Graph Mining and Graph Classification : application to cadastral map analysis

Romain Raveaux

THÈSE DE DOCTORAT

présentée devant

L'Université de La Rochelle
Ecole Doctorale en Sciences et Ingénierie pour l'information (S2I)

en vue de l'obtention du grade et du titre de

Docteur de l'Université de La Rochelle

soutenue publiquement le 25 novembre 2010 par

Romain Raveaux

Fouille de graphes et classification de graphes : Application à l'analyse de plans cadastraux.

(Graph Mining and Graph Classification: Application to cadastral map analysis.)

sous la direction de M. Jean-Marc Ogier et M. Jean-Christophe Burie

Composition du jury

M. Josep Lladós	Université autonome de Barcelone	*rapporteur*
M. Chew Lim Tan	Université de Singapour	*rapporteur*
M^{me} Nicole Vincent	Université de Paris 5	*rapporteur*
M. Jean-Marc Ogier	Université de La Rochelle	*directeur de thèse*
M. Jean-Christophe Burie	Université de La Rochelle	*encadrant de thèse*
M. Umapada Pal	Institut de la statistique d'Inde	*examinateur*
M. Salvatore-Antoine Tabbone	Université de Nancy 2	*examinateur*
M. Rolf Ingold	Université de Fribourg	*Président*

Ce travail de thèse a été effectué au sein du laboratoire suivants:

- L3i (EA 2118), Université de La Rochelle, France

Remerciements

Je tiens à exprimer tout d'abord mes remerciements aux membres du jury, qui ont accepté d'évaluer mon travail de thèse.

Merci à M. Rolf Ingold, Professeur des universités de l'Université de Fribourg, d'avoir accepté de présider le jury de cette thèse, et à MM. les Professeurs Chew Lim Tan de l'Université de Singapour, Josep Lladós de l'Université de Barcelone et Nicole Vincent de l'Université de Paris 5, d'avoir accepté d'être les rapporteurs de ce manuscrit. Leurs remarques et suggestions lors de la lecture de mon rapport m'ont permis d'apporter des améliorations à la qualité de ce dernier.

Merci également à MM. les professeurs Antoine Tabbone de l'Université de Nancy 2 et Umapada Pal de l'Institut de la statistique d'Inde, pour avoir accepté d'examiner mon mémoire et de faire partie de mon jury de thèse.

Merci à Jean-Marc Ogier et Jean-christophe Burie, pour avoir accepté de diriger et encadrer cette thèse, leurs aides précieuses m'ont été indispensable sur le plan scientifique. Je tiens également à les remercier pour la confiance et la sympathie qu'ils m'ont témoignées au cours de ces quatre années de thèse.

Je tiens à remercier aussi Hélène Noizet d'avoir participé à mon jury de thèse, son aide sur le plan historique et ses grandes qualités humaines m'ont été d'un grand soutien dans cette thèse.

A Rémy Mullot, responsable du laboratoire L3i, de m'avoir accueilli au sein de son équipe.

Merci également à l'équipe du projet ALPAGE, qui m'a permis d'effectuer cette thèse dans de très bonnes et très agréables conditions de travail en me plongeant dans un problème passionnant, l'interprétation d'images de plans cadastraux.

Je tiens enfin à remercier les amis, thésards ou non qui m'ont aidé au cours des quatre ans de cette thèse : François pour la grande qualité de son humour, Guillaume pour sa vision différente du monde qui nous entoure, Patrice pour ses bons petits plats, ses calembours et son rire inoubliable, Abdallahi, Antoine et Cyril toujours prêts à aider, Thomas pour son angélisme et sa candeur et Mickael pour son amour de la précision. Je n'oublie pas non plus les autres parmi lesquels je remercierai plus particulièrement les amis du midi toujours très souriants et la team des basketteurs pour nos matchs palpitants.

Un remerciement particulier pour Franck qui a en grande partie contribué à la préparation audio-visuel de ma soutenance.

Une attention amusée et tendre pour les auteurs et artistes qui ont participé d'une manière ou d'une autre à construire mon cheminement de pensée. Coté livres : Isaac Asimov et ses recueils sur les robots, Marcel Proust et sa recherche du temps perdu, Bernard Werber et ses fourmis, Denis Guedj et son théorème du perroquet et Gary Hayden pour ce livre qui n'existe pas ... Coté musiques: Jill is lucky, JB's People, Glen Hansard et Markéta Irglová, Dire Straits, Mavin Gaye, Monsieur Roux ... et tous les autres cotés aussi.

Finalement j'adresse un grand merci à toute ma famille qui a toujours été présente lorsque j'en ai eu besoin, en particulier à mon frère, à ma sœur, à mon père et à ma mère.

Résumé en français

Les travaux présentés dans ce mémoire abordent la problématique de l' interprétation de plans cadastraux colorés anciens. Dans ce contexte, ils se trouvent à la confluence de différentes thématiques de recherche telles que le traitement du signal et des images, la reconnaissance de formes, l'intelligence artificielle, la communication Homme / Machine et l'ingénierie des connaissances. En effet, si ces domaines scientifiques diffèrent dans leurs fondements, ils sont complémentaires et leurs apports respectifs sont indispensables pour la conception d'un système d'interprétation fiable et adaptable. Dans ce contexte pluridisciplinaire, le mémoire est organisé en 5 parties et l'articulation des différents chapitres est présentée en figure 1.

Traitement de la couleur

S'il est vrai qu'un pixel couleur pris isolément n'a que peu de sens, à contrario, la masse de million de pixels constituant une image vaut bien des milliers de mots. Le choix d'un espace couleur pertinent est un choix crucial quand il s'agit de construire des traitements d'image tels que la segmentation couleurs ou la reconnaissance d'objets. Partant de ce constat, nous abordons de façon générique la question suivante: Quel est le meilleur espace de représentation de la couleur dans un but de traitement d'image pour une image donnée? Dans cette partie, un système de sélection d'espace couleur est proposé. Partir d'une image Rouge, Vert, Bleu (RVB) chaque pixel est projeté dans un vecteur composé de 25 couleurs primaires. Ce vecteur est alors réduit à un espace couleur hybride composé des trois couleurs primaires les plus importantes. Seules trois composantes couleur sont retenues pour être conforme avec les formats standards en image. Ainsi, le paradigme est fondé sur deux principes, les méthodes de sélection de caractéristiques et l'évaluation d'un modèle de représentation. La qualité d'un espace de couleur est évaluée en fonction de sa capacité à faire des groupes de couleur homogène et par conséquent d'accroître la séparabilité des données. Notre cadre apporte une réponse au choix d'un espace de représentation significatif dédié aux applications de traitement d'image s'appuyant sur des informations couleur. Les espaces couleurs standards ne sont pas conçus pour traiter des images spécifiques (images médicales, les images de documents), de sorte qu'il existe un besoin réel pour des modèles couleurs adaptés et spécifiques.

Interprétation de plans cadastraux

Dans cette partie, une méthode d'extraction d'objets à partir de cartes anciennes couleurs est proposée. Cette méthodologie vise à localiser le texte, les quartiers ainsi que les parcelles à l'intérieur de chaque carte cadastrale. Tout d'abord, un modèle

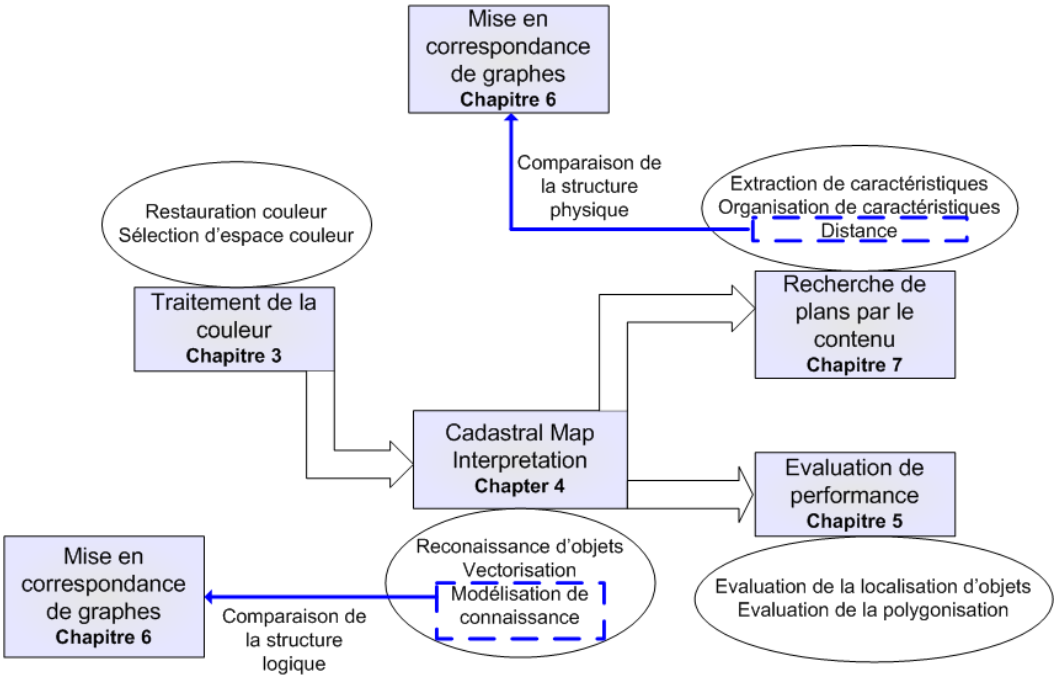


Figure 1: Organisation du manuscrit de thèse

de plan est introduit, cette représentation de la connaissance sur notre sujet a été élaborée en collaboration avec des historiens et des architectes, des experts du domaine. Puis, les aspects couleurs sont pris en compte par l'utilisation d'algorithmes de sélection d'espace couleur hybride présentés au chapitre 3. Ensuite, des traitements d'image spécifiques ont pour but de localiser les différents objets disposés dans les plans de cadastre. Ces détecteurs sont spécialement conçus pour récupérer et identifier les différents composants tels que des caractères de texte, le cadre entourant chaque plan, les quartiers, les rues, et les parcelles. Ces outils spécifiques sont exécutés séquentiellement dans l'objectif d'obtenir leurs limites. Dans une dernière phase, ces éléments sont insérés dans une représentation à base de graphe. Cette représentation structurelle est mise en concordance avec la modélisation de connaissance (méta-modèle) définie par les experts. Cette comparaison est effectuée grâce à un algorithme de mise en correspondance de graphes. La "Métamodélisation" fait référence à la construction d'un ensemble de «concepts» (classes, relations, etc) dans un domaine particulier. Un modèle est une abstraction des phénomènes du monde réel et un méta-modèle est encore une autre abstraction mettant en évidence les propriétés du modèle lui-même.

Evaluation de la vectorisation des plans de cadastre

Cette partie présente la conception d'un banc d'essai pour évaluer les systèmes de conversion Image vers Vecteur. Plus précisément, ce protocole est conçu pour évaluer la performance des systèmes de détection et d'approximation de polygones. Notre contribution est double, un algorithme de mise en correspondance d'objets pour localiser spatialement les erreurs dans les documents vecteurs, puis une distance entre graphes qui rend compte de la précision de l'approximation polygonale. L'évaluation de performance intègre de nombreux aspects et facteurs fondés sur des unités uniformes tout en restant générique et sans seuil. Ce protocole d'évaluation de performance donne une comparaison scientifique à un niveau polygone de la cohérence d'un document vectorisé. Cet outil utilise des méthodes concrètes d'évaluation de performance qui peuvent être appliquées à des systèmes complets de polygonisation. D'ailleurs, un système dédié à la vectorisation de plans cadastraux a été évalué selon ce point de vue et les résultats en termes de qualité de détection et de précision de l'approximation polygonale sont présentés dans le manuscrit. Ensuite, le comportement de notre série d'indices a été analysé en faisant augmenter le niveau de dégradation de deux jeux de tests contenant des images de documents. Par cette mise à l'épreuve d'un algorithme de polygonisation reconnu, nous démontrons que notre protocole peut révéler les forces et les faiblesses d'un système. Enfin, nous espérons que ce protocole d'évaluation permettra d'évaluer sous un autre angle, plus proche de la sémantique et des objets manipulés par l'homme, les outils de retro-conversion de documents de la communauté de la reconnaissance de graphiques.

Une mise en correspondance de graphes et une distance entre graphes fondées sur l' assignement de sous-graphes

Au cours de la dernière décennie, l'utilisation d'objet représenté à base de graphe a considérablement augmenté. En effet, la représentation d'objet par le biais de graphe a de nombreux avantage sur la représentation traditionnelle par des vecteurs de caractéristiques. Par conséquence, les méthodes pour comparer les graphes sont devenus de premier intérêt. Dans cette partie, une méthode de mise en correspondance de graphes et une distance entre graphes relationnels attribués sont définies. Les deux approches sont fondées sur la décomposition en sous-graphes. Dans ce contexte, les sous-graphes peuvent être considérés comme des caractéristiques structurelles extraites à partir d'un graphe donné, leur nature leur permet de représenter l'information locale d'un nœud racine. Étant donné deux graphes g_1, g_2, la mise en correspondance peut être exprimé comme l'adéquation minimale entre les sous-graphes de g_1 et les sous-graphes de g_2 et ceci en respectant une fonction de coût. Cette métrique entre les sous-graphes découle de distances entre graphes faisant références dans le domaine. Par des expérimentations sur 4 jeux de données différents, la distance induite par mise en correspondance de graphes a été appliquée pour mesurer l'exactitude de l'appariement de graphes. Enfin, nous démontrons une

importante accélération par rapport aux méthodes conventionnelles tout en gardant une précision pertinente.

Recherche de plans cadastraux par le contenu

Traditionnellement, lorsqu'un développeur d'applications veut interroger par l'exemple un entrepôt d'images de scènes naturelles, les méthodes classiques se contenteront de comparer, au niveau pixel, l'image requête à toutes les images du corpus. Quand on parle des images de documents, le scénario est assez différent car nous sommes en présence d'images créées par l'homme pour l'homme. Cela fait une énorme différence qui permet des comparaisons et une exploration à des niveaux supérieurs. Deux autres stades voient le jour: (a) Les images de documents peuvent être vectorisées, et la collection d'images peut être donc interrogée au niveau vecteur. (b) Les images de documents possèdent une sémantique forte et une navigation utilisant une représentation structurelle de la connaissance est devenue possible.

Abstract

This thesis tackles the problem of technical document interpretation applied to ancient and colored cadastral maps. This subject is on the crossroad of different fields like signal or image processing, pattern recognition, artificial intelligence, man-machine interaction and knowledge engineering. Indeed, each of these different fields can contribute to build a reliable and efficient document interpretation device. This thesis points out the necessities and importance of dedicated services oriented to historical documents and a related project named ALPAGE. Subsequently, the main focus of this work: Content-Based Map Retrieval within an ancient collection of color cadastral maps is introduced. The organization of this thesis paper is in five chapters. The interaction between chapters is illustrated in figure 2 and a short description of each chapter is put forward as follows:

Introduction

Chapter 1 gives the introduction to the project and provides overall concept of this thesis. We introduce a general aspect of document image analysis, the necessities and importance of historical documents and the related project named ALPAGE. Next, we focus on coloured cadastral maps and define the scope and objectives of this study.

Color Map Understanding: State of the art

In the present chapter, we discuss how to bring an automation of the single modules of a Raster to Vector conversion system to fullest possible extent. GIS can be categorized in two types, analytical and register GIS. Analytical GIS do not require an extremely high level of geometric exactness in the cartographic materials, whereas they do require fast processing of a large number of vector layers. An example of analytical GIS is GIS developed to solve territorial planning problems, while an example of a register GIS is GIS developed for a cadastral system. Mainly, we focus on the case of register GIS with the aim is to describe and highlight the global concepts and crucial points of our problem.

Color Processing

The choice of a relevant color space is a crucial step when dealing with image processing tasks (segmentation, graphic recognition...). From this fact, we address in a generic way the following question: What is the best representation space for a

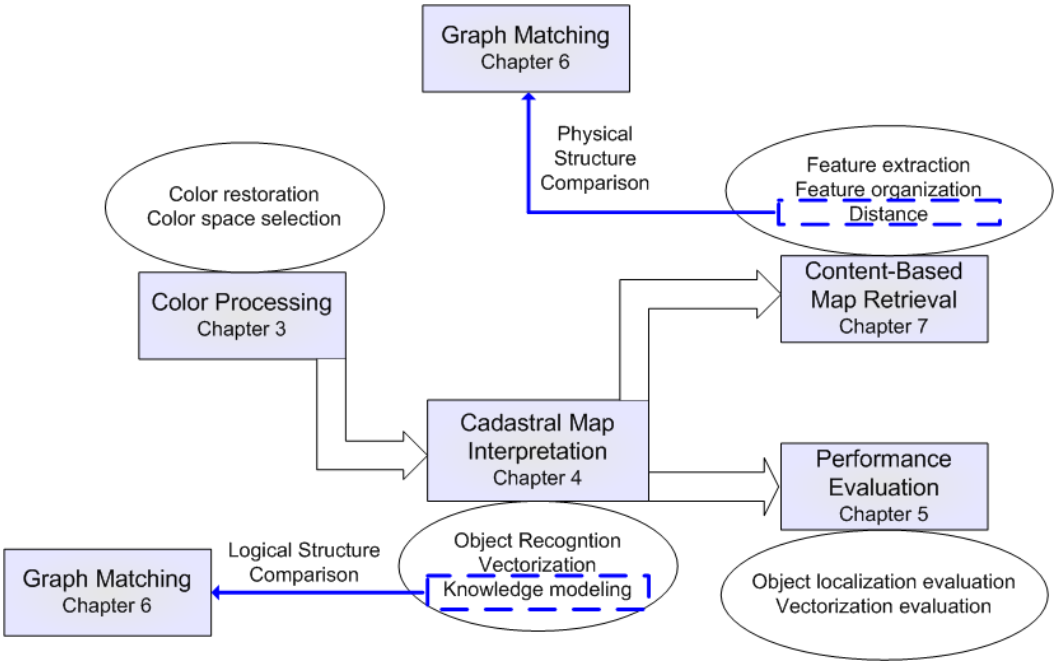


Figure 2: Thesis organization

computational task on a given image? In this chapter, a color space selection system is proposed. From a RGB image, each pixel is projected into a vector composed of 25 color primaries. This vector is then reduced to a Hybrid Color Space made up of the three most significant color primaries. Only three color components are retained to be conformed with standard image formats. Hence, the paradigm is based on two principles, feature selection methods and the assessment of a representation model. The quality of a color space is evaluated according to its capability to make color homogeneous and consequently to increase the data separability. Our framework brings an answer about the choice of a meaningful representation space dedicated to image processing applications which rely on color information. Standard color spaces are not well designed to process specific images (ie. Medical images, image of documents) so a real need has come up for a dedicated color model.

Cadastral map interpretation

In this chapter, an object extraction method from ancient color maps is proposed. It consists of the localization of frame, text, quarters and parcels inside a given cadastral map. Firstly, a model of cadastral map is introduced; this knowledge representation was elaborated in collaboration with historians and architects, experts in this domain. Secondly, the color aspect is inherited from the color restoration algorithm and the selection of a relevant hybrid color space presented in chapter 3. Thereafter, dedicated image processing aim at locating the various kinds of objects

laid out in the raster. These especially designed detectors can retrieve different components such as characters, streets, frame, quarters and parcels. These specific tools are run successively in the objective to identify boundaries of the different elements. In a last phase, these elements are put into a graph-based representation to be further compared with the meta-model defined by the experts. This comparison is carried out thanks to a graph matching algorithm. "Metamodeling" is the construction of a collection of "concepts" (things, terms, etc.) within a certain domain. A model is an abstraction of phenomena in the real world; a metamodel is yet another abstraction, highlighting properties of the model itself.

Evaluation of Cadastral Map processing

This chapter presents a benchmark for evaluating the Raster to Vector conversion systems. The benchmark is designed for evaluating the performance of graphics recognition systems on images that contain straight lines (segments) and polygons (solid) within the images. Our contribution is two-fold, an object mapping algorithm to spatially locate errors within the drawing, and then a cycle graph matching distance that indicates the accuracy of the polygonal approximation. The performance incorporates many aspects and factors based on uniform units while the method remains not rigid (threshold-less). This benchmark gives a scientific comparison at two levels of coherency and uses practical performance evaluation methods that can be applied to complete vectorization systems. It is also the opportunity to compare our unsupervised evaluation method defined in chapter 4 with a ground-truth based one. Our system dedicated to cadastral map vectorization was evaluated under this benchmark and its performance results are presented in this chapter. We hope that this benchmark will help assess the state of the art in graphics recognition and highlight the strengths and weaknesses of current vectorization technology and evaluation methods.

A Graph Matching method and a Graph Machting Distance based on probe assignments

During the last decade, the use of graph-based object representation has drastically increased. As a matter of fact, object representation by means of graphs has a number of advantages over feature vectors. As a consequence, methods to compare graphs have become of first interest. In this chapter, a graph matching method and a distance between attributed graphs are defined. Both approaches are based on subgraphs. In this context, subgraphs can be seen as structural features extracted from a given graph, their nature enables them to represent local information of a root node. Given two graphs G_1,G_2, the univalent mapping can be expressed as the minimum-weight subgraph matching between G_1 and G_2 with respect to a cost function. This metric between subgraphs is directly derived from well-known graph distances. In experiments on four different data sets, the distance induced

by our graph matching was applied to measure the accuracy of the graph matching. Finally, we demonstrate a substantial speed-up compared to conventional methods while keeping a relevant precision.

Content-Based Map Retrieval

Traditionally when facing a warehouse of natural scenes to be queried by examples; Conventional methods would jut look at the system level comparing the query images to all the images within the corpus. By system level, we mean the pixel image in its self sufficient way, pixels or a gathering of pixels. When talking about images of documents, the scenario is fairly different because we are dealing with images created by humans and dedicated to humans. This makes a huge difference and allows comparisons and an exploration at higher levels. Two more stages can be drawn : (a) Image of documents can be meaningfully vectorized, and the collection can be addressed thinking at the vector level. (b) Document images have a strong semantic and a navigation using a model representation has come true.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Forewords

This chapter provides the overall concepts of the thesis. It starts from introducing a general aspect of document image analysis. Then, it points out the necessities and importance of dedicated services oriented to historical documents and a related project named ALPAGE. Subsequently, the main focus of this work: Content-Based Map Retrieval within an ancient collection of color cadastral maps is introduced. The scope, objectives and organization of this thesis are provided at the end of this chapter.

1.2 Document Image Analysis

With the improvement of printing technology since the 15th century, there are a huge amount of printed documents published and distributed. The printed book quickly becomes a regular object in the world. By 1501 there were 1000 printing shops in Europe, which had produced 35,000 titles and 20 million copies[1]. Since that time, a vast amount of books have been falling into decay and degrading. This means not only the books themselves are disappearing, but also the knowledge of our ancestors. Therefore, there are a lot of attempts to keep, organize and restore ancient printed documents. With the best digital technology, one of the preservation methods of these old documents is the digitization. However, digitized documents will be less beneficial without the ability to retrieve and extract the information from them, which could be done by using techniques of document analysis and recognition.

[1] http://communication.ucsd.edu/bjones/Books/printech.html

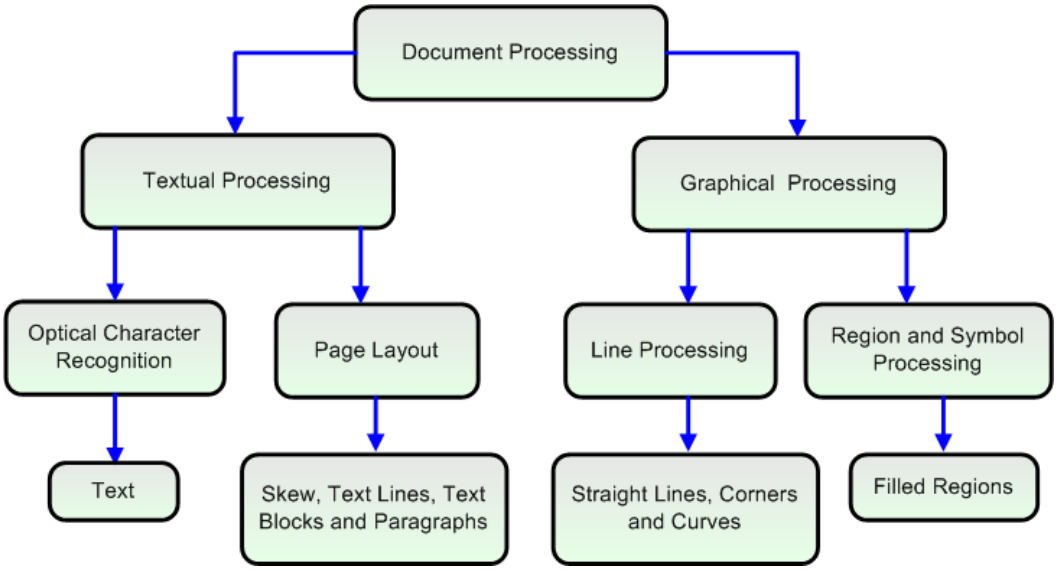


Figure 1.1: Hierarchy of document image processing; adapted from [Kasturi 2002]

Document analysis or more precisely, document image analysis (DIA), is the process that performs the overall interpretation of document images. [Nagy 2000] gave the short definition of DIA as follow. *"DIA is the theory and practice of recovering the symbol structure of digital images scanned from paper or produced by computer"*. DIA is the subfield of digital image processing that aims at converting document images to symbolic form for modification, storage, retrieval, reuse, and transmission. In practice, a document analysis system performs the basic tasks of image segmentation, layout understanding, symbol recognition and application of contextual rules in an integrated manner. The objective of document image analysis is to recognize the text and graphics components in images and to extract the intended information as a human would. Two components of document image analysis i.e. textual processing and graphical processing can be defined (see figure 1.1).

Figure 1.2 illustrates a common sequence of steps in document image analysis. After data capturing, the image undergoes pixel-level processing and feature analysis, then text and graphics are treated separately for recognition of each.

In view of an analysis of ancient documents, it requires the same concept as mentioned above. However, the task is more challenging, the text/graphic separation question is more delicate to address. A strict data flow separation between the text and graphic processing chains assumes that a "good" text/graphic segmentation within the document is always possible. This hypothesis is not always easy hold when documents become denser and denser. This is because ancient documents hold more significance and more complexities than normal one. Firstly, ancient documents have historical meanings. Some negligible details in recent document

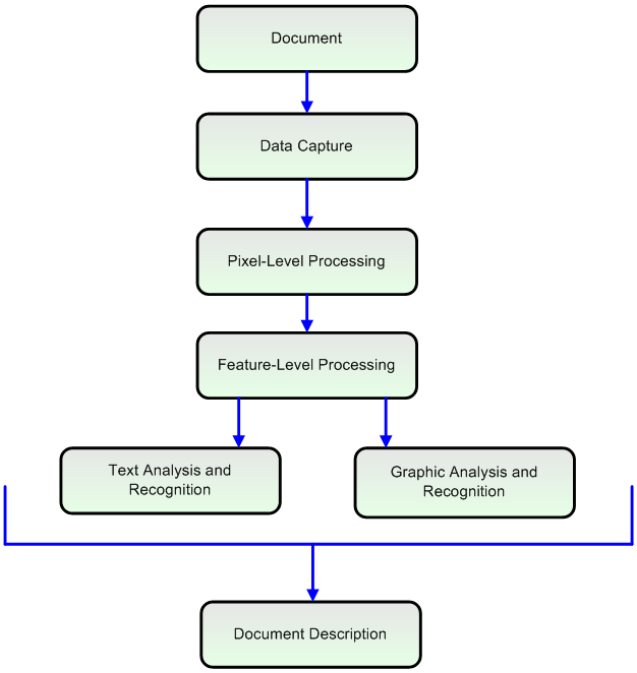


Figure 1.2: A sequence of steps for document analysis; adapted from [Kasturi 2002]

structure could be very important in historical domain. Secondly, most ancient documents are degraded and susceptible to decay over time. Thus the digitization process has to be handled carefully and precisely with higher resolution. In addition, due to the large volume of ancient documents, the capturing device must support both qualitative and quantitative problems. Thirdly, the layouts or structures of ancient documents are organized differently. There are complex arrangements of texts and graphics including the styles and fonts used in publishing. Finally, ancient documents are the targets of different users; starting from general users to experts, consequently, document understating systems should take this fact into account. This raises the question of how to structure the information extracted from ancient documents to be able to respond to different user requirements. Usages and user needs are quite hard to circumscribe due to their plurality. Usage can either individual or collective which condition the way to structure the information.

The effort to manage ancient documents so far seems to be in progress. In France, firstly, this idea was generally fragmented. There was a lack of global and strategic management tools and no common policies on handling of ancient document resources and on setting priorities in management. This results in the threat of waste in resources, efforts and investments. Digitization is also costly and needs huge budgets, often based on public funding. Fortunately, from the support of French government and the collaboration of many research laboratories, the projects called MADONNE and NAVIDOMASS were set up for the purpose of preserving and exploiting ancient documents. These pioneer projects opened the way to more and more challenging relations between ICT-HSS communities

(Information & Communications Technology - Humanities and Social Sciences), for instance, the ALPAGE project came to birth into this frame of mind. French and European initiatives such as the french digital library GALICA[2], and the British Library[3] show the engagement for this cause. Especially, to get out of the recession, a huge investment has been approved by the french government. A digitalization program supported by a 750 M€ fund is on the way. This effervescence denotes the matter of the digitization of our cultural heritage.

1.3 Historical Document Project: ALPAGE

ALPAGE[4] stands for diachronic analysis of the Paris urban area: a geomatic approach (in french, AnaLyse diachronique de l'espace urbain PArisien: approche GEomatique). It is a research project funded by the French government in the French National Research Agency (L'Agence Nationale de la Recherche)[5]. The aim of this project is to design methods for going beyond plain digitization projects of historical documents. Previously, similar projects tend to yield large databases of images with less interest in structuring, indexing and navigating on them. This is why the ALPAGE project tries to investigate the use of document image analysis methodology for providing useful indexing and browsing features in these large collections.

ALPAGE is thus focused on vectorization, indexing, organization and incremental enrichment of heritage data, in order to provide general and reliable services to users, including researchers in human and social sciences, and to work towards interoperability of data and browsing tools. The ALPAGE project was launched in 2006 for three years. The project collaboration gathers 4 laboratories.

- LAMOP from the Paris 1 University, carrying the project, which is composed of historians and medievist archeologs that are specialists of Paris.
- LIENSS from La Rochelle University grouping geomaticians
- ArcScan from Paris 10 University, grouping geomaticians that are able to deal with GIS in Archeology. Archaeologists and Art historians that are Paris experts.
- L3i from La Rochelle University, grouping computer sciences researchers, specialized in pattern recognition and vectorization.

Concerning, document understanding, research themes in Alpage project are in four directions as shown in Figure 1.3. They are (1) color processing, (2) document layout analysis, (3) graphics vectorization, (4) content-based image retrieval (CBIR).

[2] http://gallica.bnf.fr/
[3] http://www.bl.uk/
[4] http://lamop.univ-paris1.fr/alpage/index.php
[5] http://www.agence-nationale-recherche.fr

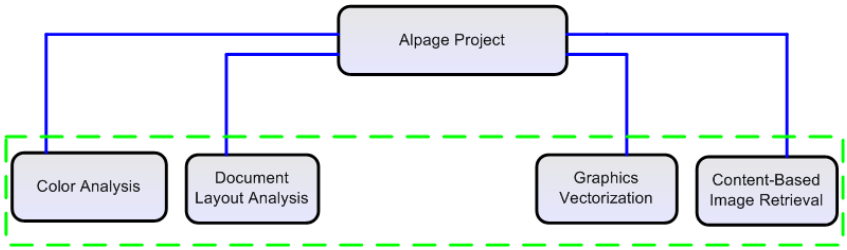


Figure 1.3: Research themes in Alpage project

1.3.1 Scientific background and objectives

This project aims at implementing mutualized working tools for both ICT-HSS communities, allowing to develop research relating to Parisian urban space, namely, PRAI software (Pattern Recognition and Artificial Intelligence) adapted to ancient cadastral maps, and a GIS (Geographical Information System) including cadastral and historical layers. It is a new approach to the urban environment, truly integrating the spatial dimension, which could be implemented thanks to the contributions of recent disciplines such as computer vision, geomatic and historians. The choice of Paris is explained both by the interest that the French capital city inspires in the scientific communities and above all by the extraordinary documentary potential: historical documents indeed exist, yet they were insufficiently utilized up to this point due to the lack of appropriate tools. The GIS allows starting from the semantic data in order to consequently consider the spatial dimension of the objects. The GIS also allows considering the urban space as a source, from which one can generate a historical discourse, having at the same time political, pedagogic and scientific implications. The political aspect consists of contributing to the management of the Parisian patrimony, allowing the public services in charge of the management operations to better integrate the patrimonial dimension of the examined projects. From a pedagogic perspective, this tool will be used as an aid to teaching in the concerned universities and schools. In the long run, we expect to ensure a broader diffusion of this tool via the internet, thanks to flexible and adapted formats. Having in its origins the will to develop the interdisciplinarity within the HSS and to set up the scientific synergies ICT/HSS, the objectives are numerous:

- To build innovative pattern recognition tools adapted to the ancient cadastral maps
- To produce inventories of Parisian urban space according to a variable scale
- To integrate the geographical and physical dimension in the societies/environments relations
- To use explanatory models in order to explain the geographical distribution of objects
- To analyze the morphology of lots/parcels at the level of the city.

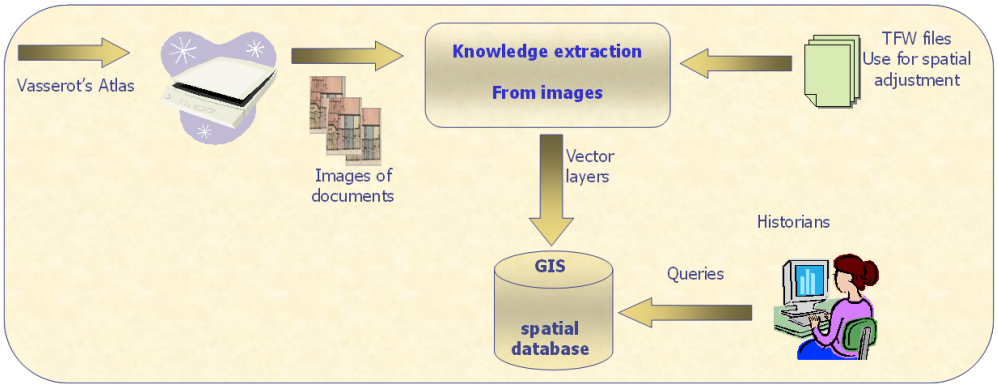


Figure 1.4: Global Methodology

1.3.2 Global Methodology

The project consists in implementing georeferenced cadastral layers, from which layers of a historical type could be created, in particular the historical topography and the medieval and modern administrative districts. The utilized source maps correspond to the land register according to small blocks found in the Vasserot Atlas (1810-1836) for the oldest twelve districts, the aim being to cover the Parisian space that is delimited by the farmer-general wall. The data processing specialists from La Rochelle and geomaticians of LIENSS work together to set up the cadastral layers: they georeference, assemble and vectorize the various raster images issuing from the source maps. This process is illustrated by the figure 1.4. In parallel, the historians primarily medievalists for now work together with the geomaticians to set up the conceptual model of data and the historical layers. This work takes into account the experiences that have already been led, especially at the Cultural Ministry (CNAU : National Urban Archeological Center).

1.4 Cadastral Map

This section provides some fundamental information related to the definition of a special image called Cadastral Map, its history and its classification. This special image is the main focus of our study.

The extraordinary potential of the automatic analysis of color documents brings new interests and represents a real challenge since color has always been considered as a strong tool for information extraction [Dorin Comaniciu 1997]. As mentioned, earlier, in the context of the project called “ALPAGE”, we are considering the digitalization of ancient maps. In this ALPAGE project, we consider cadastral maps from the 19th Century (called “Atlas VASSEROT"), on which objects are drawn by using color to distinguish parcels for instance. This project deals with the classical graphic recognition problems, to which are added difficulties due to the presence of colors and strong time due degradations of relevant information : color degradation, yellowing of the paper, pigment fading... In the context of this multi-disciplinary project,

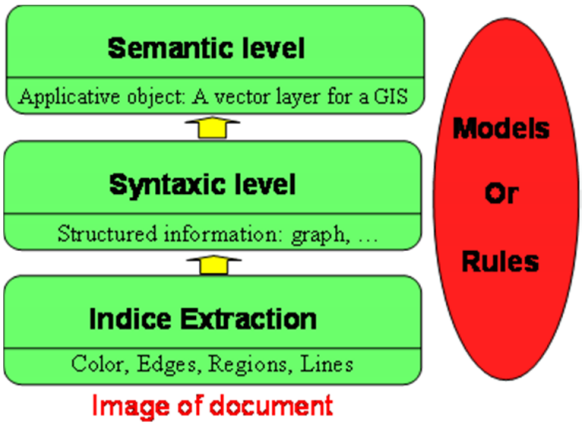


Figure 1.5: Architecture of a graphic document analysis system

the idea is to provide strategic information for historians, or students, what means that the purpose is to propose a set of processing allowing to segment/recognize all the objects of the documents. In such a topic, the number of handled objects can be counted by million. This volume of data leads to the rise of new services as intelligent indexation, document browsing and content searching. If the analysis of a given document was reduced in the digitalization of the paper document to a "bitmap" image, the problem would be commonplace. Actually the subjacent scientific problems are very complex because the objective is much more ambitious, the conversion of the paper document into its semantic interpretation [BELAID A. 1992]. The concept of retro-conversion is a semantic digitalization, from elementary data and contextual information the analysis is carried out through a color graphic recognition process where the aim is to build structured information dedicated to a GIS. A classical ascending approach from pixel to object calls various low level tools such as color segmentation or line tracking while at the top, high level methods allow the integration of a priori knowledge bringing a contribution to the interpretation process with an aim of archiving information (figure 1.5) [Lladós J. 2003].

An example of cadastral map is shown in figure 1.6. A straightforward comment points out the color specificity of these documents, hence, we need to consider the color meaning to extract cadastral information (ie: a parcel) and a closer look is given to color representation. Consequently, in this thesis, we propose a general architecture to take into account color information embedded into graphic documents in the objective to build a relevant Content-Based Map Retrieval (CBMR) system. An overview of our framework is displayed in figure 1.7 and our method relies on three major steps as follows:

1. Firstly, a preprocessing stage aims at preparing the data, so, this includes:

   (a) Finding the best color model in terms of distinction between different colors. We assume that the choice of an efficient color model will be

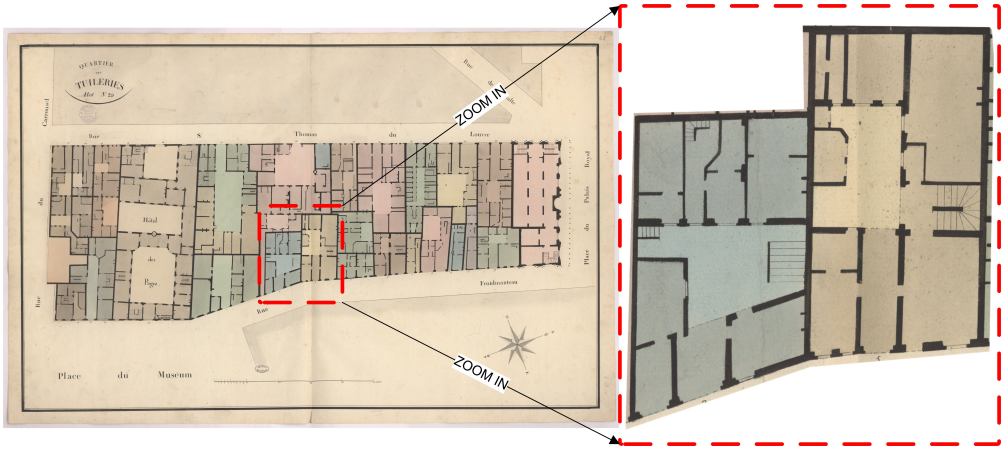


Figure 1.6: A sample of cadastral map.

decisive since the performance of any color-dependent system is highly influenced by the color model it uses.

