. Right: selected nodes obtained by combining merging and maximality criteria. The merging criterion decides that the selected pair of nodes is more significant than its ancestor.*

Figure 12.7: *Two local configurations of a dendrogram, and the selection of maximal meaningful groups. The numbers in each node correspond to the* NFA_g *of its associated clusters. The selected nodes, depicted in gray, are candidates to maximal meaningful clusters (in order to be maximal, their* NFA_g *must be lower than the ones of all their descendants and ancestors).*

12.7 Appendix: On the negative association of multinomial distributions

In this section we present the notion of *negative association* (a strong notion of negative dependence) and summarize some relevant consequences, first reported by Joag-Dev and Proschan in [JDP83]. We also complete some proofs, that were just oulined in the original paper, and we apply these general results to multinomial distributions.

Definition 12.7 (Negative association) *A set $\mathcal{X} = \{X_1, \ldots, X_n\}$ of real random variables is said to be negatively associated (NA) if for every two disjoint index sets $I, J \subset \{1, \ldots, n\}$,*

$$\mathbb{E}\left[f(X_i, i \in I) g(X_j, j \in J)\right] \leqslant \mathbb{E}\left[f(X_i, i \in I)\right] \cdot \mathbb{E}\left[g(X_j, j \in J)\right],$$

for all non-decreasing functions $f : \mathbb{R}^{\#I} \to \mathbb{R}$, $g : \mathbb{R}^{\#J} \to \mathbb{R}$ (a function $h : \mathbb{R}^k \to \mathbb{R}$ is said to be non-decreasing if $h(x_1, \ldots, x_k) \geqslant h(y_1, \ldots, y_k)$ whenever $x_1 \leqslant y_1, \ldots, x_k \leqslant y_k$).

Remark: Negative association is a natural generalization of negative correlation.

The negatively associated set $\mathcal{X} = \{X_1, \ldots, X_n\}$ verifies the following properties:

Property 12.1 *For any non-decreasing functions f_i, $i \in \{1, \ldots, n\}$,*

$$\mathbb{E}\left[\prod_{i=1}^{n} f_i(X_i)\right] \leqslant \prod_{i=1}^{n} \mathbb{E}\left[f_i(X_i)\right].$$

Proof: Define $f(x_1, \ldots, x_{n-1}) = \prod_{i=1}^{n-1} f_i(x_i)$ and $g(x_n) = f_n(x_n)$ for all $(x_1, \ldots, x_n) \in \mathbb{R}^n$. Since f and g are both non-decreasing, it follows from definition 12.7 that

$$\mathbb{E}\left[\prod_{i=1}^{n} f_i(X_i)\right] \leqslant \mathbb{E}\left[\prod_{i=1}^{n-1} f_i(X_i)\right] \mathbb{E}\left[f_n(X_n)\right].$$

Using induction yields the desired result. ∎

Property 12.2 *For all $(x_1, \ldots, x_n) \in \mathbb{R}^n$,*

$$\Pr\left(X_i \geqslant x_i \; \forall \; i \in \{1, \ldots, n\}\right) \leqslant \prod_{i=1}^{n} \Pr\left(X_i \geqslant x_i\right).$$

This follows immediately from property 12.1 for $f_i(x) = \chi_{[x \geqslant x_i]}$, the indicator function of event $[x \geqslant x_i]$. The following property is obvious from definition 12.7:

Property 12.3 *Non-decreasing functions defined on disjoint subsets of a set of NA random variables are NA.*

PROPERTY 12.4 *The union of independent sets of NA random variables is NA.*

Proof: Let $\mathbf{X}$ and $\mathbf{Y}$ be independent vectors such that for each one, its components are sets of NA random variables. Let $(\mathbf{X}_1, \mathbf{X}_2)$ and $(\mathbf{Y}_1, \mathbf{Y}_2)$ denote arbitrary partitions of $\mathbf{X}$ and $\mathbf{Y}$ respectively. Hence, the vector $(\mathbf{X}, \mathbf{Y})$ is NA if and only if $\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)g(\mathbf{X}_2, \mathbf{Y}_2)\right] \leqslant \mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)\right]\mathbb{E}\left[g(\mathbf{X}_2, \mathbf{Y}_2)\right]$. Now,

$$\begin{aligned}\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)g(\mathbf{X}_2, \mathbf{Y}_2)\right] &= \mathbb{E}\left\{\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1, \mathbf{Y}_2\right]\right\} \\ &= \sum_{(y_1,y_2)} \mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1 = y_1, \mathbf{Y}_2 = y_2\right] \Pr(\mathbf{Y}_1 = y1, \mathbf{Y}_2 = y2).\end{aligned}$$

Since $(\mathbf{X}_1, \mathbf{X}_2)$ and $(\mathbf{Y}_1, \mathbf{Y}_2)$ are independent, we have that $\{f(\mathbf{X}_1, \mathbf{Y}_1)|\mathbf{Y}_1 = y_1, \mathbf{Y}_2 = y_2\}$ and $\{g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1 = y_1, \mathbf{Y}_2 = y_2\}$ are parametric functions of random vectors $\mathbf{X}_1$ and $\mathbf{X}_2$ respectively. Thus, because of the negative association of $\mathbf{X}$,

$$\begin{aligned}\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1 = y_1, \mathbf{Y}_2 = y_2\right] \leqslant \\ \mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)|\mathbf{Y}_1 = y_1, \mathbf{Y}_2 = y_2\right]\mathbb{E}\left[g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1 = y_1, \mathbf{Y}_2 = y_2\right].\end{aligned}$$

Hence,

$$\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)g(\mathbf{X}_2, \mathbf{Y}_2)\right] \leqslant \mathbb{E}\left\{\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)|\mathbf{Y}_1, \mathbf{Y}_2\right]\mathbb{E}\left[g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1, \mathbf{Y}_2\right]\right\}$$

Now, since conditional expectations $\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)|\mathbf{Y}_1, \mathbf{Y}_2\right]$ and $\mathbb{E}\left[g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1, \mathbf{Y}_2\right]$ are respectively Y_1 and Y_2 measurable functions, it follows that

$$\begin{aligned}h_1(\mathbf{Y}_1) &:= \mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)|\mathbf{Y}_1, \mathbf{Y}_2\right] = \mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)|\mathbf{Y}_1\right], \\ h_2(\mathbf{Y}_2) &:= \mathbb{E}\left[g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_1, \mathbf{Y}_2\right] = \mathbb{E}\left[g(\mathbf{X}_2, \mathbf{Y}_2)|\mathbf{Y}_2\right].\end{aligned}$$

Finally, using that $\mathbf{Y}$ is NA, we have

$$\begin{aligned}\mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)g(\mathbf{X}_2, \mathbf{Y}_2)\right] &\leqslant \mathbb{E}\left[h_1(\mathbf{Y}_1)h_2(\mathbf{Y}_2)\right] \\ &\leqslant \mathbb{E}\left[h_1(\mathbf{Y}_1)\right]\mathbb{E}\left[h_2(\mathbf{Y}_1)\right] \\ &= \mathbb{E}\left[f(\mathbf{X}_1, \mathbf{Y}_1)\right]\mathbb{E}\left[g(\mathbf{X}_2, \mathbf{Y}_2)\right].\end{aligned}$$

■

By combining these we get the following proposition.

PROPOSITION 12.5 *A random vector $\mathbf{X} = (X_1, \ldots, X_n)$ having a multinomial distribution of index M and parameter $\mathbf{p} = (p_1, \ldots, p_n)$ (what we denote by $\mathbf{X} \sim Mult(M, \mathbf{p})$), is NA.*

Proof: We can write $\mathbf{X}$ as

$$\mathbf{X} = \sum_{k=1}^{M} \mathbf{Y}_k,$$

where each $\mathbf{Y}_k \sim Mult(1, \mathbf{p})$, and the $\mathbf{Y}_k$'s are mutually independent. Since, for all $k \in \{1, \ldots, M\}$, all elements in $\mathbf{Y}_k$ are zero except for one whose value is 1, vector $\mathbf{Y}_k$ is NA. Indeed, for all I, J disjoint subsets of $\{1, \ldots, n\}$, for all non-decreasing functions $f : \mathbb{R}^{\#I} \to \mathbb{R}$, $g : \mathbb{R}^{\#J} \to \mathbb{R}$, we have

$$\begin{aligned}
\mathbb{E}\left[f(\mathbf{Y}_{k,i}, i \in I) g(\mathbf{Y}_{k,j}, j \in J)\right] &\leqslant \mathbb{E}\left[f(\mathbf{Y}_{k,i}, i \in I)\right] \cdot \mathbb{E}\left[g(\mathbf{Y}_{k,j}, j \in J)\right] \\
&\Leftrightarrow \mathbb{E}\left[\left(f(\mathbf{Y}_{k,i}, i \in I) - f(0, \ldots, 0)\right)\left(g(\mathbf{Y}_{k,j}, j \in J) - g(0, \ldots, 0)\right)\right] \\
&\qquad \leqslant \mathbb{E}\left[f(\mathbf{Y}_{k,i}, i \in I) - f(0, \ldots, 0)\right] \cdot \mathbb{E}\left[g(\mathbf{Y}_{k,j}, j \in J) - g(0, \ldots, 0)\right],
\end{aligned}$$

and this last inequality is true: the right member is non-negative because $f(\mathbf{Y}_{k,i}, i \in I) - f(0, \ldots, 0)$ and $g(\mathbf{Y}_{k,j}, j \in J) - f(0, \ldots, 0)$ are non-negative, and the left member is zero since $(f(\mathbf{Y}_{k,i}, i \in I) - f(0, \ldots, 0)$ and $g(\mathbf{Y}_{k,j}, j \in J) - f(0, \ldots, 0)$ can not be non-zero at the same time.
Then, using property 12.4, it follows that $\{\mathbf{Y}_1, \ldots, \mathbf{Y}_M\}$ is NA. Finally, since for all $l \in \{1, \ldots, n\}$, $X_l = \sum_{k=1}^{M} \mathbf{Y}_{k,l}$ are non-decreasing functions defined on disjoint subsets of $\{\mathbf{Y}_1, \ldots, \mathbf{Y}_M\}$, we have that $\mathbf{X}$ is NA (property 12.3). ■

Remark: Applying property 12.7 to the random vector $\mathbf{X}$ proves lemma 12.1, stated in section 12.4.2.

Chapter 13

Grouping spatially coherent meaningful matches

Abstract: Up to here we have dealt with local representations of shape contents in images. Consequently, common parts between images were described in terms of matched shape elements. The recognition of "global shapes" needs for an integration of the recognized partial shapes. This integration of local information is certainly not performed as a simple conjunction of the recognized partial shapes. Indeed, the way these common shape elements are organized in the image plane can trigger another complementary recognition process, allowing to recognize "global shapes".

In this chapter we reinforce the recognition confidence of our method, by combining the spatial information furnished by matched shape elements. Each pair of matching shape element leads to a unique transformation between images, which can be represented as a pattern in a transformation space. Hence, spatially coherent meaningful matches correspond to clusters in the transformation space, and their detection can then be formulated as a clustering problem, which can be solved as a particular case of the theory developed in Chapter 12.

Résumé : Jusqu'ici nous ne nous sommes intéressés qu'aux représentations locales des formes contenues dans les images. Par conséquent, les parties communes entre les images ont été décrites en termes d'éléments de formes mis en correspondance. La reconnaissance de « formes globales » nécessite de tenir en compte les positions relatives des formes partielles reconnues, ce qui ne peut pas être réduit à une simple conjonction des informations locales. En effet, la manière dont ces éléments de formes communs sont organisés dans le plan image peut déclencher un processus de reconnaissance complémentaire, permettant de reconnaître des « formes globales ».

Dans ce chapitre nous renforçons la mesure de confiance en la reconnaissance, en combinant l'information spatiale fournie par les éléments de forme appariés. Chaque paire d'éléments de formes correspond à une unique transformation entre les images, pouvant être représentée comme un élément de l'espace des transformations. Ainsi, les appariements significatifs spatialement cohérents correspondent à des groupes dans l'espace des transformations. Leur détection se ramène donc à un problème d'identification de groupes, qui peut être résolu comme un cas particulier de la théorie développée au chapitre 12.

"Structure means recognition that unity is at the foundation of everything. To say structure is also to say Abstraction: geometry, rhythm, proportion, lines, planes, idea of object. These are elements of work – they act, they form, they construct and gain significance through the law of unity."

Joaquín Torres-García, Estructura.

13.1 Why spatial coherence detection?

Looking at Figure 13.1, one can recognize on the top left image a detail of Uccello's painting shown on the bottom left image. These two images correspond to different snapshots, and present different colors and different compression rates. However, geometrical shape information in their common part appears to be preserved, and if we ask someone if these pictures share some part, the person would answer "yes" without any doubt, and would also be able to localize the detail on the top, in the painting on the bottom.

If we now look for common shape elements between these two images by means of our meaningful matching method (Figure 13.2), we will find the results quite surprising: there are only sixteen meaningful matches, and among them there are seven false matches (Figure 13.3). Indeed, since images have been damaged by the jpeg compression and since they are at very different scales, there are not as many similar level lines in the common part as we could have imagined. Besides, looking closely at the false matches in Figure 13.2, despite not being very meaningful (their NFA are all larger than 0.45), we see that many of them are very similar shape elements (up to similarity transformations).

These remarks lead us to an obvious conclusion: spatial coherence of matched shape elements plays a major role in recognition. (Early studies in motion perception explain this phenomenon by the fact that most of the structures in the visual world are rigid or at least nearly so, see Marr [Mar82].) The meaningful matching method does not use this information at all, since its goal is to compare shape elements, no matter where they come from. Nevertheless, as a secondary effect, it gives also sure detections of common objects (see for instance, the result we obtain when thresholding the NFA at 0.1, Figure 13.2). The goal now is to incorporate spatial coherence information to our detection framework, in order to attain much more confidence on the detected structures. Indeed, if we stick to the detection of meaningful events *a contrario* to some null hypothesis, we expect that, observing "by chance" several meaningful matches at the time, presenting moreover a spatial coherence, will be still more unprobable.

In order to detect spatially coherent matches, we have to define a measure of resemblance between *groups of shape elements*. It is sound to do it, indirectly, by considering the similarity (resp. the affine) transformations between pairs of pieces of level lines defining shape elements. Hence, instead of defining a measure of resemblance between groups of shape elements, we will define a *spatial coherence measure* on groups of similarity (resp. affine) transformations. To each meaningful match be-

tween shape elements $\mathcal{S}$ and $\mathcal{S}'$, we can associate a similarity (or affine) transformation T. The *spatial coherence measure* of a group of transformations G included in $\mathcal{T}$, where $\mathcal{T}$ is the set of all similarity (resp. affine) transformations associated to meaningful matches, will be denoted by $NFA_g(G)$. Hence, we have turned the detection of spatial coherence into a clustering problem, which can be solved based on the *a contrario* detection methodology presented in Chapter 12.
Recall that in Chapter 1 we have defined a *shape* as any common part between any two images $\mathcal{I}$ and $\mathcal{I}'$, *modulo* a class of invariance (Definition 1.1). According to this definition, groups of shape elements which are present in two images define new shapes.
As we will see and illustrate with several experiments, the strategy we propose in this chapter will enable us to:

- Eliminate matches between similar but spatially uncoherent shape elements,
- Detect groups of spatially coherent shape elements that are common to a pair of images (a target image $\mathcal{I}$ and a scene image $\mathcal{I}'$),
- Increase the confidence on the detected shapes,
- Perform subsequent applications such as registration or motion estimation.

The plan of this chapter is as follows. In Section 13.2 we present the parameter space of similarity and affine transformations, mainly to introduce some notations. Then, in Section 13.3 we describe two classical methods for object detection based on spatial coherence. As we will see, these methods suffer from two common flaws: high sensitivity to discretization of the parameter space, and arbitrary choice of decision thresholds. In Section 13.4 we propose a transformation clustering method based on the general clustering ideas presented in Chapter 12, and we describe some issues that are specific to clustering of transformations. Several experiments are also shown. Then we discuss some related work in Section 13.6, before concluding in Section 13.7.

13.2 The transformation space

In Chapter 7 we presented local normalization methods for planar curves, invariant to similarity or to affine transformations. Then, in Chapter 8 we proposed a decision framework for matching encoded normalized pieces of level lines (defining so the meaningful matches). Let us denote by $\mathcal{I}$ the target image and $\mathcal{I}'$ the scene image, by Ω and Ω' their supports, which are bounded subsets of $\mathbb{R}^2$. Underlying any meaninful match between a shape element $\mathcal{S}$ in $\mathcal{I}$ and a shape element $\mathcal{S}'$ in $\mathcal{I}'$, there is a geometric transformation (a similarity or an affine transform, depending on the choice of the normalization), that can be derived from the normalization procedure of these shape elements. In what follows we describe the parameters involved in these transformations, and the way they can be estimated, for the similarity and the affine transformation cases.

Figure 13.1: *"Uccello" experiment. Original images and maximal meaningful level lines. Top: target image, bottom: scene image.*

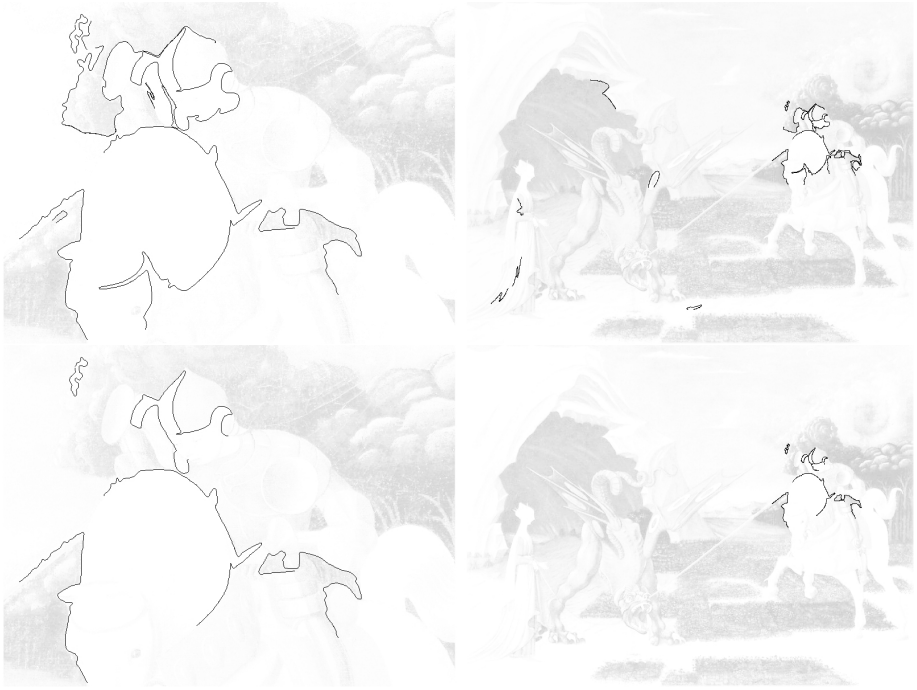


Figure 13.2: *"Uccello" experiment: meaningful matches. The number of tests was* $39\ 10^6$ *(*1022 *codes in the target image and* 38149 *in the scene image). On top: all* 16 *meaningful matches. Bottom: matches of which* $NFA < 0.1$.

13.2.1 The similarity transformation space

Let us denote by $\mathcal{B}_S$ and $\mathcal{B}'_S$ the set of local similarity frames extracted from $\mathcal{I}$ and $\mathcal{I}'$, as described in Chapter 7:

$$\mathcal{B}_S = \{\{R_1, R_2\} \text{ local similarity frame of } \mathcal{S} \text{ s.t. } \mathcal{S} \text{ is an encoded shape element in } \mathcal{I}\},$$
$$\mathcal{B}'_S = \{\{R'_1, R'_2\} \text{ local similarity frame of } \mathcal{S}' \text{ s.t. } \mathcal{S}' \text{ is an encoded shape element in } \mathcal{I}'\}.$$

We denote by (x_{R_1}, y_{R_1}) and by (x'_{R_1}, y'_{R_1}) the pair of coordinates of R_1 and R'_1, respectively, and by $\left\{\left(\{R_1^k, R_2^k\}, \{R_1^{k'}, R_2^{k'}\}\right), 1 \leqslant k \leqslant M\right\}$ the set of all pairs of frames associated to a meaningful match:

$$\left(\{R_1^k, R_2^k\}, \{R_1^{k'}, R_2^{k'}\}\right) \text{ s.t. } NFA(\mathcal{S}_k, d(\mathcal{S}_k, \mathcal{S}'_k)) \leqslant 1.$$

The 2-D similarity transformation $\mathbf{T} : \Omega \to \Omega'$ which maps R_1 into R'_1 and R_2 into R'_2 (see Figure 13.4) can be written as:

$$\forall\ (x, y) \in \Omega,\quad \mathbf{T}(x, y) = s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}, \tag{13.1}$$

where

$$s = \frac{\|R'_2 - R'_1\|}{\|R_2 - R_1\|},\quad \theta = \arctan\left(\frac{(R_2 - R_1) \times (R'_2 - R'_1)}{(R_2 - R_1) \cdot (R'_2 - R'_1)}\right) \text{ and}$$
$$\begin{pmatrix} t_x \\ t_y \end{pmatrix} = \begin{pmatrix} x'_{R_1} - s\cos(\theta)x_{R_1} + s\sin(\theta)y_{R_1} \\ y'_{R_1} - s\sin(\theta)x_{R_1} - s\cos(\theta)y_{R_1} \end{pmatrix}.$$

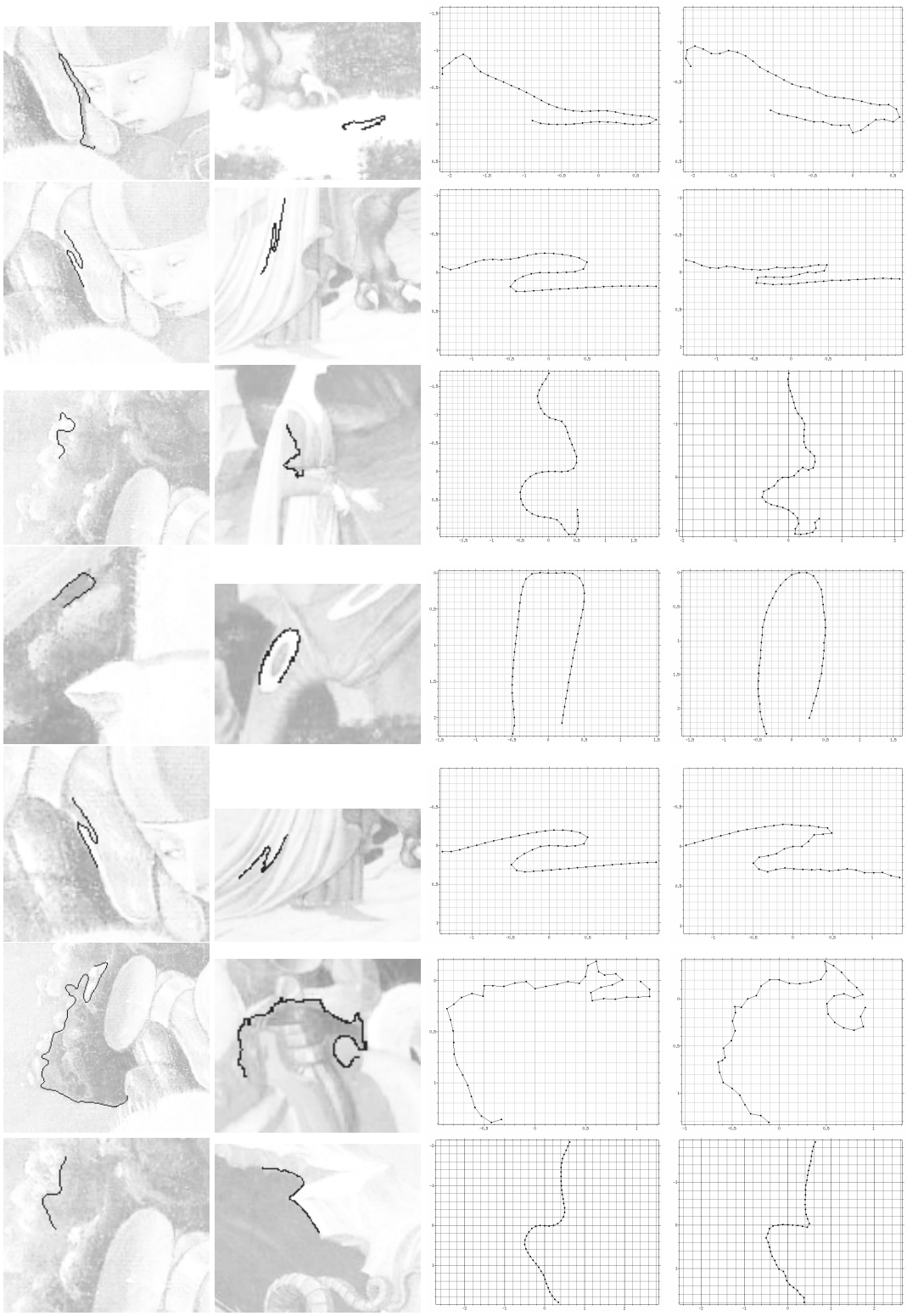


Figure 13.3: *All 7 false matches with their corresponding normalized codes. Their NFA (from top to bottom) are:* $0.45, 0.49, 0.50, 0.67, 0.77, 0.90, 0.96$. *While matched shape elements do not correspond semantically to the same objects, they are roughly similar.*

(here "$\times$" and "$\cdot$" denote cross and dot product, respectively).