(b) Removing undesirable data to preserve the true diamond (data mining), the parcels within the map.

2. Secondly, a vectorization step and its performance evaluation provide, respectively, vectorial objects to be inserted into the GIS and tools for comparing maps. The three sub-steps, composing this second part, are as follows :

   (a) A color segmentation approach dedicated to documents is presented; it is inspired by graphic construction rules of cadastral maps.

   (b) Digital curves approximation aims at transforming pixels to vectors.

   (c) Performance evaluation of the vectorization and a vectorial dissimilarity measure between maps.

3. Finally, a Content-Based Image Retrieval system adapted to cadastral maps is presented. This so called Content-Based Map Retrieval (CBMR) application lies on a vectorial distance between maps. The vectorization stage feeds the CBMR process, thus, it provides a morphological analysis and it makes possible, from a query image, to find similar cadastral maps.

   (a) Images of maps are like no others and a CBMR approach should take profit of the intrasectoral spatiality of such a document. In this way, a graph-based representation is more likely to perform better, consequently, the CBMR system should involve a graph distance when searching by similarity a map.

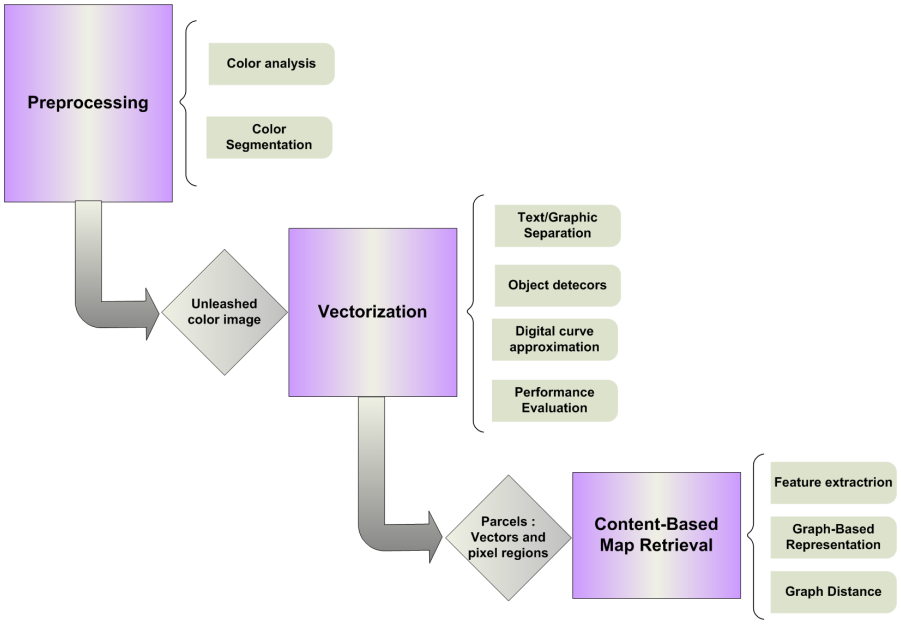


Figure 1.7: Overall methodology of our system.

1.5 Conclusion

This thesis deals with a problem of graphic detection and retrieval, specially focused on an ancient and colored cadastral maps. This thesis also belongs to a part of a historical document project named ALPAGE, which has as main objective to preserve and derive benefits from ancient documents. The main objectives of this thesis are to propose a vectorization and pattern recognition framework to a database of maps and to develop a CBIR system in order to provide a reliable accessibility and functions to that database for interested users.

The organization of this thesis paper is in five chapters. The interaction between chapters is illustrated in figure 2 and a short description of each chapter is put forward as follows:

Chapter 1 gives the introduction to the project and provides overall concept of this thesis. We introduce a general aspect of document image analysis, the necessities and importance of historical documents and the related project named ALPAGE. Next, we focus on colored cadastral maps and define the scope and objectives of this study.

Chapter 2 concerns the state of the art. This chapter reviews the literatures related to historical document analysis. The literature review gives the fundamental knowledge, global aspect and terminology, and presents recent ideas and techniques that are useful to our work.

Chapter 3 is dedicated to the color processing aspects. This project deals with the classical graphic recognition problems, to which are added difficulties due to the presence of colors and strong time due degradations of relevant information: color degradation, yellowing of the paper, pigment fading... Especially, this chapter introduces some principles on color restoration and provides a guide tour on color representation and its subsequent selection.

Chapter 4 deals with the problem of the extraction of information from cadastral maps. This part aims at presenting methods and low-level algorithms involved into the quarters and parcels retrieval. Thereafter, parcels information is structured into a graph-based representation. These data will be used to feed a CBIR stage.

Chapter 5 aims at assessing the vectorization process. The question of performance evaluation is raised and a set of metrics is defined. Theses indices reveal errors that occur in a raster to vector conversion.

Chapter 6 presents graph matching and graph classification methods. Cadastral maps are modeled by attributed relational graphs taking into account the relationships between parcels, hence, the question of finding similarities between maps turns into a graph matching problem. This chapter gives theoretical and experimental basements of graph mining methods considered for that purpose.

Chapter 7 addresses the Content-Based Image Retrieval (CBIR) topic in a general point of view (in a general way). A discussion about the suitability of structural approaches for a CBIR system is given. Finally and more specifically, a CBIR application based on polygon features is described. From a query map, the cadastral map collection browsing is aided by computer, in the objective to retrieve the most similar cadastral maps from a morphological point of view. In this way, the CBIR paradigm is derived to gives birth to what we call Content-Based Map Retrieval (CBMR).

Chapter 8 discusses and draws a conclusion of this thesis. Finally, we give perspectives and introduce possible future works.

Chapter 2

Color Map Understanding: State of the art

Contents

2.1 Forewords

The problem of automatic raster to vector (R2V) conversion is taken steadfast attention by researchers and software developers during last two decades. Numerous attempts to solve this problem have mainly originated from emerging area of automatic Geographical Information Systems (GIS). Unfortunately, completely automatic conversion system appears really challenging to be achieved perfectly, so, some authors suggested putting the operator into the loop even into the center of a conversion system. Thus, the problem of correct task division between the human and machine can be also stated [Levachkine 2003], [Levachkine S. 2000].

In the present chapter, we discuss how to bring an automation of the single modules of a R2V conversion system to fullest possible extent. GIS can be categorized in two types, analytical and register GIS. Analytical GIS do not require an extremely high level of geometric exactness in the cartographic materials, whereas they do require fast processing of a large number of vector layers [E-Cognition] [R2V]. An example of analytical GIS is GIS developed to solve territorial planning problems, while an example of a register GIS is GIS developed for a cadastral system. Mainly,

we focus on the case of register GIS with the aim is to describe and highlight the global concepts and crucial points of our problem. The chapter is divided into three parts: 1) in section 2.2, we analyze the general concepts of automation of a R2V conversion system adapted to color cartographic materials; 2) The section 2.3 gives a closer look to technical documents and briefly reports the main approaches in this topic; 3) in section 2.4 we present an overview of our approach, a Model Driven Raster to Vector conversion system. In conclusion, we emphasize the role of operator and knowledge in automation of the conversion process.

2.2 Cartographic Raster to Vector Conversion

Two main methods are currently used for map vectorization [Levachkine S. 2000]: (V1) Paper map digitization by electronic-mechanical digitizers, and (V2) Raster map (a map obtained after scanning of the paper original) digitization. The digitizing of paper maps cannot be automated; hence the only practical approach to design R2V conversions is the development of methods and software to automate vectorization of raster maps. Raster map digitization approaches can be divided into four intersecting groups [Levachkine S. 2000]: (D1) Manual; (D2) Interactive; (D3) Semi-automated, and (D4) Automatic. In practice, (D1) = (V1). In the case of "punctual" objects, the operator visually locates graphic symbols and fixes their coordinates. In the case of "linear" and polygonal objects, the operator uses rectilinear segments to approximate curvilinear contours. The manual digitization rate is one to two objects per minute. Interactive digitization uses special programs, which, once the operator indicates the starting point on a line segment, automatically follow the contours of the line (tracing). These programs are capable of tracing relatively simple lines. If the program cannot solve a graphical ambiguity on the raster map, it returns a message to the operator (alarm). Recently, vector editors capable of carrying out this digitization process have appeared, reducing the time consumption by a factor of 1.5 to 2. These can be called semi-automated systems [Levachkine 2004] [Levachkine 2003] [Levachkine S. 2000] [E-Cognition] [R2V].

In theory, automatic vector editors automatically digitize all objects of a given class, leaving to the operator the error correction in the resulting vector layers. Some vector editors use this approach [E-Cognition] [R2V] [Benz U.C. 2003]. However, in practice, the high error level resulting from any small complication in the raster map means that alternative methods should be sought to reduce the huge volume of manual corrections.

Computer aided correcting system lies on two points: (1) An automatic detection of ambiguities (low confidence zones) and (2) a pertinent graphical user interface in order to locally correct errors without perturbation of the global data coherency.

A system approach can enhance the outcome not only of processing (map recognition) but also pre-processing (preparation of paper maps and their corresponding

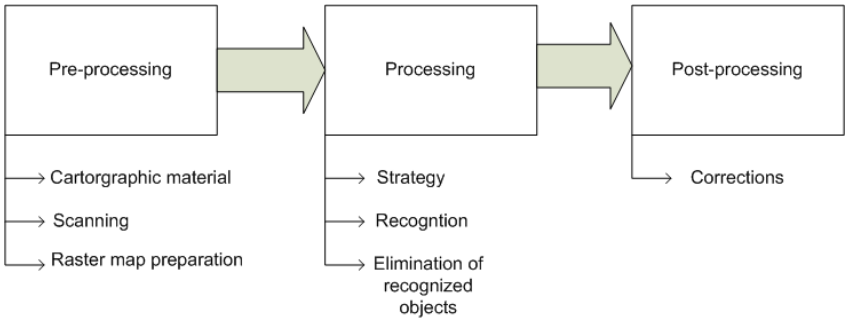


Figure 2.1: Main processing steps required in raster to vector conversion.

rasters) and post-processing (final processing of the results of automatic digitization). Thus, in the following three sections one shall consider these processing stages in the context of system approach. Figure 2.1 depicts an overview of the complete process.

2.2.1 Pre-processing

The main goal of pre-processing is to prepare raster cartographic images in such a way as to simplify them and increase the reliability of their recognition in the automatic system. The proposed sequence of operations for the preparation of raster maps for automatic recognition is:
Pre-processing=

1. Preparation of the cartographic materials for scanning
   (a) Restoration
   (b) Copying
   (c) Increasing the contrast of image objects
2. Scanning
   (a) Test scanning
   (b) Definition of the optimal scanning parameters
   (c) Final scanning
   (d) Joining raster facets
   (e) Correction of the raster image geometry by the reference points
3. Preparation of the raster maps for recognition of the cartographic objects
   (a) Edition of raster map
   (b) Elimination of the map notations and map legend
   (c) Elimination of the artificial objects and restoration covering by them images

(d) Restoration of the topology of cartographic images in pixel level

(e) Separation of basic colors of the graphical codification on a raster map

(f) Restoration of the color palette of a raster map

(g) Stratification of raster map

(h) Stratification by reduced color palette

(i) Logical stratification of the cartographic objects

The comments on all pre-processing steps can be found in [Levachkine 2004]–[Levachkine S. 2000]. Concerning this stage, let us discuss only the most important points that expose clearly our problem.

Increasing the contrast of image objects. To simplify the process of vectorization, objects of the same class can be highlighted on a copy of the map using a contrasting color. Typically such marking involves enclosing linear objects such as streets, or inside rooms. In practice, outlines of polygonal objects, which do not have explicit borders (such as water well, stairs, etc.), and are delineated only by dashed or patterned lines, should be drawn in. In particular, various polygonal objects may overlap, one represented by color, another outlined by a dashed line, and a third by patterned lines; in such cases, the objects should all be outlined explicitly.

Stratification of the raster map. A raster map, considered as a unified heterogeneous graphical image, is suitable for parallel human vision. In contrast, raster images, containing homogeneous graphical information, are suited to consecutive machine vision. Two approaches can be used for the stratification of the original raster map: 1) stratification by reduced color palette or 2) logical stratification of the cartographic objects. In the first case, maps are derived from the original raster map which preserve only those pixels that have a strongly defined set of colors corresponding to the images of one class (for example, red and yellow, say, might correspond to the icons of two distinct parcel owners). In the second case, the map only preserves fragments of general raster image context corresponding to the locations of cartographic objects of one class.

2.2.2 Processing

The main goal of this principal stage of automatic vectorization of raster maps is the recognition of cartographic images, i.e. generation of vector layers and attributive information in electronic maps. The fundamental idea of R2V automation is the development of methods, algorithms and programs that focus only on locating and identifying specific cartographic objects that constitute the map semantic structure. Each cartographic image has its own graphical representation parameters, which can be used for automatic object recognition on a raster map. The particular attributes depend on the topological class of the object. Traditionally in GIS, vector map objects are divided into three types: points, segments and polygons, representing

respectively "punctual", "linear" and area objects. Graphical images have color, geometric (location, shape), topological and attributive (quantitative and qualitative parameters, e.g. the name of object) information, which can be merged into the concept of cartographic image.

An important element of the automation of raster map vectorization is the development of an optimal sequence of steps for cartographic image recognition, successively eliminating elements already decoded from the raster map field and restoring images, which were hidden by the eliminated elements. The basic principle of this optimized ordering is from simple to complex. Nevertheless, the possibility of using information from objects already digitized (whether manually or by an automatic system) should be provided for in the development of a recognition strategy. For example, the punctual layer of street information can be successfully used for recognition of polygonal elements of the quarters. Taking this into account, it becomes clear that street names and numbers should be retrieved before the quarters are digitized. Eliminating them from the raster map, one can use their locations and attributive data to aid in recognition of elements of the quarters. This strategy can also be considered as use of different sources of evidence to resolve ambiguities in a R2V conversion.

In other words, maximal use of already existing information (directly or indirectly related to the vectorized objects) employed as general principle of automatic cartographic image recognition can increase efficiency and reliability. Summarizing the processing of raster maps, we notice that the methods and algorithms used for this process should provide complete, even redundant cartographic image recognition (no matter a number of erroneously recognized objects), since visual control and correction of the vector layers can be carried out more quickly than manual digitation of missed objects. To conclude the discussion in this section, the process of automatic cartographic image recognition (processing) often follows this scheme: Processing=

1. Development of the strategy of automatic digitization of raster maps
2. Recognition of cartographic images
   (a) Digitization of objects which have vector analogues
   (b) Digitization of objects which have not vector analogues
   (c) Elimination of superfluously recognized objects
3. Elimination of recognized images from raster map
   (a) Restoration of image covered by recognized objects
   (b) Correction of restored image

2.2.3 Post-Processing

The main goal of the post-processing of raster maps (after map image recognition) is an automatic correction of vectorization errors. For automatic correction of digitiza-

tion, two approaches can be distinguished: 1) using the topological characteristics of objects in vector layers and 2) using the sources of evidence (textual, spatial distribution information).

The first approach is based on the fact that many cartographic objects in the map have well-defined topological characteristics, which can be used for the correction of vectorization errors. Let us give just one obvious example: Parcels. The topological characteristics of parcel systems (e.g. contour lines) are:

(a) parcels are continuous, (b) they cannot overlap each other, and (c) each parcel has at least one edge on the street side. However, in a raster map these characteristics, as a rule, may be lost due to several reasons: (i) the lines are broken where a street number for a given parcel is written, (ii) some parcels are not well drawn in high density regions, (iii) the folding of the page may corrupt the raster image, and (iv) degradations due to the storage condition can cause many defects of printing the paper maps. The contrast between the parcel color and the color of the street is not strong enough to make a clear distinction, so, the boundaries are broken and they need special consideration. These elements of the map's graphical design, if not considered as the parts of the parcel system, hinder the correct assembly of the polygons and either should be eliminated or (better) detached in a separate vector layer. They can be restored on the vector map and used for the automatic attribution of polygons assembled from the contour lines.

The second approach has proven its efficiency in toponym recognition of cartographic maps and can be also used in general R2V conversion post-processing [Gelbukh A. 2003]. For instance, it is obvious to say that close to a street number, there is a parcel. This information of relation between a text component and a parcel object can be useful. The example presented shows that the characteristics of internal structure and relationships between the vector objects can be used in automatic correction of errors of the automatic vectorization. In practice, it means the development of more specific software for automatic cadastral image recognition.

Summarizing the discussion of this section, the process of automatic correction of results of automatic cadastral image recognition (postprocessing) follows the scheme: Post-processing =

1. Correction of vector layers based on peculiarities of their internal topology
2. Correction using the sources of evidence (textual, street names or numbers)
3. Final correction of vector layers in whole electronic map system

Color cadastral map interpretation is at the edge between cartographic raster to vector conversion and technical drawing understanding. From its color aspect,

ALPAGE' cadastral maps are close to cartographic images, on the other hand, symbols composing city maps are similar to those encountered in the technical drawing domain. In fact, both paradigms share common points but are slightly different. Consequently, the next section discuses state of the art works in the field of engineering drawing vectorization.

2.3 Technical drawing understanding

The choice of an effective interpretation strategy is difficult; it must include low-level image processing, the construction of cadastral entities, and reanalysis of "contaminated" data (erroneous data but locally apparently consistent). The literature offers many different approaches to technical document interpretation. They concern mechanical engineering documents [Vaxiviere 1992], electronic diagrams [Hamada 1993], [Okazaki 1988], or utility maps [Boatto 1992], [den Hartog 1996], [Ogier 1993] (telecom, power and water networks, cadastral, etc.). Roughly, two strategies have been proposed: bottom-up and hybrid strategies.

2.3.1 Bottom-up approach

In bottom-up strategies, algorithms are performed in a fixed sequence, usually starting "low-level" analysis of the gray level or black and white image, in which primitives [Kasturi 1990b] are extracted by specialized operators. Generally, these primitives correspond to segments, associated or not to polygonization algorithms [Janssen 1997], [Kasturi 1990a], to symbols and characters [Deseilligny 1995], [Fletcher L.A. 1988], textures [Ogier 1993], circles, dashed lines, arrows, arcs, etc. In the next phase, associations between all or a part of these primitives are detected, and higher level graphical entities are constructed, guided by some a priori knowledge. This knowledge is either directly written into in the source code, or it can be declarative knowledge based on explicit rules for graphical entities. An analysis of graphical entities and their relationships allows one to propose an interpretation result, in the case of strictly bottom-up approaches such as [Boatto 1992], [Deseilligny 1993], [Kasturi 1990a], [Shimotsuji 1992], [Suzuki 1990]. The main difficulty in this kind of process is in obtaining significant graphical entities from the low-level operators and reliable association rules between each primitive in order to have a correct interpretation. In fact, these systems extract low-level primitives each in the same way, without taking into account the specificity of representation of each object. As a consequence, due to the variability of the representation and the handmade support, many situations in technical documents are difficult to solve by these algorithms. These difficulties concern the connection between different entities (e.g., text, line, texture), OCR in handwriting under multi-orientation constraint (city maps, utility map), low image quality, and variability in the representation of graphical entities. For all such strictly bottom-up systems, the main problem is related to the poor adaptation of the parameters of the extractors and to the inadequacy of operators to the local features of certain objects.

2.3.2 Hybrid approach

The second approach consists in interpreting technical documents by leading the low-level processes as a function of the context. In this type of system, two approaches constitute interesting contributions. The first is proposed by Joseph [Joseph 1992] about mechanical engineering drawing interpretation. The ANON system is based on the "cycle of perception" proposed by Neisser [Neisser 1976]. This system is structured in three layers in order to separate spatial and symbolic processing. The first is composed of a large image analysis library associated to both search-tracking functions and management processes. The information extraction is adapted to the context by the second level, "schema" (prototypical drawing construct), which receives the entities from the lower layer and interprets the result as a function of the current schema. A cycle of hypothesis verification is thus proposed by the schema to the control system (highest layer). This control system analyzes the proposition as a function of the current state of the proposed schema and eventually modifies it. The knowledge directed image analysis and the construction cycle according to the context are two interesting concepts which are applied on 15 different schema classes. In the same category, den Hartog [den Hartog 1996] proposed a mixed approach based on a top-down control mechanism associated with bottom-up object recognition. The system decomposes the binary image into primitives (and not vectors) to have a good morphological representation of the information and uses template matching to recognize each of them. Then, contextual reasoning is performed based on a loop that includes inconsistency detection and search action generation in a region of interest (ROI). The control system defines an ordered search action list to search for a specific object type in the ROI. Priorities are specified by the user to define the most important search actions and to assign priority to the relationship between objects. A test of consistency is applied to each recognized object in order to verify the hypothesis defined at the system top level as a function of knowledge of the object to recognize. The knowledge framework of the methodology relies essentially on spatial relationships between primitives, without integrating and describing hierarchical relationships. In the case of particularly complex documents, this kind of system is penalized because of the drastically increasing number of relationships and the necessity to generate new search actions for the "designed objects". Another hybrid approach is the system described in [Ogier 1998] for map interpretation. In this system, features are grouped together to constitute primitive objects then these objects are assembled together to compose a larger object in the hierarchy and the process goes until it reaches the most global object which is the map itself. In this system, at a given level in the hierarchy a consistency checking is performed. Recognized objects are analyzed to verify if they are internally and externally consistent with each other. For example, a parcel is composed of segments to set up the outline, it has a number or an arrow, and it can involve a hatched area and symbols. Internal consistency means all the component constitutes the objects are successfully detected or not, if not a forward heuristic rule is used to correct this situation by re-extracting features in this region after modifying and relaxing

the parameters of the low-level image processing tools. On the contrary, external consistency takes into account the neighborhood of the treated object. If an object has all the components and responds to the semantic of the considered level, it is defined as an internal consistent; furthermore, if all the objects adjoining it are all internally consistent, this object will equally become more reliable through the construction of the superior hierarchical level (the parcel by the block, for example). It is then called externally consistent.

After this brief review on line drawing understanding, in the next section, we describe the adaptation of the processing stages to fit the ALPAGE's problem. In a context of ancient, degraded and colored cadastral maps, the bottom-up strategy has to be adapted.

2.4 ALPAGE: Reverse engineering dedicated to ancient color maps

It is difficult for an observer to recognize an object if he does not have a mental representation of it a priori [Neisser 1976], [Neisser 1989]. The problem is similar for an artificial interpretation system, and it is necessary to integrate this notion of a model if the aim is to obtain a representation close to the one processed by the cadastral agent. In order to carry out this "modeling", we start from the principle that the whole of the graphic document relates to a specific organization of all the graphical primitives (homogeneous color regions, strokes, lines, etc.) and to the grouping rules for these graphical entities for object representation. The main difficulty with such an interpretation methodology is the administration of semantic links between objects (borders between colored zones, limits between objects, overlapping objects, etc.). With regard to the cadastral maps, this physical-logical link is relatively simple, since the main relations between the different entities are neighborhood or containment relations. From the following representation of document entities (graphical and logical points of view) and the relationships within it, the model first proposes a hierarchical structure for the document. The four levels for city maps are illustrated in figure 2.2. The difficulty to correctly interpret a document comes from noise and/or perturbations (pigment fading, storage-due degradations, ...) which corrupt the document image. Hence, images of documents are no longer conform to their model and generating rules are not valid anymore. This phenomenon is showed in figure 2.3. The direct consequence of the degradations is a misleading computer-generated model, however our approach provides the opportunity to evaluate this ambiguity at a logical level and so, to find out how different is the computer model from the original model of document. This model comparison provides an external or global consistency to verify the compliance of objects generated by low level algorithms. The knowledge checking is carried out through a graph matching viewpoint to take into account spatial relationships between objects.

As discussed in section 2.3, the cadastral map is an association of graphical

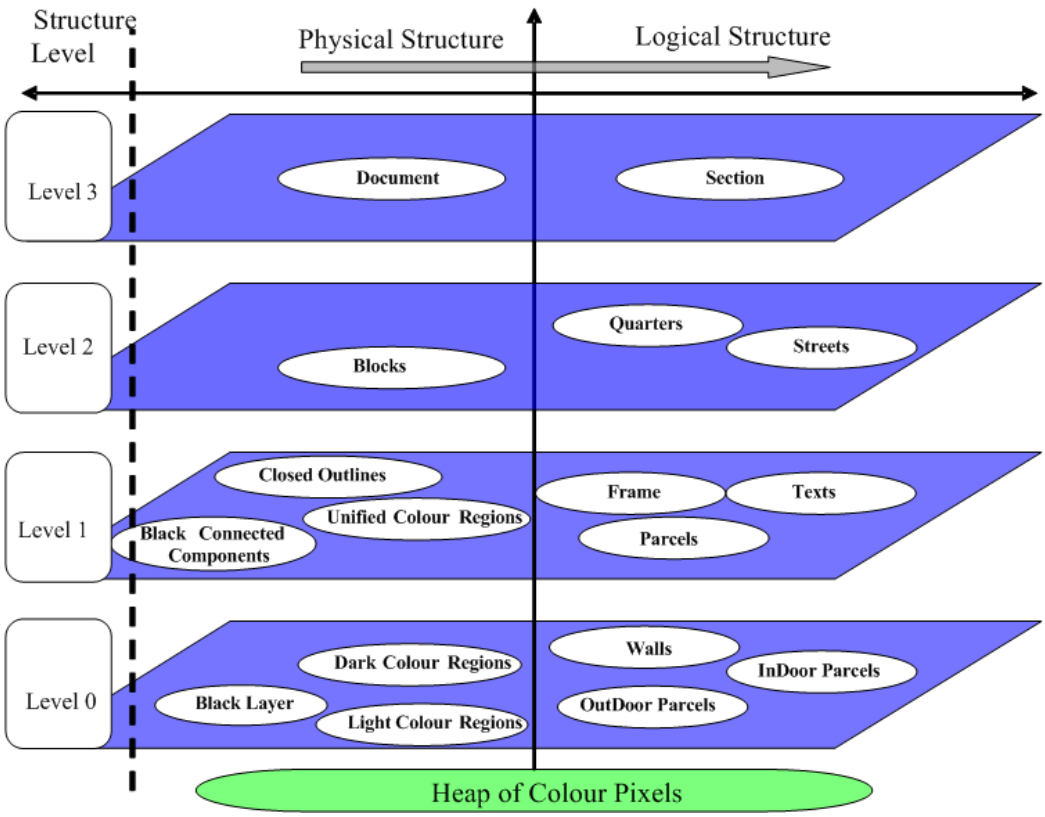


Figure 2.2: Physical and logical structures.

primitives which build a cadastral object. For the cadastral agent, these objects are parcels, streets, frame, and quarters, which are made up of lines, characters, homogeneous color regions (areas having a similar hue), symbols, etc. As can be seen in the figure 2.2, we considered four kinds of objects in relation to the French ancient city map: the parcel (of land), the street, the quarter (or district), and the section. Our bottom-up strategy consists of three stages to interpret each cadastral object. Thus, the architecture of our system is illustrated in figure 2.4. In addition, the three stages for parcel extraction from a cadastral map are put forward as follows:

1. Pre-processing

   (a) Color restoration

   (b) Color space and color representation
   Most display and color acquisition devices, such as digital cameras and scanners, have their input or output signals in the Red Green Blue (RGB) format. It is straightforwardly derived from sensor technology, the R (respectively G, B) primary in RGB corresponds to the amount of the physical reflected light in the red band (respectively green and blue bands).

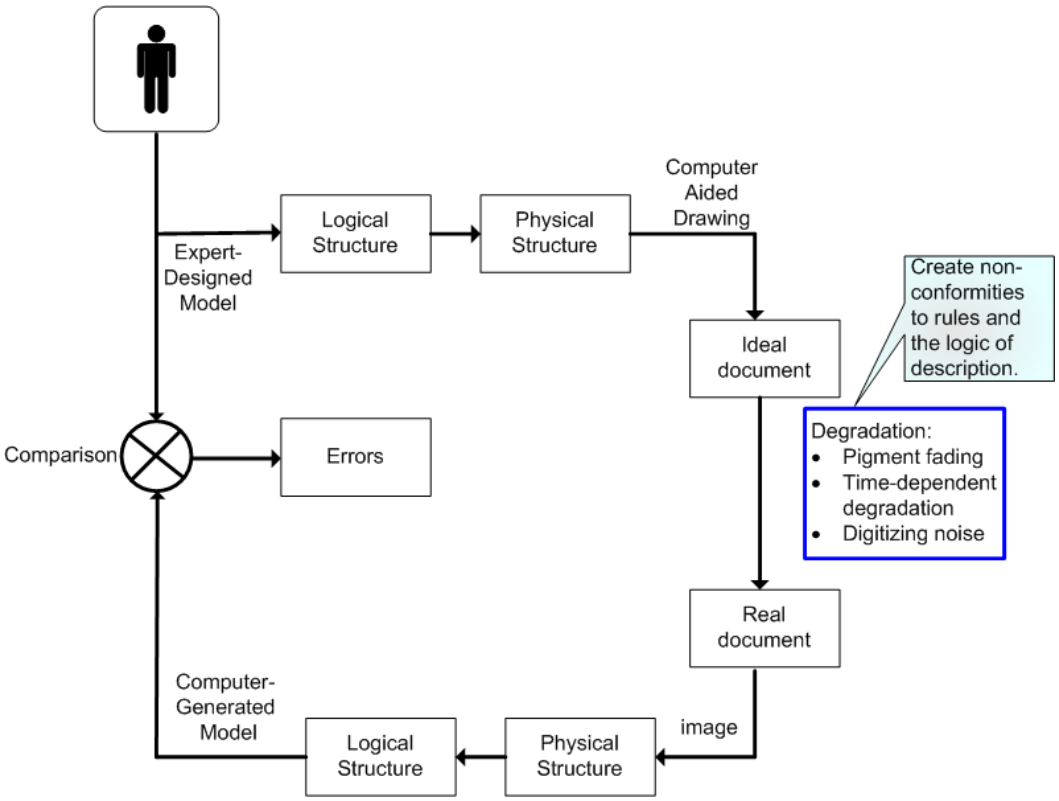


Figure 2.3: Ideal and real documents.

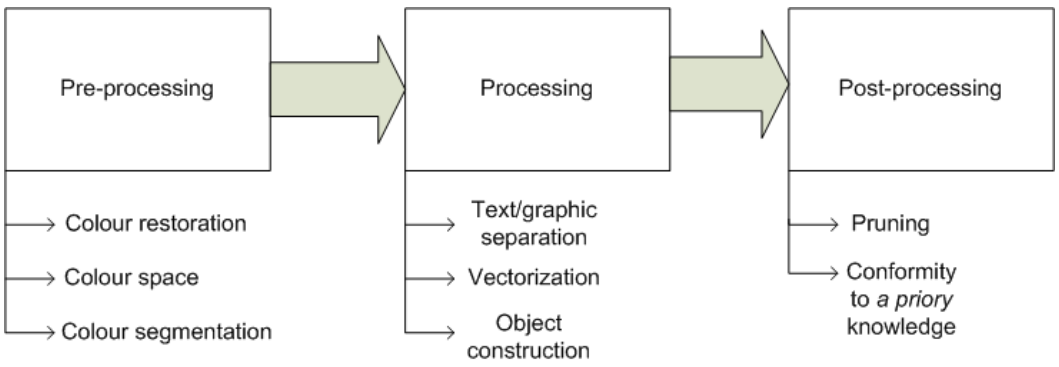


Figure 2.4: An adaptation of the bottom-up strategy to the ALPAGE's context.

This is why RGB space is widely used in the applications of image processing. However, it may not be always the most appropriate color space for computer vision algorithms. Consequently, we provide guide lines to find a suitable color model according to the kind of images we have.

(c) Color segmentation
Once the source image is transferred into a suitable hybrid color space, an edge detection algorithm is applied. This contour image is generated thanks to a vectorial gradient according to the following formalism. The gradient or multi-component gradient takes into account the vectorial nature of a given image considering its representation space (In our case a hybrid color space). The vectorial gradient is calculated from all components seeking direction for which variations are the highest. This is done through maximization of a distance criterion according to the L2 metric, characterizing the vectorial difference in a given color space. The approaches proposed by DiZenzo [Dizenzo 1986] first, and then by Lee and Cok under a different formalism are methods that determine multi-components contours by calculating a color gradient from the marginal gradients.

2. Processing

(a) Text/graphic separation
A contour image is obtained by binarization of the gradient image; this latter is issued from the color detection stage. The text/graphic segmentation is run on top of it, on every contour image to remove undesirable objects: street names, street numbers, ... The mainstream of this text/graphic separation is a graph classification principle where objects to be classified are graphs extracted from the connected components of the contour image. In a learning stage, text and graphic diversities are taken into account by a prototype selection scheme for structural data, thereafter in a decision step, a standard nearest prototype classification rule is applied to categorized instances. Finally, all connected components labeled as "graphics" are included into a so called graphic image.

(b) Object detectors
Dedicated image processing algorithms are performed on the graphic image to locate domain objects. Thus, frame, streets, quarters and parcels are delineated by black pixels.

(c) Digital curve approximation: vectorization
Black pixels are vectorized using a polygonal approximation based on a genetic algorithm. In this method, the optimization/exploration algorithm locates breakpoints on the digital curve by minimizing simultaneously the number of breakpoints and the approximation error. Using such an approach, the algorithm proposes a set of solutions at its end. This set which is called the Pareto Front in the

multi objective optimization field contains solutions that represent trade-offs between the two classical quality criteria of polygonal approximation: the Integral Square Error (ISE) and the number of vertices.

(d) Polygonizer. From lines to polygons.
Detecting polygons defined by a set of line segments in a plane is an easy step in the analysis of vectorial drawings. To perform polygon detection from a set of line segments we divide this task in four major steps. First we detect line segment intersections. Next step creates a graph induced by the drawing. The third step finds the Minimum Cycle Basis (MCB) of the graph induced in previous step. Last step constructs a set of polygons based on cycles in the previously found MCB.

3. Post-processing

(a) Correcting rules.
Polygons which are too small to be a parcel of land are merged according to their color properties.

(b) Reliability measure.
Once the vectorization is finished, we produce a graph that contains concepts and relation between them. Concepts are objects recognized in the prior operations (i.e. Text components, quarters, parcels, ...). This model instance is compared to a higher representation (a meta-model) of cadastral map thanks to a graph matching algorithm. This Model Driven Engineering angle is a well-suited candidate for image conversion by providing a common framework for representing graph structures as models conforms to meta-models elaborated by expert of the domain. The overall method provides a distance which indicates the confidence in the raster to vector conversion quality.

2.5 Discussion

The problem of automatic raster map digitization has been discussed. We conjecture that the most promising line of progress toward its solution lies in successively integrating knowledge into the system. Raster to vector process requires a human judgment to control the quality of the vectorization and to adjust algorithm parameters. This judgment can be considered as a semantic analysis. Introducing this analysis into a vectorization process must considerably improve the results. However, this integration is not a trivial task. The representation by graph formalism is a powerful tool since graphs can represent different points of view of a given image, from the region layout to knowledge configuration. Estimating the quality of the vectorization thanks to a model checking approach is one of our contribution in this domain. Thus our approach could be combined with interactive approaches.

Finally, the next section will focus on pre-processing, low level tools to unleash and extract the basic constructs of our system.

Chapter 3

Color Processing

Contents

3.1 Forewords

The choice of a relevant color space is a crucial step when dealing with image processing tasks (segmentation, graphic recognition...). From this fact, we address in a generic way the following question: What is the best representation space for a computational task on a given image? In this chapter, a color space selection system is proposed. From a RGB image, each pixel is projected into a vector composed of 25 color primaries. This vector is then reduced to a Hybrid Color Space made up of the three most significant color primaries. Only three color components are retained to be conformed with standard image formats. Hence, the paradigm is based on two

principles, feature selection methods and the assessment of a representation model. The quality of a color space is evaluated according to its capability to make color homogeneous and consequently to increase the data separability. Our framework brings an answer about the choice of a meaningful representation space dedicated to image processing applications which rely on color information. Standard color spaces are not well designed to process specific images (ie. Medical images, image of documents) so a real need has come up for a dedicated color model.

3.2 Introduction

Color representation is the basement of all color image processing applications. In fact, many color spaces were developed for graphics and digital image processing such as Red, Green, Blue (RGB) and Hue, Saturation, Intensity (HSI). Nevertheless, it is obvious that the performance of any color-dependent system is highly influenced by the color model it uses. The quality of a color model is defined by its capacity to correctly distinct color between them while being robust to variations inside a given chromatic cluster such as light changes. In term of data-mining, this problem can be addressed as maximizing the distance inter-classes while minimizing the distance intra-class. These two criteria seem to be conflicting, which represents a real challenge to any color representation scheme. Many information retrieval applications would benefit for a better representation space. The chapter is organized as follows: In the second section, we present an image filter for color restoration, in the mean time, the question of finding the best color space is introduced with a review of the related work. Thirdly, the global concept is described explaining the methodology of our contribution. Then, the fourth section presents the feature selection methods in use. The fifth section presents experimental results on color classification according different color models; in addition a comparative study on cadastral map segmentation is presented. Finally, a conclusion is given and future works are brought in the last section.

3.3 Color Restoration

The ancient map archives represent an important part of our collective memory. In introduction, we expressed the difficulties to analyze ancient documents which were deprecated due to the time, usage condition or storage environment. So clearly, a real need for image restoration has come up. A pre-process, a faded color correction [Chambah 2000] has been executed to bring colors back to original or at least to unleash color significance. It works automatically by increasing non-uniformly the color saturation of washed-out pigments without affected the dominant color. Here, we present some advances in automating the color fading restoration process, especially with regard to the automatic color correction technique. First of all, let us illustrate the particularities of our images.

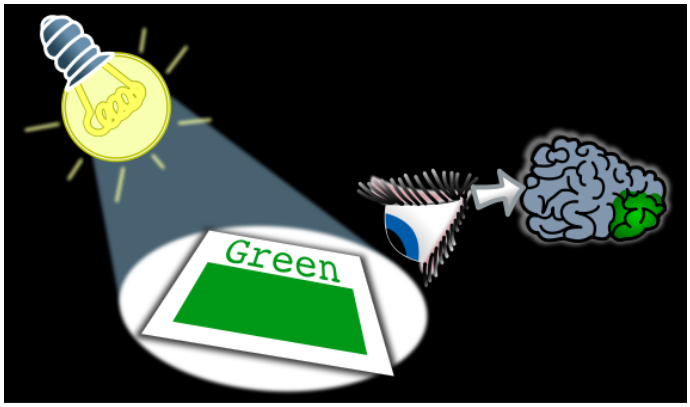


Figure 3.1: The color is a triple that depends on the source light (illuminant), the object and the sensor

3.3.1 Color illuminant

Color vision is the capacity of an organism or machine to distinguish objects based on the wavelengths (or frequencies) of the light they reflect, emit, or transmit. The nervous system derives color by comparing the responses to light from the several types of cone photoreceptors in the eye. These cone photoreceptors are sensitive to different portions of the visible spectrum. For humans, the visible spectrum ranges approximately from 380 to 740 nm, and there are normally three types of cones. What we see depends on the triple composed of the **object** reflecting the source **light** (called illuminant) and the **sensor** (our eyes), figure 3.1. An object may be viewed under various conditions. For example, it may be illuminated by sunlight, the light of a fire, or a harsh electric light. In all of these situations, human vision perceives that the object has not the same color: an apple does not always appear red, whether viewed early in the morning under the sunset or during the day. In our case, the map sheets were digitalized by a commercial scanner involving a cool-white florescent light (Standard illuminant reference: F2).

3.3.2 Image characteristics

Cadastral maps hold some important characteristics that we want to identify. Our interest was to analyze the color properties of our map collection. A complete analysis on each image would have been too time consuming, our assumption was to pick up randomly 50 images and to perform a color analysis on each of them. Among this smaller image set, we report the basic properties of a single image which is representative of our problem. It makes our discussion clearer while being still valid and likely extensible to the rest of the corpus. In the rest of the chapter we focus on the image presented in figure 3.2. Our first test was to visualize the color distribution in the RGB space. Figure 3.3 illustrates how color pixels are spread into the RGB cube. From this experiment, a first comment underlies the color points alignment along the "gray" axis, the straight line of equation $x = y = z$. Secondly, the "gray"

axis seems to give the main direction however the color cloud tends to become larger as the RGB values get higher (r>200,g>200,b>200). This fact can denote a more important variability when colors are under-saturated. Color saturation is used to describe the intensity of color in the image. A saturated image has overly bright colors. The saturation represents the "purity" of a color, with lower saturation being less pure (more washed out, as in pastels). Our next step was to visualize a color histogram of our test image. Figure 3.4 reveals the occurrence of colors in the RGB space. The color space was discretized to calculate a 3D histogram. The discretization is a simple quantification step, the RGB cube is divided into 100 smaller cubes where the amount of pixels in each cube is counted. From this histogram, we observe that most of the information is concentrated into a sphere; the center of the sphere is likely to be located at $r \simeq 220, g \simeq 220, b \simeq 220$ with a radius less than 50. Our last attempt to characterize our images is a conventional statistical framework. A Principal Component Analysis (PCA) was carried out. Let X be the color vector for a given pixel:

$$X = \begin{vmatrix} R \\ G \\ B \end{vmatrix}$$

For an image with N pixels, the covariance matrix can be written as follows:

$C =$

	R	G	B
R	cov(R)	cov(R,G)	cov(R,B)
G	cov(G,R)	cov(G)	cov(G,B)
B	cov(B,R)	cov(B,G)	cov(B)

Where $cov(,)$ is the covariance between two variables. For example, the covariance between R and G can expressed as follows:

$$\theta_{R,G} = cov(R,G) = \frac{1}{N}\sum_{i=1}^{N}(X_i^R - \mu^R)(X_i^G - \mu^G)$$

Where X_i^R denotes the R component of the i^{th} pixel. Where μ is the mean vector and μ^R the R component of the mean vector.

$$\mu^R = \frac{1}{N}\sum_{i=1}^{N} X_i^R$$

To find the eigenvectors and eigenvalues of the covariance matrix, we compute the matrix V of eigenvectors which diagonalizes the covariance matrix C:

$$V^{-1}CV = D$$

where D is the diagonal matrix of eigenvalues of C. The matrix D will take the form of an M $\times$ M diagonal matrix, where

$$D[p,q] = \lambda_m \qquad \text{for} \qquad p = q = m$$

$$V = \begin{array}{|c|c|c|}\hline \text{V0} & \text{V1} & \text{V2} \\ \hline 0.6126 & -0.5664 & -0.5513 \\ \hline 0.5895 & -0.1373 & 0.7960 \\ \hline -0.5266 & 0.8126 & -0.2498 \\ \hline \end{array}$$

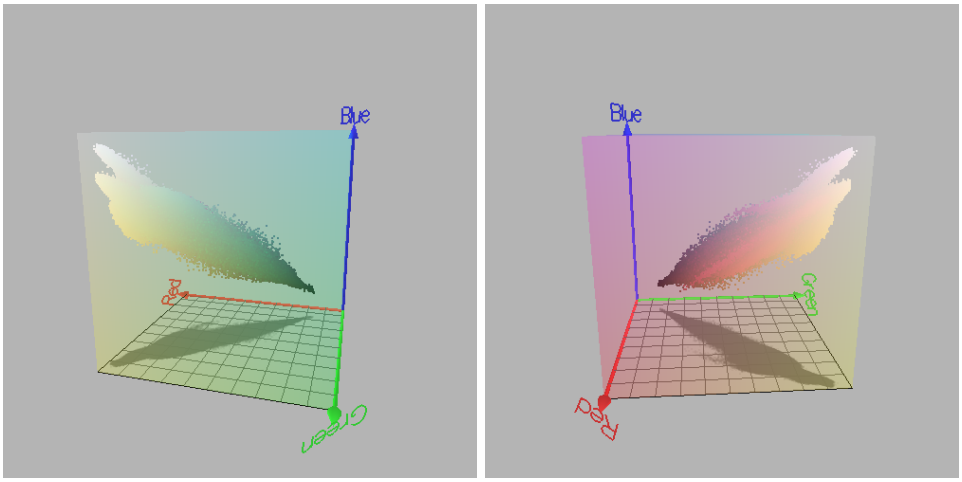
Table 3.1: Eigenvectors for the test image.

Figure 3.2: A representative image of our problem. The map sheet was digitalized by a commercial scanner involving a cool-white florescent light (Standard illuminant reference: F2)

is the m^{th} eigenvalue of the covariance matrix C, and

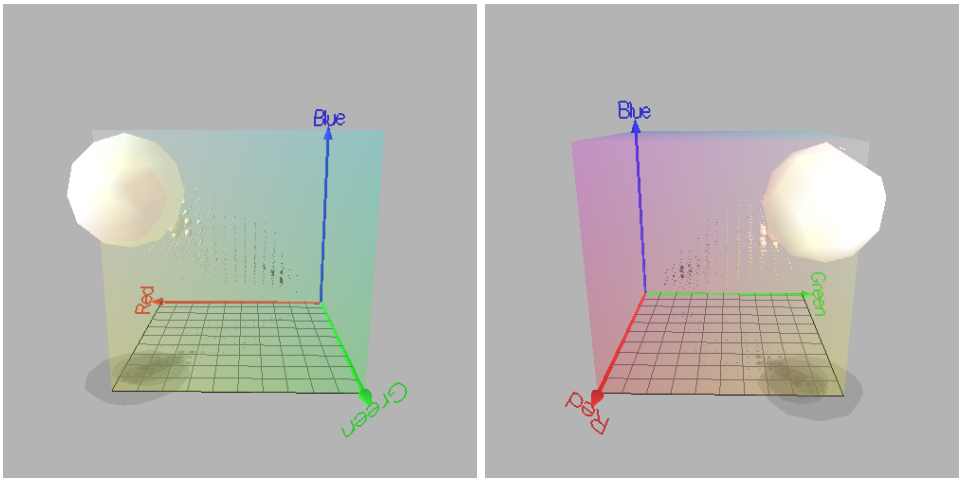
$$D[p, q] = 0 \qquad \text{for} \qquad p \neq q.$$

The matrix V, also of dimension M × M, contains M column vectors, each of length M, which represent the M eigenvectors of the covariance matrix C. The eigenvalues and eigenvectors are ordered and paired. The m^{th} eigenvalue corresponds to the m^{th} eigenvector. Eigenvector are reported in table 3.1. The eigenvalues represent the distribution of the source data's energy among each of the eigenvectors, where the eigenvectors form a basis for the data. The first axis (V0) explains 75.78% of the information while the cumulative inertia of the two first components reaches the 99%. Through this color analysis some remarkable considerations have been stated. Firstly, in the RGB space, colors are distributed along the "gray" axis while being spread all around this axis. Secondly, most of pixels, most of our data can be delineated by a sphere located into the under-saturated area of the RGB cube. Next, the PCA tends to reveal a high variability of data since two axis are required to explain significantly the information laid into the image. From this last observation we describe a color restoration based on PCA.



(a) (b)

Figure 3.3: Color pixel distribution in the RGB cube.



(a) (b)

Figure 3.4: Color pixel histogram in the RGB cube.

3.3.3 Color enhancement based on PCA

Let Y be the data in an independent system axis:

$$Y = V(X - \mu)$$

Where:

- V are the eigenvectors of the covariance matrix.
- μ is the mean vector.

Let $Y\prime$ be the data extended according the direction the main factorial axis:

$$Y' = KY$$

$$K = \begin{vmatrix} k1 & 0 & 0 \\ 0 & k2 & 0 \\ 0 & 0 & k3 \end{vmatrix}$$

The restoration matrix is given as follow:

$$M = V^{-1}KV$$

Let $X\prime$ be the vector containing the restored values:

$$X\prime = M(X - \mu) + \mu$$

The parameters k1 ,k2, k3 are calculated automatically. We want to extend as much as we can the dynamic of the factorial axis but if the parameters are pushed too high, they may cause a peek phenomena ($X'^{R,G,B} > 255$) of the color primaries and create the apparition of false-colors. To avoid this situation, we increase wisely and iteratively the parameters until the upper bound (255) is reached. To ensure this condition, for a given set of parameters k, we verify that no RGB values are above 255. The problem is formulated in the following equation 3.1. A piece of example is presented in figure 3.5. The resorted image seems visually more saturated and colors look warmer and more intense.

(3.1)

$$\begin{aligned} \max_K X' = \quad & (V^{-1}KV(X - \mu) + \mu \\ \text{under constraints} \quad & X' \leq \begin{vmatrix} 255 \\ 255 \\ 255 \end{vmatrix} \\ & K > \begin{vmatrix} k1 = 0 & 0 & 0 \\ 0 & k2 = 0 & 0 \\ 0 & 0 & k3 = 0 \end{vmatrix} \end{aligned}$$

(a) Original image (b) Restored image

Figure 3.5: Image restoration by means of non-uniform increasing of the saturation

3.4 Related works on color spaces

In this section, reference to previous works on this field of science is done starting by classical color spaces to finally present the selection of color components.

3.4.1 Standard color spaces

Most of acquisition devices, such as digital cameras or scanners, process signals in the RGB format. This is why RGB space is widely used in the applications of image processing. The R primary in RGB corresponds to the amount of the physical reflected light in the red band, the same principle holds for G and B. For example, in the color cube of RGB space the distance between blue=(0,0,255) and magenta=(255,0,255) equals the distance between magenta and white=(255,255,255). However, the human vision system considers the perceptual distance between blue and magenta less than the distance between white and magenta. RGB representation has several drawbacks that decrease the performance of the systems which depend on it. RGB space is not uniform; the relative distances between colors do not reflect the perceptual differences. Another popular model is the HSI representation. HSI space has been developed as a closer representation to the human perception system, which can easily interpret the primaries of this space. In HSI space, the dominant wavelength of color is represented by the hue component. The purity of color is represented by the saturation component. Finally, the darkness or the lightness of color is determined by the intensity component. Eq.3.2 shows the transformation between RGB and HSI spaces [J. M. Tenenbaum T. D. Garvey 1974].

$$\begin{aligned} I &= \tfrac{1}{3}(R+G+B) \\ S &= 1 - \tfrac{3}{R+G+B}[min(R,G,B)] \end{aligned} \tag{3.2}$$

$$H = \begin{cases} \theta & B \leq G \\ 360 - \theta & B > G \\ Where & \theta = \arccos \frac{0.5[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}} \end{cases} \tag{3.3}$$

Although the HSI space is suitable for lots of applications based on color images analysis, this color space presents some problems. For example, there are non-avoidable singularities in the transformation from RGB to HSI, as shown in Eq.3.2 and this representation is not a perceptual system. The XYZ color space developed by the International Commission on Illumination (CIE) in 1931 [Illumination] is based on direct measurements of the human eye, and serves as the basis from which many other color spaces are defined. The YUV color is used in the PAL system of color encoding in analogical video, which is part of television standards. The YUV model defines a color space in terms of one luminance and two chrominance components. Another alternative of YUV is the YIQ which is used in the NTSC TV standard. On the other hand, Ohta, Kanade, and Sakai [Y. I. Ohta T. Kanade 1980] have selected a set of "effective" color features after analyzing 100 different color features which have been used in segmenting eight kinds of color images. Those selected color features are usually names as I1I2I3 color model. XYZ, YUV and I1I2I3 are non-uniform color spaces; therefore CIE has recommended $L^*a^*b^*$ and $L^*u^*v^*$ as uniform color spaces, as they are non-linear transformation of RGB space [Sangwine Stephen J.; Horne 1998]. A remark comes to complete this brief review, there is no ideal color model and the possibility to combine or to mix color spaces is discussed in the next part.

3.4.2 Hybrid color Spaces

Recently, the question of finding the best color representation has generated a rich literature. In [L. Busin N. Vandenbroucke 2004], a standard color space is picked-up specifically for a given image in order to classify color pixels. The main originality of the proposed unsupervised procedure is the selection of the most relevant color space to categorize each class of pixels. This color space selection for unsupervised color image segmentation does not consider the possibility to combine color components from several spaces. To overcome this shortcoming, in [J. D. Rugna P. Colantoni 2004], dominant features from different color spaces are selected to construct a DHCS (Decorrelated Hybrid Color Space). A Principal Component Analysis (PCA) is performed from the covariance matrix composed with the total number of the candidate primaries. The 3 most significant axis are selected to reduce rate of correlation between color components. An optimization-based method [N. Vandenbroucke L. Macaire 1998] tries to compromise indices (compacity and classes dispersion) in order to assess the suitability of a color model. These indices represent two competitive constraints, in other word, two conflicting objectives, the improvement of one of them leads to the deterioration of the other. Each image is like no other, so it deserves a dedicated color representation. We believe, it is hardly possible to generalize the color pixel distribution for a given

image set. So it seems unlikely feasible to apply the same color space on all the images contained in a database. Each image must be considered independently. In [N. Vandenbroucke L. Macaire 2003], soccer players are classified, according to their color information, using supervised learning techniques, this training stage supposed to dispose of the user ground truth which is not often the case, and limit the flexibility of the system. Our framework is generic since it relies on a parsimonious use of machine learning algorithms. Furthermore, we handle different feature selection methods; we take advantages of their different ways to reach a single goal.

3.5 Methodology

The main architecture of our framework is presented in figure 3.6. It starts from an RGB image where each pixel is projected into nine standard color spaces in order to build a vector composed of 25 color components. Let C be a set of color components. $C = \{C_i\}_{i=1}^{N} = \{R, G, B, I1, I2, I3, L^*, u^*, v^*, ...\}$ with Card(C)=25. To make data homogeneous and comparable a normalization phase is carried out. Each component belongs to a finite space and is normalized between 0 and 1 using its own maximum value (i.e. $R_{norma} = R/255$). From this point, pixels represent a raw database, an Expectation Maximization (EM) clustering algorithm is performed on those raw data in order to label them. Each feature vector is tagged with a label representing the color cluster it belongs to. Thereafter, a hybrid color space is computed by feature selection methods. Finally, our approach aims at maximizing one criterion which is the color recognition rate (Eq. 3.4), the color space maximizing the recognition rate is considered as the best candidate. Thus, unlike former methods, the recognition rate is directly involved in the choice of a relevant color space. Next, the question of feature selection methods is discussed.

$$Rec = \frac{\sharp\text{Correctly classified color pixels}}{\sharp\text{Color pixels}} \tag{3.4}$$

3.6 Feature selection methods

The selection of features is a very active area in recent years, especially in the context of data mining. Indeed, the data mining in very large databases is becoming a critical issue for applications such as image processing, finance, etc. It is important to summarize and intelligently retrieve the "knowledge" from raw data. The data mining is an area based on statistics, machine learning and the theory of databases. The variable selection plays an important role in data mining especially in the preparation of data prior to processing. Indeed, the interests of the variable selection are as follows:

- When the number of variables is just too great so that learning algorithm cannot finish in a reasonable time. The selection reduces the dimension of feature space.

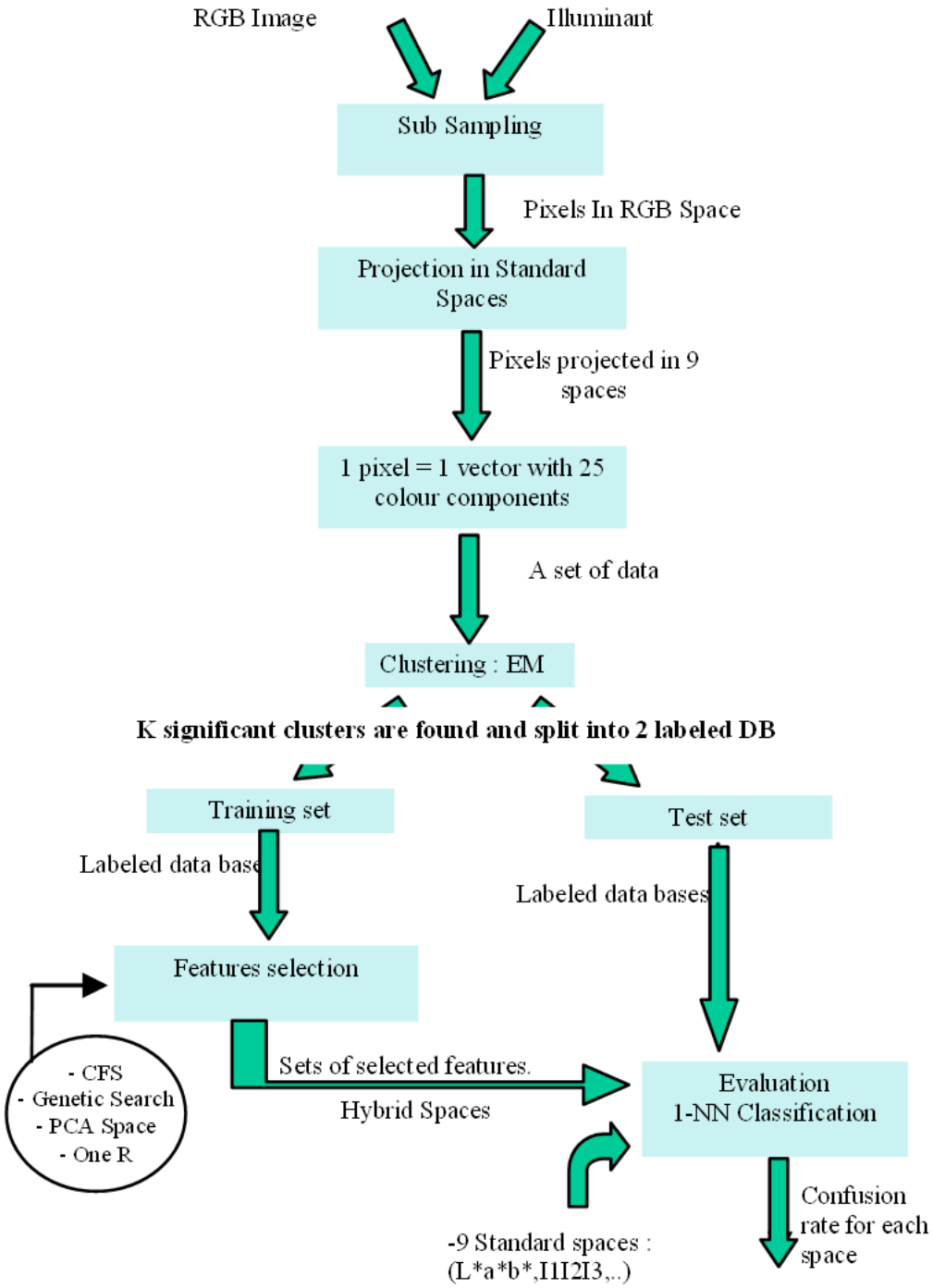


Figure 3.6: A framework for color space selection

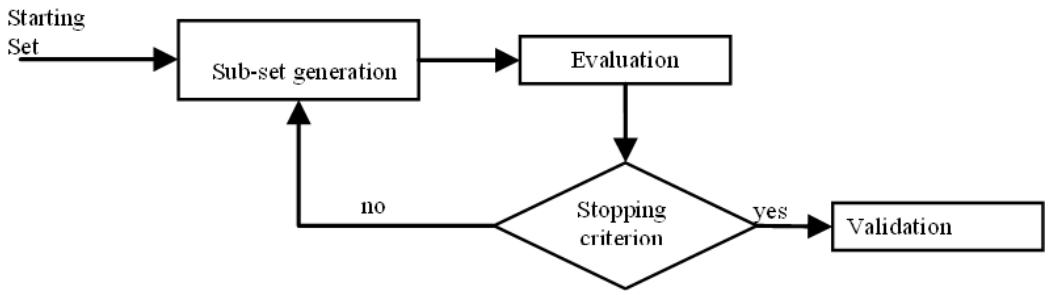


Figure 3.7: Feature selection architecture

- In terms of artificial intelligence, creating a classifier returns to create a model for the data. However, a legitimate expectation for a model is to be as simple as possible (principle of Occam's razor [Anselm Blumer Andrzej Ehrenfeucht 1987]).

Reducing the size of the feature space allows us to reduce the number of required parameters for the description of this model also avoiding the phenomenon of overfitting and emphasizing the synthesize information.

- It improves the performance of the classification, its speed and power of generalization.
- It increases the data understanding: a better view of what are the processes that give rise to them. This selection consists of:
  - The elimination of independent variables of the class,
  - The elimination of redundant variables.

3.6.1 Global concept

A general structure for selecting features can be offered in the way of figure 3.7 ([Hall 1998]). Up to a certain criterion to be satisfied, subsets are generated in browsing the feature space. The subsets generation is a searching process in the subset space of cardinality 2^N with N the number of features. All classical searching algorithms can be applied to that problem. For instance [Koller 1996] proposes the methods forward addition and backward elimination (deletion), [Dangauthier 2005] and [Yang 1998] have made a good use of evolutionary algorithms.

3.6.2 Searching algorithm and evaluation

Existing feature selection methods for machine learning typically fall into two broad categories : those which evaluate the worth of features using the learning algorithm that is to ultimately be applied to the data, and those which evaluate the worth of

features by using heuristics based on general characteristics of the data. The former are referred to as wrappers and the latter filters.

1. Wrappers use classification algorithm to evaluate the pertinence of a given subset of variables.

2. Filters are completely independent from the classification stage. They are based on statistical concepts: entropy, coherence ... A good feature subset is one that contains feature highly correlated with predictive of the class and yet uncorrelated with the others [Hall 1998].

The wrappers: Although conceptually more simple than filters, wrappers were introduced more recently by John, and Kohavi Pfleger in 1994. Their principle is to generate subsets candidates and to evaluate them thanks to a classification algorithm. The score or merit will be a combination of a trade-off between the number of variables eliminated, and the classification rate on a test file. Thus, the "assessment" stage of the selection cycle is made by a call to the classification algorithm. In fact, the classification algorithm is called several times for each evaluation because a cross-validation is frequently used. By its very intuitive principle, this method generates subsets well suited to the classification algorithm. Recognition rates are high since the selection takes into account the intrinsic bias of data. Another advantage is its conceptual simplicity: there is no need to understand how the induction is affected by the selection of variables, it is sufficient to generate and test. However, there are three reasons that the wrappers are not a perfect solution. First, they do not really have theoretical justification for the selection and they do not allow us to understand the conditional dependencies that may exist between the variables. On the other hand, the selection process is specific to a particular classification algorithm and found subsets are not necessarily valid if you change the method of induction. Finally, and this is the main defect of the method, the calculations quickly become quite long when the number of variables grows up.

The filters: Filters don't have the defects of wrappers. They are much faster, they are based on more theoretical considerations, it allows a better understanding to the dependency relationships between variables. But, as they do not take into account the biases of the classification algorithm, the subsets of variables generated give a lower recognition rate. To give a score to a subset, the first solution is to give a score to each variable independently of the others and to do the sum of those scores (OneR Selection). The alternative is to evaluate a subset as a whole [Dangauthier 2005]. However, there is an intermediary between ranking and feature subset ranking based on an idea of Ghiselli and used with good results in the context of the CFS (correlation based feature selection) by M.A. Hall [Hall 1998]. The score of a subset is constructed based on correlations variable-class and correlations variable-variable (Eq.3.5):

$$r_{zc} = \frac{k.\overline{r_{zi}}}{\sqrt{k + k(k-1)\overline{r_{ii}}}} \tag{3.5}$$

Where r_{zc} is the correlation between the summed components and the outside variable (a given color cluster – a class), k is the number of components, r_{zi} is the average of the correlations between the components and the outside variable, and r_{ii} is the average inter-correlation between components. This equation expresses that the merit of a given subset increases if the variables are highly correlated with the class and it decreases if features are highly correlated between each others. The idea is to state that a "good" subset is composed of variables highly correlated with the class (to discard independent variables) and loosely correlated between them/features (to avoid redundant components). It is an approximation since it only takes into account the interactions of order 1. The correlation or dependency between two variables can be defined in several ways. Using the statistical correlation coefficient is too restrictive because it only captures the linear dependence. However, one can use a test of independence as the statistical test of χ^2. It is also possible to combine wrapper and filter as presented in [Yang 1998].

Stopping criterion Finally, the stopping criterion may take various forms: a computation time, a number of generations (in the case of a genetic algorithm for instance), a number of selected variables or a heuristic evaluation of the subset "value".

3.6.3 Summary on feature selection methods in use

After this short review on feature selection methods, we propose to categorize and describe the approaches which are used in our color space selection framework. These different methods are mentioned because they cover the main types of attribute selection algorithm.

Hybrid color space built by genetic algorithm. GACS Basics:

- Attribute Evaluator: 1-Nearest Neighbor classifier (1-NN)
- Search method: Genetic search

Genetic algorithm (GA) is a search technique used in computing to find exact or approximate solutions to optimization and search problems. Genetic algorithms are categorized as global search heuristics and are a particular class of evolutionary algorithms (EA) that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover. Full details about Genetic algorithms in search, optimization and machine learning are presented in [Goldberg 1989]. In Hybrid Color Space (HCS) context, each individual has to encode a vector, where each component is an axis of the HCS. We consider a set C of features, $C = \{C_i\}_{i=1}^{N} = \{R, G, B, I1, I2, I3, L^*, u^*, v^*, ...\}$ with Card(C)=25. Practically, it is almost impossible to test all possible combinations; the number of feasible solutions evolves according a factorial function of the total number of the candidate primaries (combinatorial explosion). Hence, GAs are well suited to get rid off absurd

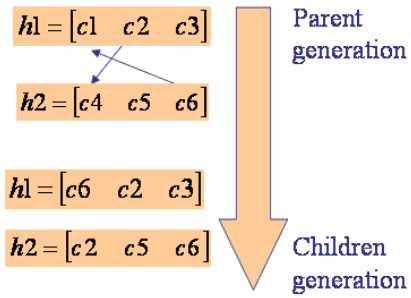


Figure 3.8: Cross over operator

combinations. Roughly, the first step is to initialize the population, each individual is made up picking randomly three elements of C. Concerning cross over operator, two individuals h1 and h2 share their genetic material, swapping one of their component; figure 3.8. Finally, to perform mutation on an individual, one component is selected and replaced at random by an element of C. Finally, the evaluation phase computes a 1-NN classifier based on a Euclidian metric.

Correlation-based Feature Subset Selection (CFS) Basics:

- Attribute Evaluator: Statistical
- Search method: Greedy stepwise

Correlation-based Feature Subset Selection (CFS) proposed by M.A. Hall [Hall 1998] evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. Subsets of features that are highly correlated with the class while having low intercorrelation are preferred. The greedy stepwise search method performs a greedy forward search through the space of attribute subsets. It may start with no attributes and stops when the addition of any remaining attributes results in a decrease in evaluation. The cardinality of the final subset can be set to three in our case.

De-correlated Hybrid Color Space (DHCS) Basics:

- Attribute Evaluator: Statistical
- Search method: Ranker

The basic idea of a De-correlated Hybrid Color Space [J. D. Rugna P. Colantoni 2004] is to combine different color components from different color spaces. Considering that there is a high redundancy between colors components it is, in a general way, quite difficult to define criteria of analysis to compute automatically the most relevant color components corresponding to a selected set of color components. That is the reason why, in order to build a hybrid color space, based on K' color components, from K selected color components,

Name	Type	Evaluation	Searching algorithm
CFS	Filter	Statistical	Greedy stepwise
DHCS	Filter	Statistical	Ranker
GACS	Wrapper	Classification	Genetic Algorithm
OneRS	Wrapper	Classification	Ranker

Table 3.2: Selection feature methods in use

such as K' << K, the proposed method: (1) computes the covariance matrix (of size KxK) of K color components selected, (2) computes the eigenvectors and the eigen values of this matrix, (3) reduces to K' the number of color components in computing the K' most significant eigen values of the covariance matrix from a principal component analysis (PCA). It can also produce a ranked list of attributes by traversing the space from one side to the other and recording the order in which attributes are selected.

One Rule Selection method (OneRS) Basics:

- Attribute Evaluator: One Rule Evaluation
- Search method: Ranker

The One Rule Selection method classes for building and using a One-R classifier; it evaluates the worth of an attribute by using the OneR classifier, in other words, uses the minimum-error attribute for prediction, discretizing numeric attributes. OneR, short for "One Rule", is a simple classification algorithm that generates a one-level decision tree. OneR is able to infer typically simple, yet accurate, classification rules from a set of instances. The OneR algorithm creates one rule for each attribute in the training data, then selects the rule with the smallest error rate as its 'one rule'. To create a rule for an attribute, the most frequent class for each attribute value must be determined. The most frequent class is simply the class that appears most often for that attribute value. A rule is simply a set of attribute values bound to their majority class; one such binding for each attribute value of the attribute the rule is based on. The error rate of a rule is the number of training data instances in which the class of an attribute value does not agree with the binding for that attribute value in the rule. OneR selects the rule with the lowest error rate. Finally, it ranks attributes according to error rate (on the training set). It treats all numerically-valued attributes as continuous and uses a straightforward method to divide the range of values into several disjoint intervals. A full report on this selection method can be found in [Holte 1993]. A summary of the different methods is reported in table 3.2.

3.7 Image Segmentation for Hybrid Color Spaces: Vector Gradient

In [L. Busin N. Vandenbroucke 2004], a color segmentation in hybrid spaces is proposed. The paradigm lies on multi-thresholds of 1D histograms. The main drawback of this approach is its marginal nature. In fact, each threshold is calculated individually for each color component. Others systems like [N. Vandenbroucke L. Macaire 2003] and [J. D. Rugna P. Colantoni 2004] only perform a pixel classification without considering the spatial information. To tackle these problems, a vector gradient segmentation is adopted. Once the source image is transferred into a suitable hybrid color space, an edge detection algorithm is processed. This contour image is generated thanks to a vectorial gradient according to the following formalism. The gradient or multi-component gradient takes into account the vectorial nature of a given image considering its representation space (In our case a hybrid color space). The vectorial gradient is calculated from all components seeking direction for which variations are the highest. This is done through maximization of a distance criterion according to the L2 metric, characterizing the vectorial difference in a given color space. The approaches proposed by DiZenzo [Dizenzo 1986] first, and then by Lee and Cok under a different formalism are methods that determine multi-components contours by calculating a color gradient from the marginal gradients. Given 2 neighbour pixels P and Q characterizing by their color attribute A, the color variation is given by the following equation:

$$\triangle A(P,Q) = A(Q) - A(P) \tag{3.6}$$

The pixels P and Q are neighbors, the variation can be calculated for the infinitesimal gap: $dp = (dx, dy)$

$$dA = \frac{\partial A}{\partial x}dx + \frac{\partial A}{\partial y}dy \tag{3.7}$$

This differential is a distance between pixels P and Q. The square of the distance is given by the expression below:

$$dA^2 = \begin{cases} = & \{\frac{\partial A}{\partial x}\}^2 dx^2 + 2\frac{\partial A}{\partial x}\frac{\partial A}{\partial y}dxdy + \{\frac{\partial A}{\partial y}\}^2 dy^2 \\ = & adx^2 + 2bdxdy + cdy^2 \end{cases} \tag{3.8}$$

$$= \begin{cases} a = & (G_x^{e1})^2 + (G_x^{e2})^2 + (G_x^{e3})^2 \\ b = & G_x^{e1}G_y^{e1} + G_x^{e2}G_y^{e2} + G_x^{e3}G_y^{e3} \\ c = & (G_y^{e1})^2 + (G_y^{e2})^2 + (G_y^{e3})^2 \end{cases} \tag{3.9}$$

Where, $E = \{e1, e2, e3\}$ can be seen as a set of color components representing the three primaries of the hybrid color model. And where G_n^m can be expressed as the marginal gradient in the direction n for the m^{th} color components of the set E. The calculation of gradient vector requires the computation at each site (x, y): the slope

direction of A and the norm of the vectorial gradient. This is done by searching the extrema of the quadratic form above that coincide with the eigen values of the matrix M.

$$M = \begin{pmatrix} a & b \\ b & c \end{pmatrix} \tag{3.10}$$

The eigen values of M are:

$$\lambda_{\pm} = 0.5(a + b \pm \sqrt{(a-c)^2 + 4b^2}) \tag{3.11}$$

Finally the contour force for each pixel (x,y) is given by the following relation:

$$Edge(x, y) = \sqrt{\lambda_+ - \lambda_-} \tag{3.12}$$

This segmentation algorithm is assessed in the next section.

3.8 Experiments

In the idea to assess our system, we perform two evaluation stages. The first one is a color classification step to test if the color representation found by our framework is interesting in term of color distinction. The second step is a segmentation phase. Indeed, a better representation system should give better segmentation results. Note that this software, called Best Color Space Finder, can be found on the L3i-ALPAGE website[1].

3.8.1 Color classification

Test image descriptions: Our approach is applied on three different types of images. An image of: natural scene, document and a synthetic image, this is depicted in table 3.3 and figure 3.9. The "Lenna" image is a conventional data source widely used in image processing. The image of cadastral map represents the problem we attempt to tackle and finally, the synthetic image is a chessboard-like image composed of 64 squares; this image is of special interest since 2 adjacent squares can be distinguished by their saturation level. For this artificial image the number of clusters is set 65 (64 boxes and the surrounding border).

Protocol: Considering the full set of attributes a clustering algorithm (EM) is applied on each image. The number of color clusters per image is reported in table 3.3. The final purpose is to find the hybrid color space which provides the most similar color partition compared to with the one discovered using the whole set of features. The merit of a color space is evaluated under this consideration. For each image, each color cluster is divided in two sub-sets, one for training and one for validation purpose. At this stage, each pixel is labeled according its cluster

[1] http://alpage-l3i.univ-lr.fr/ -> Best Colour Space Finder

Id	Image	Type	# of clusters
Im1	Lenna	Natural Scene	18
Im2	SatSnake	Synthetic image, discriminated by the saturation	65
Im3	Image of document	Ancient Cadastral Map	9

Table 3.3: Test image descriptions

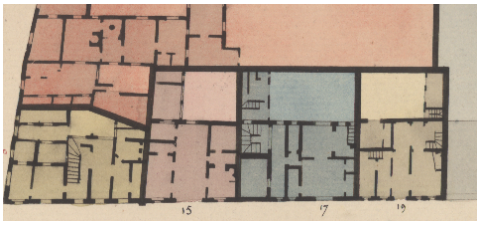
	$\|X_{training}\|$ pixels	$\|X_{test}\|$ pixels
IM1	130107	130107
IM2	100951	100951
IM3	110424	110424

Table 3.4: Training and Test Databases

number. Feature selection methods are performed on the training database and each color space is evaluated thanks to the test data set. The color space performance evaluation consists in a classification stage. Both test and training databases are projected into the color space we want to evaluate. Each color vector from the test database is classified using a nearest neighbor rule (One nearest neighbor classifier, 1-NN for short). The test object is labeled with the cluster number of the most similar color instance from the training database; the underlying vector comparison is based on a euclidean distance (L2 norm). Table 3.4 reports the number of elements handled during the classification process.

Results: A fully detailed example is reported in table 3.5. The complete list of selected attributes by feature selection methods is provided for the image 3.9(b). The selected components by the color space finder methods are quite heterogeneous, likely the reason lies on the variability of the considered approaches. In fact, they do not rely on the same principles; they all differ either from their search method or from their selection mechanism. Nevertheless, all feature selection methods adopted the saturation component to describe the content of this image. This demonstrates the pertinence of feature selection algorithms. For each real-world image, classification errors are gathered in tables 3.6, 3.7. The color space minimizing the confusion rate is elected to be the most discriminating feature space. Over the two images, color spaces built by the GACS perform the best. Consequently, generally speaking HCS outperform standard spaces. However, a closer look to the results denotes a poor achievement of selected attributes by OneRs, DHCS and CFS, they failed to overcome standard spaces. The "top 5" is often dominated by one HCS follows by four standard spaces.

(a) (b)



(c)

Figure 3.9: Images in use.

Table 3.5: Hybrid color Spaces found on the Image IM2

Attributes	CFS	GACS	DHCS	OneRs
R	0	0	0	0
G	0	0	1	0
B	0	0	0	0
I1	0	0	0	0
I2	0	0	0	0
I3	0	0	0	0
T	0	0	0	1
S	1	1	1	1
I	0	0	0	0
L*	0	0	0	0
a*	0	0	0	0
b*	0	1	0	0
L*	0	0	0	0
u*	1	0	0	0
v*	0	0	0	0
A	0	0	0	0
C1	0	0	0	0
C2	0	0	0	0
X	0	0	0	0
Y	0	0	1	0
Z	1	0	0	0
Y	0	0	0	0
I	0	0	0	1
Q	0	0	0	0
Y	0	0	0	0
U	0	1	0	0
V	0	0	0	0
# of attributes	3	3	3	3

Table 3.6: Confusion rate on Image 1

IM1			
Color Spaces	***Error***	***Color Spaces***	***Error***
GACS	0.2868	OnRS	0.3558
L*u*v*	0.29785	La*b*	0.3578
YUV	0.32764	I1I2I3	0.3683
YIQ	0.3345	XYZ	0.4650
HSI	0.3394	CFS	0.5877
AC1C2	0.3435	DHCS	0.7067
RGB	0.3529		

Table 3.7: Confusion rate on Image 3

IM2			
Color Spaces	***Error***	***Color Spaces***	***Error***
GACS	0.14065	RGB	0.1561
YIQ	0.1445	OnRS	0.1615
I1I2I3	0.1478	La*b*	0.1650
HSI	0.1488	XYZ	0.2093
L*u*v*	0.1533	CFS	0.3043
YUV	0.15387	DHCS	0.349
AC1C2	0.1557		

3.8.2 Application to segmentation and evaluation

In this section, we evaluate the worth of HCS *vs* the standard RGB space in a segmentation context. Firstly, we describe two datasets for these experiments: (1) the well-known and publicly available Berkley database, this later allows an evaluation on a large ground-truthed corpus; (2) an image of document on which a segmentation algorithm was applied. Next, the question of the segmentation evaluation is discussed and finally, the segmentation performance is evaluated to figure out which color space offers the best image representation.