Plane similarity transformations define a 4-D parameter space; we refer to it as the similarity transformation space. Each pair of frames $(\{R_1, R_2\}, \{R'_1, R'_2\})$ defines a point $T = (t_x, t_y, \theta, s)$ in the similarity transformation space, corresponding to the transformation $\mathbf{T}$. Let us denote by $\mathcal{T} = \left\{T_k = (t_{x_k}, t_{y_k}, \theta_k, s_k),\ 1 \leqslant k \leqslant M\right\}$ the set of points in the similarity transformation space associated to the meaningful matches of all corresponding frames $\left(\{R_1^k, R_2^k\}, \{R_1^{k'}, R_2^{k'}\}\right)$.

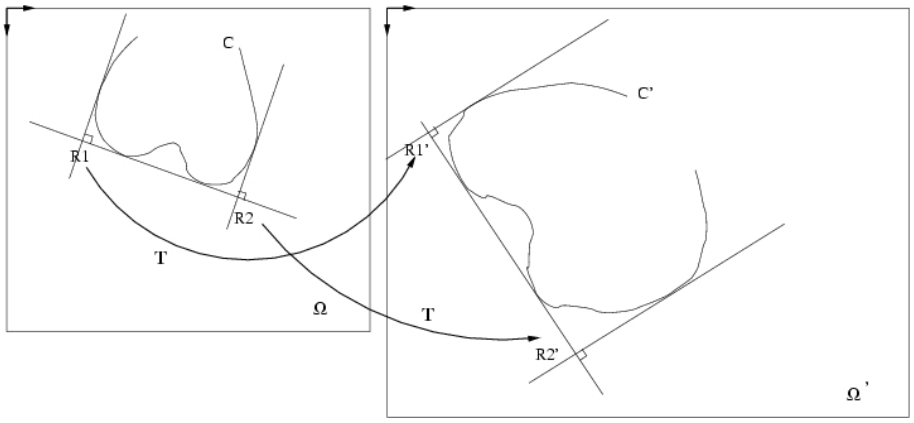


Figure 13.4: *Two pieces of level lines and their corresponding local similarity frames. The similarity* $\mathbf{T}$ *maps* R_1 *into* R'_1 *and* R_2 *into* R'_2.

13.2.2 The affine transformation space

We now consider the local affine invariant normalization described in Chapter 7. Affine normalization of a piece of curve was performed by mapping its local frame $\{R_1, R_2, R_3\}$ into the triplet $\{(0,0), (1,0), (0,1)\}$ (we use the same name for these points and from those involved in the similarity normalization in order to be consistent with the notation in Chapter 7, but there is of course no relation between them). Let us denote by $\mathcal{B}_A$ and $\mathcal{B}'_A$ the set of local affine frames extracted from $\mathcal{I}$ and $\mathcal{I}'$:

$$\mathcal{B}_A = \{\{R_1, R_2, R_3\} \text{ local affine frame of } \mathcal{S} \text{ s.t. } \mathcal{S} \text{ is an encoded shape element in } \mathcal{I}\},$$
$$\mathcal{B}'_A = \{\{R'_1, R'_2, R'_3\} \text{ local affine frame of } \mathcal{S}' \text{ s.t. } \mathcal{S}' \text{ is an encoded shape element in } \mathcal{I}'\}.$$

We denote by (x_{R_1}, y_{R_1}) and by (x'_{R_1}, y'_{R_1}) the pair of coordinates of R_1 and R'_1, respectively, and by $\left\{\left(\{R_1^k, R_2^k, R_3^k\}, \{R_1^{k'}, R_2^{k'}, R_3^{k'}\}\right), 1 \leqslant k \leqslant M\right\}$ the set of all pairs of frames associated to a meaningful match:

$$\left(\{R_1^k, R_2^k, R_3^k\}, \{R_1^{k'}, R_2^{k'}, R_3^{k'}\}\right) \text{ s.t. } NFA(\mathcal{S}_k, d(\mathcal{S}_k, \mathcal{S}'_k)) \leqslant 1.$$

The planar affine transformation $\mathbf{T} : \Omega \to \Omega'$ which maps R_1 into R'_1, R_2 into R'_2 and R_3 into R'_3 can be uniquely expressed as follows (uniqueness comes from the Cholesky decompositon):

$$\forall\ (x,y) \in \Omega,\quad \mathbf{T}(x,y) = \underbrace{\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & \varphi \\ 0 & 1 \end{pmatrix} \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix}}_{\mathbf{M}} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}. \tag{13.2}$$

Given $\{R_1, R_2, R_3\}$ and $\{R'_1, R'_2, R'_3\}$, solving for $\mathbf{M} = ((m_{ij}))$ is straightforward. Then, one can compute the transformation parameters in (13.2) by means of the following formulas:

$$\begin{aligned} \theta &= \arctan(m_{21}/m_{22}), \\ \varphi &= (m_{11}m_{12} + m_{21}m_{22}) \,/\, (m_{11}m_{22} - m_{12}m_{21}), \\ s_x &= \sqrt{m_{11}^2 + m_{21}^2}, \\ s_y &= (m_{11}m_{22} - m_{12}m_{21}) \,/\, \sqrt{m_{11}^2 + m_{21}^2}, \\ \begin{pmatrix} t_x \\ t_y \end{pmatrix} &= \begin{pmatrix} x'_{R_1} \\ y'_{R_1} \end{pmatrix} - M \begin{pmatrix} x_{R_1} \\ y_{R_1} \end{pmatrix}. \end{aligned}$$

We call affine transformation space the 6-D parameter space defined by these six variables. Each pair of frames $(\{R_1, R_2, R_3\}, \{R'_1, R'_2, R'_3\})$ defines a point $T = (t_x, t_y, \theta, \varphi, s_x, s_y)$ in the transformation space. We will denote by $\mathcal{T} = \left\{T_k = (t_{xk}, t_{y_k}, \theta_k, \varphi_k, s_{xk}, s_{y_k}),\ 1 \leqslant k \leqslant M\right\}$ the set of points in the affine transformation space associated to the meaningful matches corresponding to frames $\left(\{R_1^k, R_2^k, R_3^k\}, \{R_1^{k\prime}, R_2^{k\prime}, R_3^{k\prime}\}\right)$.
Note that, for sake of commodity, we use the same notation for points in the similarity and in the affine transformation spaces. Since most part of what we present in the following sections holds for both groups of transformations, unless precised, we will use this notation to refer to them indistinctly.

13.3 Two classical methods for object detection based on spatial coherence

In this section we discuss some issues of the generalized Hough transform [Bal81], of which variations are probably the most widely used techniques in object detection. Then we will describe another frequently used technique for robust transformation estimation: the RANSAC algorithm [FB81].

13.3.1 The generalized Hough transform

In [Bal81], Ballard proposed a generalization to the Hough transform [Hou62] allowing to detect arbitrary planar shapes undergoing similarity transformations. Most of object detection and recognition systems using transformations clustering are based on the generalized Hough transform. The

basic idea is to quantize the transformation space into D-dimensional cells. Each transformation point T_i is quantized, and then votes for one of these cells. In practice, noise and image quantization induce localization errors in the extracted features, and one has to take into account uncertainty in computing T_i. Thus, each pairing of model and image features defines a volume of possible transformations, so it should cast a vote into each cell intersecting this volume (see [GH90] for an error analysis when using line segments as features).
As like as all techniques based on histograms in multidimensional spaces, the generalized Hough method is very sensitive to the choice of quantization precision (this remark also holds for Lamdan and Wolfson's Geometric Hashing [WR97, LW88], described in Chapter 2). Most of the time, the cell size is chosen by problem specific *ad hoc* arguments (see [Low99] for an example). However, in the general case, quantization effects may lead to several problems:

- Similar transformation points may vote for different cells. In order to reduce this problem, either votes are counted by adding the votes of neighboring cells (using a sliding window) in the case of no uncertainty in T_i, or, when uncertainty is considered, a vote is casted into each cell intersecting the uncertainty volume.

- In the plane similarity case, for instance, if one wants to do a fine discretization of the 4-D transformation space in order to perform accurate detection, the search space is too large for exhaustive search. Coarse to fine techniques applied to transformation clustering, first introduced by Stockman [SKB82], can deal with this complexity problem, but there is no reason why the most voted cells at the finer scale correspond to the most voted ones at coarser scales.

- From the detection viewpoint, the cells size is also crucial. Indeed, if quantization is too fine, cells will not have enough votes and correct instances will be missed (false negatives). On the other side, choosing a very coarse quantization increases the likelihood of large clusters occurring at random (false positives). Moreover, as pointed out by Grimson and Huttenlocher in [GH90], using a local neighborhood or casting multiple votes, and reducing the number of cells to reduce the search space, both methods greatly increase the chance of large random clusters.

These remarks partially motivate our decision of using the clustering techniques described in Chapter 12, along with the validity assessment method proposed in the same chapter. Indeed, the proposed methodology does not suffer from quantization problems.

13.3.2 A RANSAC based approach

The "RANdom SAmple Consensus" algorithm by Fischler and Bolles [FB81] (RANSAC), is certainly one of the most popular robust estimators in computer vision. It has proved very successful in stereo vision tasks, such as the estimation of homographies and fundamental matrices [HZ00]. The main reasons of its success are its quite general nature, and its ability to deal with large proportions of

outliers. Roughly speaking, in its general form, the RANSAC procedure to fit a model consists in randomly selecting a minimal subset of the data (*i.e.* a subset allowing to instantiate the model), then computing the number of inliers consistent with the instantiated model. These two steps are repeated for N minimal subsets of the data. The model having the largest number of inliers is chosen, and it is refined by re-estimating it from the corresponding set of inliers.
In our framework, we deal with M meaningful matches, and usually M is small enough to test for all corresponding similarity or affine transformations. Hence, using the same ideas, an elementary algorithm would be as follows:

- For each element in the set of M pairs of local frames corresponding to meaningful matches:
  1. Compute the associated transformation T.
  2. Apply T to all target local frames, and compute their distances to their corresponding scene local frames.
  3. Compute the number of inliers consistent with T, *i.e.* the pairs for which the distance is less than d pixels.
- Choose the transformation T having the largest number of inliers.
- Re-estimate T for all pairs of local frames determined as inliers (with a least squares method, for instance).

One can iterate this procedure on the set of outliers, in order to find other (less dominant) transformations.
Even for this simple version of the algorithm, two problems arise: the choice of the distance threshold d, and the minimum number of inliers a model should have in order to be valid. The distance threshold d is usually chosen empirically; otherwise, it can be chosen by considering a significance level α, corresponding to the probability that a point is an inlier [HZ00], what requires hypothetizing a model for the distribution of distances. Concerning the minimum number of inliers to assess model validity, generally it is also fixed by means of arbitrary rules. It seems reasonable to us that this minimum number of inliers depends on the distance threshold, but up to our knowledge, no effort has been done to establish this relation.
In the next section we propose a transformation clustering method based on the general clustering concepts presented in Chapter 12. This method is able to detect multiple transformations, and does not suffer from arbitrary or *ad hoc* choices like the ones we have just described.

13.4 Meaningful clusters of transformations and shape detection

Since corresponding pairs of frames that are part of the same match of a model to an image will result in approximately the same transformations, we can formulate the problem of shape detection as a transformation clustering problem. According to Chapter 12, in order to do it we have to define:

1. A dissimilarity measure between points in the transformation space,
2. A grouping strategy.

Defining a metric between transformations is not trivial, mainly because of two reasons. The first one is that the transformation space consists in directions of which magnitudes are not directly comparable. This problem is not specific to transformation clustering but general to clustering of any kind of data, and was discussed in Chapter 12. The second reason is that the (similarity or affine) transformation space does not have a vector space structure, which means we cannot even derive metrics from norms.
We can easily get rid of these two drawbacks by defining dissimilarity between transformations based on distances in the image plane. A possible distance function between *similarity transformations* is

$$d_T(T_i, T_j) = \max\left\{\|\mathbf{T}_i(R_1^k) - \mathbf{T}_j(R_1^k)\|,\ \|\mathbf{T}_i(R_2^k) - \mathbf{T}_j(R_2^k)\|,\ 1 \leqslant k \leqslant M\right\},\ 1 \leqslant i, j \leqslant M,$$

where $\|\cdot\|$ denotes the ℓ_2-norm, $\left\{\{R_1^k, R_2^k\}, 1 \leqslant k \leqslant M\right\}$ is the set of all local similarity frames of shape elements $\mathcal{S}_k$ such that $(\mathcal{S}_k, \mathcal{S}'_k)$ is a meaningful match, and $\mathbf{T}_k$ its corresponding transformation defined by (13.1). Note that this distance is actually a true metric adapted to the considered transformations; symmetry, non negativity and triangle inequality properties clearly hold, and $d_T(T_i, T_j) = 0$ if and only if $T_i = T_j$ because, since $\mathbf{T}_i$ and $\mathbf{T}_j$ are similarity transformations,

$$\left[\mathbf{T}_i(R_1^i) = \mathbf{T}_j(R_1^i) \text{ and } \mathbf{T}_i(R_2^i) = \mathbf{T}_j(R_2^i)\right] \Leftrightarrow \mathbf{T}_i = \mathbf{T}_j.$$

In practice, we will consider the following dissimilarity function, since its computation is less time consuming than for the distance we have just described :

$$d_T(T_i, T_j) = \max\left\{\|\mathbf{T}_i(R_1^k) - \mathbf{T}_j(R_1^k)\|,\ \|\mathbf{T}_i(R_2^k) - \mathbf{T}_j(R_2^k)\| \ :\ k \in \{i, j\}\right\},\ 1 \leqslant i, j \leqslant M. \tag{13.3}$$

Notice that here the maximum is taken only over the pairs of local frames $\{R_1^i, R_2^i\}$ and $\{R_1^j, R_2^j\}$ that were used to define $\mathbf{T}_i$ and $\mathbf{T}_j$. Although this dissimilarity measure does not satisfy the triangle inequality and is therefore neither a norm nor a true metric, it is always non negative, symmetric, and equals zero only if $T_i = T_j$.
A dissimilarity function between affine transformations can be defined in the same way, by considering the affine local frames.

Once a dissimilarity measure between transformation points has been defined, a grouping strategy has to be chosen. In Chapter 12 we discussed partitional and hierarchical clustering methods, and we

presented several reasons for prefering hierarchical methods. Basically, this decision was based on the ability to deal with differently shaped clusters, the possibility to detect all natural clusters even when their number of points or their variances were different, as well as to extract dense clusters embedded in less dense ones. All these requirements were accomplished by hierarchical clustering methods, in particular by the single-linkage method, but not by partitional methods. Last but not least, we also saw that the problem of cluster validity assessment was better posed for hierarchical methods. Local stopping rules, as the validity criteria for meaningful groups that we proposed in Chapter 12, should be able to avoid, in many cases, undesirable chaining effects inherent to the single-linkage method. Thus, in a general setting, the single-linkage method combined with the detection of maximal meaningful groups, would be a reasonable choice.

Let us now summarize some features that are particular to clusters of transformations, to show that the proposed set up is certainly suitable for transformation clustering. Figure 13.5 shows two 2-D projections of the transformation points T_k corresponding to the meaningful matches between a pair of images. These are the typical clusters we may observe in our framework. Notice that here natural clusters are not isotropic at all, and surrounding points are more sparsely distributed than in its core. This kind of noisy points are a consequence of some slight instability in the extraction-normalization-encoding of pieces of level lines.

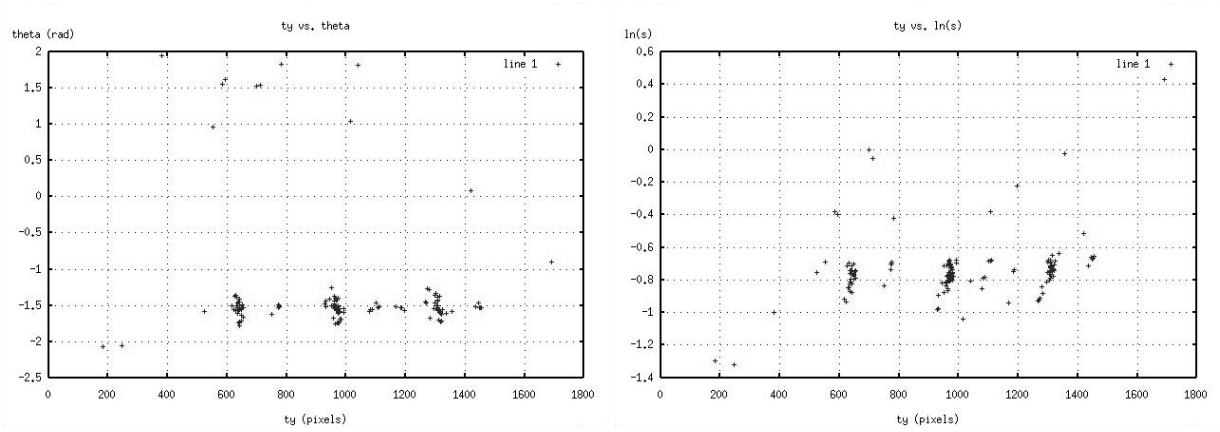


Figure 13.5: *Two 2-D projections of a typical cloud of transformations associated to meaningful matches.*

13.4.1 *A contrario* definition of meaningful groups

The background model

The aim of this section is to define an accurate background model, allowing to detect spatial coherence *a contrario*, that is, by rejecting the hypothesis that clusters have been generated by the background process. That is why the background model must take into account all artifacts inherent to the data generation process (these models are the so called "data-influenced" null models, which perform better than the "standard" null models; see Chapter 12, Section 12.3). Thus, in our transformation clustering framework, the background model should describe the following situation (which

we describe here for the similarity case, and can be easily derived for the affine case):

- images shape contents are represented by encoded pieces of level lines (shape elements), along with sets of local similarity frames, localized nearby meaningful level lines;
- no shape coincidence between the target image $\mathcal{I}$ and the scene image $\mathcal{I}'$ is observed.

In other words, our background process is responsible for the generation of all local similarity transformations defined by sets of local frames $\mathcal{B}_S$ and $\mathcal{B}'_S$, in the absence of global shape coincidence between images $\mathcal{I}$ and $\mathcal{I}'$. Hence, we may assume that:

(A1) Parameters t_x, t_y, θ and s are random variables, θ and s are statistically independent, and the "region" probability of any hyper-rectangle $R = R_x \times R_y \times R_\theta \times R_s$ in the transformation space,

$$p(R) := \Pr\left((t_x, t_y, \theta, s) \in R\right) \tag{13.4}$$

$$= \int_{R_\theta \times R_s} \Pr(t_x \in R_x, t_y \in R_y | \theta, s) dP(\theta) dP(s) \tag{13.5}$$

can be estimated from the pair of images $\mathcal{I}$ and $\mathcal{I}'$ (in a way we immediately precise).

(A2) Points $T_j = (t_{xj}, t_{yj}, \theta_j, s_j)$, $j \in \{1, \ldots, M\}$, are mutually independent, identically distributed random variables, following the joint law of (t_x, t_y, θ, s).

In assumption (A2) we are also implicitly assuming that, under the *a contrario* model, the distribution of transformation points does not depend on the meaningfulness of the match that generates it. Concerning assumption (A1), in order to estimate the *a contrario* region probability $p(R)$ from the pair of images, no instance of the target shapes should be present in the scene. Actually we do not know in advance if it is the case or not: this is one of the questions we want to answer. However this is not really a problem since in detection problems, the background strongly dominates the image statistics. Hence, one can estimate the region probabilities as follows:
Assume domains R_θ and R_s are "small" (otherwise, partition these domains and write (13.5) as a sum of integrals over small domains). Under reasonable regularity conditions on $(\theta, s) \mapsto \Pr(t_x \in R_x, t_y \in R_y | \theta, s)$, if R_θ and R_s are small, for all $(\theta, s) \in R_\theta \times R_s$ one can consider the following approximation:

$$\Pr(t_x \in R_x, t_y \in R_y | \theta \in R_\theta, s \in R_s) \simeq \Pr(t_x \in R_x, t_y \in R_y | \bar{\theta}, \bar{s}),$$

where $\bar{\theta}$ and $\bar{s}$ denote respectively the centers of intervals R_θ and R_s. Replacing this approximation in (13.5) yields

$$p(R) \simeq \Pr(t_x \in R_x, t_y \in R_y | \bar{\theta}, \bar{s}) \Pr(\theta \in R_\theta) \Pr(s \in R_s).$$

Hence, we are led to compute:

1. $\Pr(\theta \in R_\theta)$ and $\Pr(s \in R_s)$: Given the sets $\mathcal{B}_S$ and $\mathcal{B}'_S$ of all local frames from images $\mathcal{I}$ and $\mathcal{I}'$, compute the $\#\mathcal{B}_S \times \#\mathcal{B}'_S$ similarity transformations mapping a frame $\{R_1, R_2\}$ into a frame $\{R'_1, R'_2\}$. Then estimate $\Pr(\theta \in R_\theta)$ and $\Pr(s \in R_s)$ as empirical frequencies over all $\#\mathcal{B}_S \times \#\mathcal{B}'_S$ possible similarities.

2. $\Pr(t_x \in R_x, t_y \in R_y | \bar{\theta}, \bar{s})$: The idea is to infer what (t_x, t_y) should be if $\bar{\theta}$ and $\bar{s}$ are given, and local frames are distributed in images $\mathcal{I}$ and $\mathcal{I}'$ according to points R_1^i $(1 \leqslant i \leqslant N)$ and $R_1'^j$ $(1 \leqslant j \leqslant N')$, respectively. Hence, one can estimate the *a contrario* probability $\Pr(t_x \in R_x, t_y \in R_y | \bar{\theta}, \bar{s})$ as an empirical frequency over all possible translations:

$$\left\{ R_1'^{j\,\mathrm{T}} - \bar{s} \begin{pmatrix} \cos\bar{\theta} & -\sin\bar{\theta} \\ \sin\bar{\theta} & \cos\bar{\theta} \end{pmatrix} R_1^{i\,\mathrm{T}} \text{ s.t. } l \leqslant \bar{s}\|R_2^i - R_1^i\| \leqslant L \ : \ 1 \leqslant i \leqslant N, 1 \leqslant j \leqslant N' \right\} \tag{13.6}$$

   (l and L are thresholds related to the minimal and maximal sizes of considered shape elements. Indeed, since too short shape elements are not informative and the too long ones are too global, they were not encoded and consequently, they cannot be considered in this estimation).