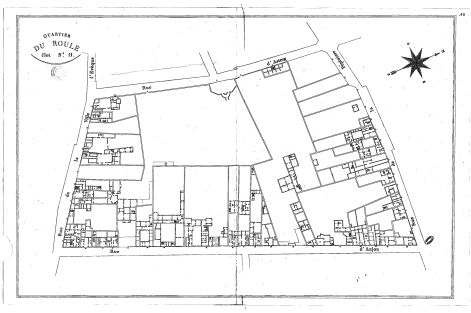
Database description

- Ancient color cadastral map
  - In the context of the ALPAGE project, we are considering the digitalization of ancient maps on which objects are drawn by using color to distinguish parcels for instance. We believe that such a problem would take advantages of a dedicated color space. The color segmentation of cadastral maps relies on the edge values defined in Eq.3.12. These edge values are then filtered using a two class classifier based on an entropy principle in order to get rid off low gradient values. At the end of this clustering stage a binary image is generated. This image will be called as contour image through the rest of this chapter. Finally, regions are extracted by finding the white areas outlined by black edges. The gradient and the binary images are displayed in figure 3.10.

- Berkeley segmentation data set
  - Research on early vision problems such as edge detection and image segmentation has traditionally been critiqued on the grounds that quantitative measurements of performance are rare. It is therefore difficult to evaluate the effect of different design choices and the superiority (or inferiority) of various novel heuristics that have been proposed in the literature. Recently the availability of the Berkeley Segmentation DataSet [Martin 2001], [Segmentation 2002] has allowed the quantitative measurement of performance on boundary finding and the relative power of various pairwise similarity cues. While this is, of course, not the first example of quantitative measurement in segmentation the availability of this large data set containing a wide variety of images and segmentations by multiple human observers (11,0000 segmentations of 1000 images), allows one to draw conclusions with greater "statistical confidence". A sample of the kind of images that composes the database is shown in figure 3.11 along with a human segmentation.

(a) Original RGB image

(b) Color Gradient Image



(c) Binary image

Figure 3.10: Map segmentation

Segmentation evaluation method

- Berkeley data set (with ground-truth)
  - The human segmented images provide our ground truth boundaries. We consider any boundary marked by a human subject to be valid. Since we have multiple segmentations of each image by different subjects, it is the collection of these human-marked boundaries that constitutes the ground truth. In figure 3.11, the output of our algorithm is presented for a given image. Let us assume that this output is a soft boundary map with one pixel wide boundaries, valued from zero to one where high values signify greater confidence in the existence of a boundary. The task is to determine how well this soft boundary map approximates the ground truth boundaries.
- Cadastral map subset (without ground-truth).
  - Another way to assess a segmentation process is to compute the Levin and Nazif (LN) criterion. Without ground-truth for our images, a unsupervised evaluation is required. LN criterion combines intra-class and inter-class disparities. Inter-class disparity score computes the sum of contrasts of the regions balanced by their surfaces while the intra-class uniformity score computes the sum of the normalized standard deviation

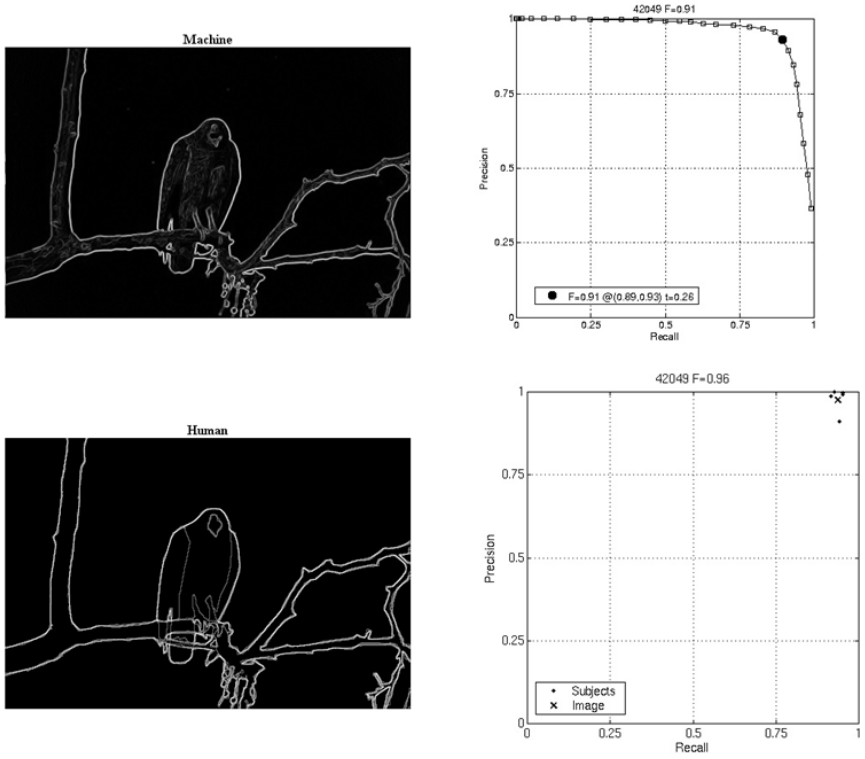


Figure 3.11: Boundary detection: Machine vs Human. Precision and Recall curve.

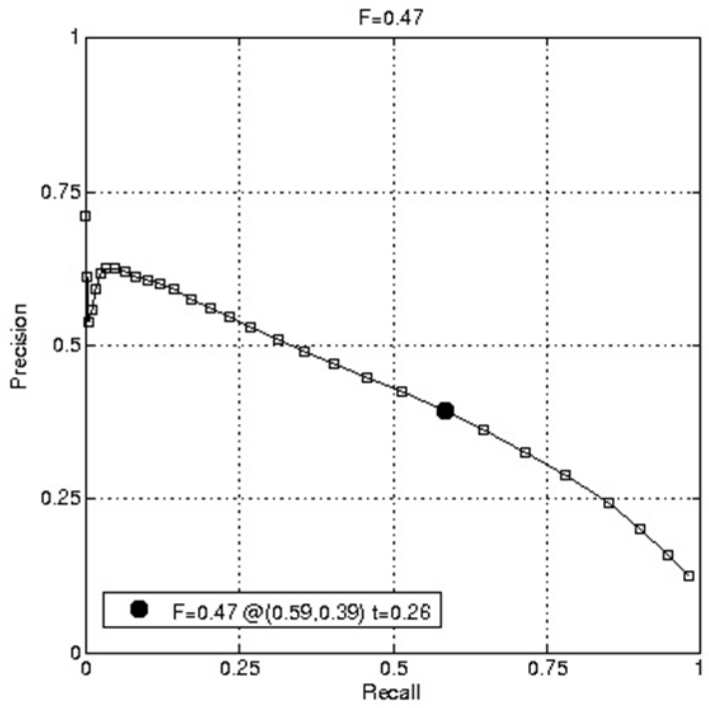
of each region. It takes into parameters the segmented image and the original image and returns a score, the higher the better. This comparison is carried out on 50 pairs of maps. Levin and Nazif criterion [Rosenberger 2006] is the union of two principles, the variability intra and inter regions.

Results

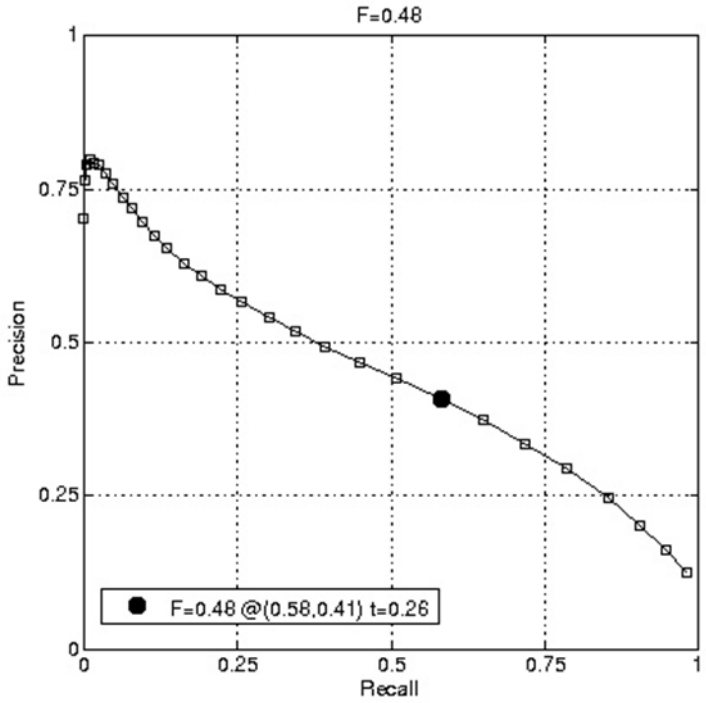
- Berkeley data set (with ground-truth)
  - The vector gradient, defined in section 3.7, is performed as a boundary detector on the Berkeley Dataset. Segmentation evaluation in RGB space and HCS space are illustrated in figure 3.12. In figure 3.12, Precision and Recall (PR) curves are presented. In this graph, Recall is a measure of how well the detector performs in finding relevant contours while Precision is a measure of how well the detector performs in not returning nonrelevant contours. The F-measure is the harmonic mean of precision and recall. One remarkable consideration extracted from the Precision/Recall curves is the fact that the PR curves in HCS is always above or equal to the RGB PR curve. Consequently, the F-measure is slightly higher when performing in HCS space. This is an encouraging observation, it means that at least a HCS will not degrade the segmentation results and compare to RGB, HCS will tend to improve the precision when the recall is low.
- Cadastral map subset (without ground-truth)
  - In table 3.8 LN criterion for the segmentations based on RGB and HCS is reported. As mentioned in the prior paragraph, HCS reveal to be slightly better than RGB, it means that regions found when processing the color gradient in HCS are more uniform and more contrasted than in RGB space. However, this remark must be tempered by the fact that the improvement is not significant. Somehow the approach reminds of the ”killing butterflies with missiles” paradigm, a complex framework to achieve a bit better than the original image.

3.9 Conclusion

The quality of a color model is judged by two decisive factors: "Robustness" and "Distinction". The robustness of the color representation is an indication of the sensitivity of color values to illumination and brightness variations. The “Distinction” capacity of a color model is directly linked to its capacity to separate one color from the others. The color space minimizing the error rate classification is the most discriminating space for a given image (Tables 3.6, 3.7). The space generating the



(a) On RGB space



(b) On HCS space

Figure 3.12: Overall results: Precision/Recall curves on Berkeley benchmark

Table 3.8: Comparison HCS and RGB spaces on a segmentation process using LN criterion.

Dizenzo segmentation color cadastral maps	*LN Criterion on 50 images*	
	Average	**Std deviation**
RGB	0.4770375	0.005396543
HCS	0.480325	0.007211647

least mistake will be retained to continue treatments on the image. The chosen space is minimizing the distance intra-class, within the same unit chromatic while maximizing the distance inter-classes. Such properties are helpful in post-processing stages such as segmentation, or graphics recognition. The LN criterion results lead to the same conclusion, showing that the contrast inter regions and the homogeneity intra-region are slightly better in HCS than in the RGB case. These results are encouraging and they demonstrate how important it is to choose a "good" color model. To take the stock, in this chapter, we have presented a color space selection framework. Our contribution focuses on a "all-in-one" system to find a suitable color space. Our tool can be seen as a pre-process to any color information retrieval application (Segmentation, graphic recognition ...). Our approach aims at maximizing one criterion which is the color recognition rate to unleash the color information. Each image is like no other, so a dedicated color representation is required. We believe, it is hardly possible to model a unique color space from a given image set and then to apply this "mean model" individually, that's why our method computes independently a dedicated model to each image. Our framework relies on a wise use of different feature selection methods in order to take advantages of their diverse ways to reach a single goal. Finally, Hybrid Color Spaces are particularly well suited while dealing with very specific images, such as medical images, images of documents where CIE spaces are not particularly well designed. We believe that much color image software would get profit to the use of an adapted color space. A future work is envisaged by comparing Hybrid Color Spaces to Support Vector Machine approaches such as Multidimensional scaling (MDS).

Chapter 4

Cadastral map interpretation

Contents

4.1 Introduction

In this chapter, an object extraction method from ancient color maps is proposed. It consists of the localization of frame, text, quarters and parcels inside a given cadastral map. Firstly, a model of cadastral map is introduced; this knowledge representation was elaborated in collaboration with historians and architects, experts in this domain. Secondly, the color aspect is inherited from the color restoration algorithm and the selection of a relevant hybrid color space presented in chapter 3. Thereafter, dedicated image processing aim at locating the various kinds of objects laid out in the raster. These especially designed detectors can retrieve different components such as characters, streets, frame, quarters and parcels. These specific tools are run successively in the objective to identify boundaries of the different elements. In a last phase, these elements are put into a graph-based representation to be further compared with the meta-model defined by the experts. This comparison is carried out thanks to a graph matching algorithm. "Metamodeling" is the construction of a collection of "concepts" (objects, terms, etc.) within a certain domain. A model is an abstraction of phenomena in the real world; a metamodel is yet another abstraction, highlighting properties of the model itself.

Technical documents have a strategic role in numerous organizations, composing somehow a graphic representation of their heritage. In the context of the project called “ALPAGE”, a closer look is given to ancient French cadastral maps related to the Parisian urban space during the 19th century (figure 4.1). Hence, the map collection is made up of 1100 images issued from the digitalization of Atlas books. On each map a vast number of domain-objects are drawn by using color to distinguish them, i.e. parcels, water collection points, stairs, windows/doors, ... Within the scope of the thesis, we focus on the following objects: text components, quarters and parcels. However, our methodology can be easily extended to a wider range of items. From a computer science point of view, the challenge consists in the extraction of information from color documents in the objective of providing a vector layer to be inserted in a GIS.

4.1.1 Image processing and knowledge representation

In the last fifty years, a lot of image processing applications have been developed in many fields (medicine, geography, robotic, industrial vision, ...). We know that image processing specialists design applications by trial errors cycles. They do not enough reuse already developed solutions and design new ones nearly from scratch. The lack of application formulation modeling and formalization is a reason of this behavior. Indeed, image processing experts do not realize a complete and rigorous formulation of the applications. Only the solutions are used as their definitions. Therefore, the reusability of these applications is very poor because the limits of the solution applicability are not explicit. Moreover they often suffer from a lack of modularity and the parameters are also often tuned manually without giving explanations on the way they are set.

Knowledge based systems such as OCAPI [Clement 1993], MVP [Chien 1996] or BORG [Clouard 1999] were developed to construct automatically image processing applications and to make explicit the knowledge used to solve such applications. However, a priori knowledge on the application context (sensor effects, noise type, lighting conditions, ...) and on the goal to achieve was more or less implicitly encoded in the knowledge base. This implicit knowledge restricts the range of application domains for these systems and it is one of the reasons of their failure [Draper 1996].

More recent approaches bring more explicit modelling [Maillot] [Hudelot 2003]) [BOMBARDIER] [Town] but they are all limited to the description of business objects for detection, segmentation, image retrieval, image annotation or recognition purposes. Some of them use ontologies that provide the concepts needed for this description: a visual concept ontology for object recognition in [Maillot], a visual descriptor ontology for semantic annotation of images and videos in [Bloehdorn 2005] or image processing primitives in [Hudelot 2003]. Others capture the business knowledge through meetings with the specialists: use of a conceptual modeling method (NIAM/ORM) method in [BOMBARDIER] to collect and map the business knowledge to the vision knowledge. But they do not completely tackle the problem of the application context description (or briefly as in [Maillot]) and the effect of this context on the images (environment, lighting, sensor, image format). Moreover they do not define the means to describe the image content when objects are a priori unknown or unusable (e.g. in robotic, image retrieval or restoration applications). They also suppose that the objectives are well known (to detect, to extract or to recognize an object with a restrictive set of constraints) and therefore they do not address their specification.

To overcome these problems, in [Renouf 2007], Renouf et al aim at building a methodology and a guideline for the development of such applications in order to make it easier and more reliable. To reach this goal, they have to make explicit the formulation of the problem to be solved, and the knowledge used by image processing experts during the design. This solution is really promising but it is still an ongoing work and cannot be used to build a real cadastral map understanding system.

The literature offers many different approaches to technical document interpretation. They concern geographical charts [Deseilligny 1993], mechanical engineering documents [Vaxiviere 1992], electronic diagrams [Hamada 1993], or utility maps [Boatto 1992] (telecom, power and water networks, cadastral, ...). Roughly, two strategies have been proposed: bottom-up and mixed strategies. In bottom-up strategies, algorithms are performed in a fixed sequence, usually starting "low-level" analysis of the gray level or black and white image, in which primitives [Kasturi 1992] are extracted. Figure 4.2 illustrates the three steps of a map understanding system.

The rest of chapter is organized as follows: In section 4.3, we describe the cadastral map meta-model. Section 4.4 is dealing with the problem of quarter and parcel extraction. Then, the section 4.5 explores the ability of evaluating the quality of the system by means of a model oriented approach. Section 4.6 shows experimental results on quarter retrieval. Finally, section 4.7 contains the chapter' conclusion.

Figure 4.1: Example of cadastral map (7616 x 4895 pixels, 200 dpi, 24 BitsPerPixel)

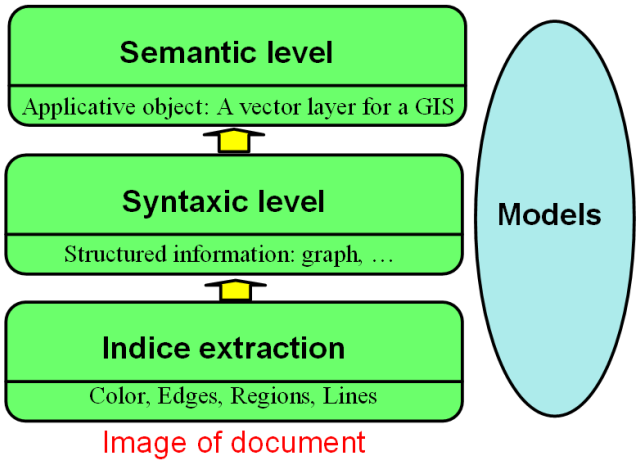


Figure 4.2: Architecture of a graphic document analysis system

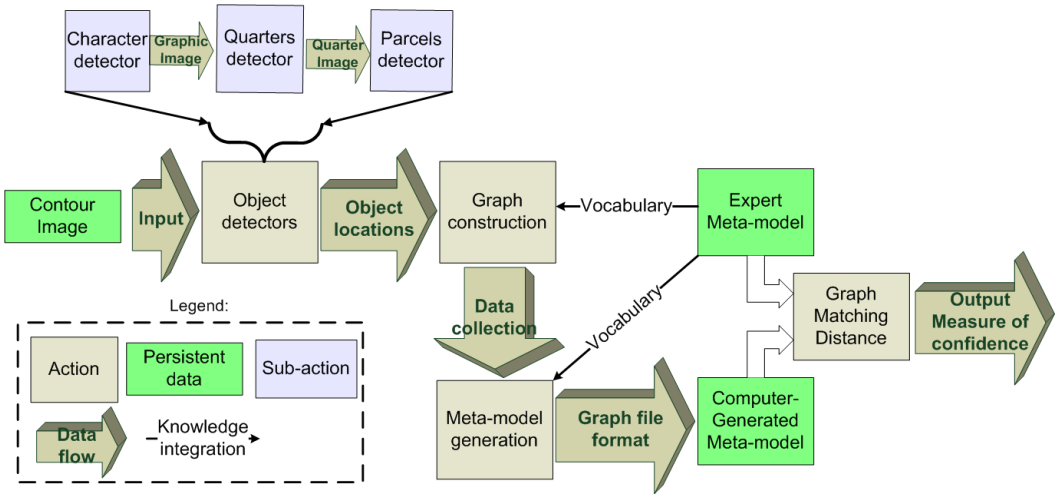


Figure 4.3: Map interpretation strategy

4.2 Global methodology

The cadastral map interpretation strategy is described in this section. Figure 4.3 depicts the overall data flow process. On a given raster, a contour image is created by performing an edge detection algorithm in a hybrid color space. This contour image inherits the color meaning of the original raster. Object detectors are run sequentially to locate frame, characters, streets, quarters and parcels within the raster. Thereafter, this information feeds a higher level which elaborates a graph structure. In this data structure, nodes relate the presence of objects found during the detection step and edges represent the spatial relation between the objects. Terms, words and appellations to qualify node and edge labels are so called concepts. Concepts are defined into a knowledge representation formalism named ontology. This latter contains the vocabulary and the logic of description of each element required to model a cadastral map. An ontology offers a formal description of the knowledge by a set of concepts within a domain and the relationships between those concepts. It is used to gather the information on a given topic from experts in ancient cadastral maps. Therefore, the produced graph can be seen as a model instance of the map. On the other hand, the graph generation is not constraint by the ontology and non-conform variations to the knowledge base can be introduced into the graph structure. A higher level of representation is required to answer the question: "how conform to the expert knowledge the raster to vector result is ?" To reach this goal and without loss of information, the graph is translated into an eXtensible Markup Language (XML) format to be further handled.

Definition 1. *(XML document) An XML document is tuple* $< E, N_E, R_E >$*, Where* N_E *is the set of element names,* R_E *is the distinguished root element of the XML document,* E *is a sequence of elements. Each element* $e \in E$ *is a triplet*

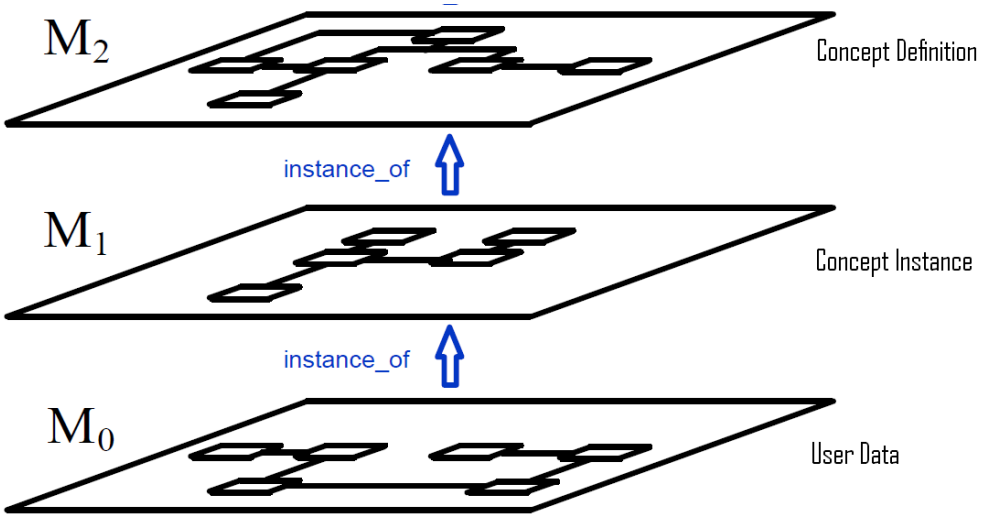


Figure 4.4: Traditional Modeling Infrastructure

$< eid, name, content >$, *where eid is the identifier of element, we assume that every element has a unique ID attribute.* $name$ *is a function name:* $E \longrightarrow N_E$. *The content is either a character string, or a list of name* : *eid pairs.*

The structure of such a file is analyzed with the joint use of a cadastral map ontology to produce a meta-model corresponding to the instance data. An ontology concentrates two paradigms: (i) Description logic to describe each concept from the knowledge base; (ii) Metamodeling to organize and structure the relationship between concepts.

Figure 4.4 illustrates the traditional three layer infrastructure that underpins the metamodeling framework.

This infrastructure consists of a hierarchy of model levels, each (except the top) being characterized as "an instance" of the level above. The bottom level, also referred to as M0 is said to hold the "user data", i.e., the actual data objects the software is designed to manipulate. The next level, M1, is said to hold a "model" of the M0 user data. This is the level at which user models reside. Level M2 is said to hold a "model" of the information at M1. Since it is a model of a (user) model, it is often referred to as a metamodel.

An ontology generator from XML documents is used to elaborate a computer-generated meta-model. This translation proceeds to a generalization describing the structure of an XML document. This automated knowledge representation is derived from a collection of instances and a source ontology. In other words, this transformation builds a meta-model on top of a model. Afterward, the computer-generated meta-model can be mapped with an expert meta-model using a graph matching algorithm. This results in a measure of quality of the document interpretation.

4.3 Meta-model of cadastral map

Here after, we present the fruit of many discussions, collaborations and meetings with our partners from Humanities And Social Sciences. Especially, the architect Michel Denès[1] was the investigator of a dictionary which explains how the maps were elaborated and the meaning of many symbols inside the maps. To identify the elements described in the atlas, it is obviously the vocabulary of this historical period that has been favored. This dedicated vocabulary is inspired from various texts of this period contemporary of the city drawn by Philibert Vasserot. P. Vasserot was the chief geometer in charge of the elaboration of the cadastral map (topological record). The Cécile Souchon's article published in 2005 lights up several points; the author points out that the maps indicate the divisions between the ground, stairs, courts, gardens, water wells, and ovens. Moreover, the Souchon's article deals with the paper layout called "Grand Aigle" (105x75cm) and the scale was changed by the geometer if the concerned quarter was too large to fit the paper sheet. Commonly, the documents are recorded using a range of scales from 1/200 to 1/500. Lastly, the article expresses that parcel boundaries walls are stippled with black ink. Now, from the raster image itself, M. Denès made some remarkable observations. Are drawn by instruments: the property limits, walls and gardens. The thickness is set once and for all: frontage walls and dividing walls have the same thickness. Finally, most of painted details are hand-drawn: stairs, common ovens, latrines, etc. From this work, a taxonomy of the cadastral map has arisen providing a natural and textual description of objects that can be found in the rasters. To take note of this knowledge, we created a meta-model of cadastral map that can be handled by computer algorithms. The meta-model is displayed in figure 4.5 under the Unified Model Language (UML) formalism. This latter denotes inter-relationship of different elements inside the system. The data semantics of cardinality, categorization, N-ary relationship are represented and from this complete expression, we focus our attention on a subpart which deals with frame, street, quarter, parcel and text elements. This restraint meta-model is shown in figure 4.6. Simply, this diagram expresses the basic content of the map and how items are laid out: Cadastral map has a frame which encircles the street names, they are materialized as text components. These connected components surround each quarter and text cannot be localized inside a quarter. Finally, a quarter contains many parcels, there is no defined cardinality for that.

4.3.1 Concept definitions

Before explaining our methodology, we want to visualize what a parcel is for an historian. In figure 4.7, a parcel was manually vectorized by an expert. This parcel

[1]The École nationale supérieure d'architecture de Versailles is a French architectural school located at the ancient stables of the Versailles Palace. The school was founded in 1969 after the suppression of the École des beaux-arts architecture section. Architect Jean Castex was one of the school's founders, while Nicolas Michelin (founder of the group Labfac) is its current managing director.

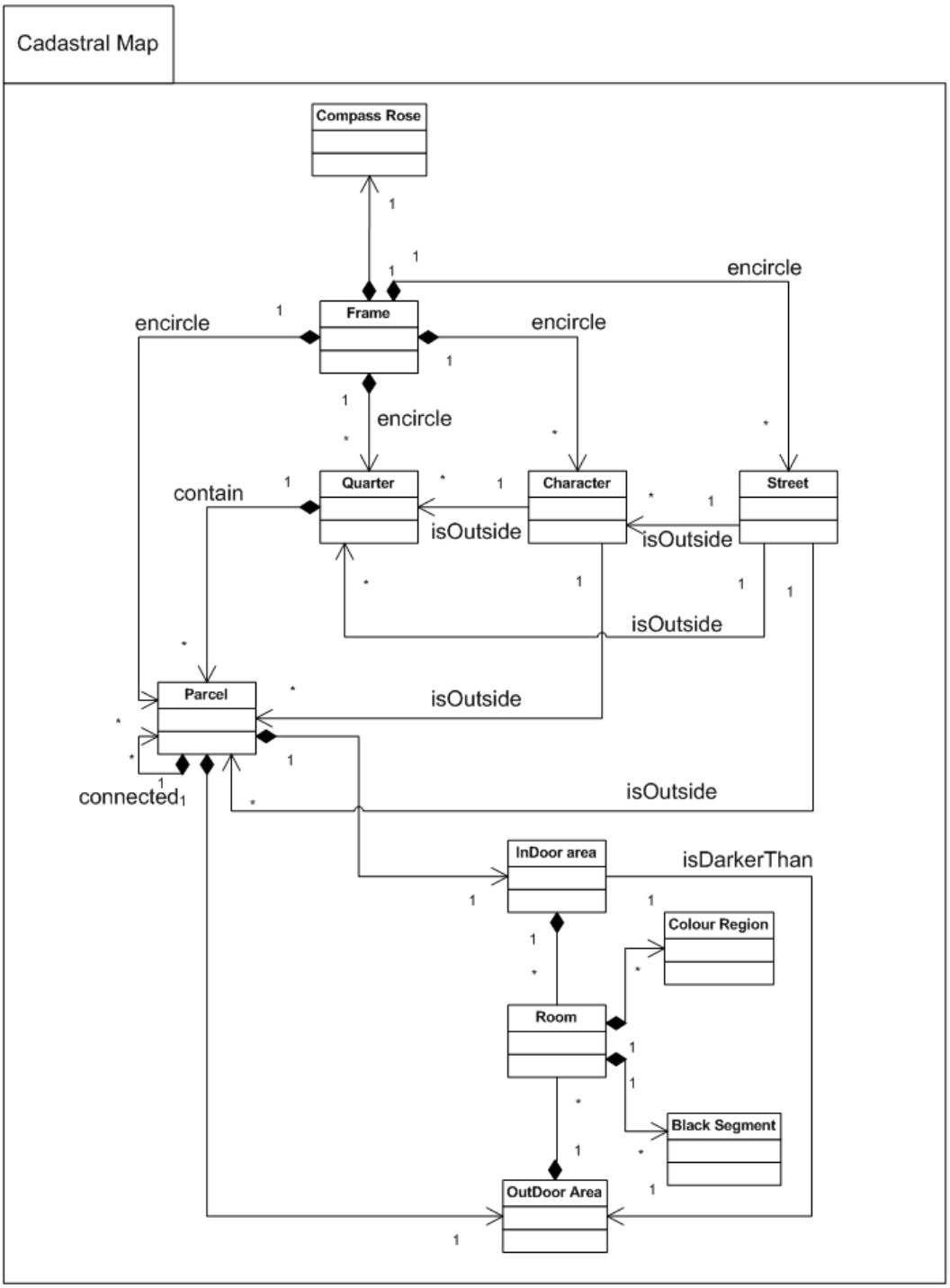


Figure 4.5: Expert-designed meta-model of cadastral map

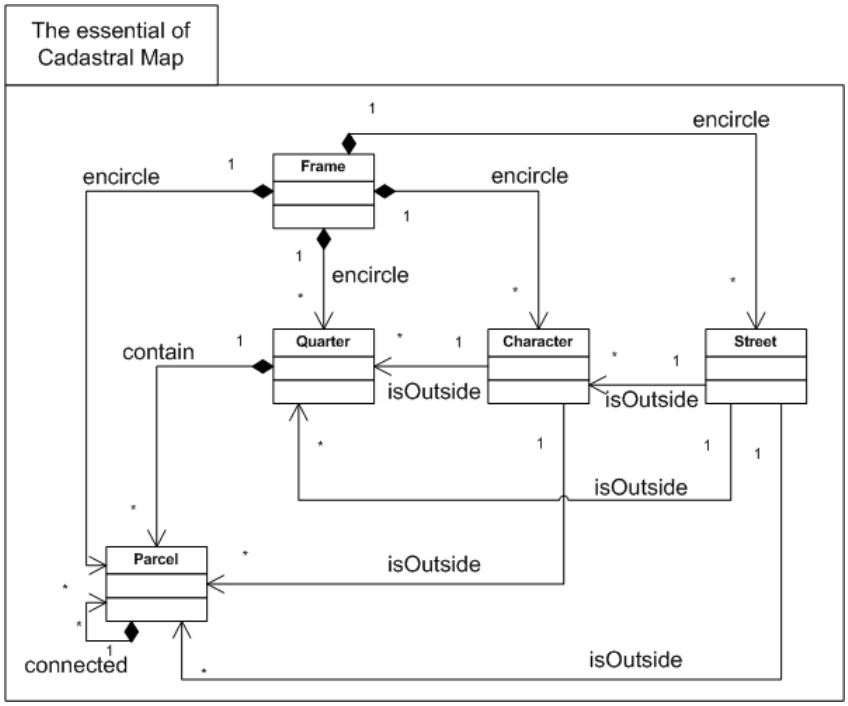


Figure 4.6: Limited cadastral map meta-model. Logic of description.

specification leads to the description of the bricks constituting a parcel.

Definition 2. *Parcel properties:*

- *is inside a quarter*
- *is outlined by thick lines*
- *two adjacent parcels cannot have the same color*

From fine to coarse, figure 4.8 brings to view examples of a map with two quarters. In fact, an image can hold from one to four quarters. With the time and when working with the maps for a quite long time, clear conclusions can be drawn about quarters:

Definition 3. *Quarter properties:*

- *is surrounded by streets*
- *is text-less.*

By extension, a group of quarters has some basic properties:

Definition 4. *A set of quarters:*

- *is surrounded by streets*

(a) Source image (b) Parcel polygonized by an historian

Figure 4.7: Parcel: a visual specification

- *is centered in the middle of the map*
- *is apart from others items (rulers, compass rose...)*

Here, the notations used to model our problem are defined.

Definition 5. *(Frame, Quarter, Street, Character, Parcel) Is defined as Frame, Quarter, Street, Character, Parcel any objects identified by the Frame, Quarter, Street, Character, Parcel detectors, respectively.*

4.3.2 Relation definitions

Definition 6. *(isOutside) An object isOutside another object if they do not share any features.* $obj1$ *and* $obj2$ *have no common points. This fact can be represented by the following statement:*

$$|obj1 \cap obj2| = \emptyset$$

and,

$$|obj1 \cup obj2| = |obj1| + |obj2|$$

This relation is symmetric and a binary relation is symmetric if it holds for all a and b that if a is related to b then b is related to a.

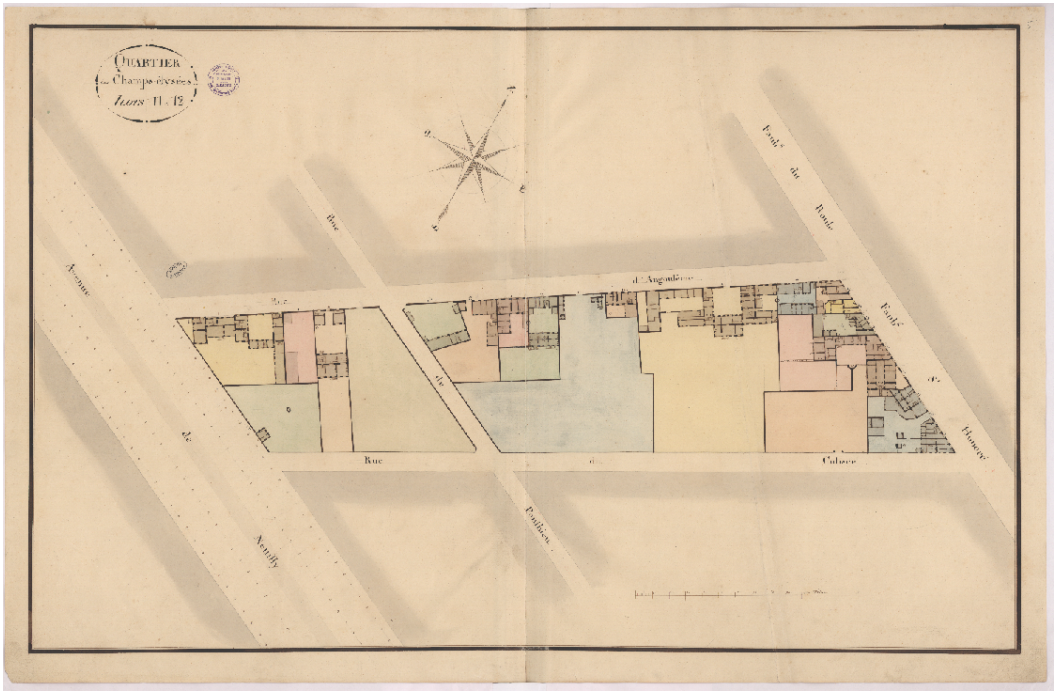
Definition 7. *(contain) This relationship expresses the fact that an object is a composition of others objects. An object overlaps another. We define that an* $obj1$ *contains* $obj2$ *if they verify the following relation:*

$$|obj1 \cap obj2| = |obj2|$$

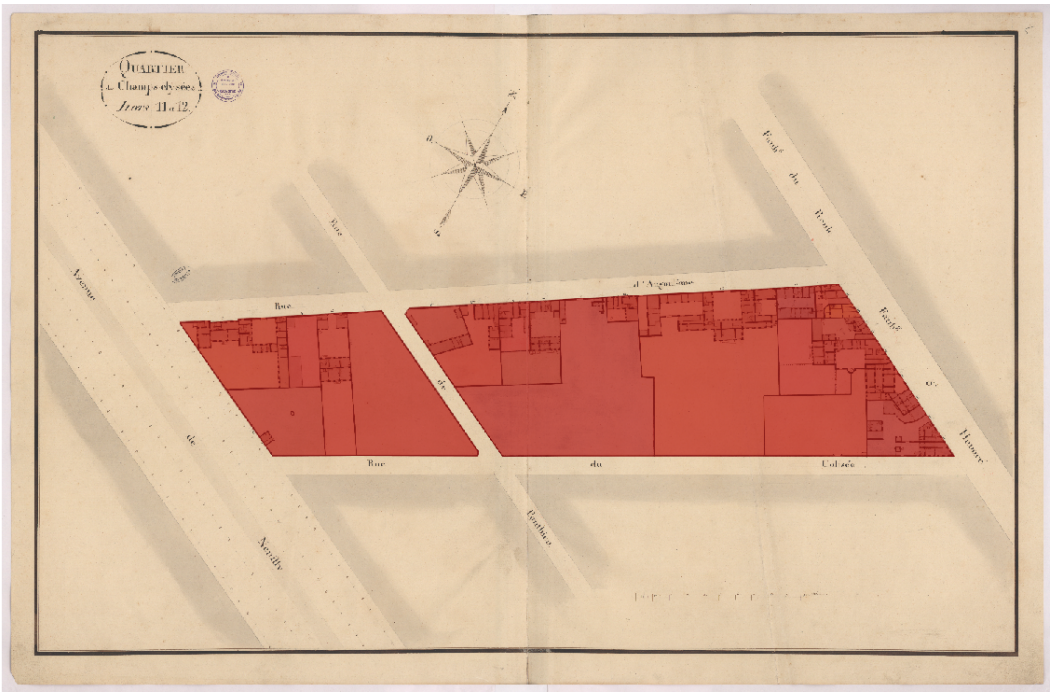
and,

$$|obj1 \cup obj2| = |obj1|$$

This relation is transitive, a logic relation between three elements such that if it holds between the first and second and it also holds between the second and third it must necessarily hold between the first and third.



(a) Source image



(b) Quarter polygonized by an historian

Figure 4.8: Quarter: a visual specification

Definition 8. *(connected) When two white areas are only separated by a black pixel border of one pixel width.*

Definition 9. *(encircle) When an object surrounds another without intersection. obj*1 *encircles obj*2 *if they verify the following relation:*

$$|BoundingBox(obj1 \cup obj2)| = |obj1|$$

and,

$$obj1 \cap obj2 = \emptyset$$

This relation is transitive ans is applied to the elements related to obj1.

Finally, the cardinality of the relations between concepts was analyzed as follows:

Many To Many:

- A case study: Courses are taught by one or many teachers. One teacher can teach on one or many courses.
- In our case : Characters are outside one or many Quarters. One Quarter can have one or many Characters outside.

One To Many:

- A case study: A course has one or many students.
- In our case : A Frame encircles one or many Quarters.

Many To One:

- A case study: Many students belong to one department.
- In our case : Many Parcels belong to one Quarter.

One To One:

- A case study: Office numbers are associated with a unique address.
- In our case : One cadastral map has a unique Frame.

From this model, a list of objects to be retrieved has been enumerated. In the next section, we present the image processing tools especially designed to locate and isolate the characters, the streets, the frame, the quarters and the parcels.

4.4 Object extraction

In this section, a description of the different object detectors run on the contour image is presented. In a first step, we just recall the main steps of the color processing where the goal is to unleash the graphical and textual information composing the map.

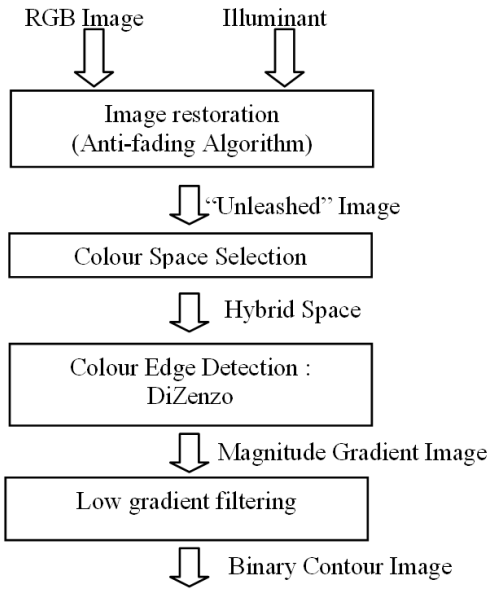


Figure 4.9: Colour analysis flow chart

4.4.1 Reminder on color processing

Figure 4.9 points out the color processing organization. This analysis can be seen as a primitive extraction scheme: Color restoration, color space selection and edge detection. These stages aim to extract homogeneous regions of our documents. From the chapter 3, we have described in section 3.7 an image segmentation method for hybrid color spaces. The contour force for each pixel (x,y) is given by the equation 3.12. This equation is recalled below:

$$Edge(x, y) = \sqrt{\lambda_+ - \lambda_-}$$

In this equation, $\lambda_\pm$ are eigen values of the vectorial gradient of 2 neighbor pixels. These edge values are filtered using a two class classifier based on an entropy principle in order to get rid off low gradient values. At the end of this clustering stage a binary image is generated. This image will be called as contour image through the rest of this chapter. Finally, regions are extracted by finding the white areas outlined by black edges (figure 4.10). Note that this binary contour image (figure 4.10) is much more consistent than a usual image given by the binarization of the luminance channel.

At this point, we describe our object extraction strategy from the so called contour image. Image processing tools are run from a double expertises : (i) Knowledge on the data; (ii) Knowledge on the image processing. The conjunction of both information should lead to a model driven image processing scenario. Where image detectors are automatically adapted to extract a set of defined graphic elements. A lack of time prevented us to achieve this investigation however a beginning of answer is presented in [Raveaux 2010]. In cite[Raveaux 2010], a method integrat-

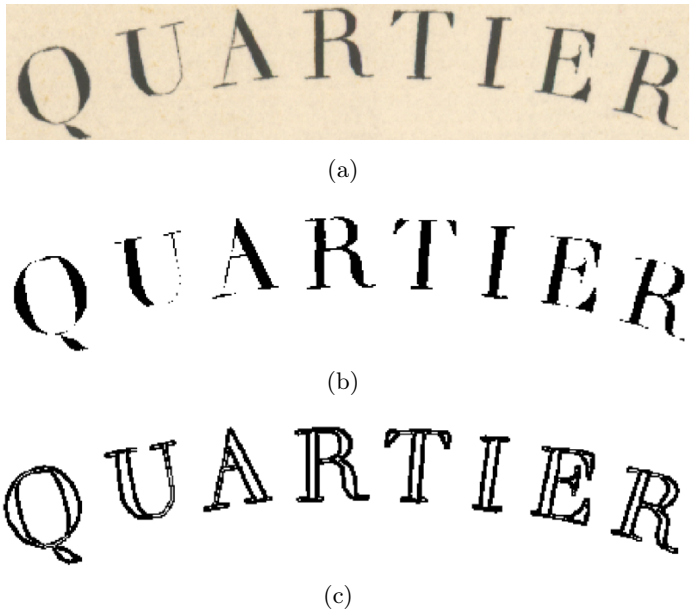


Figure 4.10: From top to bottom: source image ; Binary from luminance ; Contour image

ing efficiently a semantic approach into an image segmentation process is proposed. A graph based representation is exploited to carry out this knowledge integration. Firstly, a watershed segmentation is roughly performed. From this raw partition into regions an adjacency graph is extracted. A model transformation turns this syntactical structure into a semantic model. Then the consistence of the computer-generated model is compared to the user-defined model. A genetic algorithm optimizes the region merging mechanism to fit the ground-truth model.

4.4.2 Methodology

A peeling the onion approach is adopted. Image processing is carried out one at the time and sequentially, removing step by step a layer of information. The algorithm sequence is presented in figure 4.11. Tiny elements and characters make the drawing too dense and hard to interpreter. To reduce the complexity of the problem, the first step has as an objective to separate characters from and graphic elements. Our approach is a selective method operating on the graphic layer. Dedicated image algorithms locate and filter objects surrounding the quarters. Once quarters are isolated, they are put into different images, one binary image per quarter to be further analyzed. Each quarter is a parcel container but algorithms has to be more sensitive and more locally applied to precisely retrieve parcels within a given quarter. This refine method takes into the color information contained into the contour image but also, the background of the map representing the thick walls of the buildings. Combining the contour image and the black layer leads to a robust parcel detection. Finally, when detected parcel edges are vectorized. This “peeling the onion” strategy

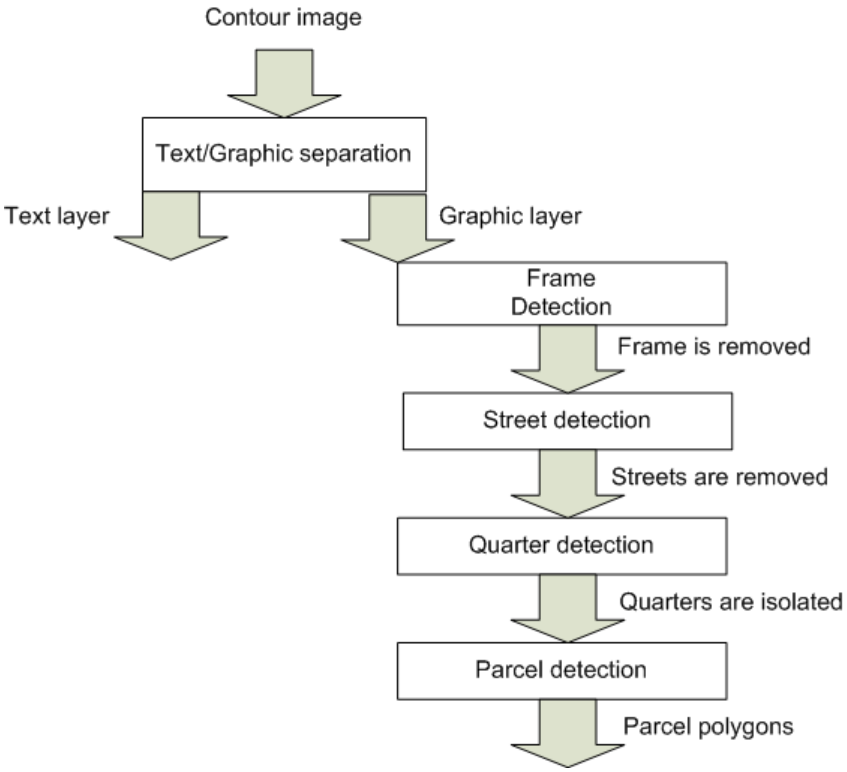


Figure 4.11: Object extraction process.

starts by a text/graphics segmentation presented in the next part.

4.4.3 Text/graphic separation

In our case, we assume that characters do not overlap the graphics. Considering this assumption the segmentation is performed from the contour image where connected components are treated as the basic construct of our approach. The four necessary steps to achieve this task are proposed in figure 4.12.

Pre-processing: clustering Connected components are clustered into two groups according to their number of pixels, the CLARA [Kaufman 1990] algorithm is involved in this process. Black areas are then labeled as small or large. The rest of the method will only focus on connected components tagged as “small”. From this point the question can be stated as a two class problem. To categorize a given CC as text or graphic, a complete classification chain is carried out.

Representation: Graph data set In a first step, considering each "small" CC as a binary image, both black and white connected components are extracted. These connected components are then automatically labeled with a partitional clustering

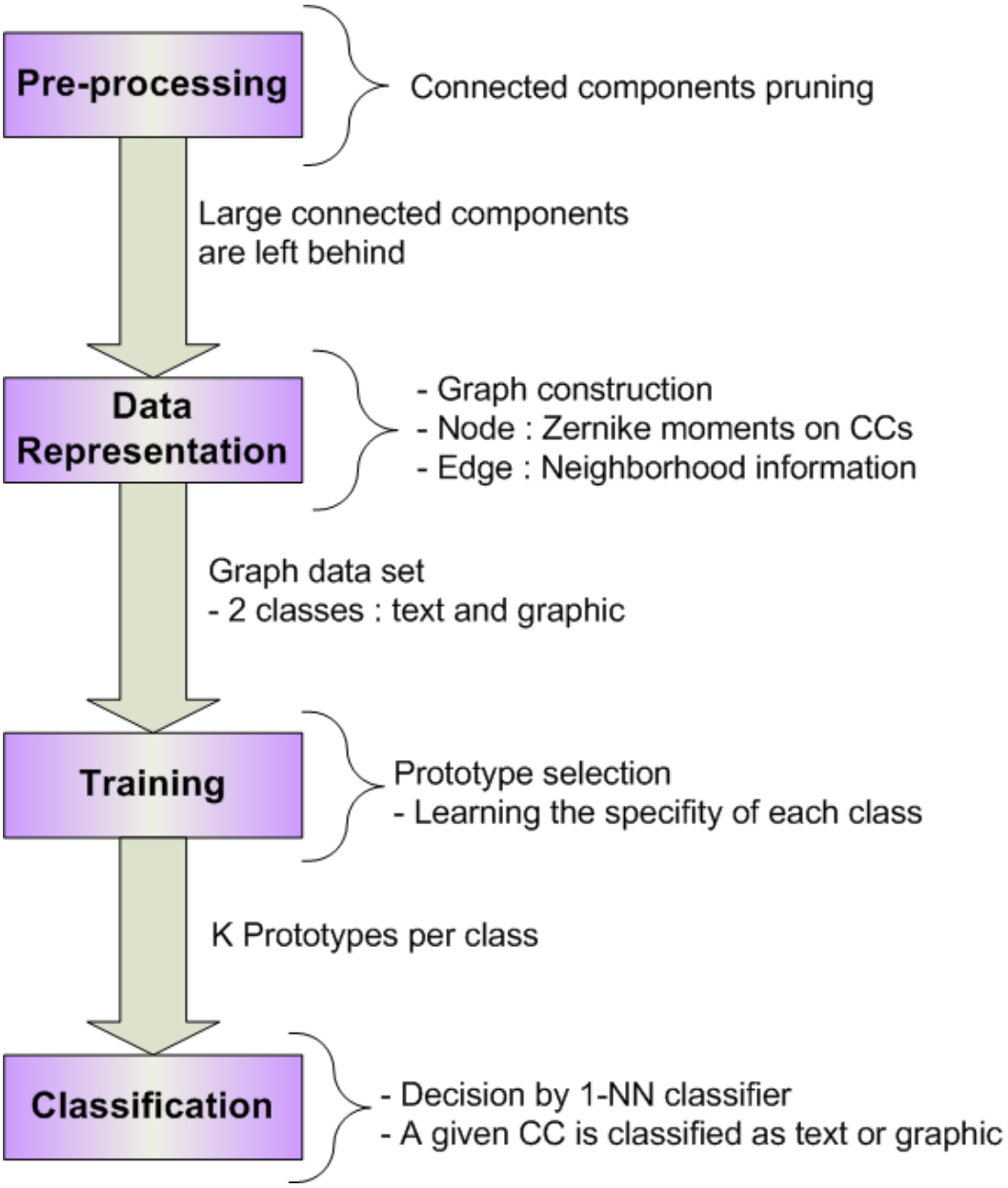


Figure 4.12: Text/Graphic separation scheme: An overview.

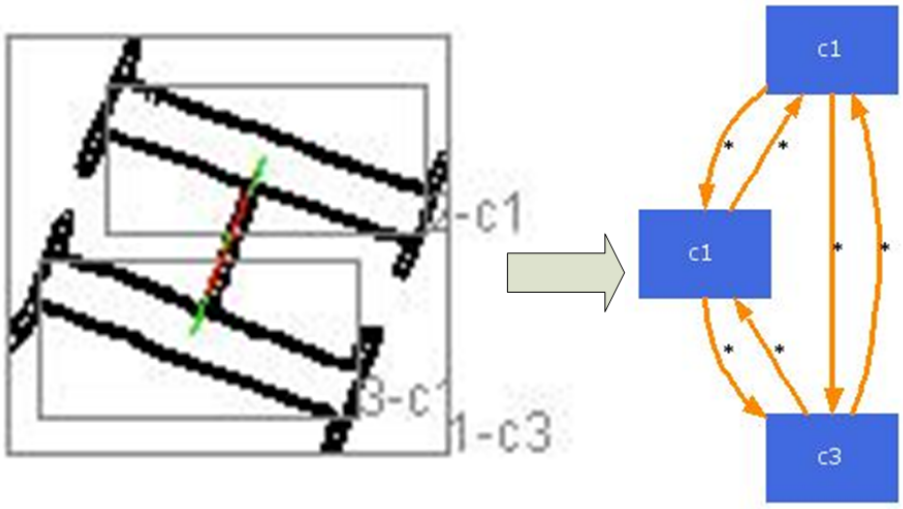


Figure 4.13: From image to graph

algorithm [Kaufman 1990] using Zernike moments as features [Khotanzad 1990][2]. Using these labeled items, a graph is built. Each connected component corresponds to an attributed vertex in this graph. Then, edges are built using the following rule: two vertices are linked with a directed and unlabeled edge if one of the vertices is a neighbor of the other vertex in the corresponding image. This neighborhood is decided according to the distance between the centroids of each connected components with respect to a predefined number of neighbors (h). The two values h and c, concerning respectively, the number of clusters found by the clustering algorithm and the number of significant neighbors, are issued from a comparative study. An example of the association between drop cap image and the corresponding graph is illustrated in figure 4.13.

Discussion on parameters h, and c. We chose the couple $< c, h >$ according to one criteria: the minimization of the silhouette index [Rousseeuw 1987]. We have tuned the number of clusters from one to 10 and the number of neighbors from one to three.

Definition 10. *(Silhouette) For each observation i, the silhouette width $s(i)$ is defined as follows: Put $a(i)$ = average dissimilarity between i and all other points of the cluster to which i belongs (if i is the only observation in its cluster, $s(i) := 0$ without further calculations). For all other clusters c, put $d(i,c)$ = average dissimilarity of i to all observations of c. The smallest of these $d(i,c)$ is $b(i) := \min_C d(i,c)$, and can be seen as the dissimilarity between i and its "neighbor" cluster, i.e., the nearest one to which it does not belong. Finally, $s(i) := \dfrac{(b(i) - a(i))}{max(a(i), b(i))}$.*

Observations with a large $s(i)$ (almost 1) are very well clustered, a small $s(i)$

[2]Zernike moments will be further described in chapter 7

(around 0) means that the observation lies between two clusters, and observations with a negative $s(i)$ are probably placed in the wrong cluster. For each configuration the silhouette index was computed. Experimentally, we found that the pair $< c, h >$ that minimizes the most this index is $< c = 5; h = 2 >$.

Training: Prototypes selection The learning algorithm consists in the generation of K graph prototypes per class for a group of N classes. These prototypes are produced by a graph based Genetic Algorithm [Raveaux 2007b], it aims to find the near optimal solution of the recognition problem using the selected prototypes. In such a context, each individual in our Genetic Algorithm (GA) is a vector containing K graphs per class, that is to say K feasible solutions (prototypes) for a given class. Hence, an individual is composed of KxN graphs. The fitness (the suitability) of each individual is quantified thanks to the classification rate obtained using the corresponding prototypes and a test database. The classification is processed by a 1-Nearest Neighbor classifier using the graph probing distance [Lopresti 2003][3]. Then, using the operators described in [Raveaux 2007b], the GA iterates, in order to optimize the classification rate. The stopping criterion is the generation number. At the end of the optimization task, a classification step is applied on a validation database in order to evaluate the quality of the selected prototypes.

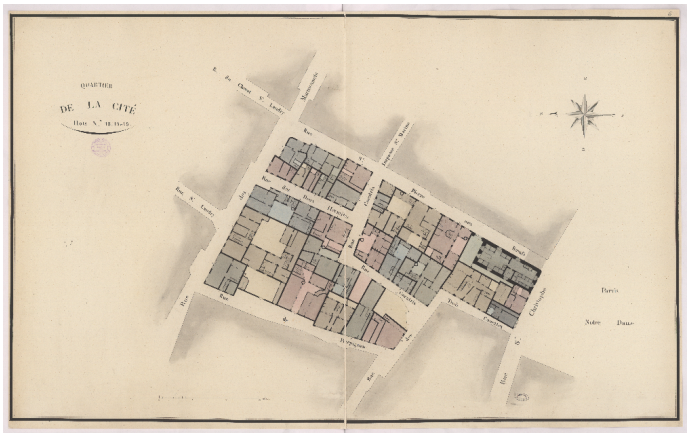
Classification Presenting an unknown CC as an input, the 1-NN classifier trained with the prototypes learned during the training phase takes the decision to categorize the given CC as Text or Graphic. An example of text/graphic segmentation is illustrated in figure 4.14. The full system, Prototype Based Reduction Scheme for Structural Data Classification is available online at the L3i-ALPAGE website[4]. In addiction, a complete description of algorithms and data structures involve in the graph prototype search are described in appendix B.

4.4.4 Frame detection

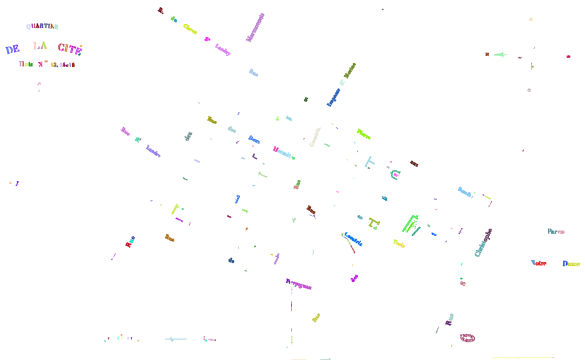
Each map is encircled by a frame. A frame is a thick stippled rectangle made with black ink. To detect the coordinates of the 4 segments defining the rectangle, a probe based approach is chosen. The continuity of the frame can be corrupted due to the folding problem or some additive noise, so a robust approach to determine segment coordinates and thickness is required. For each side of the image, a number np of probes are spread along a given axis at the middle according to a centered normal distribution. Thereafter, each probe follows a line crossing the image from one side to the other, orthogonally to the axis of distribution. When a probe hits a black pixel its progression is stopped and the coordinates are recorded, then the given probe restarts from its breakpoint until it reaches a white pixel, at this point, the coordinates are stored and the probe is destroyed. For each border, probe's

[3]The graph probing distance is fully described in chapter 6.

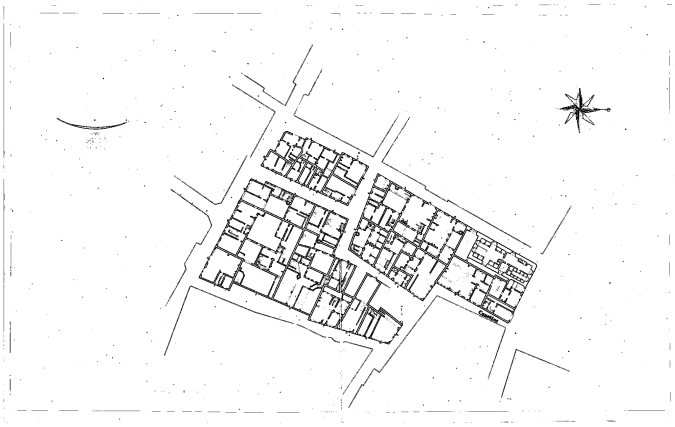
[4]http://alpage-l3i.univ-lr.fr/ -> A Prototype Based Reduction Scheme for Structural Data Classification



(a) Original image

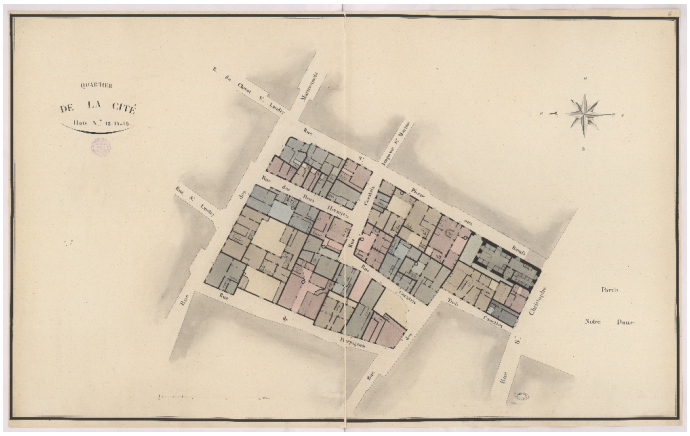


(b) Text part

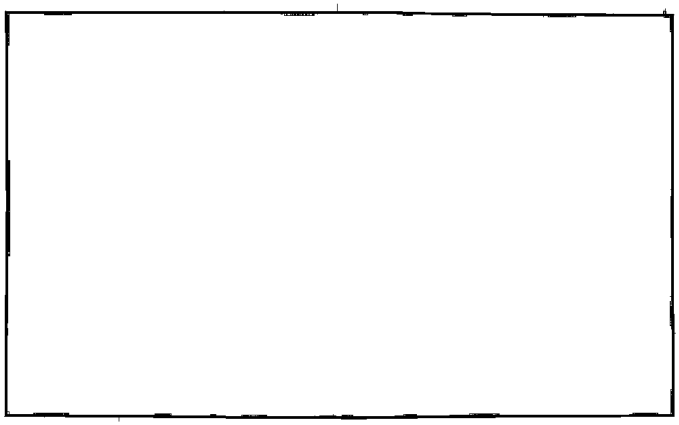


(c) Graphic image

Figure 4.14: Text/graphic decomposition.



(a)



(b)

Figure 4.15: Frame extraction stage.

coordinates and thickness are submitted to a voting scheme to determine the values that appear the most frequently. Figure 4.15 reflects the frame obtained after the detection phase.

4.4.5 Street detection

The graphic image is built by subtracting the text image from the original image. Therefore many small CCs and punctual information remain on this graphic image. Run-Length Smoothing Algorithm (RLSA) helps grouping together homogeneous CCs. Creating a more coherent graphic image. The basic idea of RLSA is to take advantage of the white runs existing in the horizontal and vertical directions. For each direction, RLSA eliminates white runs whose lengths are smaller than threshold smoothing values (sv, sh). Next, streets are easily detected and then removed according the criteria of density. In fact, a street is a long and thin stroke delineating a large surface, consequently the corresponding bounding box has a low

density of black pixels, figure 4.16.

4.4.6 Quarter extraction

This stage aims to locate where quarters are inside a given cadastral map. At first a merging algorithm gathers closely couples CCs in a unique structure, thus, outliers elements are left behind and isolated. In fact, one missing pixel is enough to break into pieces a connected component. To mend that collection, a neighborhood graph assembles closely located and concentrated CCs, but dislocated yet, in a single piece. Map decorations, compass rose and ruler are considered as outliers since they do not settle into our framework and they are most likely to be found on the edge of the map. These objects are spread into the map beyond our regions of interest that constitute quarters. Finally, only strongly coherent CCs are put into a single graph structure preserving the main information, which is to say the quarters. The whole principle is presented in figure 4.17. This strategy begins by the construction of a graph. This latter is arranged by a node merging process then a pruning algorithm removes small regions to keep only the significant information. Finally, an active contour is applied to delineate boundaries of each remaining regions.

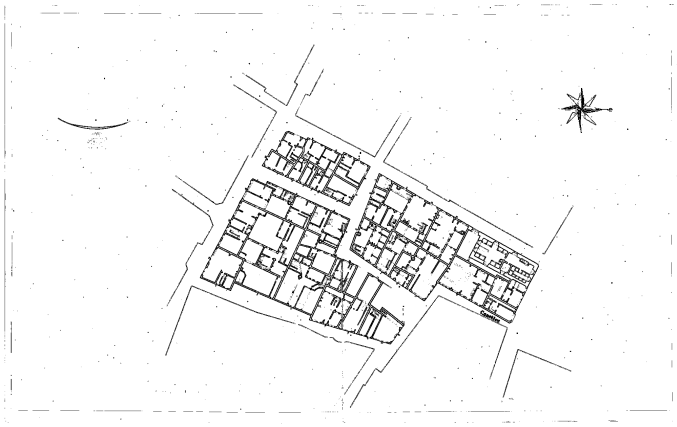
Neighborhood graph Each connected component represents an attributed vertex in this graph. Then, edges are built using the following rule: two vertices are linked with an undirected and unlabeled edge if one of the vertices is a neighbor of the other vertex in the corresponding image. The h value, concerning the number of significant neighbors, is issued from a comparative study. The value h is set to two, this value is inherited from section 4.4.3.

Merging This step aims to merge CCs spatially close to each others. When a given node $n1$ is merged with another one $n2$, the merged node $n1$ is deleted and its edges are linked to $n2$. The algorithm 1 makes a reference to a distance between nodes corresponding to a distance between CCs. This problem can be stated as follows: A given set of CCs is composed of N pixels: $P = \{P_i \equiv (x_i, y_i)\}_{i=1}^{N}$, therefore :

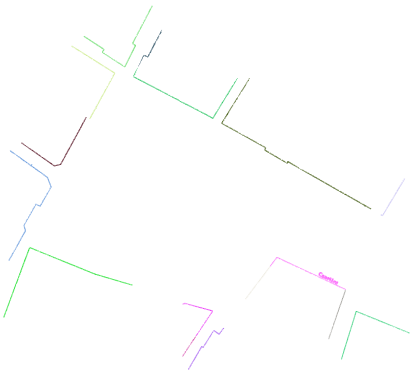
$$d(Node1, Node2) = d(n1, n2) = \min_{d} \left(\sum_{i=1}^{N} \sum_{j=1}^{N} d\left(P_i^1, P_j^2\right) \right) \tag{4.1}$$

This neighborhood is decided according to the distance between the centroids of each connected components with respect to a predefined threshold.

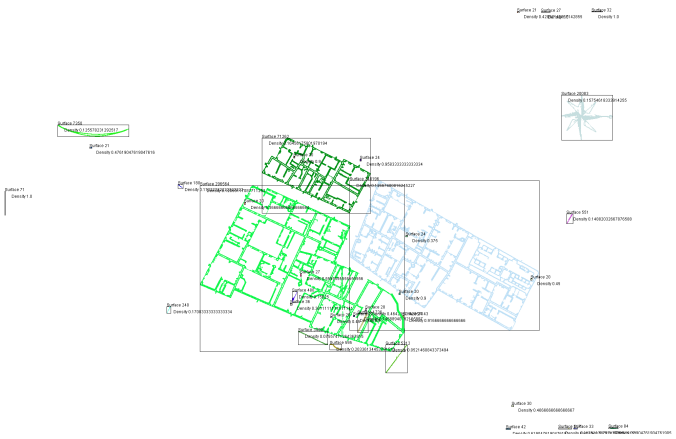
Significant CCs After the merging process, a given node is a set of CCs that will be named region. From the merged neighboring graph, only the biggest nodes in term of surface are considered and analyzed. Behind this assumption relies the hypothesis that the quarters represent the main information into a cadastral map, so, quarter regions have the biggest surfaces. When keeping the biggest nodes, we preserve the main information, figure 4.18. This simple pruning algorithm is summarized in Algo. 2.



(a) Graphic image



(b) Streets identified by colors



(c) Bounding Boxes of remaining pieces

Figure 4.16: Street detection using density criteria.

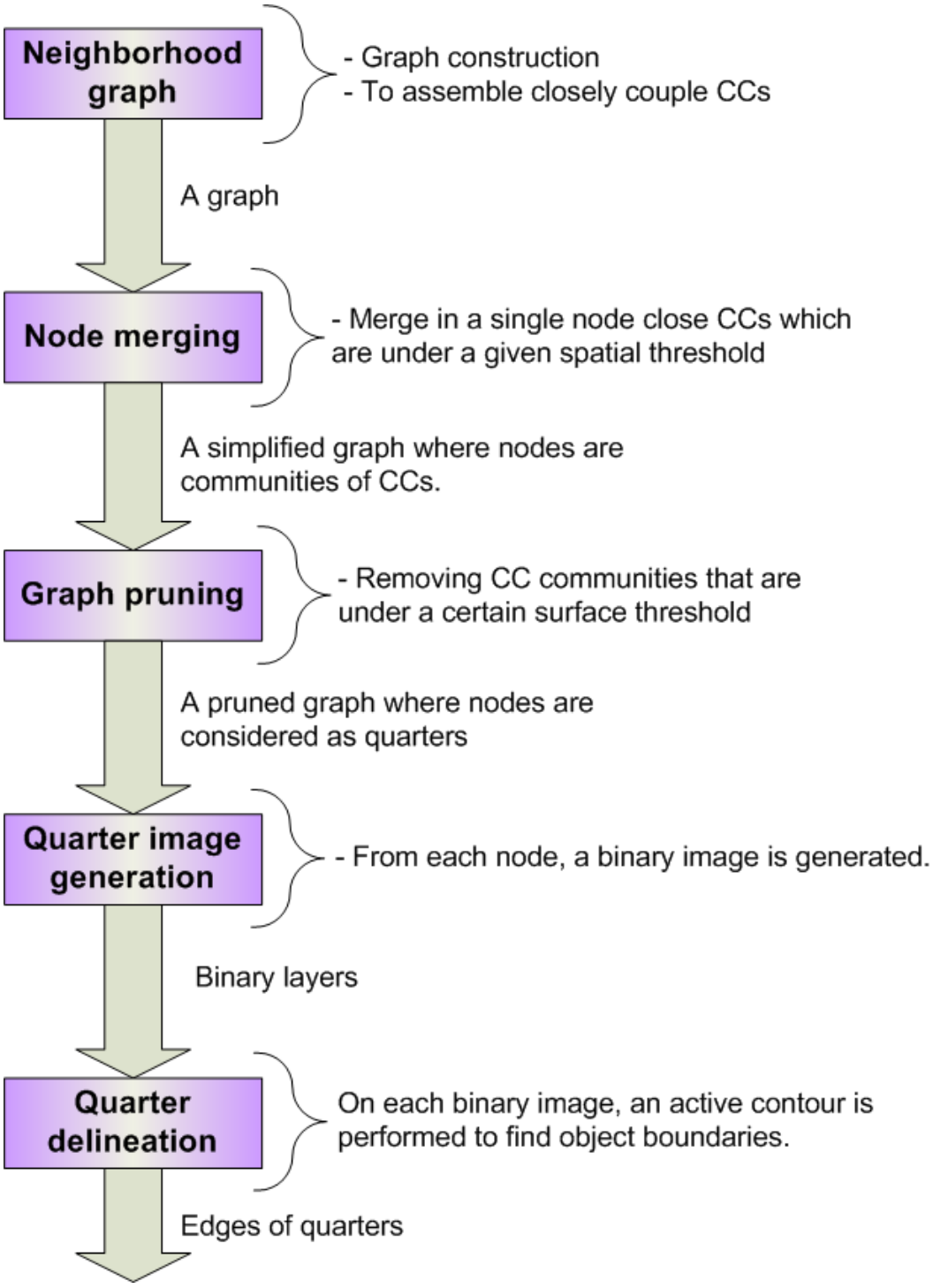


Figure 4.17: Overview of the quarter extraction scheme

Algorithm 1 Merging scheme for graph data structure

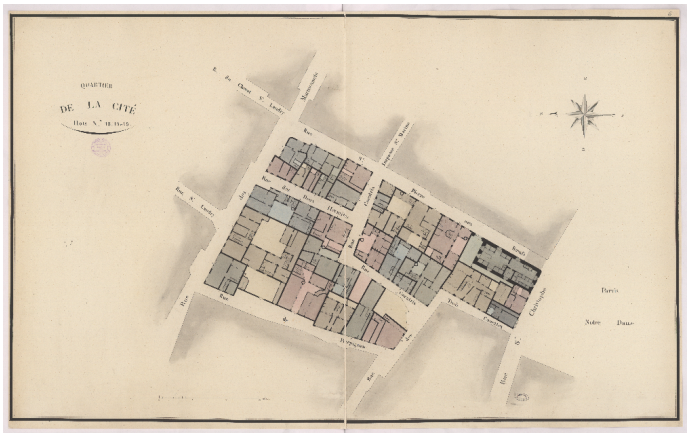
Require: A spatial distance threshold: T
Require: The number of significant neighbors: h
Ensure: A simplified graph.

```
Start
MergingFlag = true
while MergingFlag == true do
  MergingFlag=false
  for i = 1 TO Number Of Nodes do
    N(i): the i^th node.
    for j = 1 TO h do
      N(j): the j^th node of the neighboring.
      if d(N(i),N(j)) < T then
        MergeNodesInGraph(N(i),N(j))
        MergingFlag = true
      end if
    end for
  end for
end while
End
return The merged graph
```

Algorithm 2 Surface pruning algorithm for a neighborhood graph

Require: A surface threshold: S
Ensure: A pruned graph.

```
Start
for i = 1 TO Number Of Nodes do
  N(i): the i^th node.
  s : getSurface(N(i))
  if s < S then
    DeleteEdgesFromTheNode(N(i))
    DeleteNodeFromTheGraph(N(i))
  end if
end for
End
return The pruned graph
```

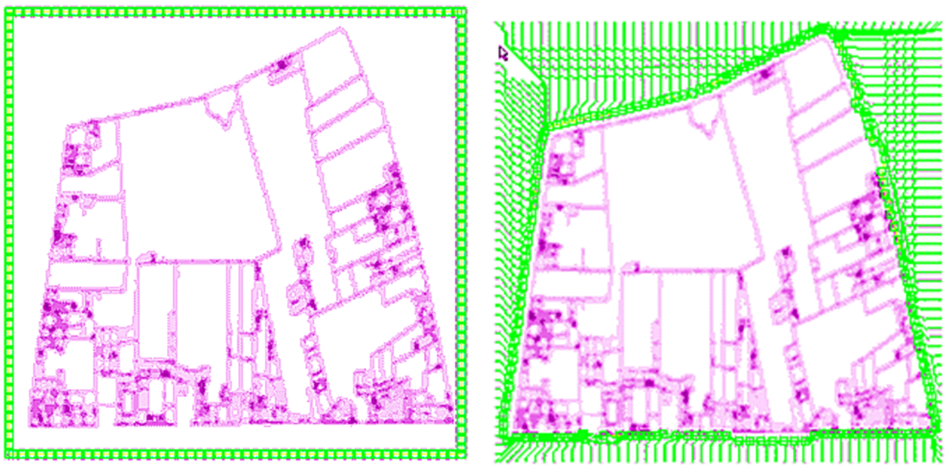


(a) Source image

(b) Connected components their bounding box

(c) Remaining connected components after pruning mechanism based surface criteria

Figure 4.18: Connected Components pruning.



(a) (b)

Figure 4.19: (a) Snake initialization; (b) Snake progression

Active contour: Snake Once the significant regions are identified from the former step, we still need to delineate the contour of the quarters. Active contours, or snakes, are computer-generated curves that move within images to find object boundaries, see [Kass 1988] and [Leymarie 1993] for more details. On each quarter an active contour is performed to smoothly find the edges. An application of snake is depicted in figure 4.19.

Contours are initialized with a distance between adjacent snake points ('snaxels') of approximately 3 pixels. Each snaxel moves under the influence of internal and external constraints:

- *Continuity*: An internal constraint that specifies that a snaxel should locate itself so as to make the distance between itself and its neighbors close to the average distance between snaxels. This version of the continuity constraint tends to cause the snake to shrink.

- *Curvature*: An internal constraint that specifies that a snaxel should locate itself so as to minimize the amount of curvature it introduces; that is, it should maximize the angle it defines between its neighbors.

- *Gradient*: An external constraint that specifies that a snaxel should locate itself in areas where the gradient in the image is large. Here, on the specific case of binary image, the gradient is extremely high (1.0) when a black pixel is encountered and is worth zero for a white pixel.

To initialize a contour, the quarter bounding box is adopted. The contour and the points that are used in the contour are shown in green in figure 4.19. The snake moves 100 iterations at maximum. When the snake reaches steady state and no more points are moved on an iteration, the program no longer refreshes the contour and stops. The three parameters above the image (continuity, curvature, and gradient)

can be changed from zero to 'very high'. Each of these affects the behavior of the snake in a particular way. After changing the parameters, we have adopted empirically the most suitable solution to best fit the quarter digital curve while preventing noisy/disturbed contours. The best trade-off was found when continuity and curvature are set to "Low" while gradient is set to "High". Note that it is the values of these weights relative to each other that is important; setting all three values to "High" will give exactly the same results as setting them all "Low".

Discussion on parameters Merging and pruning the neighborhood graph involve two parameters. The first one, T is related to the distance between regions. In the algorithm 1 two regions are merged if they are closed to each others which is to say if the minimum distance (simple linkage) between the regions is less than T. In the merging phase, we aim at assembling CCs that are broken because of the time due degradation. Experimentally, a value for T equal to 15 pixels is enough to merge closely coupled CCs without creating aggregating unwanted elements. This can be executed thanks to the removal of small CCs during the text/graphic phase, the graphic layer is quite clean and light in term of information. The flip side of coin is that small pieces of graphics can be missed. The second parameter is related to the surface of the quarters. In the algorithm 2, regions having a surface less than S are removed. Behind this assumption lies the fact quarters are the main objects drawn into the map and so, they have a significant surface. This pruning is helpful to remove ruler, compass rose, decoration and noise. After a series of tests, a value of $S = 60000\ pixel^2$ was convincing enough to remove undesirable objects and to keep the quarters into the graph. Finally, the quarter extraction according the explained thresholds is evaluated in section 4.6.

4.4.7 Parcel Extraction

Once the quarter information is isolated, the next stage is to locate parcel information. In order to achieve that goal a series of processes have to be done. This data flow process is brought to view in figure 4.20.