Let us say a few words about the estimation of these three probabilities for the background model. For $\Pr(\theta \in R_\theta)$, one would expect θ to be uniformly distributed in $[-\pi, \pi)$, and this belief was experimentally confirmed (see Figure 13.6(a) for an example of empirical θ distribution from the "Uccello" experiment of Figures 13.1 and 13.2). Concerning $\Pr(s \in R_s)$, the distribution of the zoom factor s is far from being uniform. Figure 13.6b shows an example of $\ln(s)$ empirical distribution from the same "Uccello" experiment. This distribution also confirms that the size distribution of shapes in images is far from being uniform (see [AGM99] for a study on the distribution of scales in natural images). Indeed, one can show that if size distribution was uniform, the histogram $h_s(x)$ of $\ln(s)$ should be proportional to $\mathrm{e}^{-|x|}$. Finally, for the estimation of $\Pr(t_x \in R_x, t_y \in R_y | \theta \in R_\theta, s \in R_s)$ we can also say that, just as for the distribution of s, if we do not have any *a priori* information on the scene image, we can by no means assume a realistic *a priori* distribution for (t_x, t_y) given (θ, s). Since this distribution depends on the location of local frames, it can not be pre-computed for a representative set of $(\bar{\theta}, \bar{s})$ unless the distribution of local frames in the images is well known. Figure 13.7 shows the distribution of points $\{R'_1, R'_2\}$ of local frames in image $\mathcal{I}'$ for the "Uccello" experiment (Figure 13.1). Assuming an *a priori* distribution for these points makes no sense if one wants to detect shapes *a contrario* from these particular target and scene images.

On the other hand, in specific settings such as object detection in fixed kind of scenes, one can manage to learn all necessary distributions once for all. For instance, one can learn the distribution of s by direct computation, and compute that of (t_x, t_y) conditionnally to $(\bar{\theta}, \bar{s})$ for any value of $(\bar{\theta}, \bar{s})$ by estimating the distribution of frames in target and scene images and using formula (13.6).

Remark: The ideas we have presented here also hold for the affine transformation clustering. For this case, θ, φ, s_x and s_y are considered to be mutually independent. Then, as much as we did here, the joint probability of (t_x, t_y) given $(\theta, \varphi, s_x, s_y)$ is computed.

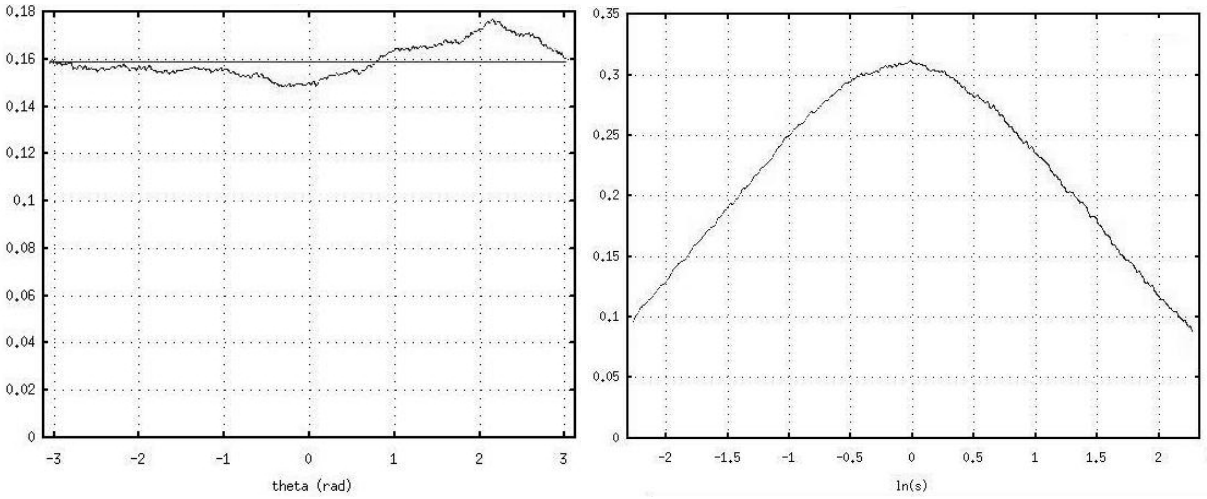


Figure 13.6: *histograms for θ (left) and $\ln s$ (right) from the "Uccello" experiment. Notice θ is (almost) uniformly distributed. The histogram of $\ln s$ is very image dependent, so it is impossible to assume an* a priori *distribution unless dealing with specific applications where image statistics are well known.*

Figure 13.7: *Distribution of frames points R'_1 (left) and R'_2 (right) on the "Uccello" $\mathcal{I}'$ image of Figure 13.1. Points are distributed nearby meaningful level lines.*

Meaningful groups

Now that we have an accurate background model, we are in position to evaluate the probability of a given cluster of transformation points as a realization of the background process. Hence, we are able to detect relevant clusters by Helmholtz principle: those clusters being unlikely to be observed by chance will be considered as meaningful groups. Let us give an example to illustrate this idea. In Figure 13.8, we display the six 2-D projections of the transformation points T_k corresponding to the "Uccello" meaningful matches (Figure 13.2). The "high density" cluster we observe reveals a conspicuous coincidence. Indeed, the probability of its being a realization of the background process should be very low, and one would expect it to be an exception to randomness. Then, a better explanation for this group of matches will be shape coincidence, which is actually the case here.
Let us make things more precise. For any hyper-rectangle R in the transformation space, we know how to compute the "region probability" $p(R)$, the probability that a transformation point generated by the background process falls in R. The next step prior to detection of relevant clusters, is to define a set of reasonable cluster candidates. This choice was discussed in Chapter 12. For the sake of clarity, let us summarize for the particular case of transformation clustering, some definitions and results that were presented for the general case in Chapter 12.

DEFINITION 13.1 (ε-MEANINGFUL GROUP) *We say that a group of matches is ε-meaningful if*

$$\#\mathcal{R} \cdot \mathcal{B}(M, k, p(R)) \leqslant \varepsilon.$$

Remark: Clusters of transformations correspond to groups of meaningful matches. Therefore, from now on, we refer indistinctly to *meaningful groups of matches* or *meaningful clusters of transformations.*

PROPOSITION 13.1 *The expected number of ε-meaningful groups in $\mathcal{R}$ is less than ε.*

DEFINITION 13.2 (NUMBER OF FALSE ALARMS) *Given a group G of k meaningful matches among M with transformations located in cell R, we call number of false alarms of G the number*

$$NFA_g(G) = \#\mathcal{R} \cdot \mathcal{B}(M, k, p(R)),$$

where $p(R)$ is the probability that a transformation point falls in R, the smallest hyper-rectangle in $\mathcal{R}$ containing all k transformation points.

The number of false alarms of G is a measure of how likely it is that a group having at least k meaningful matches and a region probability p, was generated "by chance", as a realization of the background process. The lower is $NFA_g(G)$, the more unlikely G is generated by the background, and hence, the more meaningful is G.
While each meaningful group we detect will be relevant by itself, the whole set of meaningful groups will probably exhibit high redundancy in the sense that we will get many nested meaningful groups.

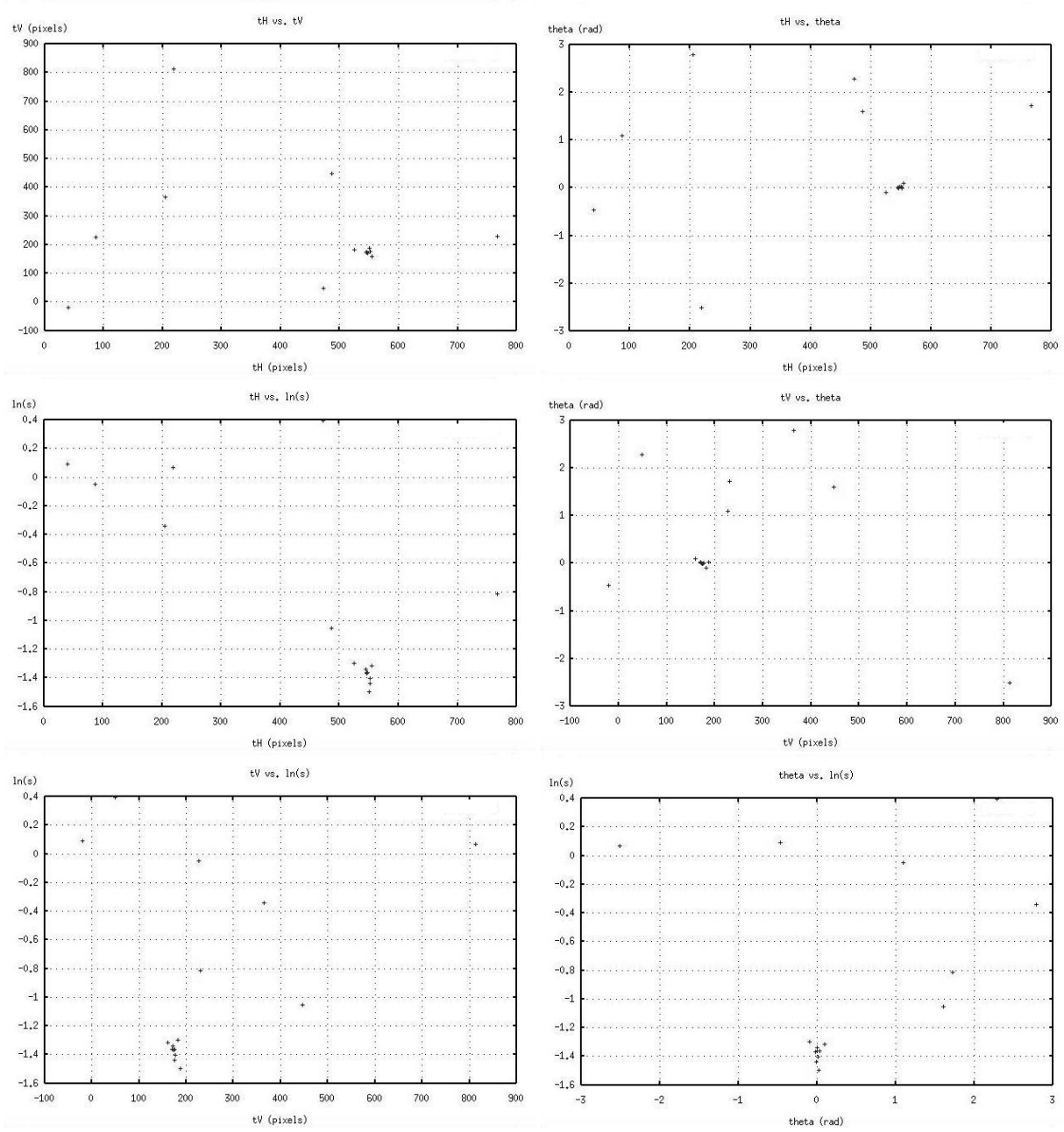


Figure 13.8: *"Uccello" experiment: transformation points of meaningful matches. While good matches define a cluster, the six false detections lead to non coherent transformations. Surrounding points of the cluster are more sparsely distributed, due to some instability in the extraction-normalization-encoding of pieces of level lines.*

Our strategy to reduce this redundancy combines the inclusion tree given by the hierarchical clustering procedure, and the measure of meaningfulness given by NFA_g. At each step of the hierarchical clustering procedure, two clusters are merged. Nevertheless, this merging is not necessarily a better data representation than the two separate clusters. By using the complete dendrogram $\mathcal{D}$ of $2M-1$ clusters, we decide *a posteriori* whether pairs of clusters should be merged or not. Let us denote by G, G_1 and G_2 the groups of matches corresponding respectively to a node and its two children nodes in $\mathcal{D}$. Roughly speaking, we will accept merging if, under the *a contrario* model, the expected number of groups like G we would observe is smaller than the one of observing groups like G_1, G_2, or the pair G_1 and G_2. Hence, we will say that G is valid if it verifies this criterion. Before giving the definition of a valid group, let us define $NFA_{gg}(G_1, G_2)$, the number of false alarms of the pair (G_1, G_2) as:

$$NFA_{gg}(G_1, G_2) = \frac{\#\mathcal{R}\,(\#\mathcal{R}-1)}{2} \sum_{i=k_1}^{M} \sum_{j=k_2}^{M-i} \binom{M}{i,j} {p_1}^i {p_2}^j (1-p_1-p_2)^{M-i-j}, \tag{13.7}$$

where k_1 and k_2 are the number of elements in G_1 and G_2 (resp.), and p_1 and p_2 their associated region probabilities. $NFA_{gg}(G_1, G_2)$ is an estimate of the number of occurrences, under the *a contrario* model, of the event $\mathcal{E}$: "there are two non overlapping groups A and B, with region probabilities p_1 and p_2 (resp.), containing at least k_1 and k_2 matches (resp.) among M". Indeed, $\#\mathcal{R}\,(\#\mathcal{R}-1)\,/2$ is the number of pairs of clusters of $\mathcal{R}$, and the probability of event $\mathcal{E}$ is given by the joint tail of the trinomial probability distribution.
Let us continue recalling some elements of Chapter 12.

DEFINITION 13.3 (VALID GROUP) *Let G, G_1 and G_2 be the groups of matches corresponding respectively to a node and its two children nodes in $\mathcal{D}$. We say that G is a valid group if both following inequalities hold:*

$$NFA_g(G) < \min\left\{NFA_g(G_1), NFA_g(G_2)\right\}, \tag{13.8}$$

$$NFA_g(G) \leqslant NFA_{gg}(G_1, G_2). \tag{13.9}$$

PROPOSITION 13.2 *If G is a valid group, then $NFA_g(G) < \frac{1}{2} \cdot NFA_g(G_1) \cdot NFA_g(G_2)$.*

DEFINITION 13.4 (MAXIMAL ε-MEANINGFUL GROUP) *We say that a group of matches G is a maximal ε-meaningful group if and only if:*

1. $NFA_g(G) \leqslant \varepsilon$,
2. *G is valid,*
3. *for all valid descendant F, $NFA_g(F) > NFA_g(G)$,*
4. *for all valid ancestor F, $NFA_g(F) \geqslant NFA_g(G)$.*

The maximal meaningful groups of matches correspond to our notion of shape: groups of shape elements we can recognize in a pair of images.

In Figures 13.9 and 13.10 we display the maximal meaningful groups for the "Uccello" and "Casablanca" experiments. In the "Uccello" experiment, the algorithm detects one maximal meaningful group, with $NFA_g = 2.7\ 10^{-41}$. Comparing this number of false alarm with the NFA associated to the best match, which was $1.2\ 10^{-8}$, we confirm that grouping can greatly increase confidence in detections. For the "Casablanca" experiment, two maximal meaningful groups are detected. The merging criterion (eq. (13.8)) decides that two separate groups (the actors' faces on one hand and the word "Casablanca" on the other hand) are a better representation than a single big group containing both groups. Indeed, while the big group in Figure 13.11 has the lowest NFA_g ($7.8\ 10^{-17}$), it is not a valid cluster since its NFA_g is not as small as half the product of its two children's NFA_g ($3.6\ 10^{-14}$ and $9.8\ 10^{-11}$).

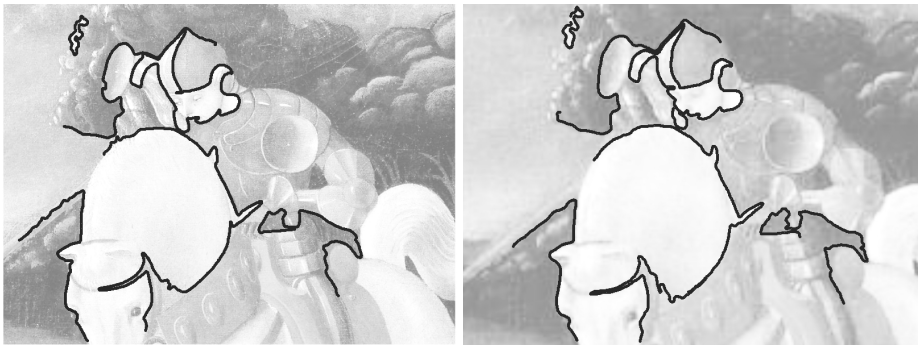
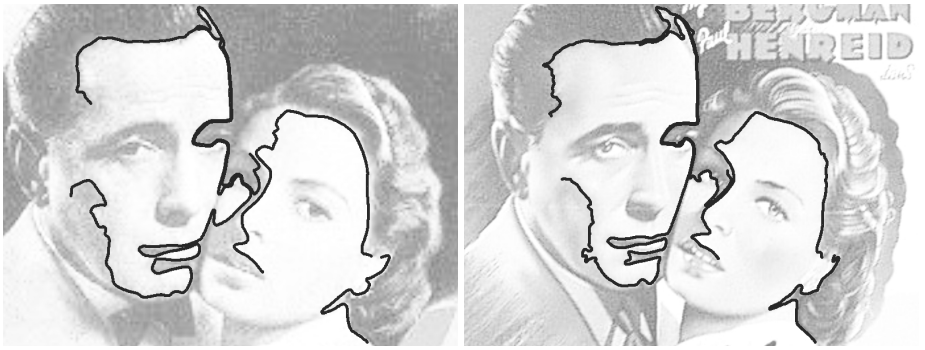


Figure 13.9: *"Uccello" experiment: a single maximal meaningful group was detected. Zoom on the matches of the group for the target image (left) and the scene image (right). The group is composed by all nine good matches, and its* NFA_g *is* $2.7\ 10^{-41}$.



4 meaningful matches, $NFA_g = 3.6\ 10^{-14}$

3 meaningful matches, $NFA_g = 9.8\ 10^{-11}$

Figure 13.10: *"Casablanca" experiment: maximal meaningful groups. Zoom on the matches of the group for the target image (left) and the scene image (right). The merging condition leads to the detection of the two maximal meaningful groups, instead of the group defined by the merging of these groups.*

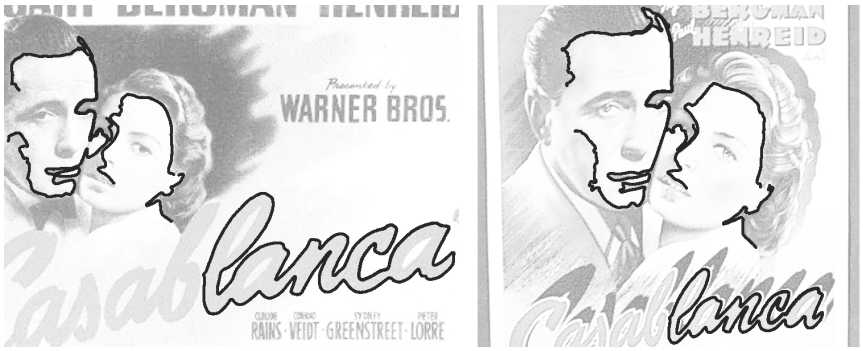


Figure 13.11: *"Casablanca" experiment. Meaningful group corresponding to the merging of groups in Figure 13.10. This group contains 7 meaningful matches, and its NFA_g is $7.8\ 10^{-17}$. According to Definition 13.3, it is not a valid cluster in terms of maximality.*

13.4.2 Experiments

The detection framework we presented in former sections is completely general and can be applied to any kind of images, provided objects are well described by planar shapes and transformations are close to similarities (or affinities). Besides the "Uccello" and "Casablanca" experiments, in this section we show some examples of different kind and nature. All experiments were done using the single-linkage algorithm (see Chapter 12, Section 12.2.2). Using other hierarchical clustering procedures yields essentially the same results. The detection threshold ε was fixed to 1 in all experiments, meaning we allow, at most, one false group detection on the average. Hence, the detection method does not need any parameter tuning, what makes it a parameter free method.

Object in clutter

Figure 13.12 shows the target and scene images. There are five instances of the target object in the scene, at different scales and locations. Two of them are strongly occluded. In Figure 13.13 we display the meaningful matches. We do not observe any false alarm and all objects are represented by matched shape elements. For one of the occluded objects we get a single meaningful match. Consequently, this object is not detected as a meaningful group (Figure 13.14).

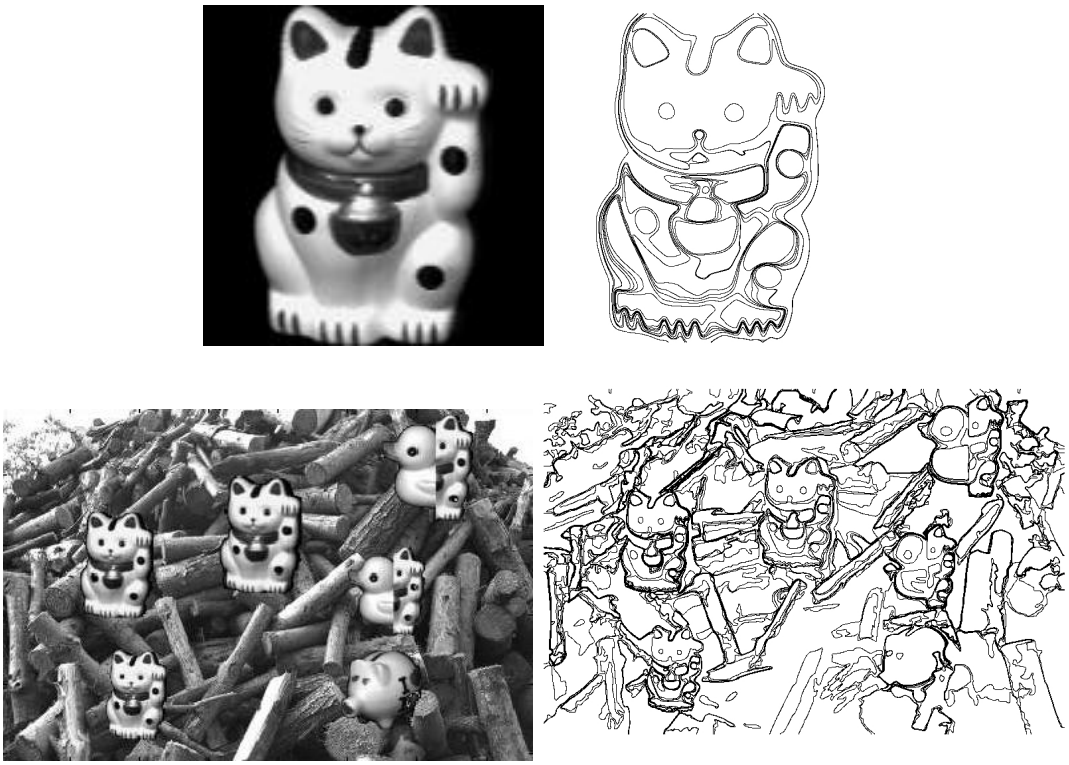


Figure 13.12: *"Object in clutter" experiment: Original images (from Columbia Object Image Library) and maximal meaningful level lines. Top: target image, bottom: scene image. Five instances of the target object are present in the scene, at different scales. Two of them are strongly occluded.*

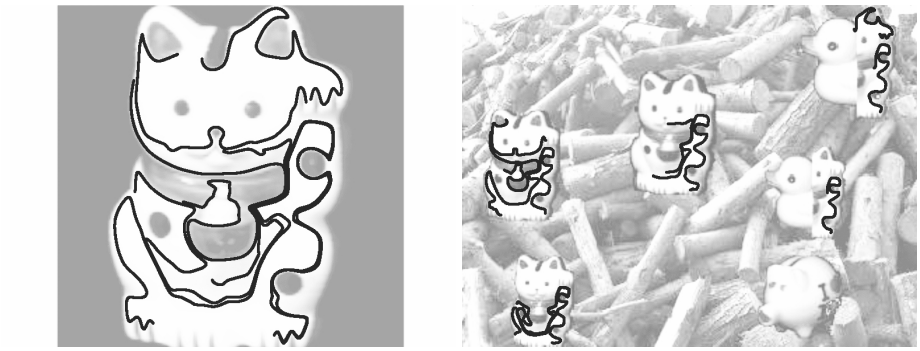
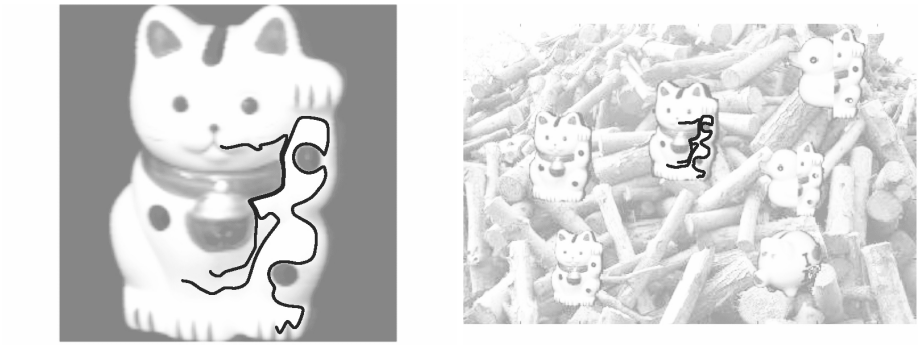
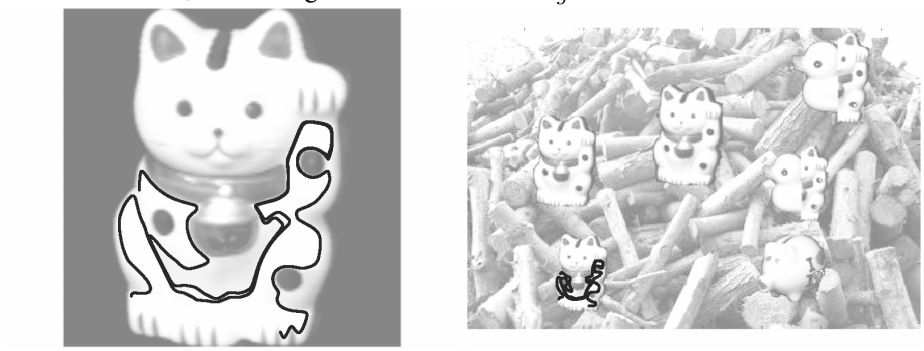


Figure 13.13: *"Object in clutter" experiment: meaningful matches. Number of tests:* $4.3\ 10^6$ *(*510 *codes in the target image,* $8,565$ *in the scene image). There are* 19 *meaningful matches, no false detection. The best match has* $NFA = 1.1\ 10^{-9}$*, and* 15 *matches have their* NFA *below* 0.1.