Methodology: The main motivation is to obtain parcel polygons to be inserted into a Geographical Information System. Such an application has some geometric requirements. In fact, polygons must be topologically consistent. By consistent, we mean that two adjacent parcels must share at least a common edge. The focus is given to each quarter area in order to refine the analysis. Images are processed independently from other objects (Frame, Characters, Streets). To better identify the problem, a binary image is retrieved from the automatic thresholding of the luminance channel. The binarization of the gray scale image issued from the luminance is performed thanks to the Otsu's algorithm. Otsu's method is used to automatically perform histogram shape-based image thresholding [Sezgin 2004] or, the reduction of a graylevel image to a binary image. The algorithm assumes that

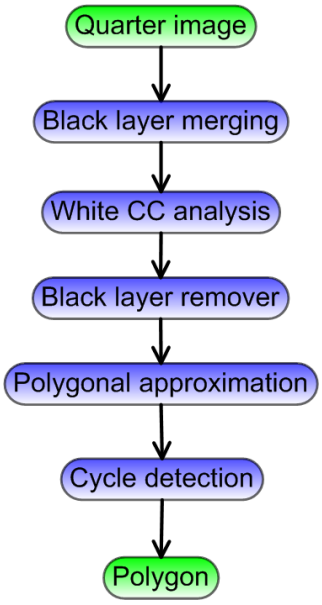


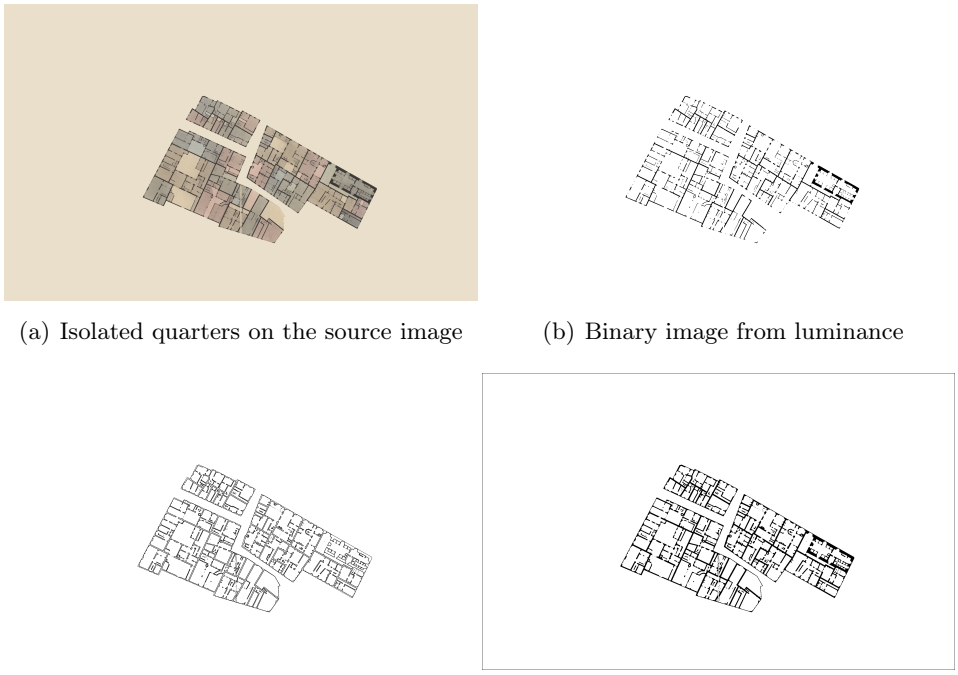
Figure 4.20: Overview : From Quarter image to vectorized parcels

the image to be thresholded contains two classes of pixels (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal [Ostu 1979]. Contour and binary images are merged together to improve the parcel detection. An example of this merging operation is illustrated in figure 4.21.

The objective by merging the black layers is to better define the contour of parcels and not to lose information. On the merged image a white component analysis is performed. Each white connected component is a continuous and closed area representing a parcel. Unfortunately, these objects do not respond to the topological constraints that are required by GIS. To respect this topological constraints, the black layer is removed iteratively. Median axis is found by a thinning algorithm performed on black pixels. This is illustrated on figure 4.22. Thereafter, parcels contour are processed to find junction points thanks to an image chaining analysis, see figure 4.23.

Hence, parcel contours are vectorized which is to say transformed into a set of segments using a digital curve approximation method, see figure 4.24. Finally, the next paragraph will describe more precisely the different algorithms that are involved in the parcel extraction.

Black layer remover: A quick reminder brings us to mention that parcels are found by a white connected component analysis. Unfortunately, in this configuration, parcels are not touching each others, black pixels are separated them. The black layer is progressively removed by using an adapted median filter. This modified version of the median filter operates only on black pixels. When a black pixel is encountered a voting scheme is set up. On the neighborhood, each non-black pixel



(a) Isolated quarters on the source image

(b) Binary image from luminance

(c) Contour image

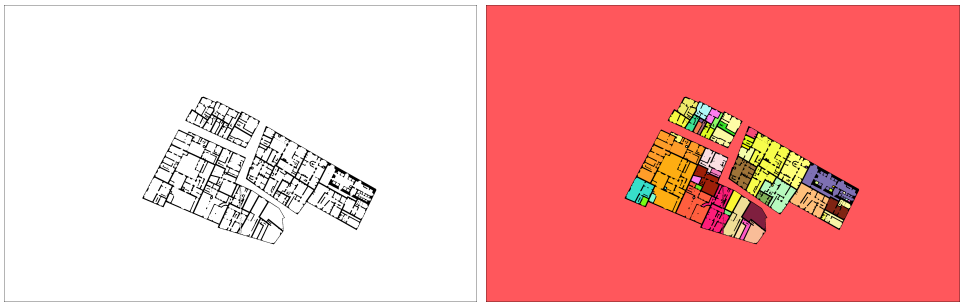
(d) Binary and contour images are merged

Figure 4.21: Refined quarter analysis

is considered and expresses its vote by giving its color value. The color that appears the most often is elected and stored in the output image. When the entire image is processed, the output image becomes the current image and the voting filter is iterated until no black pixels remain. Step by step, the black pixels are substituted by the color that is the most likely to appear in the neighborhood, conventionally, a 5x5 mask size is adopted. A sample image before and after this image processing stage is showed in figure 4.25.

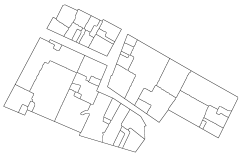
After this operation, each parcel has unique color label. From the label image, parcel contours are exacted to be further handled, which is to say polygonized. Figure 4.26 denotes the binary image where contours are ready to be polygonized.

Digital curve approximation; vectorization: Black pixels are vectorized using a polygonal approximation based on a genetic algorithm. In this method, the optimization/exploration algorithm locates breakpoints on the digital curve by minimizing simultaneously the number of breakpoints and the approximation error. Using such an approach, the algorithm proposes a set of solutions at its end. This set which is called the Pareto Front in the multi objective optimization field and it contains solutions that represent trade-offs between the two classical quality criteria of polygonal approximation : the Integral Square Error (ISE) and the number of vertices (nv) (i.e. in [Locteau 2006]). This method is threshold-less and postpones the choice of a specific solution at the end of vectorization process. Figures 4.27 and



(a) Binary and contour images are merged (b) White connected components analysis

(c) Average color region (d) Black layer removal



(e) Parcel Contour

Figure 4.22: Parcel location

(a) Full view (b) Zoom in

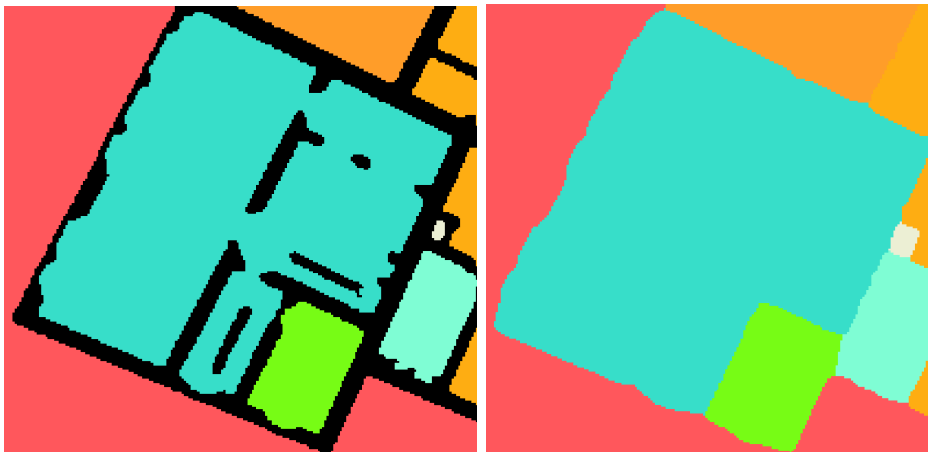
(c) Intermediate point display (d) End point display only

Figure 4.23: Image Chaining

(a) Full view (b) Zoom in

(c) Raster and vector (d) Break points visualization

Figure 4.24: Polygonization



(a) Before (b) After

Figure 4.25: Black layer removal

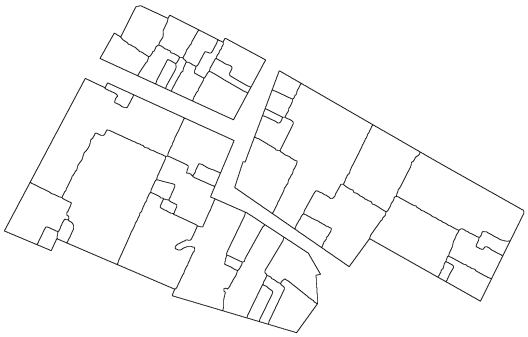


Figure 4.26: The parcel contours are materialized. Digital parcel curves ready to be polygonized

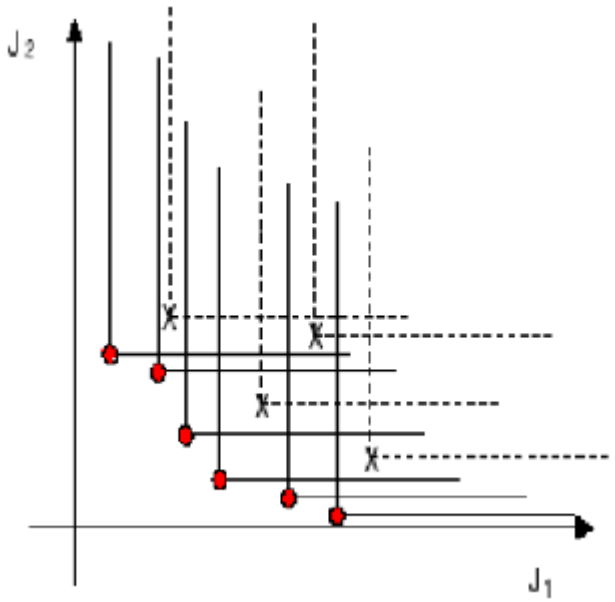


Figure 4.27: An illustration of the Pareto-front principle

4.28 illustrate the principle of Pareto front.

At the end of this stage we obtain a set of dominant solutions, they are all valuable and pertinent according a certain point of view, however in order to retrieve a unique solution, we have to compromise. On the Pareto curve, we compute the average ISE (AISE) as the area under the ISE-nv curve. This average value can be represented as a horizontal line on the solution plot. Finally, we choose the solution which is the closest to the point created by the intersection between the AISE line and the Pareto curve. An example of polygonization is depicted in figure 4.29.

Polygonizer. From lines to polygons: Detecting polygons defined by a set of line segments in a plane is an important step in the analysis of vectorial drawings. To perform polygon detection from a set of line segments we divide this task in four major steps. First we detect line segment intersections using the Bentley-Ottmann algorithm [J.L.Bentley 1979]. Next step creates a graph induced by the drawing, where vertices represent endpoints or proper intersection points of line segments and edges represent maximal relatively open subsegments that contain no vertices. The third step finds the Minimum Cycle Basis (MCB) [Syslo 1981] of the graph induced in previous step, using the algorithm proposed by Horton [Horton 1987]. Last step constructs a set of polygons based on cycles in the previously found MCB. This is straight-forward if we transform each cycle into a polygon, where each vertex in the cycle represents a vertex in the polygon and each edge in the cycle represents an edge in the polygon (i.e. in [Jr 2003]). Each polygon can be considered as a parcel. The polygon reconstruction from Line cross-sections is displayed in figure 4.30.

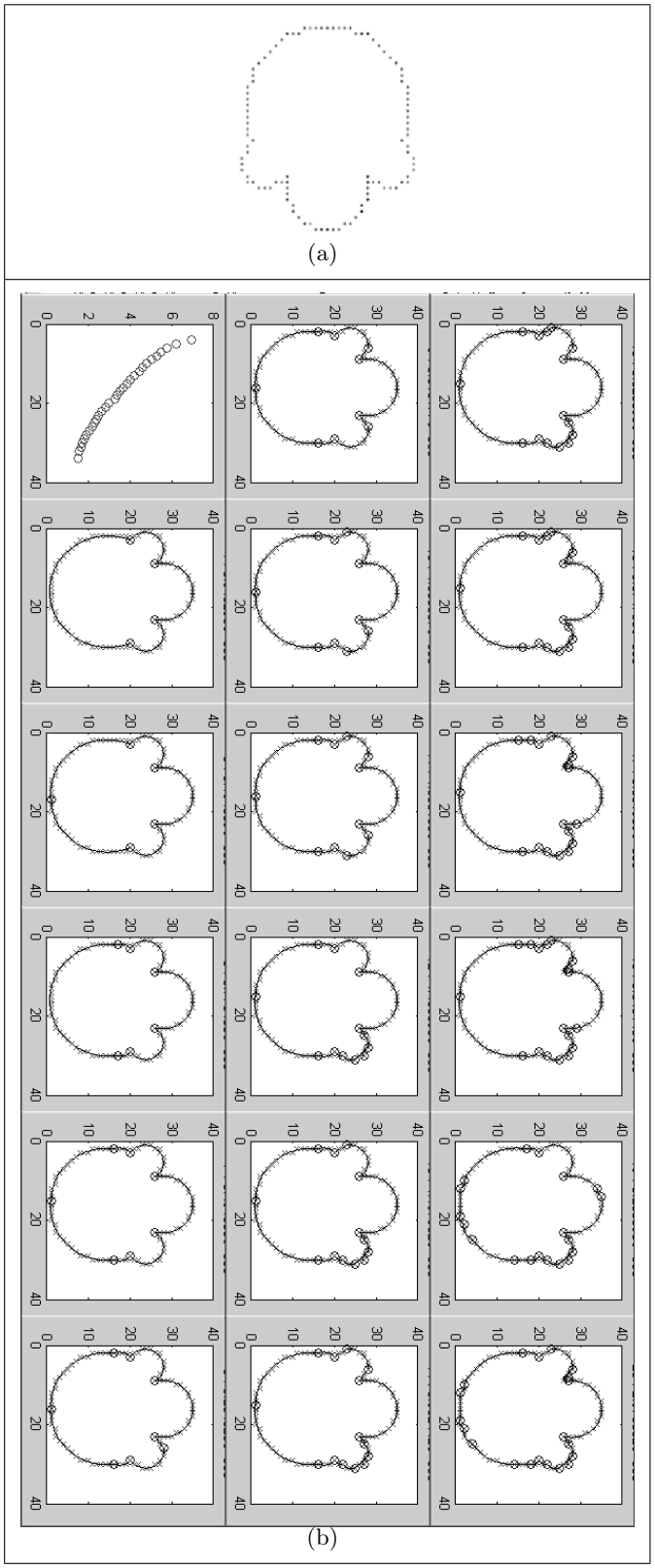
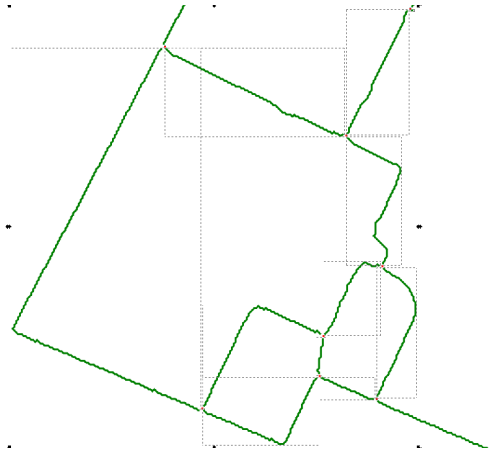
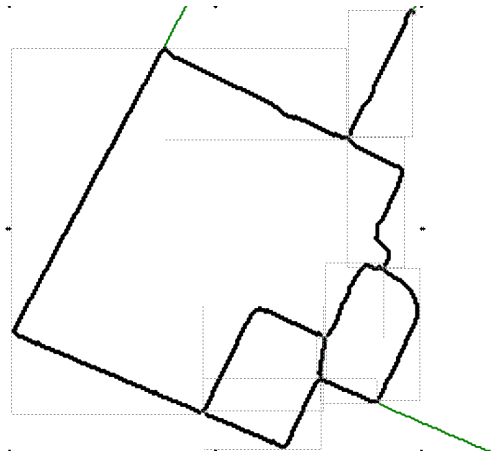


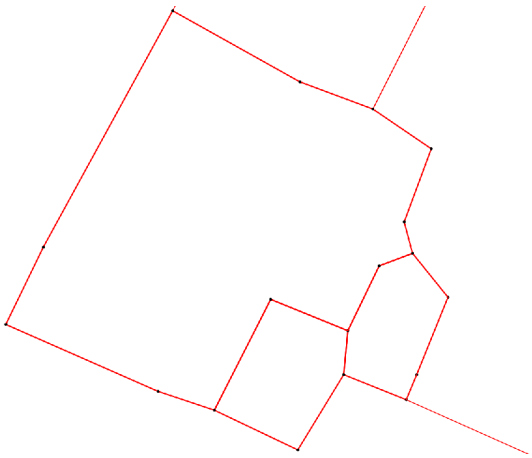
Figure 4.28: (a) Figure named: semi-circle; (b) Pareto front for the semi-circle image



(a) Zoom in

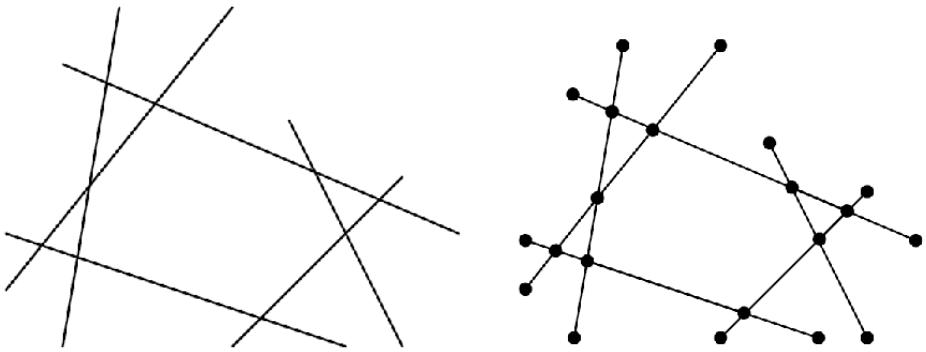


(b) Intermediate point display



(c) Polygonization

Figure 4.29: Digital curve approximation



(a) Zoom in (b) Ending and crossing points are displayed

Figure 4.30: Inferring polygons from lines. Image taken from [Jr 2003]

All the image processing algorithms, required in our approach, to extract objects from cadastral maps have been discussed. So, in the next section, the question of the quality of our approach is evoked through the viewpoint of knowledge integration. Without supervision, without ground truth, we want to know if the extracted objects are coherent with the meta-model proposed by the expert. The complete evaluation on the vectorization is postponed in chapter 5.

4.5 Quality measure by knowledge integration

Here, we aim at measuring the quality of the object extraction scheme in a self-sufficient way. It means that no ground-truth data are needed to assess the proposed object retrieval method. On the contrary, a wise integration of the expert knowledge is envisaged through the comparison of the expert meta-model and the computer-generated meta-model.

4.5.1 Methodology

First of all, let us make a kind precision. It is an obvious remark to mention that a direct comparison of two cadastral models would be misleading. The structure of two cadastral maps is very versatile, for instance a given map can be composed of only 4 parcels when another has over 40 parcels. Consequently, a real need has arisen to compare cadastral maps at a higher level, at a meta-model point of view. An overview of the main actions carried out at this stage is shown in figure 4.31. Basically, from low level image processing a model of cadastral map is built. This structured model relies on a domain-dependent ontology which defines its node and edge labels. An ontology is a receptacle for knowledge. It unifies in a single entity the structure and the definitions of domain-specific concepts. Thus, an ontology contains a meta-model resuming the organization of concepts and a logic of description to explicitly define those concepts. It is a generic framework

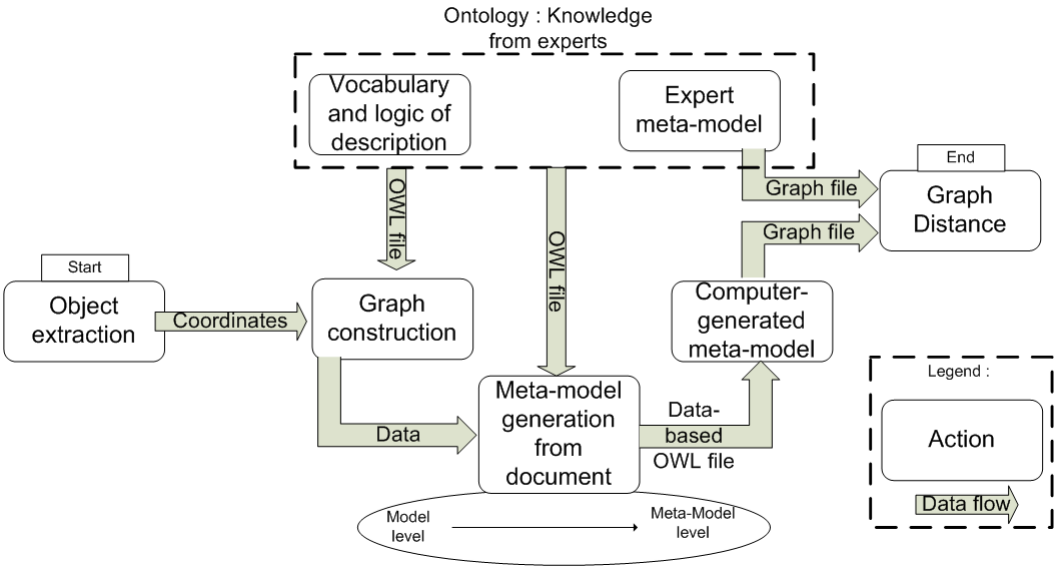


Figure 4.31: Data flow process for meta-model inference from a model instance

to gather and to handle knowledge on a topic. Thereafter, the ontology based model is generalized to reach a higher level of abstraction called meta-model. Meta-model handling and ontology-based method are emerging technologies derived from a paradigm named Model Driven Engineering, MDE for short. The applicability of MDE tools to image processing constitutes a real challenge and an open way to knowledge driven process.

4.5.2 Model Driven Engineering (MDE)

MDE [Bézivin 2005] is a new paradigm of software development that tries to fill the semantic gap faced in the data-mining field of science by the means of a higher representation called models. As an introduction, we present a "vanilla plain" case study on a 'flower' description. The logic of description is illustrated in fig 4.32. MDE lies on a three level architecture, the first stage is a raw point of view of a system in terms of regions and edges. This basement feeds a syntaxic echelon which aims to structure the primitive information. Finally, the semantic level comes to build domain specific objects (ie. Fig.4.33). A model is defined according to a certain meta-model and it is said to be a knowledge representation.

Definition 11. *A directed graph* $G = (N_G, E_G, \Gamma_G)$ *consists of a finite set of nodes* N_G*, a finite set of edges* E_G*, and a mapping function*
$\Gamma_G : E_G \rightarrow N_G \times N_G$ *mapping edges to their source and target nodes.*

Definition 12. *A model* $M = (G, \omega, \mu)$ *is a triple where:*

- $G = (N_G, E_G, \Gamma_G)$ *is a directed graph,*

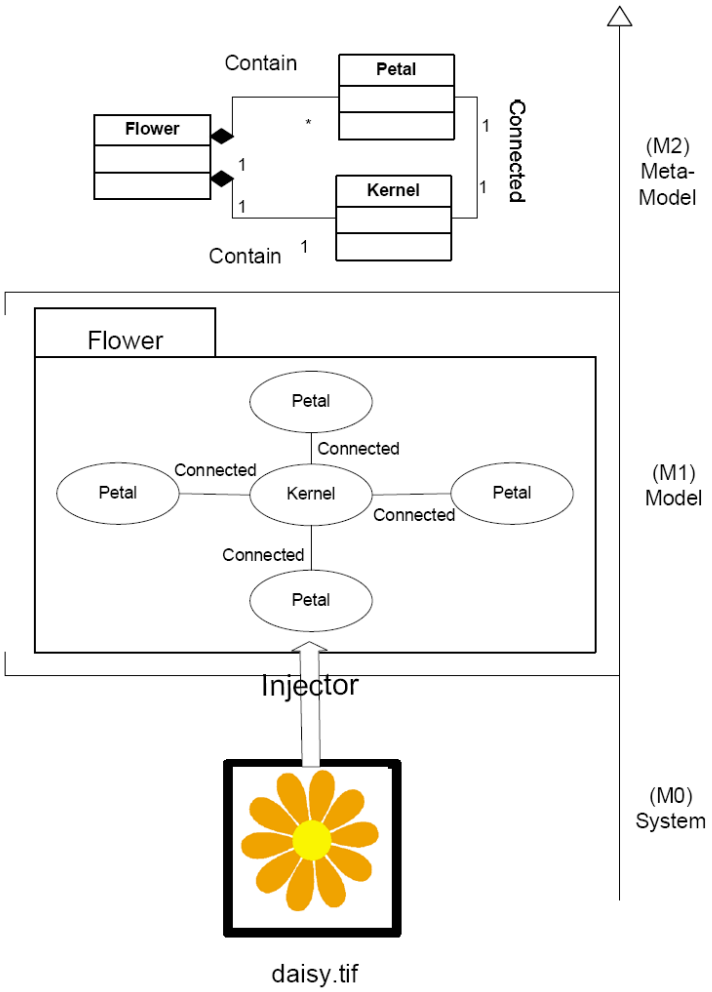
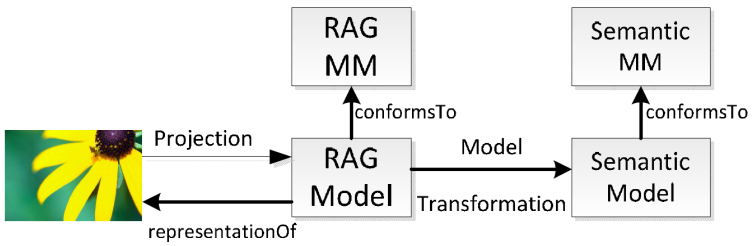


Figure 4.32: Logic of description: flower



(a)

Figure 4.33: Basic relations in Model Driven Engineering

- *ω is itself a model (called the reference model of M) associated to a graph $G_\omega = (N_\omega, E_\omega, \Gamma_\omega)$,*
- *$\mu : N_G \sqcup E_G \rightarrow N_\omega$ is the conformance function associating elements (nodes and edges) of G to nodes of G_ω.*

Definition 13. *A model transformation MT is conform to a metamodel of transformation MMT and can be defined by a function:*

$$MT : MMT(MA : MMA) \rightarrow MB : MMB$$

where:

- *MA is the transformation input model conforms to a metamodel MMA,*
- *MB is the transformation output model conforms to a metamodel MMB.*

MDE is a well-suited candidate for image interpretation by providing a common framework for representing graph structures as models conforms to meta-models. In the "flower" example, we consider image as a system which is represented by a model conforms to a meta-model. This meta-model called Region Adjacency Graph defines basic constructs for image representation under generic concepts like Region and Pixel. Models conform to the Region Adjacency Graph (RAG) meta-model represent the exact structure of the image under study.

The segmentation mechanism ends by producing a partition into regions. The graph built on top of it is rigorously conformed to the RAG meta-model. To fill the gap between computer objects and semantic concepts a mapping must exist between the two meta-models (RAG and semantic meta-models). This mapping is a function associating each node of the RAG meta-model to one or many nodes of the semantic meta-model. This linkage also called model transformation is made manually and it requires both computer scientist and expert cooperation. The figures 4.33 and 4.32 depicts the overall approach for Model Driven Image Interpretation.

In the case of cadastral map, the level 0 is the pixel image whereas the injector is the entire object extraction scheme completed by a graph construction step. Level M1 is an instance model of cadastral map as presented in figure 4.34. Finally, the M2 level is a meta-model level which makes possible the comparison between maps.

The fantastic increase of interest for knowledge engineering is pushed forward thanks to a great effort or normalization and standardization. Next part puts forward a chronological overview of formalisms dedicated to knowledge management.

4.5.3 Meta-model representations

A meta-model can be handled in several formats. We propose to blow away some obscure terms and to demystify the word "ontology". We start our discussion from XML files to finish with the Web Ontology Language (OWL) which is the most common implantation for ontology representation. All these technologies are designed for use by applications that need to process the content of information instead of

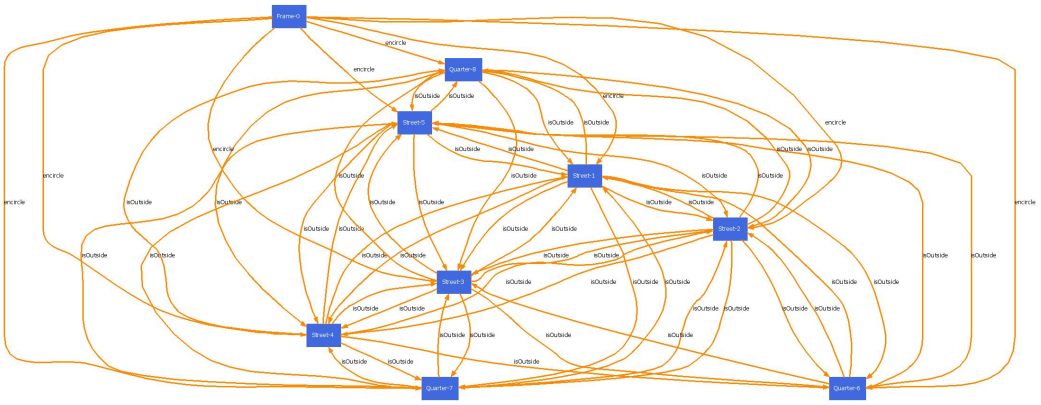


Figure 4.34: An instance model of cadastral map. Text and parcels are not expended in this view in order to obtain a comprehensive image.

just presenting information to humans. XML files on their own are just raw data, inert data without semantic, figure 4.35.

This snippet of XML code (Figure 4.35) can denote the containment notion, a FRAME is linked to a QUARTER and a QUARTER is linked to two PARCELs ... However, the nature of the relation between items is not clearly stated. A semantic information is lost, we attempted to describe that a FRAME include completely a QUARTER, spatially talking, but this aspect does not appear anymore. This way is misleading, a richer knowledge representation paradigm must be adopted, one where relations and concepts are clearly stated as object classes.

To bring the meaning back to this data and to fill the gap between data and document, a data model is required. A model aims at giving a "type", a concept name to elements encountered into the XML files. An XML Schema Definition (XSD) file can play such a role. Clearly, an XML + XSD file is a document when XML on its own is not.

Definition 14. *(XSD) A XSD is a tuple* $< T, N_T, R_T >$*, where* T *is a sequence of types,* N_T *is the set of type names. Every meta-model has a unique root type* R_T *. Each type* $\tau \in T$ *is a triplet* $< tid, name, content >$*, where tid is the type identifier, name is the name of type* τ*, it is a function* $name : T \longrightarrow N_T$*.*

A recommendation of the World Wide Web Consortium (W3C), specifies how to formally describe the elements in an eXtensible Markup Language (XML) document. This description can be used to verify that each item of content in a document adheres to the description of the element in which the content is to be placed. In general, a schema is an abstract representation of object's characteristics and relationship to other objects. An XML schema represents the interrelationship between the attributes and elements of an XML object (for example, a document or a portion

```
<FRAME id="1">
    <QUARTER id="2">
        <PARCEL id="3">
        ...
        </PARCEL>
        <PARCEL id="4">
        ...
        </PARCEL>
    </QUARTER>
</FRAME>
```

Figure 4.35: Sample of an XML file.

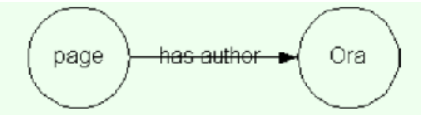


Figure 4.36: as a triple

of a document). To create a schema for a document, an analyze of its structure is necessary, defining each structural element as it is encountered.

From this point, the couple XML+XSD works perfectly well to represent knowledge. However, it does not offer a query language and no meaning extraction methods are proposed. This is why we take a look to the Resource Description Framework (RDF), an environment for constructing languages for describing resources. The RDF structure is based on the three main concepts: a resource, a property, and a statement (Subject - Predicate - Object).

Let us take as an example a single RDF assertion. Let's try "The author of the page is Ora". This is traditional. In RDF this is a triple

$$triple(author, page, Ora)$$

which you can think of as represented by the diagram displayed in figure 4.36.

The main advantage of RDF over the basic XML is its simplicity. Unlike the order of elements in XML, the order of RDF properties does not matter. RDF offers a very appealing and flexible solution to any semantic model designer. Continuing in the same direction and going even further about knowledge integration into computer software, the last decade gave birth to ontology. Towards knowledge representation, the Web Ontology Language (OWL) is a family of knowledge representation languages for authoring ontologies, and is endorsed by the World Wide Web Consortium. This family of languages uses a semantic model intended

to provide compatibility with RDF Schema. OWL ontologies are most commonly serialized using RDF/XML syntax. OWL is considered as one of the fundamental technologies underpinning the Semantic Modeling. OWL is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDFS) by providing additional vocabulary along with a formal semantics. OWL adds semantics to the schema. It allows specifying far more about the properties and classes. It is also expressed in triples. For example, it can indicate that *"If A isMarriedTo B"* then this implies *"B isMarriedTo A"*. Or that if *"C isAncestorOf D"* and *"D isAncestorOf E"* then *"C isAncestorOf B"*. Another useful aspect, OWL adds is the ability to say two things are the same, this is very helpful for joining up data expressed in different schemes. You can say that relationship *"sired"* in one schema is *owl:sameAs "fathered"* in some other schema. You can also use it to say two things are the same, such as the *"Elvis Presley"* on wikipedia is the same one on the BBC. This is very exciting as it means you can start joining up data from multiple sites, from multiple sources (this is "Linked Data"). You can also use the OWL to start creating new facts, such as *"C isAncestorOf E"*. OWL and RDF are much of the same thing, but OWL is a stronger language with greater machine interpretability than RDF. OWL comes with a larger vocabulary and stronger syntax than RDF. Ontology is about the exact description of things and their relationships. For the cadastral map, ontology is about the description of map information and relationships between map information.

Let us review the different types of knowledge representation in the particular case of image of cadastral map. As a starting point, we first present the description of a cadastral map meta-model under the UML formalism, figure 4.37.

Then, this class diagram can be exported into a XML Schema (XSD) format, figure 4.38.

Finally, this knowledge representation can be seen as an ontology. OWL (Web Ontology Language) is a language to express an ontology in a computer science way. OWL syntax is based on the Resource Description Framework (RDF).

Definition 15. *(RDF) RDF is a syntax organized into triples. A triple consists of a subject, a predicate, and an object. The subject is, well, the subject. It identifies what object the triple is describing. The predicate defines the piece of data in the object we are giving a value to. The object is the actual value.*

The cadastral map ontology is expressed in OWL and serialized using RDF/XML syntax. A graph visualization of RDF schema and an OWL syntactical viewpoint are proposed in figure 4.39. This portion of code denotes the basics syntax of OWL. Concepts also named entities are defined as "Class" in OWL. While a relation between two OWL classes is materialized with an "Object Property".

- OWL Class

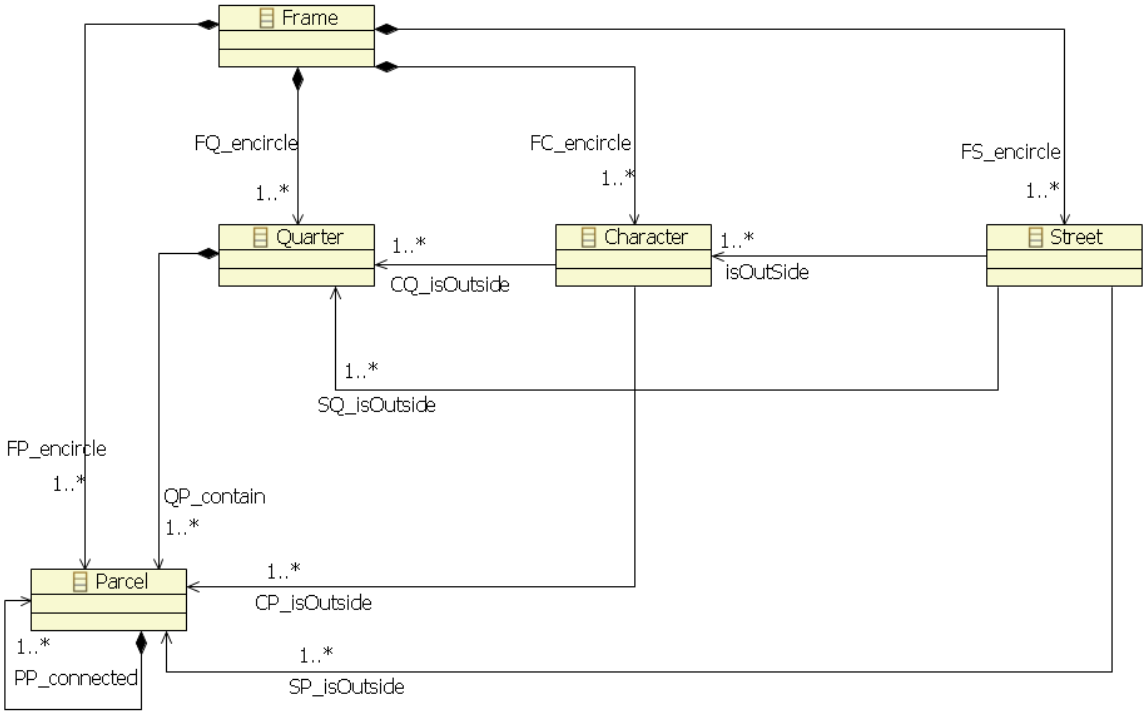


Figure 4.37: UML representation designed thanks to the Eclipse Modeling Framework (EMF)

```
<xsd:complexType name="Frame">
    <xsd:sequence>
        <xsd:element maxOccurs="unbounded" name="FQ_encircle"
        type="cadastralmaplimited:Quarter"/>
    </xsd:sequence>
</xsd:complexType>
```

Figure 4.38: XSD representation. This snippet extracted the full XSD file indicates that a "Frame" is link to "Quarter" by a relation calls "include"

  - A Class is a group of subjects or objects with similar meanings that can be classified as a single type.

- Object Property
  - An Object Property is an object so it can include methods or attributes. More specifically to OWL, an Object Property has two mandatory attributes, a range and a domain.
  - Domain
  - Range

- Domain
  - For a property one can define "domain" axioms. Syntactically, a "domain" axiom is a built-in property that links a property (some instance of the class) to a class description. The "domain" represents the source node of an arc in a RDF graph.

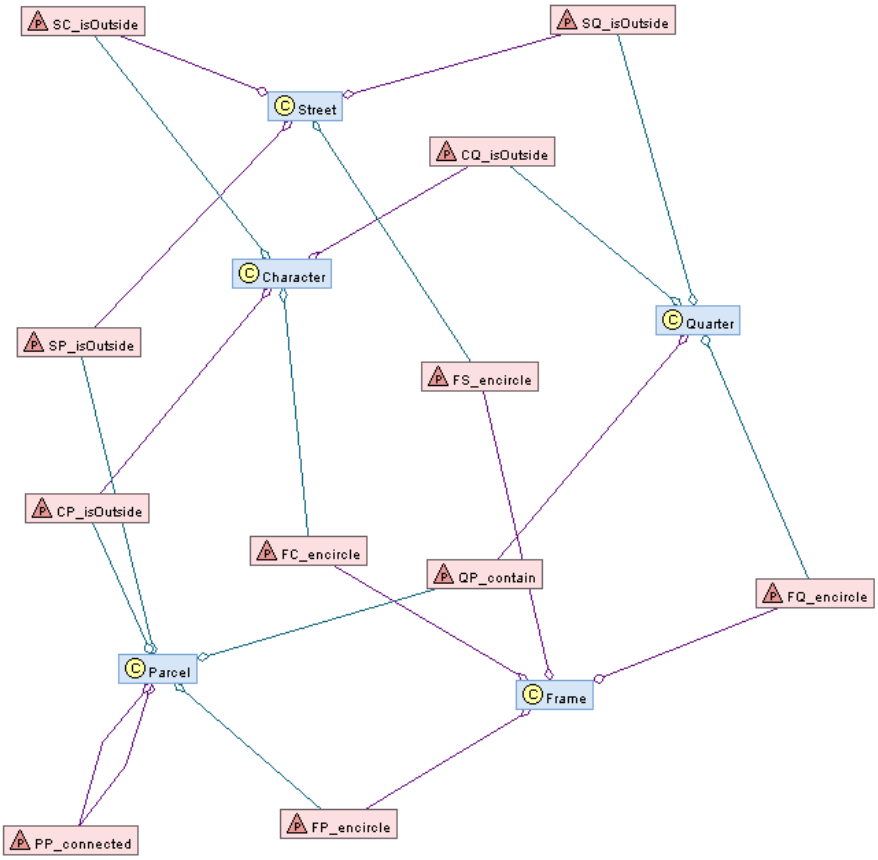
- Range
  - For a property one can define "range" axioms. Syntactically, a "range" axiom is a built-in property that links a property to a class description. A "range" axiom asserts that the values of this property must belong to the class extension of the class description. The "range" represents the target node of an arc in a RDF graph.

OWL uses open-world semantics, every single OWL class belongs to a super class where the top level class is called "Thing". In OWL, every class is derived from "Thing". On the contrary, relational databases are based on closed-world.

Thanks to this model design, we can construct a graph which is an instance model of cadastral map. A graph where nodes and edges make reference to the expert-designed ontology.

4.5.4 Graph construction

In this part, the principle describing the graph construction is presented. The algorithm 3 provides the guide lines to create a semantic graph considered as an instance model of the cadastral map. Roughly, all objects are added as nodes into the graph. Nodes are labeled according to the vocabulary defined in the ontology. Thereafter, given as parameters an ontology and a set of objects, relations between two objects are checked out sequentially until one relation is verified. In this positive case, an edge is inserted in the graph and this new edge is labeled following the resources described in the OWL file. This directed attributed relational graph is created from the object detected by our low level processing methods. Consequently, there is no guaranty that the model is conform to the meta-model, which is to say conform to the ontology. Errors could corrupt the model. To find out how different



(a)

```
<owl:Class rdf:about="♯Frame"/>
<owl:Class rdf:about="♯ Quarter "/>
<owl:ObjectProperty rdf:about="♯ FQ_encircle ">
    <rdfs:domain rdf:resource="♯ Frame "/>
    <rdfs:range rdf:resource="♯ Quarter "/>
</owl:ObjectProperty>
```

(b)

Figure 4.39: (a) Graph RDF representation; (b) RDF file snippet. It represents the declaration of two classes "Frame" and "Quarter", in addition a relation named "encircle" between Frame and Quarter is defined.

Algorithm 3 Semantic graph construction

Require: Set of frames: F
Require: Set of quarters: Q
Require: Set of streets: S
Require: Set of characters: C
Require: Set of parcels: P
Require: Sets of objects: $T < F, Q, S, C, P >$
Require: Ontology: o
Ensure: A semantic graph.

```
Start

G=CreateGraph()
addAllObjectsAsNodes(T,G,o)
for each t1 ∈ T do
  for each t2 ∈ T do
    if t1 ≠ t2 then
      n1=GetCorrespondingNode(t1,G)
      n2=GetCorrespondingNode(t2,G)
      if CheckRelation_Encircle(t1,t2) == true then
        addEdge(G,n1,n2,o : ♯ encircle)
        BREAK
      end if
      if CheckRelation_contain(t1,t2) == true then
        addEdge(G,n1,n2,o : ♯ contain)
        BREAK
      end if
      if CheckRelation_isOutside(t1,t2) == true then
        addEdge(G,n1,n2,o : ♯ isOutside)
        BREAK
      end if
      if CheckRelation_connected(t1,t2) == true then
        addEdge(G,n1,n2,o : ♯ connected)
        BREAK
      end if
    end if
  end for
end for
End
```

return The semantic graph:G

is the model from the meta-model, we need to create a new ontology derived from the semantic graph and the expert designed ontology.

As explained at the beginning of the section, a model graph is not directly comparable with the expert-knowledge. A transformation has to be performed on the model graph to measure the distance between this instance model generated automatically and the rules and constraints expressed in the expert meta-model. In this goal, an inference mechanism is involved to extract a meta-model from a model, in other words, to generate a schema from an RDF document.

4.5.5 Meta-model inference from a RDF document

The semantic graph is an instance model. A translation is performed to write it into a RDF file format. Nodes and relations between nodes are represented as RDF triples (a resource, a property, and a statement). This RDF data are coupled with the expert ontology to infer a new ontology driven by data. The individuals (data) lead the process to the elaboration of a computer generated ontology. We consider the RDF document as forming a "knowledge base" or repository with collected meta-data about a number of resources. An ontology is derived from a collection of instances. The RDF instance documents will produce a valid ontology structure, an RDF Schema (RDFS).

This part of our work was realized under the guidance of Árpád Tamási, CEO and owner of Progos Kft. and author of the ontology generator service. Progos[5] is an ontology generator from RDF documents. Progos generates a new ontology from instances and a source ontology. The Progos's ontology generator architecture is proposed in appendix C. The design of our Ontology Generator is exposed in figure 4.40. This modular architecture is inspired from the Oshani Seneviratne's work[6] which aim to generate a relational database schema from a RDF store. To our knowledge, on the topic of Ontology inference from RDF documents, this approach is the only existing one.

Our architecture of the ontology generator is shown in figure 4.40. It is composed of three modules:

1. RDF Store: This module is in charge of parsing RDF files and their corresponding RDF schema. When the files are processed, the RDF Store generates iterators so that following modules can manipulate this data. It also constructs a predicate table structure so that the schema generator can collect statistics.

2. Schema Generator: This module takes the predicate table as input and executes an algorithm in order to determine a good ontology schema for the RDF data set.

[5] http://progos.hu/tools/og/
[6] http://code.google.com/p/r-store/

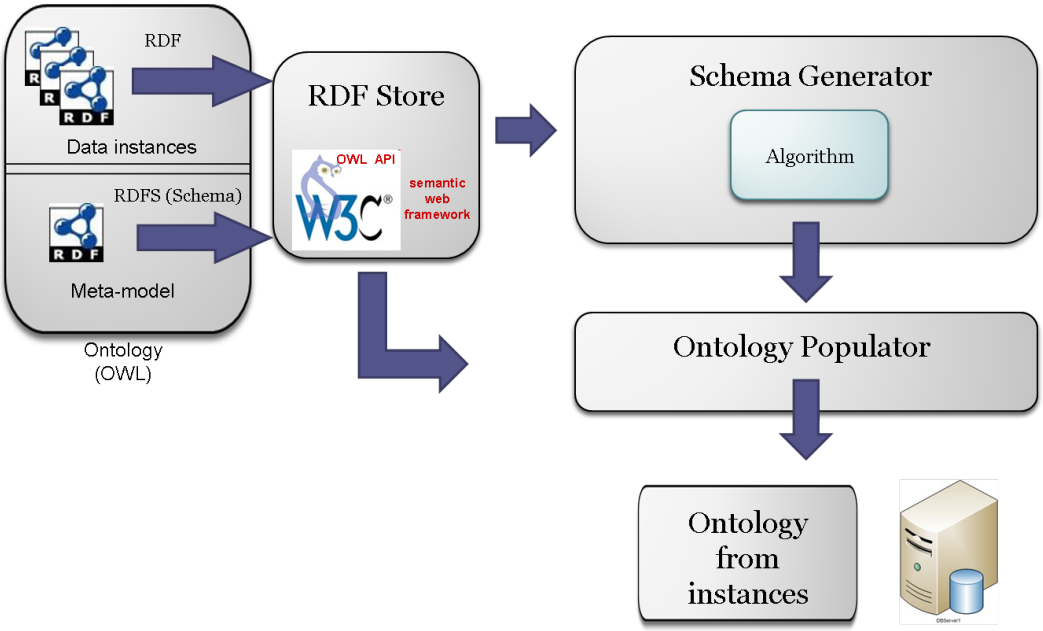


Figure 4.40: Design

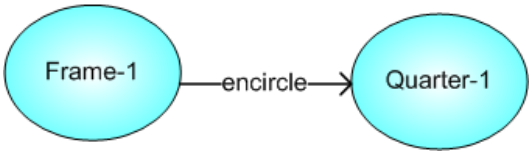
3. Ontology Populator: When the data driven schema is generated, the Ontology Populator gets the triple iterator from the RDF Store and copies the RDF triples into the new ontology.

RDF store: The RDF Store is in charge of parsing RDF files and RDF schemas. It provides resources to the Schema Generator and the DB Populator to analyze the data. OWL API Semantic Web Framework [Horridge 2008] is extensively used for this task. It provides an Open Source API that allows manipulating triples and schemas. The RDF Store:

- Generates triple iterators.
- Extracts Subject Classes from the RDF schema and classifies triples with the same subject.
  - Maps Subjects -> Subject Classes
  - Given a Subject Class returns all the Subjects of that Class.
- Constructs the Predicate table.
  - For each predicate generates the map: (qualified Predicate) -> (Subject Class, Object Class)

$$triple(Subject, Predicate, Object)$$

$$triple(Frame-1, encircle, Quarter-1)$$



Predicate table for this entry : Syntax → Predicate(key) : <Subject Class, Object Class>(pair);Frequency(scalar) Sample →: encircle : <Frame ,Quarter>;1

Figure 4.41: Predicate table construction

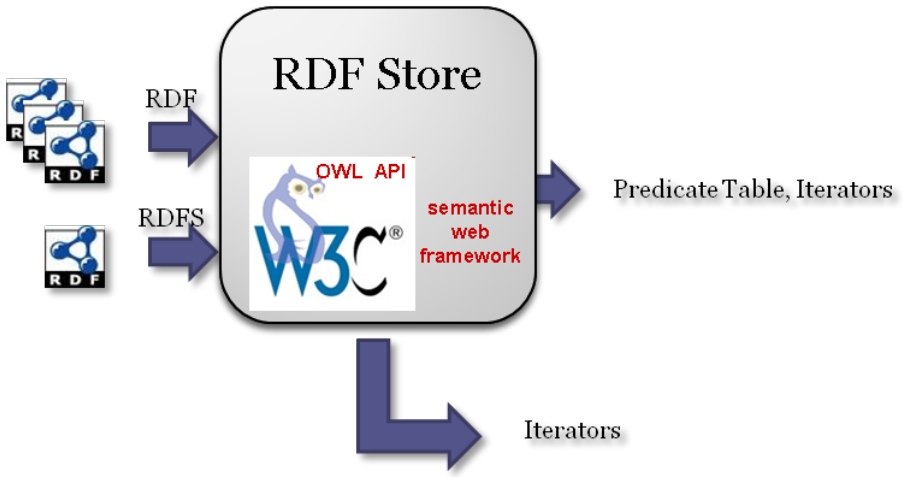


Figure 4.42: RDF store

- This structure is sent to the Schema Generator. The Schema Generator infers a data driven schema by obtaining statistics from the predicate table and from the data itself.
- An example of the predicate table construction is presented in figure 4.41. A triple insertion:

This tool provides resources to the Schema Generator and Ontology Populator to analyze RDF triples (figure 4.42).

Schema Generator The Schema Generator uses a Statistics-Based algorithm in order to produce a data-based ontology schema. This phase analyzes the RDFS and RDF data triples to produce a good meta-model (figure 4.43), one which is data compliant. The main idea behind this algorithm is to discover the underlying structure inferred by the RDF data. A RDF Schema that would represent any data that falls within this new ontology. This algorithm takes into account the

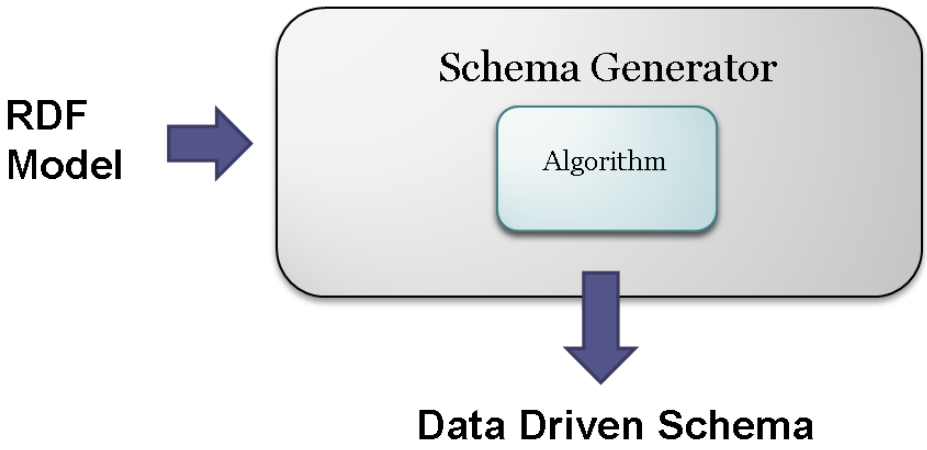


Figure 4.43: Schema Generator

semantics expressed in the RDF Schema. Based on the sub class relationships, and the property constraints, it will determine the classes which are likely to be represented in the generated ontology. Potentially, all classes are accepted to be part of the output ontology. Nevertheless, if a class never appears into the RDF triples then it will not be introduced in the target ontology. This is called "ontology pruning". The reversal is true, when browsing the Predicate table and encountering a Predicate which is not in the source RDF Schema a choice has to be made. Since OWL relies on open-world semantics, the unknown triple can be taken into account by the following steps:

- $$triple(Subject, Predicate, Object)$$
- Creation of missing classes in the output Ontology.
- Linkage of the missing classes with the "Thing" top level class.
- Creation of an Object Property named by the Predicate.
  - The Domain = Subject Class
  - The Range = Object Class

In our configuration, this latter situation should not appear. The production of all relations and concepts are limited by our low level procedure (the injector). From the Predicate table, Classes and Object properties are generated into the new ontology. The only remaining problem is the cardinality question.

Cardinality: To determine the cardinality of relationships, this algorithm does not look at the RDFS property constraints, on the contrary, the algorithm relies on statistics derived from data. Properties have domain and range constraints, which

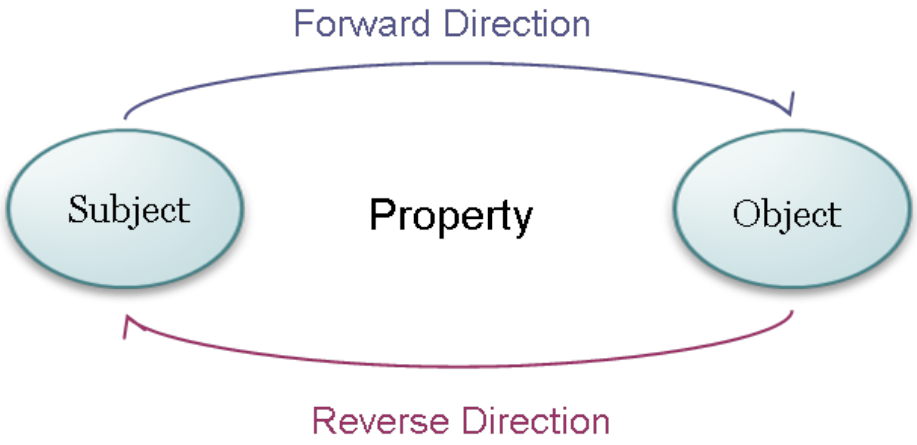


Figure 4.44: Arc Reversal

typically map the corresponding predicate in a triple from a subject to an object. The subject should be an instance of the domain, and the object should be an instance of the range. For example, if there is an RDF property called "contain" and it's domain is "Quarter" and the range is "Parcel", this means that in the RDF data there could be a triple such as <Quarter contain Parcel>. Based on these domain-range constraints, the type of the relationship as to whether it is ONE-TO-ONE, ONE-TO-MANY and MANY-TO-MANY is made based on the following algorithm.

The Statistics-Based algorithm is based on one statistic on each type of arc in the RDF graph: the cardinality. This statistic is gathered in two iterations over the RDF store.

The cardinality is determined by two iterations over the RDF store, once over the triples as they have been parsed, and once over the triples with the Subjects and Objects reversed. In both cases, the triples are traversed in sorted order, on the Subject field. That means that in the reversed set, the triples are really sorted on their Object field. It is possible to determine whether a Predicate has cardinality one-to-one or one-to-many by checking the types of arcs (Predicate) coming out of each Subject. Cardinality natures were defined in section 4.3.2.

Because the triples are sorted on Subject, only those arcs which refer to a given Subject must be stored in memory at once. As soon as a different Subject is encountered, the algorithm is certain that it will never encounter that Subject again, and can make a decision about the cardinality of each arc. The decision is recorded in a two dimensional mask table, indexed on Subject Class and Predicate. In order to find out about the other two cardinalities, the algorithm repeats the same procedure on the reversed set of triples. This is valid because by reversing the Subject and Object, we have reversed the direction of all the arcs as in figure 4.44.

Arcs that were one-to-many have become many-to-one. Arcs that were many-to-many or one-to-one have not changed their cardinality. In light of this, if the

algorithm determines that a reversed arc relation is one-to-many, it looks in the mask table to check if the prior iteration also determined it to be one-to-many. If so, the cardinality of this arc is many-to-many. Otherwise, it is many-to-one. Here is an example, let us consider the following triple repository:

$$triple(Frame-1, encircle, Quarter-1)$$

$$triple(Frame-1, encircle, Quarter-2)$$

$$triple(Frame-1, encircle, Quarter-3)$$

The first iteration takes a look at the Direct parsing. According to the subject "Frame-1" the relation is as follows :

$$< Frame\ encircle\ Quarter > : ??? - to - many$$

Reverse Parsing (Second iteration): The data set becomes :

$$triple(Quarter-1, encircle, Frame-1)$$

$$triple(Quarter-2, encircle, Frame-1)$$

$$triple(Quarter-3, encircle, Frame-1)$$

Triples are analyzed by subject.

$triple(Quarter{-}1, encircle, Frame{-}1) \longrightarrow < Frame\ encircle\ Quarter > :\ one{-}to{-}???$

, next Subject

$triple(Quarter{-}2, encircle, Frame{-}1) \longrightarrow < Frame\ encircle\ Quarter > :\ one{-}to{-}???$

, next Subject

$triple(Quarter{-}3, encircle, Frame{-}1) \longrightarrow < Frame\ encircle\ Quarter > :\ one{-}to{-}???$

. The final relation is :

$$< Frame\ encircle\ Quarter > :\ one - to - many$$

At the end of this module, all classes, relations and cardinalities are defined regarding the RDF triples. Thus, we produce a new empty Ontology, with only the meta-model of data (the RDF schema). This good Ontology in term of data is given to the last stage in order to be populated.

Ontology Populator: In one iteration over the RDF store, the triples are copied into the output OWL file. In this way, the new Ontology is filled up with data. This new Ontology is data compliant. It means that all data are conformed to the meta-model describing concepts and relations between concepts. An example of Computer-Generated schema is shown in figure 4.45. To underline the differences with the expert-designed meta-model, the figure 4.46 and 4.47 reveal the knowledge representation with and without symmetric and transitive relations.

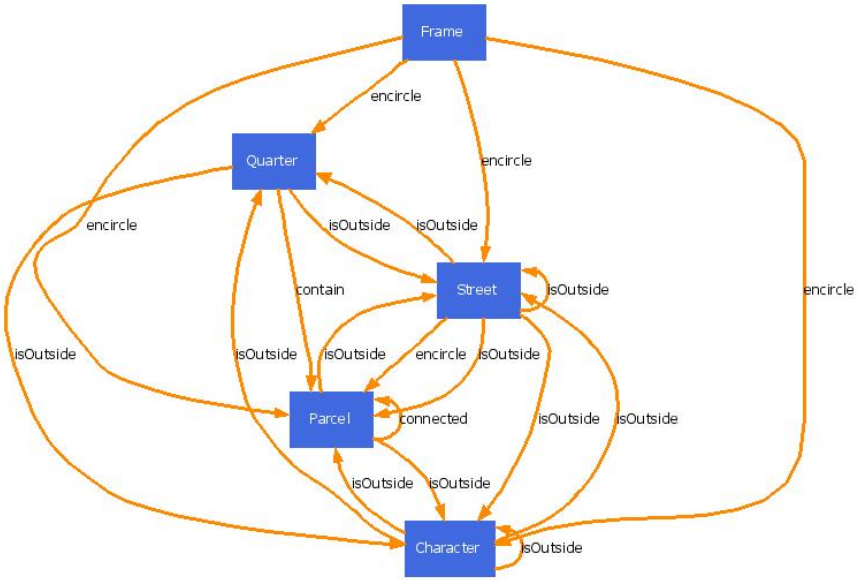


Figure 4.45: Computer-Generated Meta-model

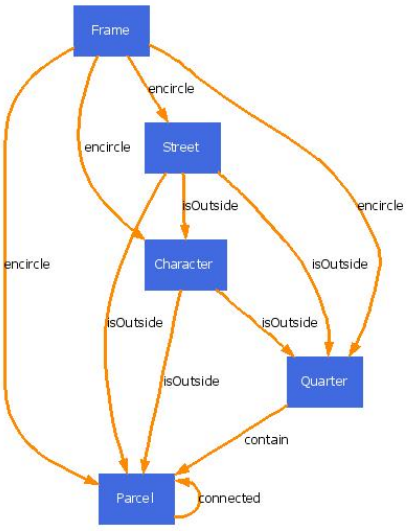


Figure 4.46: Expert Meta-model

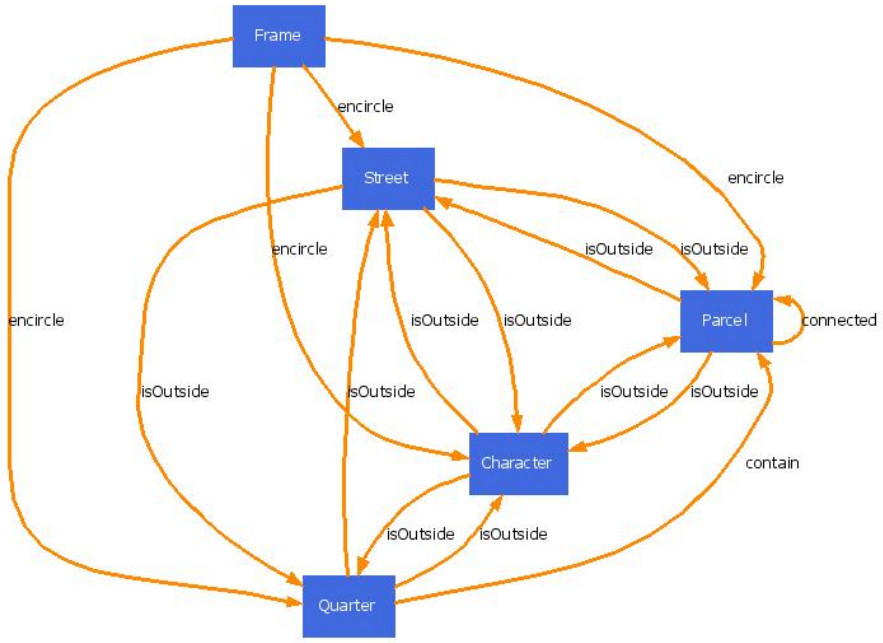


Figure 4.47: Expert Meta-model extended. Transitive and symmetric properties are drawn.

4.5.6 Meta-model comparison

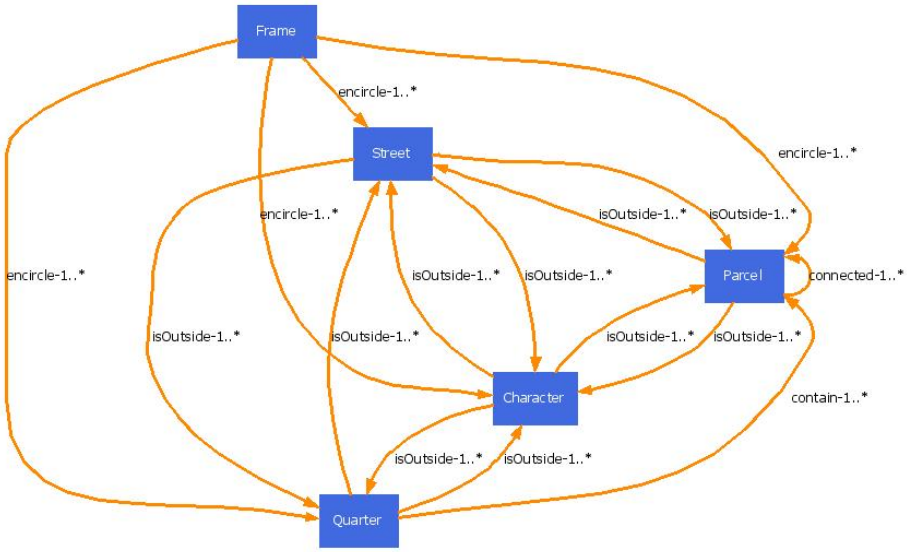
Meta-models need to be compared for several reasons, one of them is to represent the differences between two similar meta-models. We propose to address this question through the tropism of graph matching.

The transformation from meta-model to graph is quite explicit. Each concept of the meta-model is node into the graph. Edges represent reference between concepts. L_E is a set of labels and each edge is labeled with two nominal values:

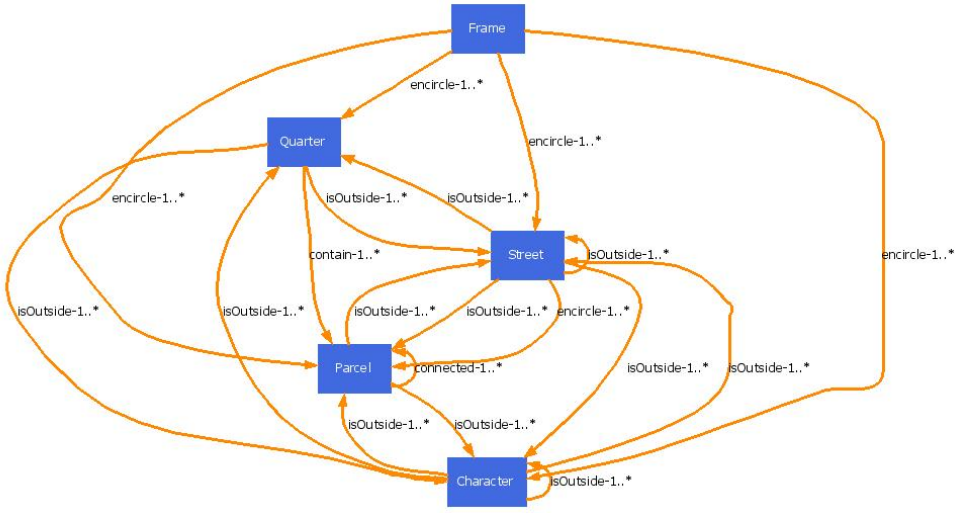
- $L_E = L_E^1, L_E^2$
- L_E^1 : The nature of the relation, a string value that belongs to the set encircle, isOutside, contain, connected
- L_E^2 : This label denotes the cardinality of the relation, a string value among 1..*, 1..1,*..1, *..*

Our graph representation preserves the structure, the relationship and the cardinality of the meta-model. An illustration of the meta-models to be compared is displayed in figure 4.48. In this work, the problem which is considered concerns the matching of directed labeled graphs. Such graphs can be defined as follows:

Definition 16. *(Graph) Let L_V and L_E denote the set of node and edge labels, respectively. A labeled graph G is a 4-tuple $G = (V, E, \mu, \xi)$, where*



(a)



(b)

Figure 4.48: Computer Generated and Expert–Designed meta-model under a graph formalism.

- *V is the set of nodes,*
- *$E \subseteq V \times V$ is the set of edges*
- *$\mu : V \rightarrow L_V$ is a function assigning labels to the nodes, and*
- *$\xi : E \rightarrow L_E$ is a function assigning labels to the edges.*

The question of comparing two meta-models is turned into a graph distance problem. A graph distance is a function $X \times X \longrightarrow \mathbb{R}$ where X is a graph as defined in Def. 16. The definition of such a distance is fully explained in chapter 6. This latter describes a SubGraph Matching Distance (SGMD) for graph comparison. A distance between two graphs is defined by solving for a max matching in a bipartite graph spanning the nodes in two graphs. A probe is computed for each node, which describes the neighborhood structure of a node out to a distance of 1 edge (along all incident edges). The cost assigned to an edge in the bipartite graph spanning two nodes is computed as the edit distance between their probes. The resulting approximation to computing the largest isomorphic subgraph is $O(n^3)$, the complexity of the bipartite matching problem. To use SGMD distance, we need to define the specific case where every edge is labeled with two values. The similarity between two edge attributes is then defined as follows :

Definition 17. *(Edge distance) Let LA_E and LB_E be two labels from two graphs G_A, G_B, respectively. Consequently, LA_E^1, LA_E^2 are the two values constituting LA_E and belonging to G_A. In this way, we express d_E as a function $d_E : L_E \times L_E \longrightarrow \mathbb{R}$.*

$$d_E(LA_E, LB_E) = \begin{cases} 0 & LA_E^1 \text{ is equal to } LB_E^1 \text{ and } LA_E^2 \text{ is equal to } LB_E^2 \\ 0.5 & LA_E^1 \text{ is not equal to } LB_E^1 \text{ or } LA_E^2 \text{ is not equal to } LB_E^2 \\ 1 & \text{otherwise} \end{cases} \quad (4.2)$$

In the next chapter, this Meta-Model Based Distance (MMBD) will be assessed with a ground-truth measure of performance. In this way, we want to confront a ground-truth based approach with our knowledge inspired method. The underlying question is to figure out if MMBD is relevant, if it behaves like a performance evaluation tool.

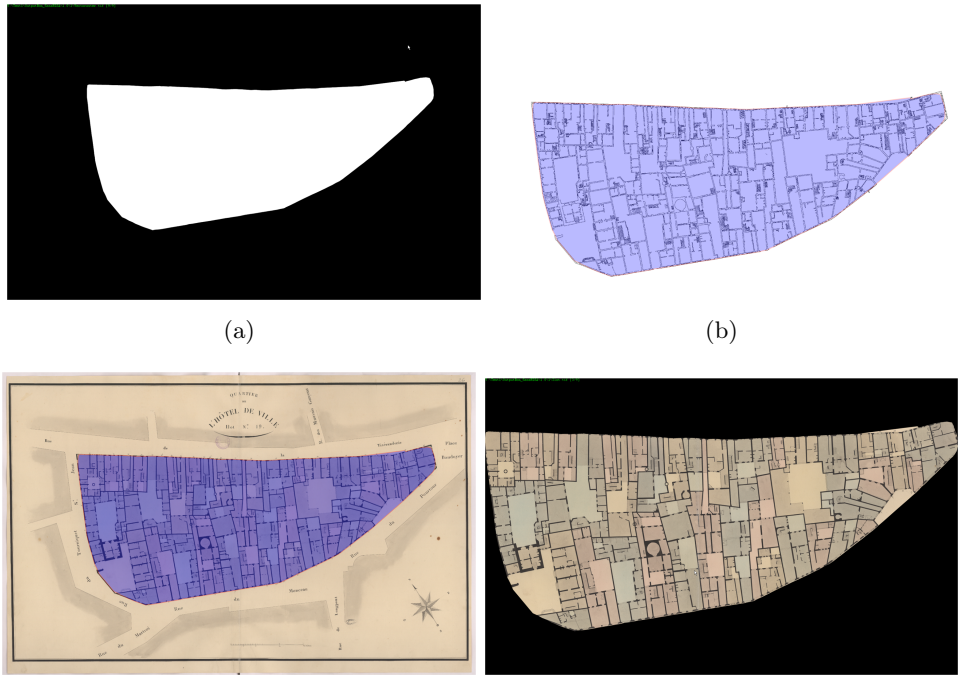
4.6 Experimental results

In this section, we evaluate the text/graphic segmentation and the quarter extraction.

4.6.1 Quarter extraction Experiments

4.6.1.1 Methodology

The assessment phase aims to compare the contour obtained with our method and the contour described by a user. To illustrate our saying, we kindly refer the reader



(a) (b)

(c) (d)

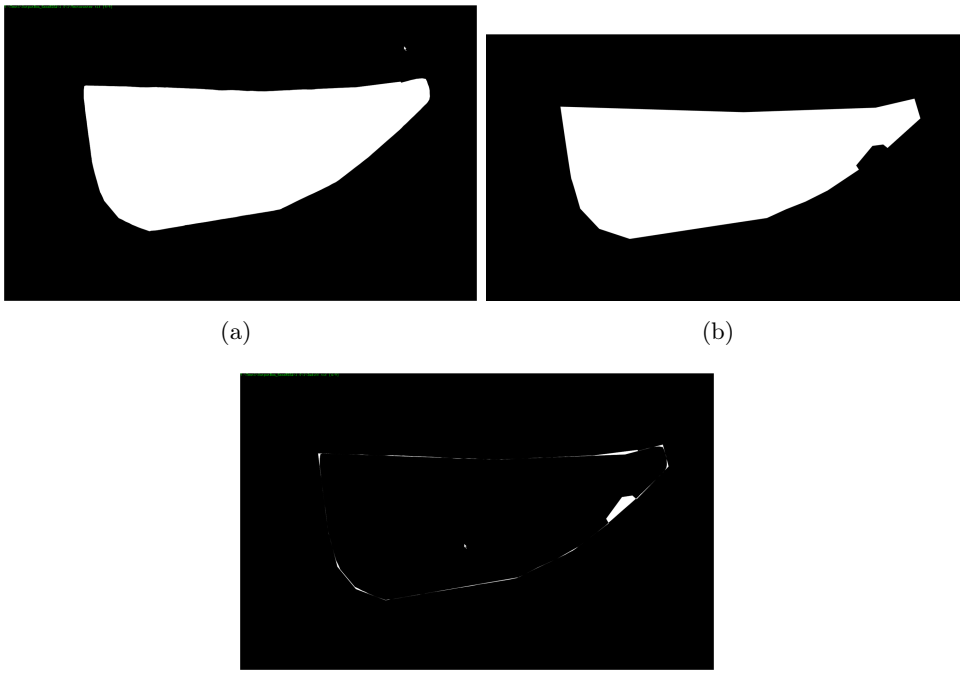
Figure 4.49: (a) Extracted quarter ; (b) Polygon given by the Snake; (c) Polygon overlapped on the source image; (d) Isolated Quarter

to figure 4.49. This evaluation is carried out by image difference between the user-defined ground truth (I2) and the computer-based contour built automatically (I1). Image differences are depicted in figure 4.50. The error is expressed in equation 4.3.

$$Error = \frac{1}{\sum_x \sum_y I2(x,y)} \left[\sum_x \sum_y (|I1(x,y) - I2(x,y)|) \right] \tag{4.3}$$

4.6.1.2 Evaluation

The error rate is computed on 30 ancient cadastral maps and results are reported in table 4.1. The errors can be categorized into two groups. (1)Quarter vectorization accuracy and (2)loss information from the retrieval system. Firstly, the active contour does not stick strictly (pixel to pixel) the input image. An accuracy error is introduced by the snake which proceeds to a polygonal approximation of the digital curve. This first kind of error is somehow minor since it represents only 1% of the whole mistake, in average over the 30 maps. Secondly, the biggest source of error is the missing information due to our selective method. Mainly, through the text/graphics separation stage some small CCs can be removed from the graphic layer while they represent part of the quarter structure. i.e.: edges or corners. Nevertheless, we would like to precise that the error rate remains quite low.



(a) (b)

(c)

Figure 4.50: (a) Rasterized polygon contour given by the snake; (b) Ground-truth; (c) Difference image

	Min	Mean	Max
Error (%)	1.58	4.91	12.80

Table 4.1: Error rates

	Training data set	*Test data set*	*Validation data set*
$\sharp elements$	4118	4118	5000
$\sharp text$	2791	2791	2500
$\sharp graphic$	1327	1327	2500

Table 4.2: Databases in use for the text/graphics segmentation

	Precision		***Recall***		***CCI***		Rec
	Class1	**Class2**	**Class1**	**Class2**	**Class1**	**Class2**	(%)
K=1	0.838	0.783	0.764	0.852	1910	2130	80.8
K=10	0.875	0.785	0.756	0.892	1884	2236	82.4
K=100	0.856	0.898	0.904	0.848	2122	2258	87.6

Table 4.3: Classification rate. Class1 and class2 represent graphics and text respectively.

4.6.2 Test on Text/Graphic segmentation

Methodology The text graphic segmentation is assessed according to the number of correctly classified CCs as text or graphic. In this objective, three databases were employed and are described in table 4.2, this latter shows the data set characteristics. The training and test sets are involved during the training phase by the genetic algorithm while the validation database is only used once to assess the whole system.

Results

Table 4.3 takes the stock of the recognition rates (Rec) on the validation database. Results deal with the need to generate prototypes instead of just finding them among the graph corpus. Moreover, increasing the number of generated prototypes helps to improve the number of correctly classified instances (CCI). Respectively, Class1 and Class2 stand for Graphics and Text. The class variabilities are better taken into account as the number of prototypes increases. The problem is better modeled with generalized prototypes which maximize the classification rate.

Figure 4.51 deals with a comparison between the well-known Fletcher and Kasturi method and our approach on a cadastral map. Of course, this is a single and unique image, but anyway, it reflects the behavior of the two paradigms. Our approach is more complex however it gives a better representation of the text layer. In addition, a comparative study is reported in table 4.4. It presents a quantitative assessment.

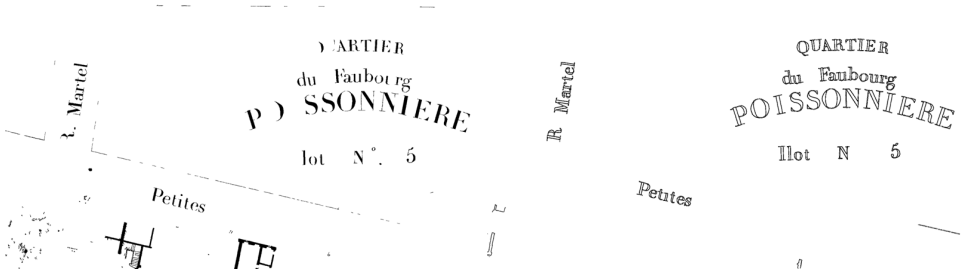
4.7 Conclusion

This chapter discussed four important phases.