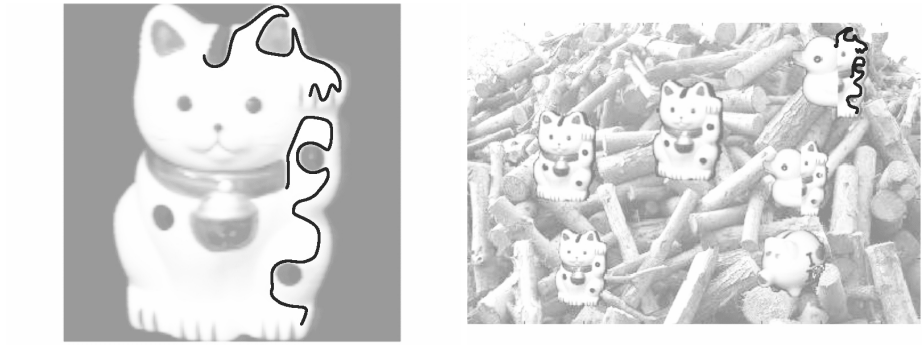


3 meaningful matches, $NFA_g = 7.7\ 10^{-13}$

5 meaningful matches, $NFA_g = 7.8\ 10^{-23}$



3 meaningful matches, $NFA_g = 2.3\ 10^{-9}$



2 meaningful matches, $NFA_g = 1.6\ 10^{-6}$

Figure 13.14: *"Object in clutter" experiment: the four maximal meaningful groups. One occluded object is missed. This object was represented by a unique meaningful match, and that was not enough to be detected as a meaningful group. The other four instances were detected with low NFA_g.*

Coca-Cola 1

In this example we look for the "Coca-Cola" logo in an advertisement (Figure 13.15). The word "Coca-Cola" is twice present in the advertisement. Notice there are also some strobe effects, since there are common parts between the level lines surrounding characters "oca" and "ola". The group of characters on the bottom part of the advertisement is very close to a similarity transformed version of the characters in the target logo. This remark does not hold for the characters written on the bottle. In Figure 13.16 we show the meaningful matches and in Figure 13.17 we display the detected maximal meaningful groups (see images legends for details). Notice how the NFA_g of groups drops down with respect to the NFA of matches, in particular for the dominant group. Notice also that the false matches on the top of the bottle have been rejected.

Figure 13.15: *"Coca-Cola 1" experiment: original images and maximal meaningful level lines. Top: target image, bottom: scene image. The word "Coca-Cola" is written twice in the scene image. The one on the bottle is not a similarity transformed version of the target image.*

Figure 13.16: *Meaningful matches. Number of tests:* $3.6\ 10^6$ *(*558 *codes in the target image,* $6,409$ *in the scene image). There are* 22 *meaningful matches, two false detections with* $NFA = 0.21$ *and* $NFA = 0.67$*. The best match has* $NFA = 3.5\ 10^{-13}$*, and* 19 *matches have their* NFA *below* 0.1*.*

(a) *9 meaningful matches,* $NFA_g = 5.4\ 10^{-35}$

(b) *7 meaningful matches,* $NFA_g = 3.4\ 10^{-16}$

(c) *2 meaningful matches,* $NFA_g = 8.5\ 10^{-5}$

Figure 13.17: *Maximal meaningful groups. Three groups were found. (a) The most meaningful groups; matches are very coherent, since the transformation between the two images parts is almost a perfect symilarity. (b) This coherent group is not as meaningful as the group in (a), since it contains less matches, and matches are less coherent because the deformation is not really given by a similarity transformation. (c) A strobe effect: characters "oca" and "ola" have some common parts.*

Coca-Cola 2

This example illustrates the performance of the proposed methodology in detecting multiple groups in an image. Here we look for the Coca-Cola logo in a scene with some Coca-Cola cans (Figure 13.18). The detection of meaningful matches leads to the matches we display in Figure 13.19. In Figure 13.20 we display the location of the corresponding transformation points in the transformation space. Figure 13.21 shows the maximal meaningful groups. The black rectangles correspond to hits and the white ones are misses. One can quickly see that these misses are not related to the cluster detection by looking at Figure 13.22. Two additional groups were detected as maximal meaningful. One of them is a consequence of the strobe effect in the word "Coca-Cola" (Figure 13.23); the other one (Figure 13.24) corresponds to a casual detection revealing a conspicuous coincidence ($NFA_g = 1.3\ 10^{-2}$). Indeed, the two matched characters "a" are coherent in scale and rotation, and there is also enough coincidence in translation. In table 13.1 we show the NFA_g of the detected groups. We can see that all groups are highly detectable, except for group 1 where the deformation can not really be modeled as a similarity transformation of the original logo.

Group nb.	1	2	3	4	5	6
nb. of matches	2	7	3	3	7	6
NFA_g	$3.9\ 10^{-2}$	$1.6\ 10^{-22}$	$3.7\ 10^{-11}$	$6.6\ 10^{-8}$	$2.2\ 10^{-24}$	$5.0\ 10^{-21}$
Group nb.	7	8	9	10	11	12
nb. of matches	7	6	7	8	6	9
NFA_g	$7.6\ 10^{-17}$	$3.6\ 10^{-19}$	$8.5\ 10^{-24}$	$3.8\ 10^{-20}$	$3.7\ 10^{-20}$	$5.6\ 10^{-32}$
Group nb.	13	14	15	16	17	17bis
nb. of matches	9	10	6	8	6	2
NFA_g	$2.4\ 10^{-29}$	$1.4\ 10^{-28}$	$4.0\ 10^{-30}$	$2.5\ 10^{-32}$	$1.0\ 10^{-18}$	$1.7\ 10^{-5}$

Table 13.1: *"Coca-Cola 2" experiment: NFA_g for the maximal meaningful groups in Figure 13.21.*

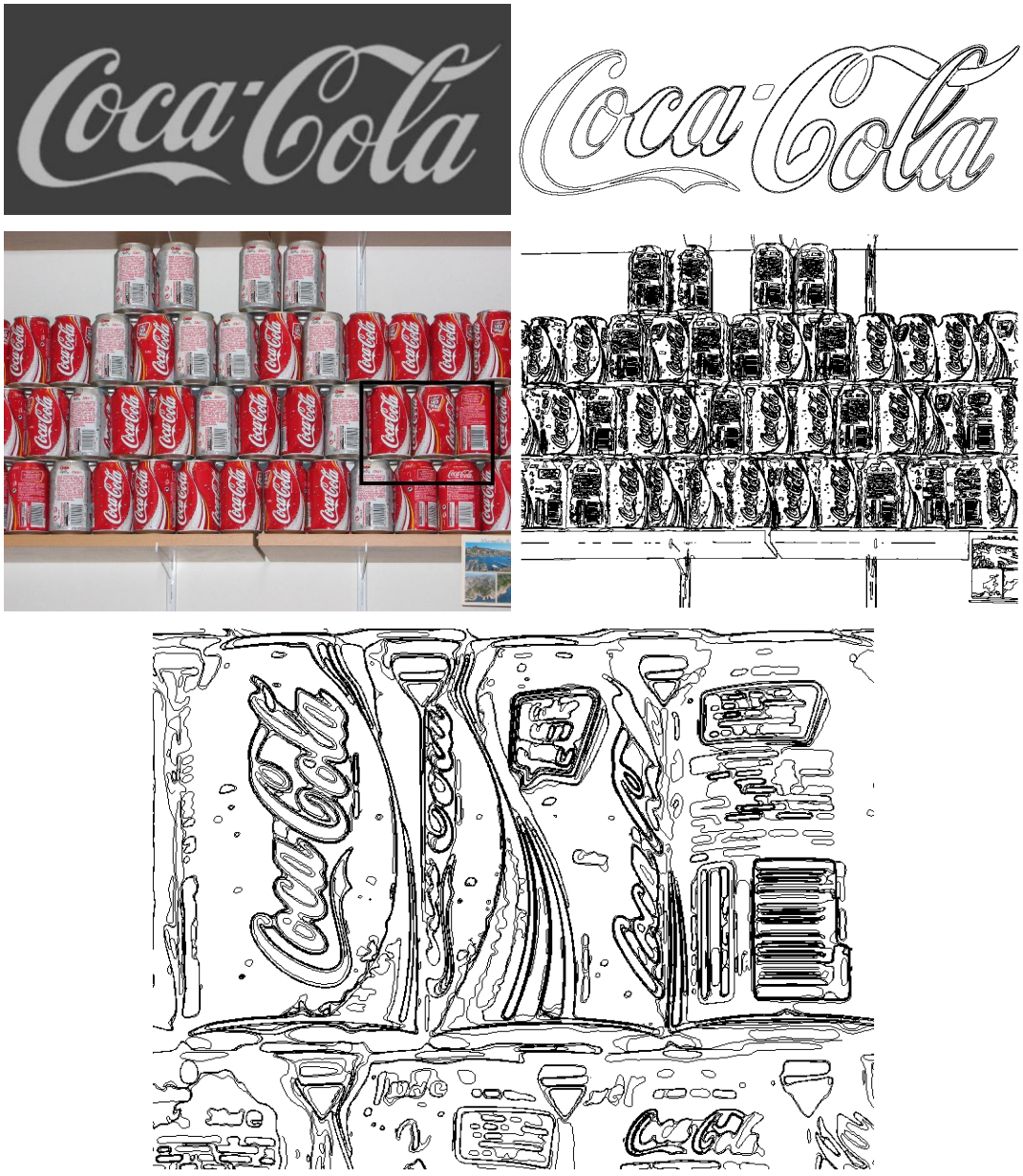


Figure 13.18: *"Coca-Cola 2" experiment: original images and maximal meaningful level lines. Top: target image, middle: scene image. Bottom: detail of the maximal meaningful level lines corresponding to the box drawn on the original image.*

Figure 13.19: *"Coca-Cola 2" experiment: meaningful matches. Number of tests:* $3.3\ 10^6$ *(*359 *codes in the target image,* $9,274$ *in the scene image). There are* 151 *meaningful matches, six of them are false detections with* NFA *of* $0.84, 0.78, 0.45, 0.60, 0.30$ *and* 0.21*. The best match has* $NFA = 4.6\ 10^{-7}$.

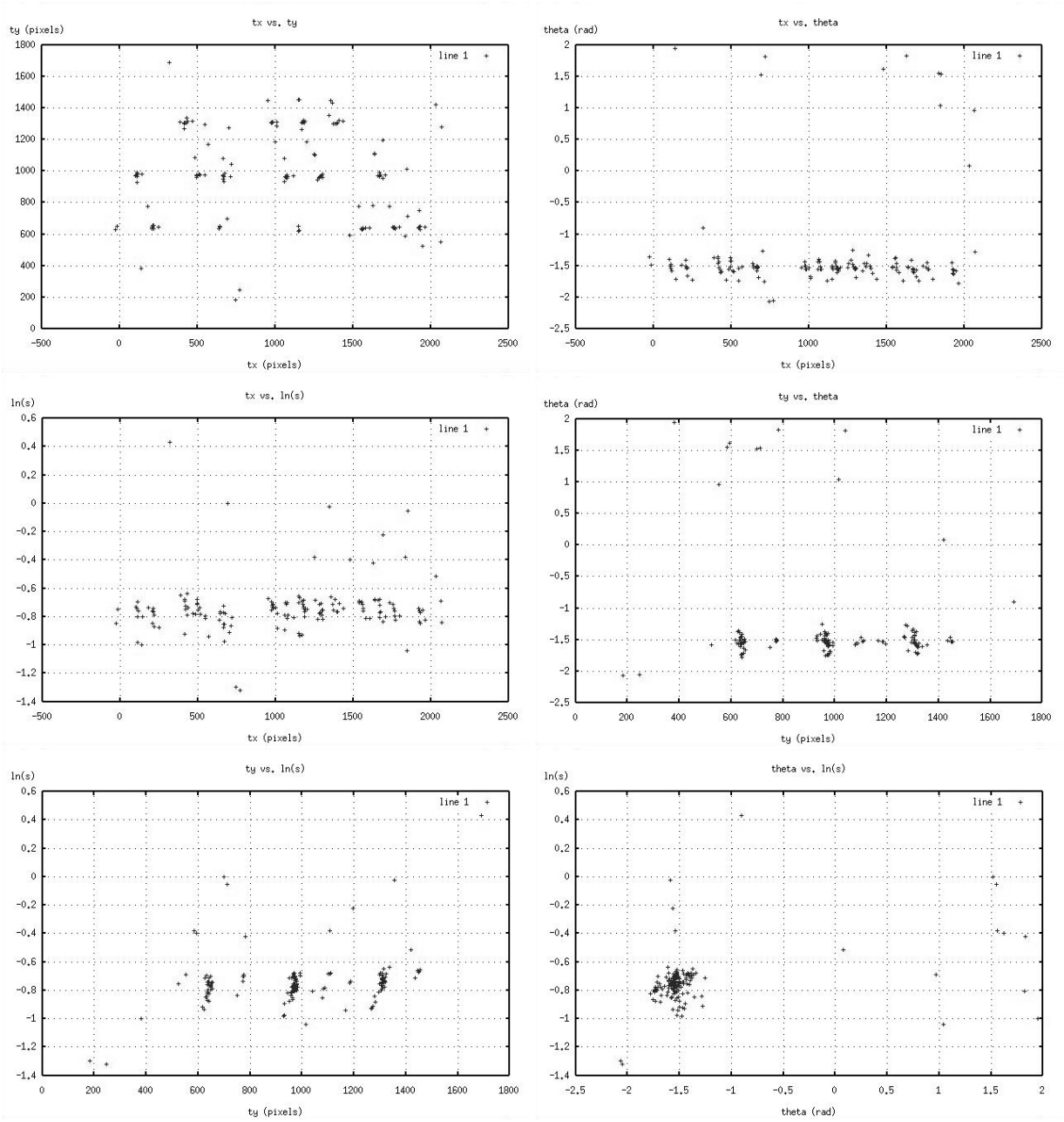
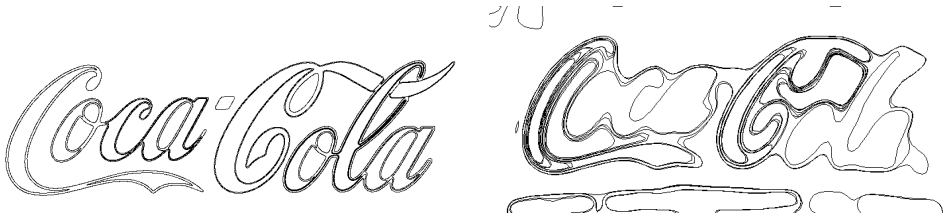


Figure 13.20: *"Coca-Cola 2" experiment: transformation points of meaningful matches. The image on the top left corresponds to the projection on the (t_x, t_y) plane, and one can see clusters of points on the cans where the logo is visible. On the right top we show projection on the (t_x, θ) plane; one can recognize the $90°$ orientation of the cans, and some outliers (false matches). In the image corresponding to the projection on the (t_y, θ) plane (middle, right) we can see three big clusters, one per row, and the $90°$ orientation of the cans. We can also recognize the three rows in the bottom left plot (projection on the $(t_y, \ln(s))$ plane, as well as the zoom factor ($\ln(s) \simeq -0.8$).*

Figure 13.21: *"Coca-Cola 2" experiment: maximal meaningful groups.* 19 *groups were detected as maximal meaningful groups. Among them,* 17 *correspond to groups in the black rectangles (see table 13.1 for their NFA_g). In rectangle nb. 17 there is also a group due to strobe effect (Figure 13.23). The other maximal group is a casual detection (Figure 13.24). The misses shown in white rectangles are explained in Figure 13.22.*



(a)

(b)

Figure 13.22: *"Coca-Cola 2" experiment: misses explanation. (a) The word "Coca-Cola" on the right is small and thus it was highly distorted by smoothing. No meaningful matches between shape elements in both images were found. One possible way to overcome this problem could consist in considering target representation at a few different scales. (b) The word "Coca-Cola" is distorted by projection on the can and perspective effect. Despite of that three meaningful matches were found, but they were not spatially coherent. Hence, no meaningful group was detected.*

Figure 13.23: *"Coca-Cola 2" experiment: strobe effect ($NFA_g = 1.7\ 10^{-5}$). Characters "oca" and "ola" share some common parts.*

Figure 13.24: *"Coca-Cola 2" experiment: casual detection ($NFA_g = 1.3\ 10^{-2}$). The image on the right has been rotated 90 degrees for a better understanding.*

Église de Valbonne

Figure 13.25 shows two different views of the *Église de Valbonne*, with their corresponding maximal meaningful level lines. The meaningful matches between these two views, up to similarity invariance, are shown in Figure 13.26. Some of them are false matches, but all of them showed a NFA greater than 0.1, as predicted by the experimental results in Chapter 8. There are also many casual matches that correspond to the same structures in the images (*e.g.* the pieces of rectangle in the fence in the left image in Figure 13.26 match with rightmost piece of curve in the right image). In Figure 13.27 we display the two maximal meaningful groups that are detected (see caption for details). Within each of these two groups, a global affine transformation was estimated by means of a least squares procedure, over the corresponding matched shape elements. These transformations were used to map the target image into the scene image. The superimpositon of the transformed target image and the scene image shows, in both cases, that the estimated transformation is a good approximation, in the neighborhood of the matched shape elements.
In Figure 13.28 we display the six 2-D projection of the transformation points associated to the meaningful matches. The red points correspond to the group in Figure 13.27(a), and the blue points to the one in Figure 13.27(b). The rest of the points, depicted in green, were not assigned to any maximal meaningful group. As we can see from these projections, finding "natural" clusters in this cloud of points is not a trivial problem. The method we propose in this work detects two (and only two) reasonable clusters. Detecting these two clusters as different entities by means of partitional methods, or without considering local merging criterion, does not seem to be possible. Indeed, the clusters are too much close, and their number of points are significantly different.

Figure 13.25: *Two frames of the* Église de Valbonne *sequence, with its corresponding meaningful level lines. The image on the left was considered as target.*

Figure 13.26: *Église de Valbonne:* 63 *meaningful matches were found, for* $15,151$ *codes in the target image and* $19,083$ *in the scene image. All false detection have* NFA *larger than* 0.1*, as expected. The best match has* $NFA = 2.8\ 10^{-12}$.

(a) 32 *meaningful matches,* $NFA_g = 1.8\ 10^{-111}$

(b) 2 *meaningful matches,* $NFA_g = 0.4$

Figure 13.27: *Église de Valbonne: two maximal meaningful groups were detected. All false matches and spatially uncoherent matches were rejected. The group in (a) corresponds to the principal group. The rightmost image shows the result of superimposing the two original images, according to a mean affine transformation. This transformation was estimated by a least-squares procedure over the matching shape elements within the group. This image shows that the global transformation between the common shapes is relatively well approximated by an affine transformation. The group in (b), which is barely detectable, corresponds to a match between far away trees. The rightmost image show that the estimated affine transform is a good approximation in the neighborhood of the matched shape elements.*

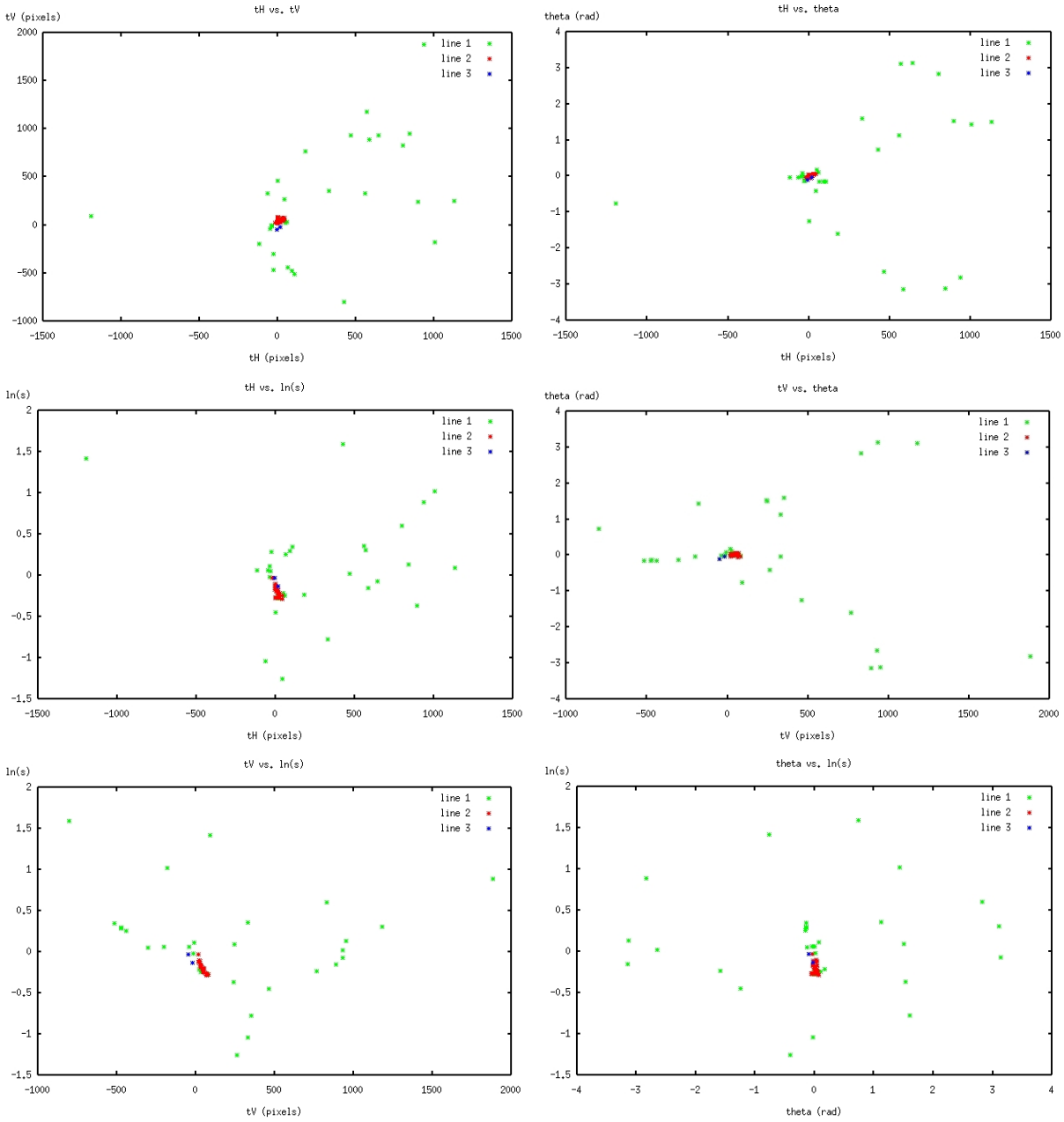


Figure 13.28: *Église de Valbonne: transformation points of meaningful matches (the six 2-D projections). The red points correspond to the group in Figure 13.27(a), and the blue points to the one in Figure 13.27(b). The rest of the points, depicted in green, were not assigned to any maximal meaningful group. Notice that the actual distance between transformations is not computed as the Euclidean distance between points in the transformation space, but using the dissimilarity measure in (13.3).*

13.5 Discussion on the definition of meaningful groups

Up to now we have considered that every transformation point T_k was equally relevant, as the family of random variables $\{T_k, 1 \leqslant k \leqslant M\}$ was assumed to be mutually independent, identically distributed. In this section we propose another definition of group meaningfulness, which associates a measure of confidence to each point, as much as we did by considering a saliency measure in Chapter 12 (section 12.5). Naturally, this measure of confidence will be related to the meaningfulness of the match that corresponds to the transformation point. Indeed, what we want to evaluate here is the probability that, just "by chance", several pairs of normalized shape elements are meaningful matches, and their respective transformation points fall into a transformation space region R. Let us make things more precise by defining the corresponding background model.

13.5.1 The background model

As we pointed out in Section 13.4.1, in our transformation clustering framework, the background model should describe the following situation:

- images contents are represented by encoded pieces of level lines (shape elements), along with sets of local similarity (or affine) frames, localized nearby meaningful level lines;
- no shape coincidence between the target image $\mathcal{I}$ and the scene image $\mathcal{I}'$ is observed.

Now we will define the background process by means of the following assumptions:

(A1) Parameters t_x, t_y, θ and s are random variables, θ and s are statistically independent, and the probability of the "region" $R = R_x \times R_y \times R_\theta \times R_s$

$$\begin{aligned} p(R) &:= \Pr\left((t_x, t_y, \theta, s) \in R\right) \\ &= \Pr(t_x \in R_x, t_y \in R_y | \theta \in R_\theta, s \in R_s) \Pr(\theta \in R_\theta) \Pr(s \in R_s) \end{aligned}$$

can be estimated from the pair of images $\mathcal{I}$ and $\mathcal{I}'$ (as was described in Section 13.4.1).

(A2) Points $T_j = (t_{xj}, t_{yj}, \theta_j, s_j)$, $j \in \{1, \ldots, M\}$, are mutually independent random variables, and for any hyper-rectangle R in the transformation space,

$$\begin{aligned} &\Pr\left(\mathcal{S}' \text{ s.t. } d(\mathcal{S}_k, \mathcal{S}') \leqslant d(\mathcal{S}_k, \mathcal{S}'_k),\ T_k \in R \mid NFA(\mathcal{S}_k, d(\mathcal{S}_k, \mathcal{S}')) \leqslant 1\right) \\ &\quad = NFA(\mathcal{S}_k, d(\mathcal{S}_k, \mathcal{S}'_k)) \times p(R). \end{aligned} \tag{13.10}$$

(here $\mathcal{S}_k$ and $\mathcal{S}'_k$ stand for shape elements of which pairing generates T_k).

Assumption (A1) is the same assumption (A1) we made for the background model in Section 13.4.1. The situation is a bit different for assumption (A2). The first difference is that here we introduce conditioning to take into account the fact that points T_k correspond to meaningful matches. We are also implicitly assuming, as we did for the former background model, that the meaningfulness of

a match and the location of its transformation point are independent under the *a contrario* model. Hence, because of this independence assumption, we have:

$$\begin{aligned}
&\Pr\left(\mathcal{S}' \text{ s.t. } d(\mathcal{S}_k,\mathcal{S}') \leqslant d(\mathcal{S}_k,\mathcal{S}'_k),\ T_k \in R \mid NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant 1\right)\\
&\quad = \Pr\left(\mathcal{S}' \text{ s.t. } d(\mathcal{S}_k,\mathcal{S}') \leqslant d(\mathcal{S}_k,\mathcal{S}'_k) \mid NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant 1\right)\\
&\qquad \times \Pr\left(T_k \in R \mid NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant 1\right)\\
&\quad = \Pr\left(\mathcal{S}' \text{ s.t. } d(\mathcal{S}_k,\mathcal{S}') \leqslant d(\mathcal{S}_k,\mathcal{S}'_k) \mid NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant 1\right) p(R)
\end{aligned}$$

Now, recall from Chapter 8 that:

$$NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant 1 \ \Leftrightarrow\ d(\mathcal{S}_k,\mathcal{S}') \leqslant \delta^*, \text{ where } \delta^* := \sup\left\{\delta > 0 \ :\ \text{PFA}(\mathcal{S}_k,\delta) \leqslant 1/(N\cdot N')\right\}.$$

Then, since we already know $d(\mathcal{S}_k,\mathcal{S}') \leqslant \delta^*$ because $(\mathcal{S}_k,\mathcal{S}')$ is a meaningful match, conditioning by $NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant 1$ just means:

$$\begin{aligned}
&\Pr\left(\mathcal{S}' \text{ s.t. } d(\mathcal{S}_k,\mathcal{S}') \leqslant d(\mathcal{S}_k,\mathcal{S}'_k) \mid NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant 1\right)\\
&\quad = \frac{\text{PFA}(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}'_k))}{\text{PFA}(\mathcal{S}_k,\delta^*)} = \frac{NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}'_k))/(N\cdot N')}{1/(N\cdot N')}\\
&\quad = NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}'_k)),
\end{aligned}$$

what finally yields (13.10). (For the second to last equality we consider $\text{PFA}(\mathcal{S}_k,\delta^*) = 1/(N\cdot N')$, by assuming $N\times N'$ is large enough to have an almost continuum evaluation of $\delta \mapsto \text{PFA}(\mathcal{S}_k,\delta)$.)
Notice that if instead of considering 1-meaningful matches, we had considered η-meaningful matches, conditioning would have been done with respect to $NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant \eta$, leading to

$$\Pr\left(\mathcal{S}' \text{ s.t. } d(\mathcal{S}_k,\mathcal{S}') \leqslant d(\mathcal{S}_k,\mathcal{S}'_k) \mid NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}')) \leqslant \eta\right) = \frac{NFA(\mathcal{S}_k, d(\mathcal{S}_k,\mathcal{S}'_k))}{\eta}.$$

13.5.2 Meaningful groups taking into account the meaningfulness of matches

For the sake of clarity, let us summarize, for the particular case of transformation clustering based on meaningful matches, some definitions and results that were presented in Chapter 12, for clustering of any kind of data and for any saliency measure (Section 12.5).
Let us denote by χ_i, $1 \leqslant i \leqslant M$, the indicator function of event $\mathcal{E}_i$: "$(\mathcal{S}_i,\mathcal{S}'_i)$ is an η_i-meaningful match, and T_i falls in R" (we already know $(\mathcal{S}_i,\mathcal{S}'_i)$, $1 \leqslant i \leqslant M$, are all meaningful matches).

DEFINITION 13.5 (ε-MEANINGFUL GROUP) *Let R be a transformation space hyper-rectangle in $\mathcal{R}$, containing a group of k among M transformation points associated to meaningful matches. We say the group is ε-meaningful if*

$$k \geqslant k^*(R,\eta_1,\ldots,\eta_M) := \min\left\{n \in \mathbb{N} \ :\ \Pr\left(\sum_{i=1}^{M}\chi_i \geqslant n\right) \leqslant \frac{\varepsilon}{\#\mathcal{R}}\right\}.$$

PROPOSITION 13.3 *The expected number of ε-meaningful groups in $\mathcal{R}$ is less than ε.*

Proposition 13.4 (Sufficient condition of ε-meaningfulness) *Let R be a transformation space hyper-rectangle from $\mathcal{R}$, containing a group of k among M transformation points associated to meaningful matches. We say the group is ε-meaningful if*

$$k \geqslant p(R) \sum_{i=1}^{M} \eta_i + \sqrt{\frac{M\left(\ln(\#\mathcal{R}) - \ln \varepsilon\right)}{h\left(p(R)\sum_{i=1}^{M}\eta_i/M\right)}}, \tag{13.11}$$

where

$$h(\mu) = \begin{cases} \frac{1}{1-2\mu}\ln\left(\frac{1-\mu}{\mu}\right) & \text{if } 0 < \mu < \frac{1}{2} \\ \frac{1}{2\mu(1-\mu)} & \text{if } \frac{1}{2} \leqslant \mu < 1. \end{cases}$$

It can be shown that the right hand side of (13.11) is an increasing function of $\sum_{i=1}^{M}\eta_i$. Applying Hoeffding inequality to the former case (where the meaningfulness of matches was not taken into account), comes to take $\eta_i = 1$ for all $1 \leqslant i \leqslant M$, that is $\sum_{i=1}^{M}\eta_i = M$. Hence, when we take into account the meaningfulness of matches, the minimal number of points a group must have in order to be meaningful decreases.

Definition 13.6 (Number of False Alarms) *Given a group G of k meaningful matches among M, we call number of false alarms of G the number*

$$NFA_g(G) = \#\mathcal{R} \times \left(\frac{p(R)\sum_{i=1}^{M}\eta_i}{k}\right)^k \left(\frac{M - p(R)\sum_{i=1}^{M}\eta_i}{M-k}\right)^{M-k},$$

where $p(R)$ is the probability that a transformation point falls in R, the smallest hyper-rectangle in $\mathcal{R}$ containing all k transformation points, and $\eta_1, \ldots, \eta_M$ are the meaningfulness of the M meaningful matches.

Proposition 13.5 *If $NFA_g(G) \leqslant \varepsilon$, then G is an ε-meaningful group.*

The proof is straightforward from Hoeffding inequalities (see Chapter 12, Section 12.5.2, Lemma 12.2). This sufficient condition is finer than (13.11), and will be used, in practice, to detect the meaningful groups when the meaningfulness of matches is taken into account.

13.5.3 Experiments

In this section we present some tables comparing the NFA_g of maximal meaningful groups obtained using the two proposed definitions, for the experiments we have shown up to here. Model A corresponds to the case were the meaningfulness of matches is not taken into account (Section 13.4.1). Model B uses the information given by the meaningfulness of matches and corresponds to the model we have just presented in Section 13.5.2.
The same maximal meaningful groups were found for both methods (in general, this is not necessary the case). Some relevant NFA_g:

- In "Casablanca", $5.9\ 10^{-20}$ instead of $7.8\ 10^{-17}$ for the big group in Figure 13.11.
- In "Coca-Cola 2", the casual match in Figure 13.24 has a NFA_g of $1.5\ 10^{-3}$ instead of $1.3\ 10^{-2}$.

nb. of matches	9
NFA_g Model A	$2.7\ 10^{-41}$
NFA_g Model B	$2.8\ 10^{-46}$

Table 13.2: *"Uccello" experiment: NFA_g for the maximal meaningful groups resulting from the two proposed definitions.*

nb. of matches	4	3
NFA_g Model A	$3.6\ 10^{-14}$	$9.8\ 10^{-11}$
NFA_g Model B	$3.0\ 10^{-15}$	$1.1\ 10^{-11}$

Table 13.3: *"Casablanca" experiment: NFA_g for the maximal meaningful groups resulting from the two proposed definitions.*

nb. of matches	3	5	3	2
NFA_g Model A	$7.7\ 10^{-13}$	$7.8\ 10^{-23}$	$2.3\ 10^{-9}$	$1.6\ 10^{-6}$
NFA_g Model B	$2.8\ 10^{-15}$	$1.7\ 10^{-26}$	$2.4\ 10^{-11}$	$7.7\ 10^{-8}$

Table 13.4: *"Object in clutter" experiment: NFA_g for the maximal meaningful groups resulting from the two proposed definitions.*

These experiments confirm that introducing the meaningfulness of matches makes good group detections more sure, by diminishing its number of false alarms. But, concerning this approach, maximality issues are not completely solved. Indeed, the NFA_{gg} of a pair of groups used for the validity criterion, defined in (13.7), only holds when patterns T_i are independent, identically distributed, so new definitions have to be explored. We have not addressed this problem yet. A first attempt, consisting in replacing the validity criterion by the necessary condition of cluster validity, given by Proposition 13.2 (Section 13.4.1), was unsuccessful in giving the "good" maximal groups. Indeed, this necessary condition may be too strong, specially when the clusters to be merged concentrate the majority of the points.

13.6 Related work

The use of spatial coherence for shape or object detection has been the subject of intensive research, in particular since Ballard's work on the generalized Hough transform [Bal81]. In his paper, Ballard proposed a method extending the Hough transform to any kind of planar shape, not necessarily described by an analytic formula. Stockman [SKB82] presented another early work based on the same

nb. of matches	2	7	9
NFA_g Model A	$8.5\ 10^{-5}$	$3.4\ 10^{-16}$	$5.7\ 10^{-35}$
NFA_g Model B	$6.5\ 10^{-6}$	$3.1\ 10^{-21}$	$3.6\ 10^{-41}$

Table 13.5: *"Coca-Cola 1" experiment: NFA_g for the maximal meaningful groups resulting from the two proposed definitions.*

Group nb.	1	2	3	4	5	6
nb. of matches	2	7	3	3	7	6
NFA_g model A	$3.9\ 10^{-2}$	$1.6\ 10^{-22}$	$3.7\ 10^{-11}$	$6.6\ 10^{-8}$	$2.2\ 10^{-24}$	$5.0\ 10^{-21}$
NFA_g model B	$4.6\ 10^{-3}$	$6.4\ 10^{-27}$	$9.4\ 10^{-13}$	$1.7\ 10^{-9}$	$8.7\ 10^{-29}$	$1.0\ 10^{-24}$
Group nb.	7	8	9	10	11	12
nb. of matches	7	6(7B)	7	8	6(7B)	9
NFA_g model A	$7.6\ 10^{-17}$	$3.6\ 10^{-19}$	$8.5\ 10^{-24}$	$3.8\ 10^{-20}$	$3.7\ 10^{-20}$	$5.6\ 10^{-32}$
NFA_g model B	$3.0\ 10^{-21}$	$7.1\ 10^{-23}$	$3.4\ 10^{-28}$	$2.9\ 10^{-25}$	$1.8\ 10^{-24}$	$8.0\ 10^{-38}$
Group nb.	13	14	15	16	17	17bis
nb. of matches	9	10	6	8	6	2
NFA_g model A	$2.4\ 10^{-29}$	$1.4\ 10^{-28}$	$4.0\ 10^{-30}$	$2.5\ 10^{-32}$	$1.0\ 10^{-18}$	$1.7\ 10^{-5}$
NFA_g model B	$3.4\ 10^{-35}$	$6.1\ 10^{-34}$	$8.2\ 10^{-34}$	$1.9\ 10^{-37}$	$2.0\ 10^{-22}$	$2.0\ 10^{-6}$

Table 13.6: *"Coca-Cola 2" experiment: NFA_g for the maximal meaningful groups resulting from the two proposed definitions.*

principle (recognize a target shape by finding clusters in the transformation space), where he introduced a coarse to fine technique allowing to reduce the search complexity. Other voting schemes, like Geometric Hashing [WR97, LW88] or the Alignment method [HU87], are frequently used in detection or recognition problems.

In [GH90, GH91] Grimson and Huttenlocher present a study on the likelihood of false peaks in the Hough parameter space. Their work is particularly interesting from the detection viewpoint we adopt in this chapter, since they also propose a detection framework where recognition thresholds are derived from a null model. Using occupancy models (Maxwell-Boltzmann or Bose-Einstein models), they characterize the probability that several transformation points will fall in a transformation space region at random (actually, they do not consider transformation points but transformation volumes, that take into account the uncertainty involved in the feature extraction procedure; here we omit these details for the sake of clarity). Then, they use this probability to fix a threshold on the minimum fraction of target features that must be matched in order to consider that this match was not generated by "*the conspiracy of random*". Previous recognition methods generally associated a single threshold with each target image, independent of the scene complexity, leading to worse performance when scenes were more complex. Contrarily to this methods, the derived threshold satisfies an important property: it is a function of the scene complexity and of the uncertainty in feature extraction.

The method we propose in this chapter shares these fundamental ideas with Grimson and Huttenlocher's work, but the approach is quite different since it is based on a hierarchical representation of the transformation points, and uses a data-dependent null model (Grimson and Huttenlocher's method assumes features are uniformly distributed in the image, what can be a non relevant assumption in many situations; see [Pen98]).

13.7 Conclusion

In the first part of this thesis we have addressed the correspondence problem between shape elements and we defined the notion of meaningful matches. Then, in this chapter, we proposed a method to define shapes as groups of spatially coherent meaningful matches. Hence, our computational approach to recognition is based on a recursive grouping strategy: similar shape elements are defined as those shape elements having similar subparts (the $5 + 1$ pieces considered in Chapter 8), and corresponding shapes are defined as shapes sharing spatially coherent shape elements. This strategy is sound from the perceptual organization point of view, since it is based on two gestaltist principles of grouping [Wer23]: *similarity* (the meaningful matches) and *familiar configuration* (the meaningful groups).

The spatial coherence of meaningful matches was detected, indirectly, by applying the clustering method proposed in Chapter 12, to the transformation points associated to meaningful matches. This method is parameterless, and is also in keeping with the general *a contrario* detection methodology adopted in this thesis. Clusters of transformation points, corresponding to groups of matches, were

detected *a contrario* to a data-influenced null model, as large deviations from this model.
By using the spatial coherence of meaningful matches, the proposed method enhances the confidence on the detected structures, as shown by the low values of NFA_g reached in the experiments in Section 13.5.3. These experiments show also that the method performs well in very different situations, and can deal with several groups of matches at the same time. In all cases, the rejection of false or spatially uncoherent matches was successful. In the next chapter we will present several experiments, that confirm the usefulness of our shape detection / recognition method.

Chapter 14

Experimental results

Abstract: The grouping of spatially coherent meaningful matches has been extensively studied in the previous chapter, and several experiments were presented and discussed. In this chapter we illustrate the whole recognition process by presenting some more experiments over different kinds of images.

Résumé : Dans le chapitre précédent, nous avons étudié largement le groupement d'appariements significatifs, basé sur la cohérence spatiale, et nous avons présenté et discuté plusieurs expériences. Dans ce chapitre nous illustrons tout le processus de reconnaissance, en présentant d'autres expériences sur des images de differente nature.

14.1 The visualization of the results

Almost all the experiments we present in this chapter are illustrated with the following images:

1. *The two original images.*

2. *The smoothed maximal meaningful boundaries of the original images*, extracted using the algorithm described in Chapter 4, then smoothed with Moisan's implementation of the affine curve shortening equation (Chapter 5).

3. *Detection of meaningful matches between shape elements.* We consider here the 1-meaningful matches, despite the fact that a few of them may correspond to false detections; indeed, as we saw in Chapter 8, the constraints imposed by the encoding methods and by the non-intersection of level lines introduce a certain amount of dependence between the distances used as features in the *background model* (which were assumed to be independent). Thresholding the NFA at 0.1 ensures that no detection can occur in white noise images. However, since the detection of meaningful matches is followed by a grouping process based on spatial coherence, in the experiments we prefer to keep these few false matches in order to test the robustness of the grouping algorithm.

A fundamental hypothesis for the *a contrario* detection of groups is that, under the *background model*, transformation points are mutually independent. In order to comply with this hypothesis, a greedy algorithm that eliminates matched shape elements which share a large piece of curve with other shape elements presenting lower NFA. More precisely speaking, if a pair of shape elements $(\mathcal{S}_1, \mathcal{S}'_1)$ is an ε_1-meaningful match, and there exists another pair $(\mathcal{S}_2, \mathcal{S}'_2)$ matching ε_2-meaningfully, with $\varepsilon_2 < \varepsilon_1$, such that $\mathcal{S}_1$ shares at least half of its length with $\mathcal{S}_2$, and the same for $\mathcal{S}'_1$ and $\mathcal{S}'_2$, the pair $(\mathcal{S}_1, \mathcal{S}'_1)$ is eliminated from the output list of matches.

The detection of 1*-meaningful matches is illustrated by superimposing the matched shape elements to the original images.*

4. *Grouping of spatially coherent meaningful matches.* For each meaningful group of matches that is detected (the maximal 1-meaningful groups defined in Chapter 13), four images are shown:

   - *The shape elements that match within a group are shown, superimposed to the original images.*
   - Given the set of transformations corresponding to the matches within a group, the best affine transformation (in the least squares sense) that maps the shape elements in the target image to the ones in the scene image is computed. Then, the target image is mapped using this transformation. *The superimposition of the transformed target image and the scene image is presented.*
   - Gradient orientation comparison. The orientation of the gradient of both the scene image and the transformed target image are computed and compared, in order to show the quality of the registration of the target image into the scene image. Let us denote by $D_1(i,j)$ the gradient of the transformed target image at pixel (i,j), and by $D_2(i,j)$ the gradient of the scene image at (i,j). *The comparison of the gradient orientation of both images is illustrated by an image for which the pixel values may be* 0 *(black),* 128 *(grey) or* 255 *(white).* If $\|D_1(i,j)\| < 2$ or $\|D_2(i,j)\| < 2$, the orientation of the gradient is not considered to provide reliable information, and pixel (i,j) is painted in grey. If the gradient norms are greater than 2, then pixel (i,j) is painted in white if $|\text{angle}(D_1(i,j), D_2(i,j))| < 15$ degrees, and in black else. (We thank José Luis Lisani for communicating this algorithm).

In what follows, we only show experiments based on the semi-local encoding procedures.

14.2 Checking the consistency of grouping: two unrelated images

We have checked the consistency of the grouping algorithm by comparing some pairs of different, completely unrelated images. In section 9 we presented an experiment of this kind, to detect matches between the shape elements of two unrelated images (see Figure 9.19 in section 9.1.4). On these two

images, the grouping method did not led to the detection of any meaningful group. An interesting experiment would consist in considering large databases of images and finding meaningful groups of matches between pairs of images from these databases. Defining a systematic approach of this kind will maybe lead to detect "typical shapes" in images.

14.3 Subjective contours and contrast changes

Figure 14.1 shows two different versions of Saint Jerôme by Georges De la Tour. The two paintings represent Saint Jerôme in the same pose. There are many elements in common in these two painting, but many differences can be seen. The maximal meaningful boundaries extracted from the original images are also shown in Figure 14.1. Looking at these lines, we have a feeling that many shape information is left. However, most of the missing lines do not really exists: they are subjective contours introduced by the pictural technique (the use of Caravaggesque *chiaroscuro* lighting).

Figure 14.1: *Two versions of* Saint Jerôme *by Georges de la Tour. The image on the left was considered as target. The majority of the contours that are missed are actually subjective contours and do not exist.*

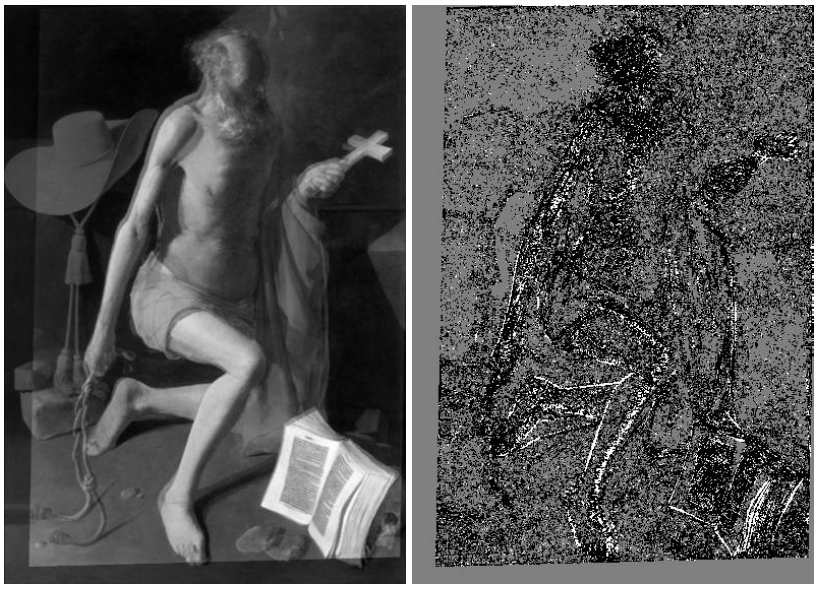
Our perception is able to assert that this two images represent almost the same scene, but this is certainly a consequence of perceptual grouping of different cues and not of the closeness of the level lines in both images, which are not that similar. This last issue is shown in Figure 14.2, where the 1-meaningful matches between shape elements detected with the similarity invariant version are displayed. The NFA of the best match is $6.0\,10^{-7}$, which is not as low as the ones obtained when comparing snapshots in Chapter 9. Moreover, some false matches can be seen (they all show NFAs larger than 0.1).

The detection of spatial coherent groups of meaningful matches led to a single maximal meaningful group, for which $NFA_g = 1.1\,10^{-15}$. The six matched shape elements within this group are displayed in Figure 14.3(a). Notice that all false matches have been rejected. In Figure 14.3(b) we show the superimposition of the "scene image" and the transformed "target image", as well as the gradient

Figure 14.2: *Saint Jerôme:* 1*-meaningful matches. The* NFA *of the best match is* $6.0\,10^{-7}$*. Some false detections can be observed; their* NFA *is above* 0.1*.*

orientation comparison image. Looking at the superimposed images, we can notice that many parts are accurately registrated, but many differences between the two paintings can be seen. This explains why the NFA_g of the group does not reach lower values. The gradient orientation comparison image illustrates the difficulty of the registration problem. The majority of the points show small contrast. Gradient orientation coincidence can be observed along some of the contours.

(a) *The six matched shape elements within the spatially coherent group. False matches have been rejected by the grouping procedure.*



(b) *Left: superimposition of the "scene image" and the transformed "target image". Right: image of gradient orientation comparison.*

Figure 14.3: *Saint Jerôme: the only detected maximal meaningful group. The group is composed by six matches, and its NFA_g is $1.1\,10^{-15}$. The superimposition of the "scene image" and the transformed "target image" reveals many slight differences between the two paintings. This explains why the NFA_g of the group does not reach lower values. Gradient orientation coincidence can be observed along some of the contours.*

14.4 Dealing with strong zooms

We call "Hitchcock 1" the experiment we present in this section. The original images and their corresponding maximal meaningful boundaries are show in Figure 14.4. We present two different examples. The first one, which we call "Hitchcock 1A", consists in considering as target image the top image in Figure 14.4, and as scene image the one on the bottom. In the second example ("Hitchcock 1B"), the role of the images is inverted: the bottom image is considered as target, and the top image as scene image.

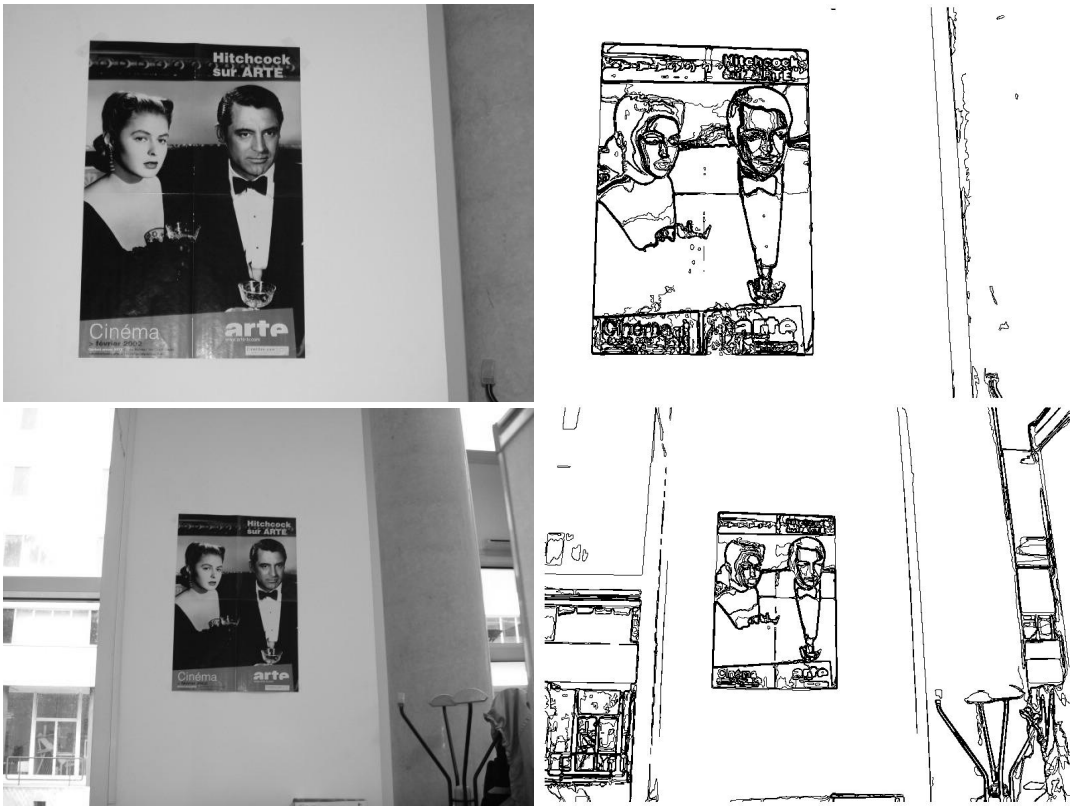


Figure 14.4: *Hitchcock 1 experiment: original images, and corresponding maximal meaningful boundaries.*

Hitchcock 1A

Figure 14.5 displays the 1-meaningful matches between shape elements. The lowest NFA is $7.0\,10^{-9}$, and two false matches are be observed (NFAs 0.25 and 0.36). The grouping algorithm (similarity invariance version) yields one maximal meaningful group containing 15 shape elements, for which $NFA_g = 3.2\,10^{-63}$. Figure 14.6(a) shows the matched shape elements that are within the group; false matches have been rejected by the grouping algorithm.

In Figure 14.6(b) we present the superimposition of the scene image and the transformed target image. The highlighted part corresponds to the region where the target image has been mapped. The estimated zoom factor was 0.56. The accuracy of the registration is illustrated by the gradient orientation comparison image.

Figure 14.5: *Hitchcock 1A experiment:* 1*-meaningful matches. Left: target image with matched shape elements. Right: scene image.*

(a) *The* 15 *matched shape elements within the spatially coherent group. False matches have been rejected by the grouping procedure.*

(b) *Left: superimposition of the "scene image" and the transformed "target image". Right: image of gradient orientation comparison.*

Figure 14.6: *Hitchcock 1A experiment: the only detected maximal meaningful group. The group contains* 15 *matches and its* NFA_g *is* $3.2\,10^{-63}$.

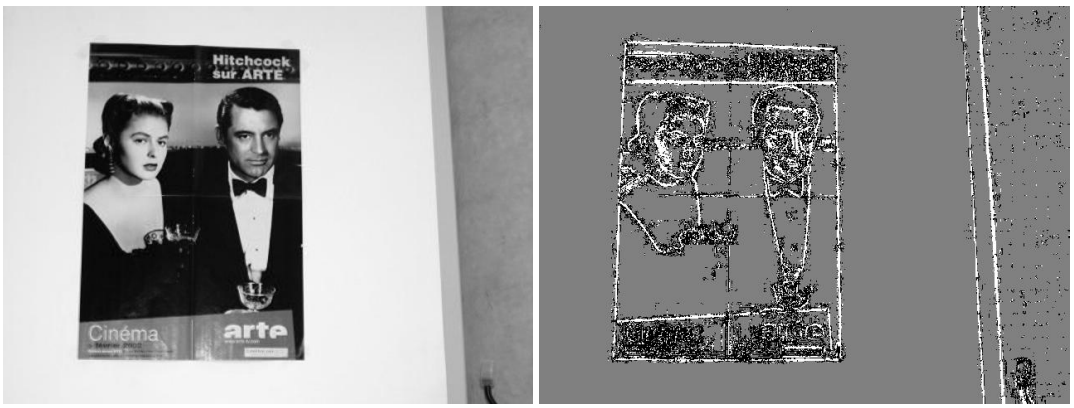
Hitchcock 1B

Figure 14.7 displays the 1-meaningful matches between shape elements, when we invert the "target" and the "scene" roles. The results are equivalent (three false detections showing NFAs larger than 0.1, the same order of magnitude for the good matches). The grouping algorithm (similarity invariance version) yields also one maximal meaningful group, containing 16 shape elements, and showing an $NFA_g = 1.1\,10^{-56}$. Figure 14.8(a) shows the matched shape elements that are within the group; false matches have been rejected by the grouping algorithm.

Figure 14.7: *Hitchcock 1B experiment:* 1*-meaningful matches. Left: target image with matched shape elements. Right: scene image.*

The superimposition of the transformed target image and the scene image, displayed in Figure 14.6(b) is very accurate: we cannot perceive any difference between this image and the original scene image. The estimated zoom factor was 1.78 (what is almost equal to $1/0.56$). The high accuracy of the registration is illustrated by the gradient orientation comparison image.

(a) *The* 16 *matched shape elements within the spatially coherent group. All false matches have been rejected.*



(b) *Left: superimposition of the "scene image" and the transformed "target image". Right: image of gradient orientation comparison.*

Figure 14.8: *Hitchcock 1B experiment: the only detected maximal meaningful group. The group contains* 16 *matches and its* NFA_g *is* $1.1\,10^{-56}$.

14.5 Dealing with occlusions

In this section we present two examples where the region of interest in the scene is occluded by the foreground.

Las Meninas by Velazquez

The pair of images considered for this experiment are shown in Figure 14.9, with their corresponding maximal meaningful boundaries. The image on the top is considered as target image, and its shape contents is sought in the bottom image. In the target image, we can see a portion of Velazquez masterpiece "Las Meninas", which is occluded by some people contemplating the painting at "El Prado" museum. The scene image is a reproduction of the original painting, taken from the World Wide Web. In this experiment, the similarity version of the recognition method was used.

Figure 14.9: *Las Meninas experiment. Top row: target image and its maximal meaningful boundaries. Bottom: scene image and maximal meaningful boundaries.*

Figure 14.10 shows the detected 1-meaningful matches between shape elements. The best match shows an NFA of $4.1\,10^{-14}$. Here again, the few false matches that were found have their NFA above 0.1.

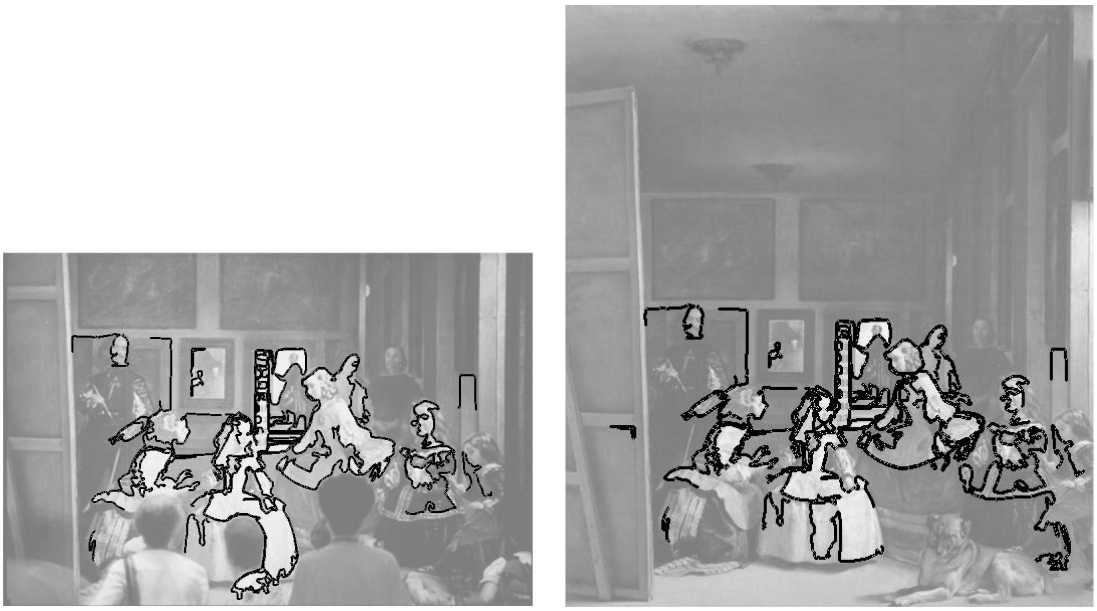


Figure 14.10: *Las Meninas experiment:* 1*-meaningful matches. The* NFA *of the best match is* $4.1\,10^{-14}$*. Some false detections can be observed; their* NFA *is above* 0.1.

A single maximal meaningful group was detected. This group contains 47 spatially coherent meaningful matches, and its NFA_g is $7.1\,10^{-155}$. In Figure 14.11 we shows the matched shape elements that are within the group; all false matches have been rejected by the grouping algorithm, except for the one displayed in Figure 14.12. While these matched shape elements are significantly different over a piece of curve (what is expressed by a large NFA: 0.96), if we look at the normalized shape element, we see that the pairs of points defining their local frames do certainly correspond by a transform that is close to the correct one. This explains why this pair of matched shape elements is not discarded in the grouping algorithm.

We end up with "Las Meninas" experiment by showing, as usual, the superimposition of the registrated target image and the scene image (Figure 14.13). The registration is very accurate.

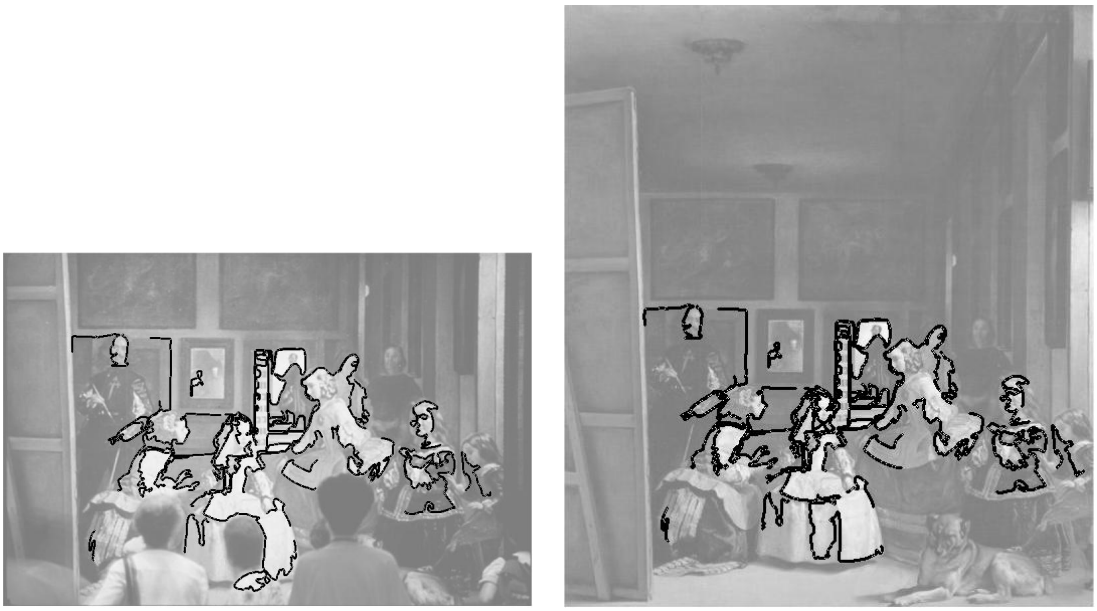


Figure 14.11: *The* 47 *matched shape elements within the spatially coherent group. All False matches have been rejected, except for the one in Figure 14.12. The NFA_g of the group is remarkably low:* $7.1\,10^{-155}$.

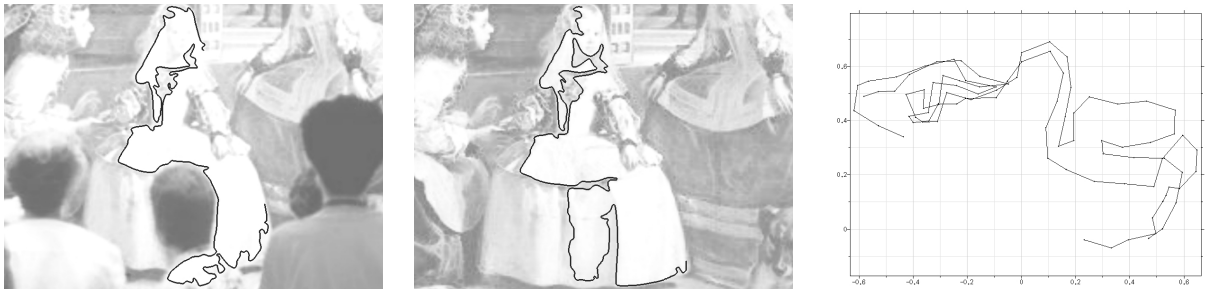


Figure 14.12: *The false match that was not rejected by the grouping algorithm. These matched shape elements are significantly different over a piece of curve (what is expressed by a large NFA:* 0.96*). However, the pairs of points defining their local frames do certainly correspond by a transform that is close to the correct one. This explains why this pair of matched shape elements is not discarded in the grouping algorithm.*

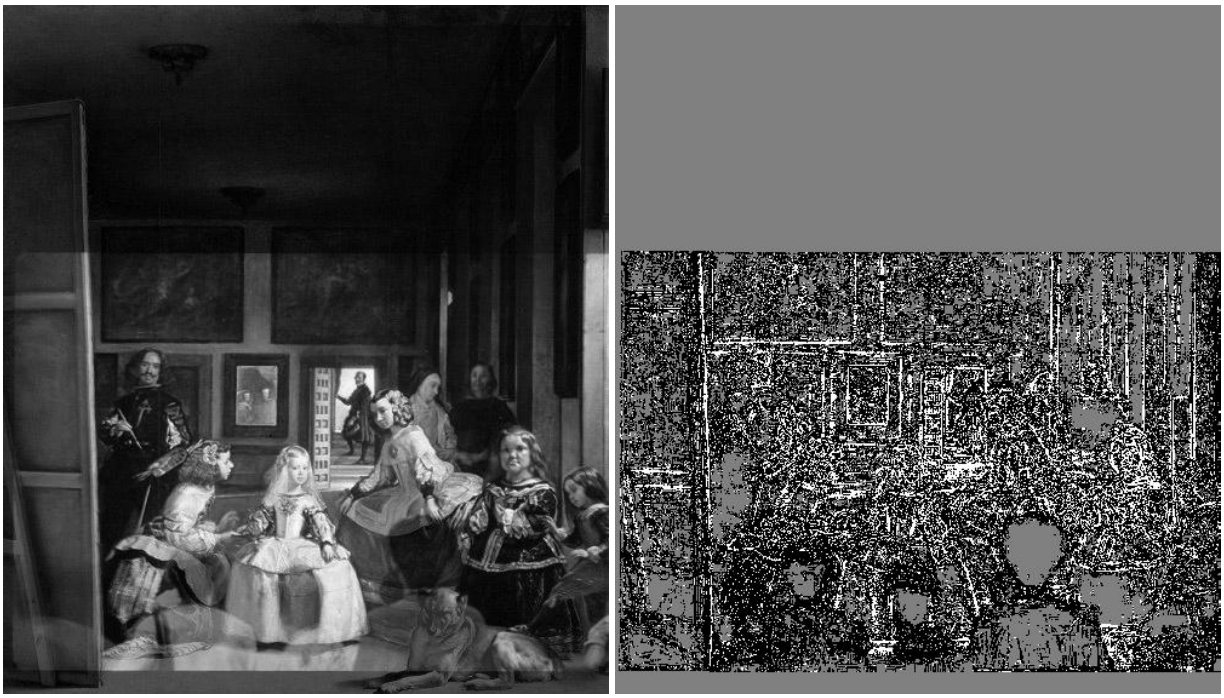


Figure 14.13: *"Las Meninas" experiment. Left: superimposition of the "scene image" and the transformed "target image". Right: image of gradient orientation comparison.*

Guernica by Picasso

We only present the final result for this experiment. The similarity invariant method was used. A single maximal meaningful group was detected, containing 36 matches between shape elements. The NFA_g of this group was also extremely low ($1.8\,10^{-154}$).

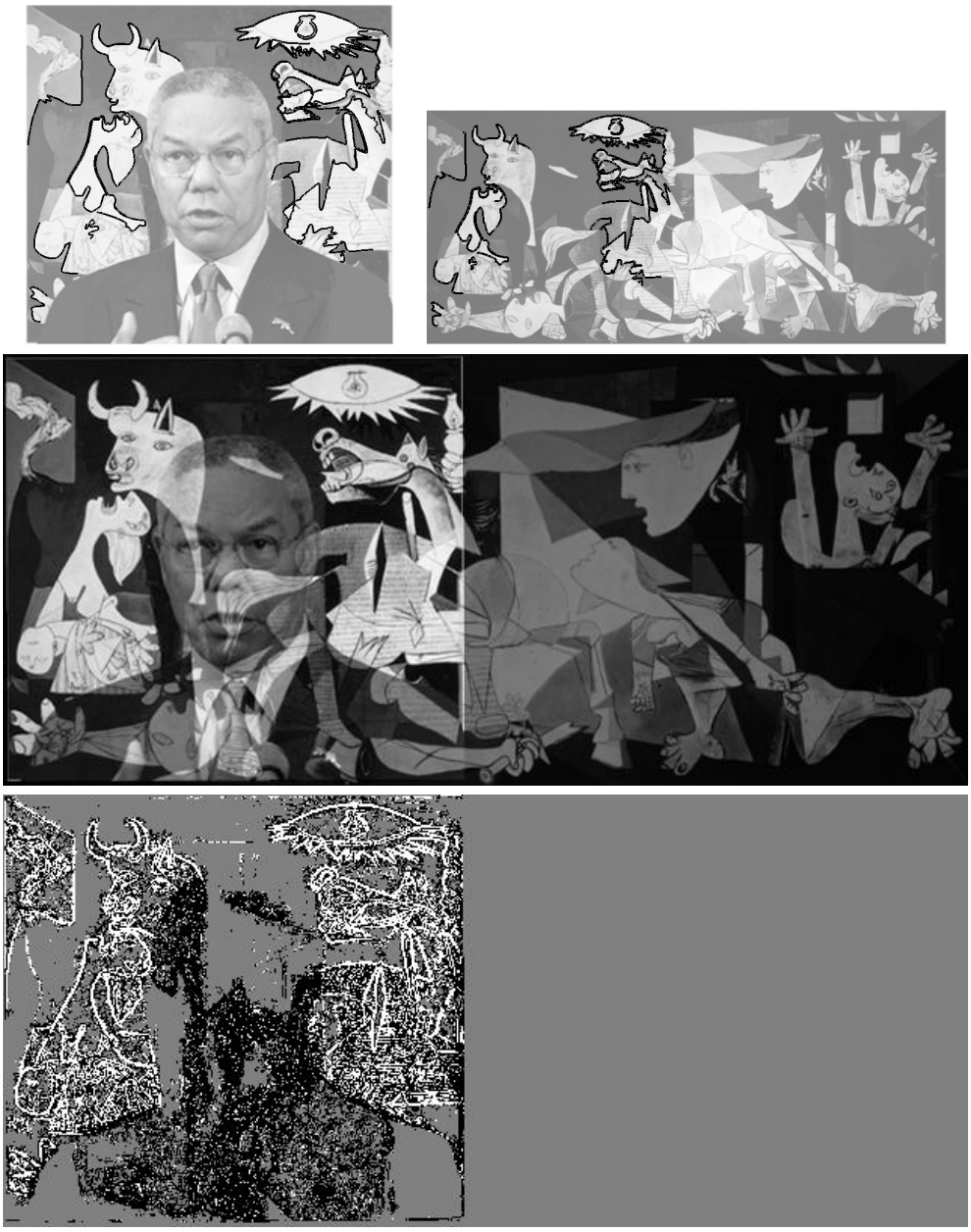


Figure 14.14: *Guernica experiment. Top: the* 36 *spatially coherent meaningful matches (the image on the left was considered as target image). Middle: the registration of the target image superimposed to the scene image. Bottom: image of gradient orientation comparison.*

14.6 Dealing with perspective distortions

Hitchcock 2

The experiment we present here has already been addressed in Chapter 9, section 9.1.2, but only the extraction, encoding and shape elements matching stages were discussed. It was shown that, as expected, the semi-local affine invariant matching method led to better results than the semi-local similarity invariant method (the meaningful matches reached lower NFAs). No false match between shape element was detected for $NFA < 1$; 16 meaningful matches were found, the lowest NFA being $6.5\,10^{-11}$. Figure 14.15 shows the original image and the meaningful matches between shape elements.

Figure 14.15: *Hitchcock 2 experiment. Top row: original images (the left image was considered as the target image). Middle: the* 16 *detected* 1*-meaningful matches. No false matches were detected, and all detections show an* NFA *below* 0.1. *The lowest* NFA *is* $6.5\,10^{-11}$.

The grouping procedure led to a unique maximal meaningful group, containing all 16 meaningful matches. The NFA_g of this group was $1.4\,10^{-75}$ (this result is remarkably better than the one obtanied using the similarity version: a single maximal group of 11 matches, with $NFA_g = 3.8\,10^{-24}$). The registration of the target image, superimposed to the scene image, is shown in Figure 14.16, as well as the image of gradient orientation comparison. Notice that the underlying perspective transform starts to be too strong to be approximated by a unique affine transform, over all the region where meaningful matches are found.

Figure 14.16: *Hitchcock 2 experiment: all* 16 *meaningful matches were spatially coherent; the group made by all of them was detected as the only maximal meaningful group ($NFA_g = 1.4\,10^{-75}$). Left: registration of the target image and superimposition on the scene image. Right: gradient orientation comparison.*

14.7 Detecting multiple groups

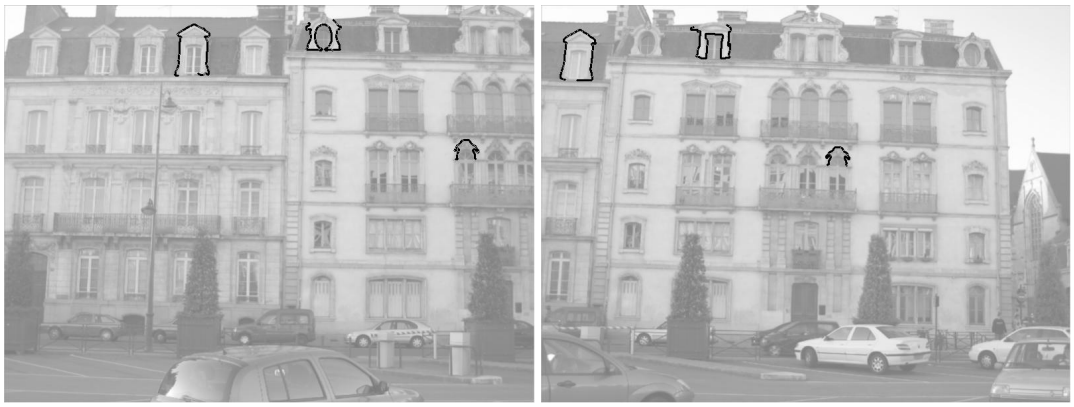
Several examples for this application were presented in previous chapters. We refer the reader to the corresponding sections:

- "Casablanca experiment": this example was presented in Chapter 1 in order to illustrate the recognition method developed in this thesis. It was also addressed in order to illustrate the validity and maximality rules in the detection of meaningful clusters of transformations (Chapter 13, section 13.4.1),
- "Object in clutter" experiment (Chapter 13, section 13.4.2),
- "Coca-Cola 1" experiment ((Chapter 13, section 13.4.2),
- "Coca-Cola 2" experiment ((Chapter 13, section 13.4.2).

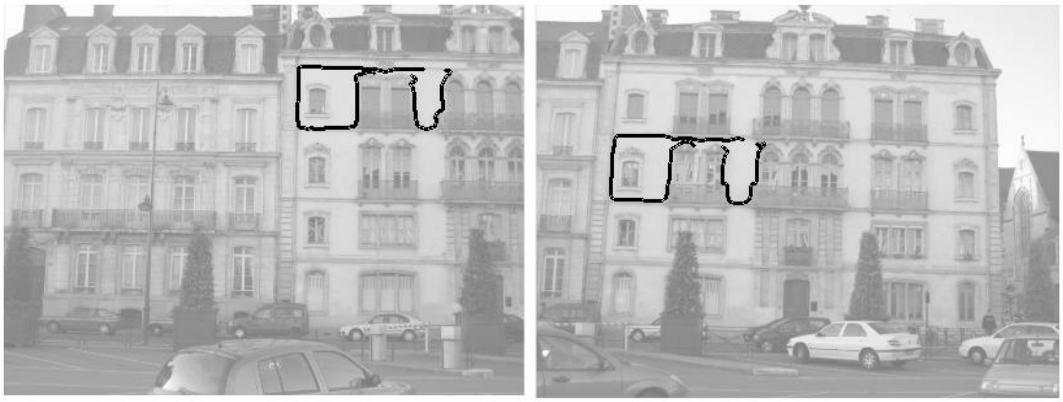
14.8 Strobe effect

This last example consist in finding groups of spatially coherent meaningful matches between the two images shown on top in Figure 14.17. Notice that, in addition to the group of matches given by the dominant motion, other groups induced by the periodicity of the buildings should be detected. This periodicity is not only present in the vertical direction but also in the horizontal direction, particularly for the right most building. The middle and bottom images in Figure 14.17 show the registration (with superimposition) images for the two maximal meaningful groups showing the lowest NFA_gs ($1.8\,10^{-144}$ and $6.4\,10^{-15}$, respectively). In Figure 14.18 we present two other maximal meaningful groups, corresponding to others (much less significant) strobe effects. Some more groups like that were detected.

Figure 14.17: *Strobe effect experiment. Top row: original images. Middle: registration and superimposition for the dominant motion; the group contains* 42 *matched shape elements, and* $NFA_g = 1.8\,10^{-144}$*. Bottom: registration and superimpositon for the most significant strobe effect;* $NFA_g = 6.4\,10^{-15}$*,* 7 *matched shape elements.*



(a) $NFA_g = 1.7\,10^{-6}$, 3 meaningful matches



(b) $NFA_g = 2.1\,10^{-3}$, 2 meaningful matches

Figure 14.18: *Two maximal meaningful groups revealing less significant strobe effects.*

14.9 Time complexity

stage	St. Jerôme	Hitchcock 1A	Las Meninas	Hitchcock 2	Strobe effect
image size target	376×597	640×480	500×333	640×480	800×600
image size scene	395×570	640×480	500×569	640×480	800×600
boundaries target	7.96	7.62	9.23	7.63	20.93
boundaries scene	8.51	7.89	14.99	6.79	23.02
smoothing target	0.79	1.31	1.80	1.32	2.76
smoothing scene	1.40	1.04	1.83	0.93	2.94
encoding target	9.26	18.27	20.11	17.97	41.90
encoding scene	22.13	9.11	19.60	11.94	38.94
matching	14.07	14.93	118.25	6.66	118.06
Nb. tests	$1.5\ 10^6$	$1.6\ 10^6$	$11.7\ 10^6$	$0.76\ 10^6$	$11.9\ 10^6$
grouping	2.49	2.86	20.76	3.25	21.39

Table 14.1: *Times (measured in seconds) for the computation of all the stages of the recognition algorithm, for some of the experiments presented in this chapter. The programs were run in a Pentium* 4, 1.7 *GHz. The row "Nb. tests" corresponds to the number of pairs of shape elements, tested in the matching stage.*

Chapter 15

Conclusion and perspectives

15.1 Main contributions of this thesis

Recognition is the ability to identify things based on prior knowledge. Visual recognition, in particular, is the process of finding correspondences between new elements and elements which had been previously seen, at least once, and live in our "world of images". In this thesis, we have focused on the problem of visual recognition based upon shape information. Two main issues are implicit in this problem: the representation of the shape contents which is present in images on the one hand, and the determination of correspondences between these representations on the other hand. Both problems have been addressed throughout this thesis.

The problem of shape representation was studied in the first part of this dissertation. Following [LMMM03], the *shape elements* defined in Chapter 1 were derived as the atoms of a shape representation which is consistent with the main classes of perturbations which do not affect recognition: contrast changes, occlusions, noise and geometrical distortions.

The problem of finding correspondences between shape elements was also addressed in the first part. We called this part "The recognition of partial shapes". Determining correspondences between shape elements not only means defining a notion of similarity between them, but also being able to decide if the two shape elements are to be paired or not. Proposing a framework that enables to come up to that kind of decisions was the main challenge of this first part.

Matching shape elements is nice, but shape elements are not what we think of when we look at images, and we do not recognize *shape elements* but *shapes*. The recognition of shapes based on shape elements thus needs for an integration of the recognized "partial shapes" given by shape elements. This integration of local information is certainly not performed as a simple conjunction of the recognized partial shapes. Indeed, the way these matched shape elements are organized in the image plane triggers another complementary recognition process, allowing to recognize "global shapes". This was the main subject of study in the second part of this thesis, which we called "Shape recognition as a grouping process". By taking into account the spatial coherence between the matched shape elements, we reinforced the confidence level on the recognition of "global shapes". As much as we did

in the first part of this thesis, a special attention was paid to decision models, allowing to assess the validity of a group of matched shape elements, and consequently to decide whether or not a target shape was present in an image scene.

Let us now summarize the main contributions of this thesis:

- We have presented a complete shape recognition method, for which all stages have been analyzed in depth. Among these stages, some of them deal with decision thresholds, and we have done significant effort in order to establish detection methods leading to sure, unsupervised decision thresholds.

- A general decision method for shape matching was proposed, and applied successfully to the correspondence problem between shape elements. The proposed method relies on the computation of the *number of false alarms* of a match (NFA), derived from a *background model* which assumes that matches occur "by chance", in a random situation. Meaningful matches are detected *a contrario* from this model, that is, as events whose probability of occurrence under the background model are extremely low. The NFA provides a measure of confidence on the detected matches. The detection of ε-meaningful matches can be performed by fixing an upper bound of value ε on the NFA. Taking $\varepsilon = 1$ or $\varepsilon = 0.1$ makes sense. Indeed, A database of shape elements being given, with each target shape element $\mathcal{S}$ and each distance δ we associate its *number of false alarms* $NFA(\mathcal{S}, \delta)$, namely the expectation of the number of shape elements at distance δ from $\mathcal{S}$ in the database. Assume that the $NFA(\mathcal{S}, \delta)$ is very small with respect to 1, and that a shape element $\mathcal{S}'$ from the database is found at distance δ from $\mathcal{S}$. This match could not occur just by chance and is therefore a meaningful detection.

- We proposed a method to find "natural" clusters in multi-dimensional data sets, based also on the *a contrario* decision framework. In this method, meaningful clusters are detected *a contrario* to a data-dependent null model. Here again, a *number of false alarms* for the groups of shape elements is defined (the NFA_g), and its value measures the confidence on the group detection. The main general contribution of the proposed method is the definition of a new local stopping rule or merging criterion (see Chapter 12), derived using statistical arguments, that proved to be very useful when applied to the transformation clustering problem.

Some other contributions have also been done, concerning the more popular problems of shape extraction and normalization procedures, namely:

- A parameterless method to detect the maximal meaningful boundaries in images. The proposed method improves the original method proposed by Desolneux *et al.* [DMM01], by introducing a multiscale approach that makes the method more robust to noise, and by proposing a more local algorithm allowing to take local contrast variations into account.

- An algorithm to detect flat pieces in curves, which coupled with bitangent lines, enables to encode nearly all meaningful level lines curves.

– Semi-local and global normalization methods, up to similarity and to affine invariance. These methods are based on bitangent lines and flat pieces (which provide robust directions on which normalization frames can be built), making the encoding very stable.

15.2 Future work

Many problems remain unsolved, or their solution has to be improved. We can classify them into two main categories: improvement of the results, and acceleration techniques.

Improving the results

The experimental results presented in this thesis are satisfactory and promising. The following points are to be explored, since they may lead to further improvements:

– The semi-local encoding methods we propose may often be not local enough, particularly the semi-local affine invariant encoding. The normalized codes shown in Figure 7.5 (Chapter 7) illustrate this problem. In fact, the main cause of non locality of the proposed semi-local normalization procedures is not the length of the encoded piece of curve (which could actually be controlled by parameter F), but the construction of the invariant frames (the lack of locality of this construction can be seen in Figure 7.4). Normalization based on more local information are thus needed, in order to perform better in the presence of occlusions. Semi-local versions of the area-based normalizations proposed for global encoding should be explored.

– Another problem of the proposed encodings follows from sampling all shape elements with a fixed number of points, independently of their lengths in the image. While this solution makes the computation of distances between normalized shape elements faster, precision problems may arise when considering long shape elements presenting strong oscillations. We can argue that these long shape elements should not have been considered in the comparison, since the fact they are long means they are not local representations. However, discarding them implies introducing a threshold on the length of shape elements, and we would not like to do that. Notice that this problem does not seem to be critical, since false matches involving such long shape elements always show NFAs close to 1 (see Figure 9.20 in Chapter 9 for an example). Anyway, solutions such as sampling shape elements at the same rate (with respect to their actual length in the image) and considering Hausdorff of Frêchet distances are to be explored. The problem of these kind of solutions is that they will certainly strongly increase the computation times. If we ever get to solve this problem, false matches involving long shape elements will no longer be meaningful matches.

– As the reader should have noticed, global meaningful matches were not considered in the grouping stage. The only reason for that are schedule constraints... There is not much to ex-

plore here; the integration of global matches to the complete recognition method just has to be implemented. We are very optimistic about that: grouping results should be greatly improved.

– Last but not least: how can we integrate the NFA associated to the meaningful matches to the NFA_g of spatially coherent groups? For instance, an isolated 10^{-20}-meaningful match will not be detected as a "shape", since a single match may not lead to a meaningful group. However, such a low NFA reveals an extremely significant match. A first proposal combining the two measures (NFA and NFA_g) has been presented in Chapter 13, section 13.5.3, but this combination was not suitable since it cannot deal with the maximality notion, which is essential here.

Reducing the time of computation

Some times for the computation of experiments presented in Chapter 14 were presented in Table 14.1. All the stages of the proposed recognition method take too much time, and important accelerations are needed if we want to consider real applications other than detection of targets in an image scene or off-line detection in general. Notice however that one of the main causes of the slowness of the method is its complete genericity and generality. Indeed, the presented method performs well on a large variety of images and problems. Thus, accelerations should be introduced for specific applications. (For instance, shape elements may be discarded *a priori*, based on prior knowledge.)

Another point we should keep in mind here is that all the detection methods we propose in this thesis are actually not only detection methods but also learning methods. Indeed, all decision thresholds involved in our method were derived ("learned") before comparison. In specific application, thresholds can be learnt once for all, and decision can be made using these fixed thresholds.

Regarding the aspects that can bring some acceleration while keeping the method as general as it is now, we should manage to:

– Reduce the redundancy of the encoding procedure. But is it possible to do it before the matching stage? Up to now we have been using a greedy algorithm to reduce redundancy in order to fit the independence requirement for the grouping background model. This can be done without any problem, since to each meaningful match an NFA is associated, providing then an ordering upon which we can base the elimination of "repeated" matches. We do not clearly see how redundancy reduction can be performed before matching, but being able to do it seems to be critical in order to accelerate the method.

– Explore new kind of independent features allowing for indexing the database of shape elements. We are not very optimistic about that. While such a kind of features can accelerate the method, we think they may lead to worst detection results. Indeed, the features we consider (the 5 pieces of normalized shape elements and the coarse description, proposed in Chapter 8) seem to give a (quite) complete representation of the shape elements, and the considered L^∞ distance is in accordance with our perceptual notion of proximity.

Bibliography

[ACDH99] N. Arnaud, F. Cavalier, M. Davier, and P. Hello. Detection of gravitational wave bursts by interferometric detectors. *Physical review D*, 59(8):082002–1 – 082002–9, 1999. 38

[ACMM01] L. Ambrosio, V. Caselles, S. Masnou, and J.M. Morel. The connected components of Caccioppoli sets and applications to image processing. *Journal of the European Society of Mathematics*, 3:213–266, 2001. 4

[AD90] N. Ansari and E. J. Delp. Partial shape recognition: A landmark-based approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):470–483, 1990. 33

[ADV03] A. Almansa, A. Desolneux, and S. Vamech. Vanishing point detection without any a priori information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):502–507, 2003. 95, 131

[AG99] H. Alt and L. Guibas. Discrete geometric shapes: Matching, interpolation, and approximation. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 121–153. Elsevier Science Publishers, 1999. 29

[AGLM93] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel. Axioms and fundamental equations of image processing: Multiscale analysis and P.D.E. *Archive for Rational Mechanics and Analysis*, 16(9):200–257, 1993. 6, 7, 13, 30, 84, 85, 116

[AGM99] L. Alvarez, Y. Gousseau, and J.-M. Morel. *Scales in natural images and a consequence on their Bounded Variation Norm*, pages 247–259. Lecture Notes in Computer Science, 1682, 1999. 266

[AKW01] H. Alt, C. Knauer, and C. Wenk. Matching polygonal curves with respect to the Fréchet distance. In *Proceedings of the 18th International Symposium on Theoretical Aspects of Computer Science*, pages 63–74, Dresden, Germany, February 15-17 2001. 34

[AL00] A.A. Adjeroh and M.C. Lee. An occupancy model for image retrieval and similarity evaluation. *IEEE Transactions on Image Processing*, 9(1):120–131, 2000. 38

[AMS02] L. Alvarez, L. Mazorra, and F. Santana. Geometric invariant shape representations using morphological multiscale analysis and applications to shape representation. *Journal of Mathematical Imaging and Vision*, 18(2):145–168, 2002. 32

[Åst94] K. Åström. Affine and projective normalization of planar curves and regions. In *Proceedings of European Conference on Computer Vision*, volume 2, pages 439–448, Stockholm, Sweden, 1994. 30

[Åst95] K. Åström. Fundamental limitations on projective invariants of planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):77–81, 1995. 7, 30

[AST98] S. Angenent, G. Sapiro, and A. Tannenbaum. On the affine heat flow for nonconvex curves. *Journal of the American Mathematical Society*, 1998. 14

[Att54] F. Attneave. Some informational aspects of visual perception. *Psychological review*, 61(3):183–193, 1954. 1, 2, 33, 40, 41

[Bal81] D.H. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981. 35, 260, 292

[BB95] L. Bottou and Y. Bengio. Convergence properties of the k-means algorithms. In G. Tesario and D. Touretzky, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 585–592, Denver, Colorado, USA, 1995. 229

[BCGM98] S. Belongie, C. Carson, H. Greenspan, and J. Malik. Color- and texture-based image segmentation using the Expectation-Maximization algorithm and its application to content-based image retrieval. In *Proceedings of the International Conference on Computer Vision*, pages 675–682, Mumbai, India, 1998. 32

[BCM03] C. Ballester, V. Caselles, and P. Monasse. The tree of shapes of an image. *ESAIM: Control, Optimisation and Calculus of Variations*, 9:1–18, 2003. 48

[BMP02] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):509–522, 2002. 34

[Boc85] H.H. Bock. On some significance tests in cluster analysis. *Journal of Classification*, 2:77–108, 1985. 15, 235, 236

[BS91] G. Barles and P.M. Souganidis. Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic Analysis*, 4:271–283, 1991. 87

[Can83] J. Canny. A variational approach to edge detection. In *National Conference on Artificial Intelligence*, pages 54–58, Washington DC, USA, 1983. 41

[Can86] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986. 30, 54, 95

[Cao03] F. Cao. Good continuations in digital images. In *Proceeding of International Conference on Computer Vision*, volume 1, pages 440–447, Nice, France, 2003. 57, 72, 79

[Cao04] F. Cao. Application of the Gestalt principles to the detection of good continuations and corners in image level lines. *Computing and Visualisation in Science*, 7(1):3–13, 2004. 132

[CBCN01] P.B. Chapple, D.C. Bertilone, R.S. Caprari, and G.N. Newsam. Stochastic model-based processing for detection of small targets in non-gaussian natural imagery. *IEEE Transactions on Image Processing*, 10(4):554–564, 2001. 39

[CCM96] V. Caselles, B. Coll, and J.-M. Morel. A Kanizsa program. *Progress in Nonlinear Differential Equations and their Applications*, 25:35–55, 1996. 55, 66

[CCM99] V. Caselles, B. Coll, and J.-M. Morel. Topographic maps and local contrast changes in natural images. *International Journal of Computer Vision*, 33(1):5–27, 1999. 4, 27, 55, 66

[CH74] T. Calinski and J. Harabasz. A dendrite method for cluster analysis. *Communications in statistics*, 3(1):1–27, 1974. 234

[CHY95] F.S. Cohen, Z. Huang, and Z. Yang. Invariant matching and identification of curves using B-splines curve representation. *IEEE Transactions on Image Processing*, 4(1):1–10, 1995. 33

[CK98] J.L. Cox and D.B. Karron. Digital Morse theory. Manuscript available from `http://www.casi.net/D.DMT/D.Overview/AcademicPressPaper14-03`, 1998. 44, 48, 49

[CKS97] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997. 77

[CLM94] T. Cohignac, C. Lopez, and J.M. Morel. Integral and local affine invariant parameter and application to shape recognition. In *International Conference on Pattern Recognition*, pages A:164–168, 1994. 33

[CMS04] F. Cao, P. Musé, and F. Sur. Extracting meaningful curves from images. *Journal of Mathematical Imaging and Vision*, 2004. To appear. 20, 54, 116

[Coh94] T. Cohignac. *Reconnaissance de formes planes*. PhD thesis, Ceremade, Université Paris IX Dauphine, 1994. 32, 122

[CV01] T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001. 54

[DBM77] S.A. Dudani, K.J. Breeding, and R.B. McGhee. Aircraft identification by moment invariants. *IEEE Transactions on Computers*, 26(1):39–46, 1977. 31

[DE84] W.H.E. Day and H. Edelsbrunner. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, 1(1):7–24, 1984. 230

[Der87] R. Deriche. Using Canny's criteria to derive a recursively implemented optimal edge detector. *International Journal of Computer Vision*, 1(2):167–187, 1987. 41

[DF90] R. Deriche and O. Faugeras. Tracking line segments. *Image and Vision Computing*, 8(4):261–270, 1990. 95

[DHS00] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. John Wiley and Sons, 2000. 28, 34, 224, 225, 226, 227, 228, 229, 230, 234

[DK82] P.A. Devijver and J. Kittler. *Pattern recognition - A statistical approach*. Prentice Hall, 1982. 133

[DLR77] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977. 227

[DMM00] A. Desolneux, L. Moisan, and J.-M. Morel. Meaningful alignments. *International Journal of Computer Vision*, 40(1):7–23, 2000. 14, 60, 96, 131, 241

[DMM01] A. Desolneux, L. Moisan, and J.-M. Morel. Edge detection by Helmholtz principle. *Journal of Mathematical Imaging and Vision*, 14(3):271–284, 2001. 4, 13, 20, 40, 53, 56, 57, 58, 59, 60, 61, 64, 99, 116, 131, 318

[DMM03a] A. Desolneux, L. Moisan, and J.-M. Morel. A grouping principle and four applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):508–513, 2003. 14, 15, 54, 57, 58, 59, 247

[DMM03b] A. Desolneux, L. Moisan, and J.-M. Morel. Variational snake theory. In S. Osher and N. Paragios, editors, *Geometric Level Set Methods in Imaging, Vision, and Graphics*. Springer Verlag, 2003. 56, 72, 77, 79

[DMM04] A. Desolneux, L. Moisan, and J.-M. Morel. *Computational Gestalt Theory*. Lecture Notes in Mathematics, Springer Verlag, 2004. To appear. 14

[DRRRD03] I. Debled-Rennesson, J.-L. Rémy, and J. Rouyer-Degli. Segmentation of discrete curves into fuzzy segments. Technical Report 4989, INRIA, 2003. 96

[Dry96] I. Dryden. General shape and registration analysis. Technical report, University of Leeds, Department of Statistics, 1996. 29

[Dub87] R . C. Dubes. How many clusters are best? – an experiment. *Pattern Recognition*, 20(6):645–663, 1987. 233, 234

[ER92] L.C. Evans and R.Gariepy. *Measure Theory and Fine Properties of Functions*. CRC Press, 1992. 55

[Fau93] O. Faugeras. *Three-dimensional Computer Vision: a Geometrical Viewpoint*. MIT Press, Cambridge, 1993. 84

[FB81] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the Association for Computing Machinery*, 24(6):381–395, 1981. 260, 261

[FB86] M.A. Fischler and R.C. Bolles. Perceptual organization and curve partitioning. *IEEE Transactions on pattern analysis and machine intelligence*, 8(1):100–105, 1986. 97

[FH98] P. Felzenszwalb and D.P. Huttenlocher. Image segmentation using local variation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 98–104, 1998. 66

[FK95] O. Faugeras and R. Keriven. Some recent results on the projective evolution of 2D curves. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 13–16, Washington DC, USA, 1995. 7, 30, 85

[FL99] P. Frosini and C. Landi. Size theory as a topological tool for computer vision. *Pattern Recognition and Image Analysis*, 9(4):596–603, 1999. 22, 31, 205

[FL01] P. Frosini and C. Landi. Size functions and formal series. *Applicable Algebra in Engineering, Communication and Computing*, 12:327–349, 2001. 22, 31, 205

[Gei03] W.S. Geisler. Ideal observer analysis. In L. Chalupa and J. Werner, editors, *The Visual Neurosciences*. MIT Press, 2003. 35

[GH90] W.E.L. Grimson and D.P. Huttenlocher. On the sensitivity of the Hough transform for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(3):255–274, 1990. 261, 294

[GH91] W.E.L. Grimson and D.P. Huttenlocher. On the verification of hypothesized matches in model-based recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(12):1201–1213, 1991. 35, 37, 38, 294

[Gir87] G. Giraudon. Chaînage efficace de contour. Technical Report 0605, INRIA, 1987. 40, 96

[GL89] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1989. 122

[GM96] G. Guy and G. Medioni. Inferring global perceptual contours from local features. *International Journal on Computer Vision*, 20(1):113–133, 1996. 97

[GMR04] F. Guichard, J.-M. Morel, and R.D. Ryan. Image analysis and P.D.E.'s. Book in preparation, 2004. 21, 83, 85, 86, 87, 88

[Gor96] A.D. Gordon. Null models in cluster validation. In W. Gaul and D. Pfeifer, editors, *From Data to Knowledge: Theoretical and Practical Aspects of Classification, Data Analysis, and Knowledge Organization*, pages 32–44. Springer Verlag, 1996. 235, 236, 247

[Gor99] A.D. Gordon. *Classification*. Monographs on Statistics and Applied Probability 82, Chapman & Hall, 1999. 15, 233, 234, 235, 236

[Gou03] Y. Gousseau. Comparaison de la composition de deux images, et application a la recherche automatique. In *proceedings of GRETSI 2003*, Paris, France, 2003. 132

[Gre93] U. Grenander. *General pattern recognition*. Oxford Science Publications, 1993. 135

[GW99] Y. Gdalyahu and D. Weinshall. Flexible syntactic matching of curves and its application to automatic hierarchical classification of silhouettes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1312–1328, 1999. 29, 33

[Har84] R. Haralick. Digital step edges from zero crossing of second directional derivatives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):58–68, 1984. 54, 67

[HC96] Z. Huang and F.S. Cohen. Affine-invariant B-spline moments for curve matching. *IEEE Transactions on Image Processing*, 5(10):1473–1480, 1996. 33

[HJ95] D.P. Huttenlocher and E.W. Jaquith. Computing visual correspondence: incorporating the probability of a false match. In *Proceedings of International Conference on Computer Vision*, pages 515–522, Cambridge, Massachussets, USA, 1995. 35

[HKR93] D.P. Huttenlocher, D. Klanderman, and A. Rucklige. Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863, 1993. 28

[HO00] A. Hyvärinen and E. Oja. Indpeendent component analysis: Algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000. 199

[Hoe63] W. Hoeffding. Probability inequalities for sums of random variables. *Journal of the American Statistical Association*, (58):13–30, 1963. 245

[Hou62] P.V.C. Hough. *Methods and means for recognizing complex patterns*, 1962. U.S. Patent 3,069,654. 95, 260

[HTF01] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2001. 139, 224, 226, 227, 230, 232

[Hu62] M.K. Hu. Visual pattern recognition by moments invariants. *IRE Transactions on Information Theory*, 8:179–187, 1962. 31, 122

[HU87] D.P. Huttenlocher and S. Ullman. Object recognition using alignment. In *International Conference of Computer Vision*, pages 267–291, London, UK, 1987. 35, 294

[Hyv99] A. Hyvärinen. Survey on independent component analysis. *Neural Computing Surveys*, (2):94–128, 1999. 199

[HZ00] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 261, 262

[HZW99] M. Handouyaya, D. Ziou, and S. Wang. Sign language recognition using moment-based size functions. In *Proceedings of the Vision Interface (VI'99) conference*, pages 210–216, Trois-Rivières, Canada, 1999. 209

[IK88] J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics, and Image Processing*, 44(1):87–116, 1988. 95

[Jac96] D.W. Jacobs. Robust and efficient detection of salient convex groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):23–37, 1996. 95

[Jay03] E.T. Jaynes. *Probability theory - the logic of science*. Cambridge University Press, 2003. 134

[JDJ00] A.K. Jain, R.P.W Duin, and M. Jiachang. Statistical pattern recognition: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–36, 2000. 224

[JDP83] K. Joag-Dev and F. Proschan. Negative association of random variables, with applications. *Annals of Statistics*, 11(1):286–295, 1983. 242, 249

[JMF99] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999. 15, 224, 225, 226, 227, 228, 229, 230

[JZ97] A.K. Jain and D. Zongker. Feature selection: evaluation, application, and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):153–158, 1997. 224

[Kan79] G. Kanizsa. *Organization in Vision: Essays on Gestalt Perception*. Praeger, 1979. 3, 30

[Kan96] G. Kanizsa. *La Grammaire du Voir*. Diderot, 1996. Original title: *Grammatica del vedere*. French translation from Italian. 54, 57

[KB01] M.I. Khalil and M.M. Bayoumi. A dyadic wavelet affine invariant function for 2d shape recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10), 2001. 31

[KB03] R. Kimmel and A.M. Bruckstein. On regularized Laplacian zero crossings and other optimal edge integrators. *International Journal of Computer Vision*, 53(3):225–243, 2003. 77

[KLS89] A. Krzyzak, S.Y. Leung, and C.Y. Suen. Reconstruction of two-dimensional patterns from Fourier descriptors. *Machine Vision and Applications*, 2:123–140, 1989. 31

[KM99] G. Koepfler and L. Moisan. Geometric multiscale representation of numerical images. In *Second International Conference on Scale Space Theories in Computer Vision*, volume 1682 of *Lecture Notes in Computer Science*, pages 339–350. Springer-Verlag, 1999. 85, 88

[Koe84] J.J. Koenderink. The structure of images. *Biological Cybernetics*, (50):363–370, 1984. 55

[KR89] T.Y. Kong and A. Rosenfeld. Digital topology: Introduction and survey. *Computer Vision, Graphics and Images Processing*, 48(3):357–393, 1989. 52

[KR90a] L. Kaufman and P. J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. John Wiley and Sons, 1990. 225, 226, 227

[KR90b] T.Y. Kong and A. Rosenfeld. If we use 4- or 8-connectedness for both the objects and the background, the Euler characteristic is not locally computable. *Pattern Recognition Letter*, 11:231–232, 1990. 52

[Kro50] A.S. Kronrod. On functions of two variables. *Uspehi Mathematical Sciences*, 5(35):24–134, 1950. (in Russian). 48

[Kur92] C. Kuratowski. *Topologie, I et II*. Jacques Gabay, 1992. 49

[KWT87] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, (1):321–331, 1987. 54, 77

[LB93] M. Lindenbaum and A. Bruckstein. On recursive, $O(N)$ partitioning of a digitized curve into digital straight segments. *Transactions on Pattern Analysis and Machine Intelligence*, 15(9), 1993. 96

[LC86] C.C. Lin and R. Chellappa. Classification of partial 2-d shapes using Fourier descriptors. In *CVPR86*, pages 344–350, 1986. 31

[Lin98] T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):77–116, 1998. 67

[Lis01] J.L. Lisani. *Shape Based Automatic Images Comparison*. PhD thesis, Université Paris 9 Dauphine, France, 2001. 14, 33, 34, 40, 96, 117

[LKK95] Z. Liu, D.C. Knill, and D. Kersten. Object classification for human and ideal observers. *Vision Research*, 35(4):549–568, 1995. 35

[LMMM03] J.L. Lisani, L. Moisan, P. Monasse, and J.-M. Morel. On the theory of planar shape. *SIAM Multiscale Modeling and Simulation*, 1(1):1–24, 2003. 2, 13, 21, 33, 34, 40, 42, 81, 83, 85, 88, 117, 317

[LMR01] J.L. Lisani, P. Monasse, and L. Rudin. Fast shape extraction and applications. Technical Report 2001-16, CMLA, ENS Cachan, 2001. 41, 42, 46, 47, 48, 49

[Lon98] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998. 29

[Low85] D.G. Lowe. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publisher, 1985. 139, 237, 247

[Low99] D.G Lowe. Object recognition from local scale-invariant features. In *Proceedings of IEEE International Conference on Computer Vision*, pages 1150–1157, Corfu, Greece, 1999. 261

[LSW88] Y. Lamdan, J.T. Schwartz, and H.J. Wolfson. Object recognition by affine invariant matching. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pages 335–344, Ann Arbor, Michigan, U.S.A., 1988. 14, 21, 33, 95

[LW67] G.N. Lance and W.T. Williams. A general theory of classificatory sorting strategies. i. hierarchical systems. *Computer Journal*, 9:373–370, 1967. 229

[LW88] Y. Lamdan and H.J. Wolfson. Geometric hashing: a general and efficient model-based recognition scheme. In *Proceedings of IEEE International Conference on Computer Vision*, pages 238–249, Tampa, Florida, USA, 1988. 35, 261, 294

[MAK96a] F. Mokhtarian, S. Abbasi, and J. Kittler. Efficient and robust retrieval by shape content through curvature scale space. In *Proceedings of International Workshop on Image Databases and MultiMedia Search*, pages 35–42, Amsterdam, The Netherlands, 1996. 29

[MAK96b] F. Mokhtarian, S. Abbasi, and J. Kittler. Robust and efficient shape indexing through curvature scale space. In *Proceedings of British Machine Vision Conference*, pages 53–62, Edinburgh, UK, 1996. 29

[Mal99] S. Mallat. *A Wavelet Tour in Signal Processing*. Academic Press, 2nd edition, 1999. 55

[Mar82] D. Marr. *Vision*. Freeman Publishers, 1982. 32, 41, 54, 67, 247, 254

[Mat75] G. Matheron. *Random Sets and Integral Geometry*. John Wiley and Sons, 1975. 42, 54

[MC85] G.W. Milligan and M.C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985. 234

[Met75] W. Metzger. *Gesetze des Sehens*. Waldemar Kramer, 1975. 1

[MG00] P. Monasse and F. Guichard. Fast computation of a contrast invariant image representation. *IEEE Transactions on Image Processing*, 9(5):860–872, 2000. 13, 42, 46, 48, 49

[MH80] D. Marr and E. Hildreth. Theory of edge detection. *Proceeding of the Royal Society of London*, (B-207):187–207, 1980. 32, 41, 54, 65

[Mil69] J. Milnor. *Morse Theory*. Number Study 51 in Annals of Mathematics Studies. Princeton University Press, 1969. 49

[MM92] F. Mokhtarian and A.K. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):789–805, 1992. 32

[MM00] F. Meyer and P. Maragos. Nonlinear scale-space representation with morphological levelings. *Journal of visual communication and image representation*, (11):245–265, 2000. 55

[Moi97] L. Moisan. *Traitement numérique d'images et de films: équations aux dérivées partielles préservant forme et relief*. PhD thesis, Université Paris 9 Dauphine, France, 1997. 85

[Moi98] L. Moisan. Affine plane curve evolution: A fully consistent scheme. *IEEE Transactions on Image Processing*, 7(3):411–420, 1998. 13, 21, 85, 87, 88, 90, 117

[Mok95] F. Mokhtarian. Silhouette-based isolated object recognition through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):539–544, 1995. 29

[Mon99] P. Monasse. Contrast invariant image registration. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, volume 6, pages 3221–3224, Phoenix, Arizona, USA, 1999. 31

[Mon00] P. Monasse. *Représentation morphologique d'images numériques et application au recalage, Morphological Representation of Digital Images and Application to Registration*. PhD thesis, Université Paris 9 Dauphine, France, 2000. 59, 60, 69

[MS89] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Communication on Pure and Applied Mathematics*, 42(4):577–685, 1989. 66

[MS95] J.-M. Morel and S. Solimini. *Variational Methods in Image Segmentation*. Birkhauser, 1995. 229

[MS04] L. Moisan and B. Stival. A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. *International Journal of Computer Vision*, 57(3):201–218, 2004. 132

[MSC+04] P. Musé, F. Sur, F. Cao, Y. Gousseau, and J.-M. Morel. Accurate estimates of false alarm number in shape recognition. Technical Report 2004-01, CMLA, ENS de Cachan, 2004. 21, 81

[MSCG03] P. Musé, F. Sur, F. Cao, and Y. Gousseau. Unsupervised thresholds for shape matching. In *Proceedings of IEEE International Conference on Image Processing*, Barcelona, Spain, 2003. 21

[MSM03] P. Musé, F. Sur, and J.-M. Morel. Sur les seuils de reconnaissance des formes. *Traitement du Signal*, 20(3):279–294, 2003. 81, 140, 192

[MTY02] M.I. Miller, A. Trouvé, and L. Younès. On the metrics and euler-lagrange equations of computational anatomy. *Annual Review of Biomedical Engineering*, 4:375–405, 2002. 34

[MTY03] M.I. Miller, A. Trouvé, and L. Younès. Geodesic shooting for computational anatomy. To appear, 2003. 34

[OH97] C. Olson and D.P. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IEEE Transactions on Image Processing*, 6(12):103–113, 1997. 32, 35, 37

[OK04] C. Olman and D. Kersten. Classification objects, ideal observers & generative models. *Cognitive Science*, (28):227–239, 2004. 35

[Ols98] C.F. Olson. Improving the generalized Hough transform through imperfect grouping. *Image and Vision Computing*, 16(9-10):627–634, 1998. 38

[Pav80] T. Pavlidis. *Structural Pattern Recognition*. Springer Verlag, second edition, 1980. 40

[PD02] N. Paragios and R. Deriche. Geodesic active regions and level set methods for supervised texture segmentation. *International Journal of Computer Vision*, 46(3):223–247, 2002. 77

[Pen98] X. Pennec. Toward a generic framework for recognition based on uncertain geometric features. *Videre: Journal of Computer Vision Research*, 1(2):58–87, 1998. 38, 40, 294

[PF77] E. Persoon and K.S. Fu. Shape discrimination using Fourier descriptors. *SMC*, 7(3):170–179, 1977. 31

[PM04] E. Le Pennec and S. Mallat. Sparse geometrical image approximation with bandelets. *IEEE Transaction on Image Processing*, 2004. To be published. 55

[RH79] B.L. Raktoe and J.J. Hubert. *Basic Applied Statistics*. Marcel Dekker Inc., 1979. 142

[Rot95] C.A. Rothwell. *Object Recognition Through Invariant Indexing*. Oxford Science Publications, 1995. 14, 21, 29, 30, 33, 117

[Rub21] E. Rubin. *Visuell wahrgenommene Figuren*. Copenhagen, Gyldendals, 1921. 3

[RZFM95] C.A. Rothwell, A. Zisserman, D.A. Forsyth, and J.L. Mundy. Planar object recognition using projective shape representation. *International Journal of Computer Vision*, 16:57–99, 1995. 33

[SC98] J. Sato and R. Cipolla. Quasi-invariant parameterisations and matching of curves in images. *International Journal of Computer Vision*, 28(2):117–138, 1998. 33

[SCD02] J.L. Starck, E.J. Candès, and D.L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684, 2002. 55

[Sch99] C. Schmid. A structured probabilistic model for recognition. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, volume 2, pages 485–490, Fort Collins, Colorado, USA, 1999. 34, 35

[Sed99] R. Sedgewick. *Algorithms in C++: Fundamentals, Data Structures, Sorting, Searching*. Addison-Wesley, 3 edition, 1999. 52

[Ser82] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982. 4, 42, 54, 55, 84

[SG80] J. Sklansky and V. Gonzalez. Fast polygonal approximation of digitized curves. *Pattern Recognition*, 12:327–331, 1980. 97

[SG00] P. Salembier and L. Garrido. Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. *IEEE Transactions on Image Processing*, 9(4):561–576, 2000. 55, 60

[SI99] D. Shen and H.H.S. Ip. Discriminative wavelet shape descriptors for recognition of 2-d patterns. *Pattern Recognition*, 32(8):151–165, 1999. 31

[Sil75] S.D. Silvey. *Statistical Inference*. Chapman and Hall, 1975. 133

[SKB82] G. Stockman, S. Kopstein, and S. Benett. Matching images to models for registration and object detection via clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(3):229–241, 1982. 261, 292

[SM97] C. Schmid and R. Mohr. Local greyvalue invariants for image retrieval. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997. 32

[SM00] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000. 66

[Sma96] C.G. Small. *The Statistical Theory of Shapes*. Springer Verlag, 1996. 7, 34, 35, 225

[SP95] S. Sclaroff and A. Pentland. Modal matching for correspondence and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):545–561, 1995. 31, 34

[SS95] P. Salembier and J. Serra. Flat zones filtering, connected operators, and filters by reconstruction. *IEEE Transactions on Image Processing*, 4(8):1153–1160, 1995. 55, 60

[ST93] G. Sapiro and A. Tannenbaum. Affine invariant scale-space. *International Journal of Computer Vision*, 11(1):25–44, 1993. 13, 84, 116

[Ste95] C.V. Stewart. MINPRAN: a new robust estimator for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):925–938, 1995. 35

[TSK03] P.N. Tan, M. Steinbach, and V. Kumar. Introduction to Data Mining. Not published yet, 2003. 228, 233

[VC00] C. C. Venters and M. D. Cooper. A review of content-based image retrieval systems. Technical Report jtap-054, University of Manchester, UK, 2000. 29

[Vel01] R.C. Veltkamp. Shape matching: similarity measures and algorithms. In *Proceedings of International Conference on Shape Modeling and Applications*, pages 188–197, Genova, Italy, 2001. 28, 29

[VH01] R. Veltkamp and M. Hagedoorn. State-of-the-art in shape matching. In M.S. Lew, editor, *Principles of Visual Information Retrieval*, volume 19. Springer Verlag, 2001. 29

[Vin93] L. Vincent. Grayscale area openings and closings, their efficient implementation and applications. In J. Serra and P. Salembier, editors, *Proceedings of the* 1^{st} *Workshop on Mathematical Morphology and its Applications to Signal Processing*, pages 22–27, Barcelona, Spain, 1993. 60

[VT00] R. Veltkamp and M. Tanase. Content-based image retrieval systems: A survey. Technical Report UU-CS-2000-34, Utrecht University, 2000. 29

[War63] J. H. Jr. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(2):236–244, 1963. 230

[Wei93] I. Weiss. Noise-resistant invariants of curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):943–948, 1993. 33

[Wer23] M. Wertheimer. Untersuchungen zur Lehre der Gestalt, II. *Psychologische Forschung*, (4):301–350, 1923. Translation published as Laws of Organization in Perceptual Forms, in Ellis, W. (1938). A source book of Gestalt psychology (pp. 71-88). Routledge & Kegan Paul. 2, 27, 54, 57, 237, 247, 294

[Wit83] A.P. Witkin. Scale space filtering. In *Proceedings of International Joint Conference on Artificial Intelligence*, pages 1019–1021, Karlsruhe, Germany, 1983. 55

[WN99] A. Winter and C. Nastar. Differential feature distribution maps for image segmentation and region queries in image databases. In *CBAIVL Workshop at Conference on Computer Vision and Pattern Recognition*, Fort Collins, Colorado, USA, 1999. 32

[Wol90a] H.J. Wolfson. Model-based object recognition by Geometric Hashing. In *Proceedings of the European Conference on Computer Vision*, pages 526–536, Antibes, France, 1990. Lecture Notes in Computer Vision 427, Springer Verlag. 35

[Wol90b] H.J. Wolfson. On curve matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):483–489, 1990. 32

[WR97] H.J. Wolfson and I. Rigoutsos. Geometric hashing: an overview. *IEEE Computational Science & Engineering*, 4(4):10–21, 1997. 35, 261, 294

[WW96] G.H. Watson and S.K. Watson. Detection of unusual events in intermittent non-gaussian images using multiresolution background models. *Optical Engineering*, 35(11):3159–3171, 1996. 39

[Zhu99] S.C. Zhu. Embedding Gestalt laws in markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1170–1187, 1999. 71

[ZR72] C.T. Zahn and R.Z. Roskies. Fourier descriptors for plane closed curves. *IEEE Transactions on Computers*, C-21(3):269–281, 1972. 31

Résumé

Cette thèse traite de la reconnaissance des formes dans les images numériques. Une représentation appropriée des formes est déduite de l'analyse des perturbations qui n'affectent pas la reconnaissance : changement de contraste, occlusion partielle, bruit, perspective. Les atomes de cette représentation, appelés *éléments de forme*, fournissent des descriptions semi-locales des formes. L'appariement de ces éléments permet de reconnaître des « formes partielles ». Les « formes globales » sont alors définies comme des groupes de formes partielles présentant une cohérence dans leur disposition spatiale.
L'aspect fondamental de ce travail est la mise en place de seuils non-supervisés, à tous les niveaux de décision du processus de reconnaissance. Nous proposons des règles de décision pour la mise en correspondance de formes partielles ainsi que pour la détection de formes globales. Le cadre proposé est basé sur une méthodologie générale de la détection dans laquelle un événement est significatif s'il n'est pas susceptible d'arriver par hasard.

Mots-Clés : reconnaissance de formes, lignes de niveau, élément de forme, normalisation, modèle de fond, nombre de fausses alarmes, détection *a contrario*, classification non-supervisée, groupement de formes.

Abstract

This thesis deals with the recognition of shapes in digital images. A suitable shape representation is derived by analyzing invariance to perturbations that do not significantly affect visual recognition: contrast changes, partial occlusion, noise, perspective distortion. The atoms of such a representation, called *shape elements*, provide semi-local descriptions of shapes. Matching shape elements enables the recognition of "partial shapes". Then, "global shapes" are defined as groups of partial shapes showing some spatial coherence.
Deriving unsupervised thresholds involved in all decision levels of the shape recognition process, is the central points of this work. We propose decision rules for both the correspondence problem of partial shapes, and for the detection of global shapes. The proposed framework is based on a general detection methodology asserting that meaningful events may be viewed as exceptions to randomness.

Keywords: shape recognition, topographic map, level lines, shape element, normalisation, background model, number of false alarms, *a contrario* detection, unsupervised classification, clustering, shape grouping.