- Firstly, a text/graphics separation was proposed. It is based on a graph rep-

	#Graphics	*#Text*	*#Elements*	*CCI*		**Rec**
				Class1	**Class2**	
Fletcher-Kasturi	763	122	855	435	57	55.59

Table 4.4: Comparative Study



(a) (b)

Figure 4.51: (a) The text layer given by the Fletcher and Kasturi approach; (b) the text layer found by our method.

resentation and a prototype selection method for structural data. A classifier trained on these prototypes categorizes connected components into two broad groups, text or graphic.

- Next, the quarter localization is carried out through a coarse to fine approach. The architecture is based on a "peeling the onion" strategy in order to remove unwanted objects from the map. This aspect was assessed on 30 maps and results tend to illustrate a reliable behavior.

- Thirdly, the parcel extraction is performed *a posteriori*. A closer look is given to parcels when quarters are individually identified. Specific image processing are carried out to delineate parcel borders. Hereafter, the joint use of a polygonal approximation method and a cycle detection transform pixels in segments and segments in polygons, respectively. The question of parcel accuracy will be highlighted in the next chapter, dealing on performance evaluation.

- In the fourth stage, a knowledge integration strategy is proposed for verifying image processing integrity. In this objective, a data driven ontology generator has been defined. Data are generated from object detectors to form an instance model where relations and concepts are specified into an expert-designed ontology. This structure stands apart from a conventional graph-based representation (a "vanilla plain" graph) because concepts and relations that compose the graph are explicitly written into a knowledge base called ontology. Thereafter, this computer-generated model comes to feed a higher level of representation. A meta-model is automatically inferred from an instance model making possible the comparison with an expert-designed knowledge

representation. This comparison is led thanks to a graph matching algorithm that provides quantitative measures about how different are two meta-models. Furthermore, another information is retrieved by the graph matching method, in this way, it gives out the nodes where errors occur into the graphs.

From this last remark an important issue arises. Object detectors and nodes are closely coupled, therefore, it is possible to spot which low level process is corrupted and does not generate the compliant information. These indications make possible a feed back on low level processes changing and adjusting their parameters (threshold) to obtain iteratively a "better" meta-model conforms to the expert knowledge.

Ongoing works are investigating the possibility to correct errors through the use of perception cycle while another future work is exploring an orthogonal direction, the use of ontology reasoners to automatically label nodes of a model graph. In this situation, relations between nodes could be considered as constraints that must be fulfilled to classify objects into broad categories.

Chapter 5

Evaluation of Cadastral Map processing

Contents

5.1 Forewords

This chapter presents a benchmark for evaluating the Raster to Vector conversion systems. The benchmark is designed for evaluating the performance of graphics recognition systems on images that contain polygons (solid) within the images. Our contribution is two-fold, an object mapping algorithm to spatially locate errors within the drawing, and then a cycle graph matching distance that indicates the accuracy of the polygonal approximation. The performance incorporates many aspects and factors based on uniform units while the method remains not rigid (threshold-less). This benchmark gives a scientific comparison at polygon level of coherency and uses practical performance evaluation methods that can be applied to complete polygonization systems.

A system dedicated to cadastral map vectorization was evaluated under this benchmark and its performance results are presented in this chapter. By stress testing a given system, we demonstrate that our protocol can reveal strengths and weaknesses of a system. The behavior of our set of indices was analyzed when increasing image degradation. We hope that this benchmark will help assessing the state of the art in graphics recognition and current vectorization technologies.

5.2 Introduction

In this chapter the question of performance evaluation is discussed.

Driven by the need to convert a large number of hard copy engineering drawings into CAD files, raster to vector conversion has been a field of intense research for the last four decades. In addition to research prototypes in several academic and industrial research centers, several commercial software products are currently available to assist users in converting raster images to vector (CAD) files. However, the process of selecting the right software for a given vectorization task is still a difficult one. Although trade magazines have published surveys of the functionality and ease of use of vectorization products [Byrnes 1997], a scientific, well designed, comparison of the auto-vectorization capability of the products was still required.

Responding to this need, the International Association for Pattern Recognition's technical committee on graphics recognition (IAPR TC10) organized the series of graphics recognition contests. The first contest, held at the GREC'95 workshop in University Park, PA, focused on dashed line detection [Kasturi 1996a], [Kong 1996], [Dori 1996]. The second contest, held at the GREC'97 workshop in Nancy, France, attempted to evaluate complete (automatic) raster to vector conversion systems [Chhabra 1998], [Phillips 1998], [Phillips 1999]. The third contest, held off-line associated with the GREC'99 workshop in Jaipur, India, also aimed to evaluate complete (automatic) raster to vector conversion systems. These contests tested the abilities of participating algorithms / systems to detect segments and arcs from raster images. They adopted a set of performance metrics based on the published line detection performance evaluation protocol [Wenyin 1997] to evaluate and compare the participating algorithms / systems on-line at the workshop site with test data of different quality and complexity. Pre-contest training images and the performance evaluation software were provided before the contests so prospective participants could try their systems and improved them for optimal performance. Test images could be synthesized and/or real scanned images.

5.2.1 Related Work

Performance evaluation and benchmarking have been gaining acceptance in all areas of computer vision and so in the graphics recognition field of science.

Early work on this topic were carried out to evaluate performance of thinning algorithms. [Haralick 1992] was the first to propose a general approach for performance evaluation of image analysis, with thinning taken as a case in point.

Evaluation and comparison of thinning algorithms have also been performed by [Lee 1991], [Lam 1993], [Jaisimha 1993] and [Cordella 1996]. Some of these evaluation and comparison works were carried out from the viewpoint of OCR, while the work of [Jaisimha 1993] is domain independent. Although thinning may also be employed as preprocessing of line detection, the latter has different characteristics and therefore requires different evaluation protocol.

The benchmark we present here is designed for evaluating the performance of graphics recognition systems on images that contain polygons within the images. Although the evaluator is limited to this entity type, it is useful, since most of engineering drawings use this geometric element.

Accurate and efficient vectorization of line drawings is essential for any higher level processing in document analysis and recognition systems. In spite of the prevalence of vectorization methods, no standard for their performance evaluation protocol exists at a polygon level. All prior works focused on a lower level of consistency (arcs and segments). We propose a protocol for evaluating polygon extraction to help compare, select, improve, and even design object detection algorithms to be incorporated into drawing recognition and understanding systems. The protocol can be seen as an extension to polygon level of related approaches by proposing an evaluation which is closer to the user requirements (i.e. at a semantic level). This new viewpoint on the problem involves two local dissimilarity measures for estimating polygon detection and approximation quality.

Vectorization and other line detection techniques have been developed to convert images of line drawings in various domains from pixels to vector form (e.g., [Kasturi 1990a], [Nagasamy 1990], [Filipski 1992]) and a number of methods and systems have been proposed and implemented (e.g., [Boatto 1992], [Vaxiviere 1992], [Dori 1995], [Dori 1993]). Objective evaluations and quantitative comparisons among the different shape detection algorithms are available thanks to protocols issued from GREC contests [Kasturi 1996b], [DBL 1998], [Chhabra 2000] that provide quantitative measurements.

Performance evaluation of vectorization and line detection has been reported by [Kong 1996], [Hori 1996], [Wenyin 1997] and [Chhabra 1998]. Kong et al. [Kong 1996] propose a quantitative method for evaluating the recognition of dashed lines. Hori and Doermann [Hori 1996] propose a quantitative performance measurement methodology for task-specific raster to vector conversion. Wenyin and Dori [Wenyin 1997] present a protocol for evaluating the recognition of straight and circular lines. Phillips and Chhabra [Chhabra 1998] define a methodology for evaluating graphics recognition systems operating on images that contain various objects such as straight lines and text blocks. All of these methods are limited in their applicability. Kong et al. [Kong 1996] have developed a protocol and a system for systematically evaluating the performance of line detection algorithms, mainly for dashed-line detection algorithms. They define the overlap criteria of the match between a ground truth and a detected line based on the angle and the distance between them, and the partial overlap is also considered. They do not allow for fragmentation of detected lines. They use several arbitrary and rigid thresholds, for

example, the angle should be less than 3° and the distance between two lines less than 5 cells.

Hori and Doermann [Hori 1996] instantiate and extend Haralick's framework for performance characterization in image analysis [Haralick 1992], in an application-dependent manner, for measuring the performance of raster to vector conversion algorithms. They provide a set of metrics (evaluation contents) which is specifically geared to vectorization of mechanical engineering drawings. The "applications" addressed in the work are thinning, medial line finding, and line fitting-all low level techniques that do not completely constitute vectorization. It is hard to extend the work to evaluate a complete vectorization system. Hori and Doermann's protocol does not distinguish between detection rate and false alarm rate. It does not include an overall evaluation metric. It does not allow for fragmentation of detected lines. The metrics for line evaluation are given in several nonuniform units. It uses length ratio, deviation, and count ratio to evaluate the line length detection, line location detection, and line quantity detection, respectively. There is lack of an overall evaluation metric which provides an overall combined performance evaluation of the algorithm under consideration.

Wenyin and Dori [Wenyin 1997] propose performance evaluation indices for straight and circular line detection. Detection and false alarm rates are defined at both the pixel level and the vector level. Use of pixel level performance indices (measures of shape preservation) is not completely appropriate for real images that contain severe distortions such as warping and/or other defects introduced in the hard copy drawing and/or defects generated by the scanning/imaging system. On such images, attempts to obtain a high pixel recovery index would unnecessarily require the detected vectors to be true to the distorted shape of the imaged lines, thereby making the detected lines fragmented. For such images, the pixel recovery index needs to be assigned less weight than the vector recovery index. However, there is no way to predetermine the right relative weights for the pixel and vector recovery indices.

Phillips and Chhabra [Chhabra 1998] present the task of evaluation from the opposite angle. They do not look at the complexity of the entities to be recognized. Instead, in their view, the true measure of performance has to be goal directed. The goal of line drawing recognition is to convert a paper copy or a raster image of a line drawing into a useful form (such as a vector CAD file). How well a graphics recognition system works should be measured by how much manual effort is required to correct the mistakes made by the system, not by how well it recognizes difficult shapes. The goal of the evaluation is to measure the cost of postprocessing operations that are necessary to correct the mistakes of vectorization. EditCost is the cost estimate for human post-editing effort to clean-up the recognition result.

An other view point, we want to present, is an auto-assessment approach where the ground-truth is not directly required. Introduced in chapter 4 in the section 4.5; Meta-Model Based Distance (MMBD) is an unsupervised method for map comparison. It takes as two inputs: (i) an ontology representing the expert knowledge under a structured formalism and (ii) the logical structure which is extracted from

a document thanks to our retrieval method. Document analysis and understanding systems decompose the document in information elements, characterized by the role they play in the document (frame, streets, parcels...), and specifies the relationships (syntactic and semantic) between these elements. This logical structure cannot be directly used for a self-evaluation purpose. The logical structure is not invariant from a document to another, in fact, the number of elements (Quarters, Parcels) is very document depend. To tackle this problem, a generalization is operated on the logical structure to obtain a higher point of view called meta-model. This meta-model gathers on a single node, all instances of a given concept and an edge does exist between two concepts if the relation is always true within the model. This meta-model, inferred from a model, confers a generic nature to the knowledge extracted from a document and so, a more stable source of information. Hence, the meta-model can be used to measure up the similarity with an expert-designed knowledge representation by finding common points and share terms. The flip-side of the coin is the approximation implied by the generic nature of the meta-modeling. This model-based approach has to partially ignore some information in order to be synthetic. In this way, an NMBD value of 0 doest not mean that the map is perfectly reconstituted. The distance will only increase if missing items from the raster to vector conversion affects the meta-model representation. If missing-elements engender "non standard" as define by the expert modeling. Another point, we want to discuss, is that the vectorization precision is not modeled, so, this aspect is not expressed by NMBD.

Based on this synthesis of performance evaluation systems, one can observe that most of these methods remains at a very low level of analysis of the information (vector level), while user requirements often concern high level analysis. From the related work which focuses on low level primitives (segments, arcs), we extend the global concept of performance evaluation of vectorized documents to polygon level. Herein, we present our work, a recovery index which combines a local overlapping metric at polygon level when data are closer to the semantic and a matching distance for evaluating the polygonal approximation correctness in term of edit operations.

5.2.2 Our approach

[Wenyin 1997] and [Chhabra 1998] are well suited tools to tackle the performance evaluation problem of vectorized documents. However, to be more realistic and closer to objects handled by humans, [Wenyin 1997] and [Chhabra 1998] also underlined the need to consider more complex structures or domain-specific objects into the assessment process. For instance, in [Chhabra 1998], Dr. Chhabra reported as a shortcoming that "The detection of polylines, polygons, objects, symbols, etc. was not tested". A step in this direction is to address the problem under the prism of grouping of vectors. Unfortunately, prior algorithms cannot be easily modified to reach higher level objects since no match was attempted between solid entities. In fact, if in the case of low-level primitives the matching can be easily resumed to an overlapping criterion for more complex elements the questions is more ambitious.

Due to fragmentation phenomena introduced by R2V tools, an entity of the ground-truth can likely be related to many elements of the auto-vectorized version of the document. Solving this ambiguity requires complex matching algorithms that are not provided by prior works because the underlying problem does not exist at a low level of analysis. Rather to consider polygon fragmentation and combination as being simply wrong, and to only allow the best match with the maximum overlap, we address the question in an optimal manner to find best polygon assignments. As a consequence, to address the performance evaluation problem at polygon level, we need to provide a robust object matching. In our our approach, this major phase is carried out by a combinatorial framework to perform polygon assignments. Secondly, from the original idea of *EditCost* explained by Phillips and Chhabra, the cost estimate for human post-editing effort to clean-up the recognition result, we propose the use of a graph matching. This paradigm provides more than a value in $\mathbb{R}$, it reveals the sequence of corrections to be made to transform a set of connected line segments into another.

Through the reading of the literature, on the topic of performance evaluation of document image algorithms, we took into account comments and limitations of former protocols to detect five desired points:

1. To consider object fragmentation
2. To provide indices in uniform units
3. A generic and a domain-independent protocol
4. An overall evaluation metric
5. To evaluate how much manual effort is required to correct mistakes made by the system

Our proposal fulfills these five points: (1) A polygon assignment method and a graph matching algorithm tackle both polygon and line fragmentation problem; (2) Our two indices are bounded between 0 and 1; (3) No assumptions about the kind of documents are made by our protocol, the only constraint is that the document must hold polygons; (4) An overall metric is provided by linear combination of the two proposed indices; (5) The EditCost representative of the manual labor to be made to correct a document is envisaged through the graph matching question in terms of basic edit operations (addition, deletion, substitution). Furthermore, working at a polygon level hold many advantages. It makes the spotting of errors easier, there are much less polygons than vectors into a drawing so the visualization of mistakes is pretty fast. This point is very important for industrial systems, since it permits to reduce the user correction time, by helping him to focus on errors directly. Furthermore, this facilitates the study of large samples of documents and new error categorizations may arise. Addressing the question from another point of view can help developers to improve and design R2V software.

`We can't solve problems by using the same kind`

```
of thinking we used when we created them.

(Albert Einstein)
```

We propose here a novel and optimal object matching for polygon comparison at a different point of view from prior works. We consider the coherency of the document at a polygon level. Our polygonization evaluation is based on a polygon mapping constrained by the topological information. While this measure appreciates the quality of the polygon overlapping, a cycle graph matching takes a closer look to a lower level of information : the segment layout within the polygons. In this way, we express the consistency of the drawing at a polygon point of view.

The smallest item that can be found in a engineering drawing is the segment.

Definition 18. *(Segment) In geometry, a line segment is a part of a line that is bounded by two end points, and contains every point on the line between its end points.*

This object is considerably versatile, the number of line segments present in a wire drawing can be very impacted by the vectorization algorithm due to the noise that occurred in the original image of documents (Noise due to the storage condition, digitization steps). On the opposite, we decide to investigate a more consistent and reliable object called polygon.

Definition 19. *(Polygon) In geometry, a polygon is traditionally a plane figure that is bounded by a closed path or circuit, composed of a finite sequence of straight line segments (i.e., by a closed polygonal chain). These segments are called its edges or sides, and the points where two edges meet are the polygon's vertices or corners. A polygon is a 2-dimensional example of the more general polytope in any number of dimensions.*

The polygons are formed by running a cycle detection algorithm on the heap of segments that composed the drawing. Invented in the late 1960s, Floyd's cycle-finding algorithm [Floyd 1967] is a pointer algorithm that uses only two pointers, which move through the sequence of points at different speeds. This polygon layer is more reliable and so it provides a better basement to build a dissimilarity measure on top of it. A conventional way of defining measures of dissimilarity between complex objects (maps, drawing issued from vectorization) is to base the measure on the quantity of shared terms. Between two complex objects o_1 and o_2, the aim is to find the matching coefficient mc, which is based on the number of shared terms. The polygon organization of a document is a good viewpoint, more stable and less subject to variations than the segment layer. In the mean time, it represents a complimentary view of the problem.

Polygonized elements issued from a raster to vector conversion method are assigned and measured up to a manually vectorized Ground Truth. The assignment problem is one of the fundamental combinatorial optimization problems in

the branch of optimization or operations research in mathematics. It consists of finding a maximum weight matching in a weighted bipartite graph.

In its proposed form, the problem is as follows:

- Let D_{GT}, D_{CG} be a Ground Truth document and a Computer Generated document, respectively.
- There are $|D_{CG}|$ number of polygons in D_{CG} and $|D_{GT}|$ number of polygons in D_{GT}. Any polygons (P_{CG}) from D_{CG} can be assigned to any polygons (P_{CG}) of D_{CG}, incurring some cost that may vary depending on the P_{CG}-P_{GT} assignment. It is required to map all polygons by assigning exactly one P_{CG} to each P_{GT} in such a way that the total cost of the assignment is minimized. This matching cost is directly linked to the cost function that measures the similarity between polygons.

Our combinatorial framework cuts down the algorithmic complexity to an $O(n^3)$ upper bound, depending on the number of polygons in the largest drawing. Hence, the matching can be achieved in polynomial time which tackles the computational barrier. We stand apart from the prior approaches by grouping low level primitives into polygons and then considering their matching at this high level point of view. Once, polygons are mapped, it is interesting to take a closer look to a lower level by checking out to segment layouts within the mapped polygons. This presents some advantages, elements are locally affected to define a local dissimilarity measure which is visually interesting; it makes easier the spotting of miss detected areas. A complete data flow process for polygonized document evaluation is proposed. Our contribution in this domain is two-fold. Firstly, we compare a ground truth document and a computer generated document thanks to an optimal framework that proceeds to the object mapping. Finally, another operator provides estimates the relation between the segments within two mapped polygons in terms of edit operations by means of a cycle graph matching. The figure 5.1 depicts an overview of our methodology.

5.2.3 Organization

The organization of the chapter is as follows: Sect. 5.3.1 describes theoretically and in terms of algorithm the polygon mapping method, Sect. 5.3.2 explains the cycle graph matching process in order to judge the quality of the polygonal approximation. Sect. 5.3.3 put forwards the type of errors that are likely to appear in object retrieval systems. Sect. 5.4 describes the experimental protocol, this section also explains how to interpret our new set of indices on a application to cadastral map evaluation. A summary is included in Sect. 5.5, followed by discussions and concluding remarks.

5.3 A set of indices for polygonization evaluation

In this section, we define the two criteria involved into our proposal for a performance evaluation tool dedicated to polygonization. In the first part, a polygon assignment

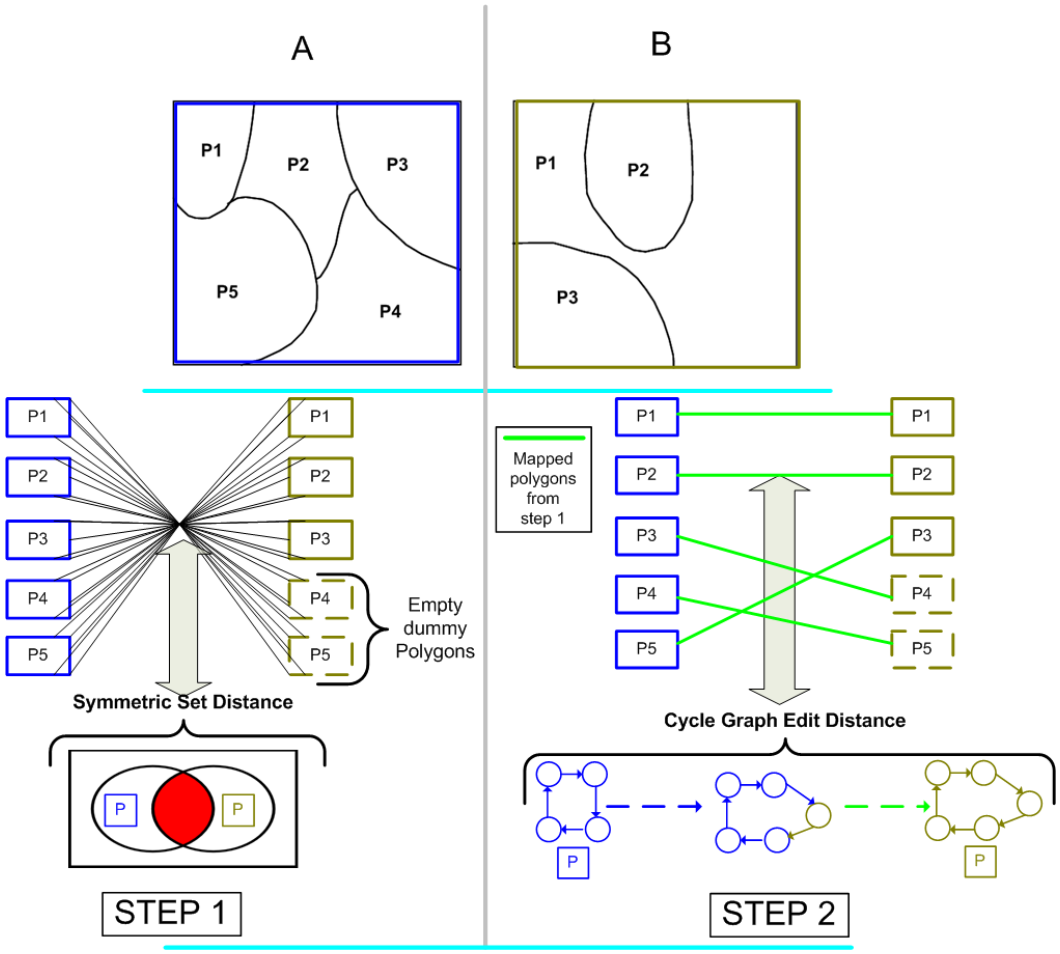


Figure 5.1: Overview of the global methodology. A bipartite graph weighed by the symmetric difference, and cycle graph edit distance applied to mapped polygons

method is described. It aims at taking into account shape distortions caused by retrieval systems. Secondly, a matched edit distance is defined. This measure represents the variations introduced when a given system approximates digital curves. It is a synthesis on vectorization precision. Finally, miss or over detection errors due to raster to polygon conversion are introduced in a third part. This decomposition leads to the definition of specific notations and error categorizations.

5.3.1 Polygon mapping using the Hungarian method

Once polygons are located into the vectorized document, it can be seen as a partition of polygons. Comparing two documents (D_1, D_2) is resumed to the matching of each polygon of D_1 with each polygon of D_2. This assignment is performed using the Hungarian method which is formally described in the next part.

5.3.1.1 Algorithmic of the Hungarian method

Our approach for vectorized document comparison is based on the assignment problem. The assignment problem considers the task of finding an optimal assignment of the elements of a set D_1 to the elements of a set D_2. Without loss of generality, we assume that $|D_1| \geq |D_2|$. The complete bipartite graph $G_{pm} = D_1 \cup D_2 \cup \triangle, D_1 \times (D_2 \cup \triangle)$, where $\triangle$ represents empty dummy polygons, is called the polygon matching of D_1 and D_2. A polygon matching between D_1 and D_2 is defined as a maximal matching in G_{pm}. We define the matching distance between D_1 and D_2, denoted by $PMD(D_1, D_2)$, as the cost of the minimum-weight polygon matching between D_1 and D_2 with respect to the cost function K. The cost function is especially dedicated to our problem and is fully explained in section 5.3.1.2. This optimal polygon assignment induces a univalent vertex mapping between D_1 and D_2, such as the function $PMD : D_1 \times (D_2 \cup \triangle) \rightarrow \mathbb{R}_0^+$ minimized the cost of polygon matching. If the numbers of polygons are not equal in both documents, then empty "dummy" polygons are added until equality $|D_1| = |D_2|$ is reached. The cost to match an empty "dummy" polygon is equal to the cost of inserting a whole unmapped polygon ($K(\emptyset, P)$). A shortcoming of the method is the one-to-one mapping aspect of the algorithm, however, this latter is performed at a high level of perception where data are less likely to be fragmented. Finally, this disadvantage should not discourage the use of the PMD distance considering the important speed-up it provides while being optimal, deterministic and quite accurate. In addition, unmapped elements are not left behind, they are considered either as "false alarm" or "false negative" according to the kind of mistakes they induced (see section 5.3.3). Formally, the assignment problem can be defined as follows.

Definition 20. *(The Assignment Problem) Let us assume there are two sets D_1 and D_2 together with an $n \times n$ cost matrix C of real numbers given. To improve the clarity of the reading $|D_1| = |D_2| = n$. The matrix elements C_{ij} correspond to the costs of assigning the i-th element of D_1 to the j-th element of D_2. The*

assignment problem can be stated as finding a permutation $p = p_1, p_2, ...p_n$ *of the integers 1,2,...,n that minimizes* $\sum_{i=1}^{n} C_{ip_i}$

The assignment problem can be reformulated as finding an optimal matching in a complete bipartite graph and is therefore also referred to as bipartite graph matching problem. Solving the assignment problem in a brute force manner by enumerating all permutations and selecting the one that minimizes the objective function leads to an exponential complexity which is unreasonable, of course. However, there exists an algorithm which is known as Munkres' algorithm [Munkres 1957][1] that solves the bipartite matching problem in polynomial time. In Algorithm 4 Munkres' method is described in detail. The assignment cost matrix C given in Definition 20 is the algorithm's input, and the output corresponds to the optimal permutation, i.e. the assignment pairs resulting in the minimum cost. In the description of Munkres' method in Algorithm 4 some lines (rows or columns) of the cost matrix C and some zero elements are distinguished. They are termed covered or uncovered lines and starred or primed zeros, respectively.

Munkres' algorithm is based on the following theorem.

Theorem 5.3.1. *(Equivalent Matrices) Given a cost matrix C as defined in Definition 20, a column vector c,* $c = (c_1, ..., c_n)$*, and a row vector* $r = (r_1, ..., r_n)$*, the square matrix C' with the elements* $C'_{ij} = C_{ij} - c_i - r_j$ *has the same optimal assignment solution as the matrix C. C and C' are said to be equivalent.*

Proof. ([Bourgeois 1971]) Let p be a permutation of the integers $1, 2, .., n$ minimizing $\sum_{i=1}^{n} C_{ip_i}$, then

$$\sum_{i=1}^{n} C'_{ip_i} = \sum_{i=1}^{n} C_{ip_i} - \sum_{i=1}^{n} c_i - \sum_{j=1}^{n} r_j$$

□

The values of the last two terms are independent of permutation p so that if p minimizes $\sum_{i=1}^{n} C_{ip_i}$, it also minimizes $\sum_{i=1}^{n} C'_{ip_i}$.

Consequently, if we find a new matrix C' equivalent to the initial cost matrix C, and a permutation p with all C'_{ip_i}, then p also minimizes $\sum_{i=1}^{n} C_{ip_i}$. Intuitively, Munkres' algorithm transforms the original cost matrix C into an equivalent matrix C' having n independent zero elements. This independent set of zero elements exactly corresponds to the optimal assignment pairs.

The operations executed in lines 1 and 2, and STEP 4 of Algorithm 4 find a matrix equivalent to the initial cost matrix (Theorem 5.3.1). In lines 1 and 2 the column vector $c = (c_1, ..., c_n)$ is constructed by $c_i = \min\{C_{ij}\}_{j=1,...,n}$, and the row

[1] Munkres' algorithm is a refinement of an earlier version by Kuhn [Kuhn 1955] and is also referred to as Kuhn-Munkres, or Hungarian algorithm.

Algorithm 4 Munkres' algorithm for the assignment problem

Require: A cost matrix C with dimensionality n

Ensure: The minimum cost polygon assignment

For each row r in C, subtract its smallest element from every element in r
For each column c in C, subtract its smallest element from every element in c
For all zeros z_i in C, mark z_i with a star if there is no starred zero in its row or column
STEP 1:
for Each column containing a starred zero **do**
    cover this column
end for
if n columns are covered **then**
    GOTO DONE
else
    GOTO STEP 2
end if
STEP 2:
if C contains an uncovered zero **then**
    Find an arbitrary uncovered zero Z_0 and prime it
    if There is no starred zero in the row of Z_0 **then**
        GOTO STEP 3
    else
        Cover this row, and uncover the column containing the starred zero **GOTO** STEP 2
    end if
else
    Save the smallest uncovered element e_{min} **GOTO** STEP 4
end if
STEP 3: Construct a series S of alternating primed and starred zeros as follows:
Insert Z_0 into S
while In the column of Z_0 exists a starred zero Z_1 **do**
    Insert Z_1 into S
    Replace Z_0 with the primed zero in the row of Z_1. Insert Z_0 into S
end while
Unstar each starred zero in S and replace all primes with stars. Erase all other primes and uncover every line in C **GOTO** STEP 1
STEP 4: Add e_{min} to every element in covered rows and subtract it from every element in uncovered columns. **GOTO** STEP 2
DONE: Assignment pairs are indicated by the positions of starred zeros in the cost matrix

vector $r = (r_1, ..., r_n)'$ by $r_j = \min\{C_{ij}\}_{i=1,...,n}$. In STEP 4 the vectors c and r are defined by the rules

$$c_i = \begin{cases} e_{min} & \text{if row } i \text{ is covered} \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

$$r_j = \begin{cases} 0 & \text{if column } j \text{ is covered} \\ e_{min} & \text{otherwise} \end{cases} \tag{5.2}$$

where e_{min} is the smallest uncovered element in the cost matrix C.

STEP 1, 2, and 3 of Algorithm 4 are procedures to find a maximum set of independent zeros which mark the optimal assignment. In the worst case the maximum number of operations needed by the algorithm is $O(n^3)$. Note that the $O(n^3)$ complexity is much smaller than the $O(n!)$ complexity required by a brute force algorithm.

Main Steps

After a formal description which explains theoretically how the Hungarian algorithm works, a case study is proposed providing a more practical apprehension of this method on the specific context of polygon mapping.

- Here are the main steps for achieving the Hungarian method. It begins with the construction of a cost matrix. Since we aim at evaluating a Computer Generated vectorization with the Ground-Truth, we will refer to these elements using the acronyms D_{CG} and D_{GT}, respectively. Each vectorized document is a set of objects, a set of polygons,

  $$D = \{Poly_1, Poly_2, ..., Poly_n\}.$$

  Let us state without loss of generality the following assumption : $|D_{GT}| \geq |D_{CG}|$, where $|X|$ stands for the number of polygons that composes the vectorization. Figure 5.2 represents the cost matrix, where every cell was filled up using a cost function (K) and represent the cost to associate a polygon of the Ground-Truth to a polygon of the Computer Generated vectorization. The cost function is especially dedicated to our problem and is fully explained in section 5.3.1.2.

- The Hungarian method is defined to find an optimal solution to the assignment problem where the input is a square matrix. How to make our matrix square ? The trick lies on the addition to empty dummy polygons. In such a way, the two sets have the same dimension:

CG \ GT	Poly 1	Poly 2	Poly 3	Poly 4
Poly 1	24	10	21	11
Poly 2	14	22	10	15
Poly 3	15	17	20	19

Figure 5.2: Original cost matrix

CG \ GT	Poly 1	Poly 2	Poly 3	Poly 4
Poly 1	24	10	21	11
Poly 2	14	22	10	15
Poly 3	15	17	20	19
Dummy	11	19	14	13

Figure 5.3: Square cost matrix

$$\begin{aligned} GT &= \{Poly_1, Poly_2, ..., Poly_n\} \\ CG &= \{Poly_1, Poly_2, ..., Poly_m, \\ &\quad Dummy_{m+1}, Dummy_{m+2}, ..., Dummy_n\} \end{aligned}$$

Figure 5.3 is a representation of the matrix being squared.

- **Step 1**: Subtract the entries of each row by the row minimum. Hence, each row has at least one zero and all entries are positive or zero (Figure 5.4).
- **Step 2**: Subtract the entries of each column by the column minimum. Each row and each column has at least one zero (Figure 5.5).
- **Step 3**: Select rows and columns across which you draw lines, in such a way

CG \ GT	Poly 1	Poly 2	Poly 3	Poly 4	Reduced
Poly 1	14	0	11	1	10
Poly 2	4	12	0	5	10
Poly 3	0	2	5	4	15
Dummy	0	8	3	2	11

Figure 5.4: Row reduction.

		GT			
		Poly 1	Poly 2	Poly 3	Poly 4
CG	Poly 1	14	0	11	0
	Poly 2	4	12	0	4
	Poly 3	0	2	5	3
	Dummy	0	8	3	1
	Reduced	0	0	0	1

Figure 5.5: Cost matrix after line and column reduction

		GT			
		Poly 1	Poly 2	Poly 3	Poly 4
CG	~~Poly 1~~	~~11~~	~~0~~	~~11~~	~~0~~
	Poly 2	4	12	0	4
	Poly 3	0	2	5	3
	Dummy	0	8	3	1

Figure 5.6: Three lines for covering the zeros

that all the zeros are covered and that no more lines have been drawn than necessary (Figure 5.6).

- **Step 4**: A test for optimality.
  - If the number of the lines is n then choose a combination from the modified cost matrix in such a way that the sum is zero.
  - If the number of the lines is less then n, go to step 5.

  In our case the number of lines is 3 (<4).

- **Step 5**: Find the smallest element which is not covered by any of the lines (figure 5.7). Then subtract it from each entry which is not covered by the lines (figure 5.8) and add it to each entry which is covered by a vertical and a horizontal line (figure 5.8). Go back to step 3 (figure 5.9).

- **Optimal results**: The final result is found by choosing the cells where the costs are minimum and zeros are applied (figure 5.10). Poly 1 and Dummy can not be picked because the affectation of zeros would not be optimal.

  Finally, the cells are summed up (figure 5.11). In our case study, the minimum cost to associate the polygons of GT with the polygons of CG is 48.

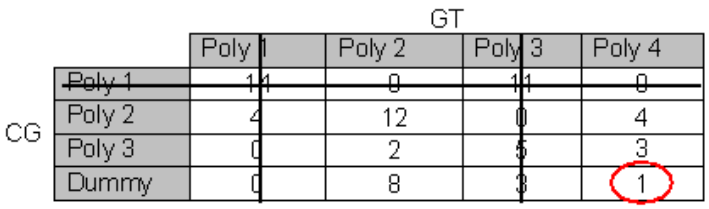


Figure 5.7: Minimal value not covered by any lines

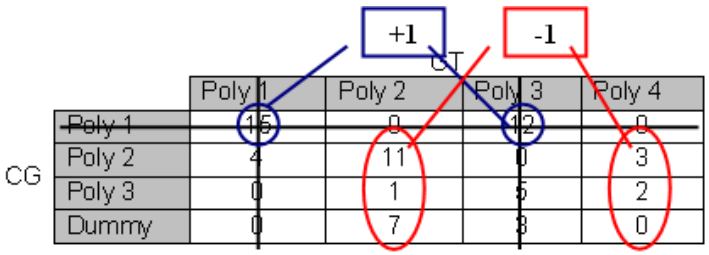


Figure 5.8: Matrix adjustment

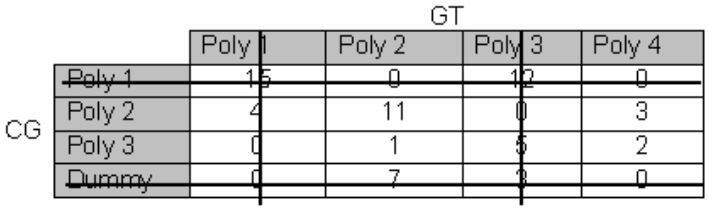


Figure 5.9: Finding the smallest element which is not covered by any of the lines (4)

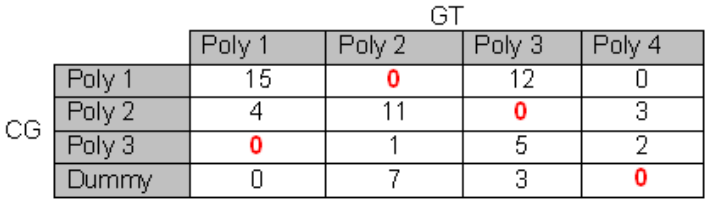


Figure 5.10: Optimal result

GT	CG	
Poly 1	Poly 2	10
Poly 2	Poly 3	10
Poly 3	Poly 1	15
Poly 4	Dummy	13
Cost		48

Figure 5.11: Final result

5.3.1.2 Cost function for polygon assignments

Munkres' algorithm as introduced in the last section provides us an optimal solution to the assignment problem in $O(n^3)$ time. In its generic form, the assignment problem considers the task of finding an optimal assignment of the elements of a set GT to the elements of a set CG assuming that numerical costs are given for each assignment pair. In fact, a cost function does exist between each pair of polygons to express numerically their similarity, likely a "zero" will represent two identical polygons and "one" two polygons not sharing any common features. The polygon overlay, inspired by the theory of sets, measures the similarity between polygons. When polygons are compared into the same axis system, the overlay takes into account spatial adjustment between polygons. The process of overlaying polygons shares common points with set theory. Let's assume that A and B are two sets, the intersection can be reformulated through the set theory. *Intersection*, where the result includes all those set parts that occur in A and B. A way to compare them is to find out how A differs from B (see figure 5.12): In mathematics, the difference of two sets is the set of elements which are in one of the sets, but not in both. This operation is the set-theoretic kin of the exclusive disjunction in Boolean logic. The symmetric difference of the sets A and B is commonly denoted by $A\Delta B$. The symmetric difference is equivalent to the union of both relative complements, that is:

$$A\Delta B = (A \setminus B) \cup (B \setminus A)$$

and it can also be expressed as the union of the two sets, minus their intersection:

$$A\Delta B = (A \cup B) \setminus (B \cap A) \tag{5.3}$$

The symmetric difference is commutative and associative:

$$A\Delta B = B\Delta A$$

The empty set is neutral, and every set is its own inverse:

$$A\Delta\emptyset = A$$

$$A\Delta A = \emptyset$$

From the original idea of the symmetric difference between two sets stated in equation 5.3, we derive a dissimilarity measure applicable to polygons. Let $P1$, $P2$ be two polygons and then, let us define a function K that induces a mapping of $P1$ and $P2$ into $\mathbb{R}$ ($K : P1 \times P2 \rightarrow \mathbb{R}$) :

$$K(P1, P2) = P1 \Delta P2 = 1 - \frac{|P1 \cap P2|}{|P1| + |P2| - |P1 \cap P2|} \tag{5.4}$$

Where $|P|$ denotes the area of the polygon P. Now, let us take a closer look to the basic properties of this dissimilarity measure :

$$K(P1, P2) >= 0 \ \ \forall P1, P2 \tag{5.5}$$

$$K(P1, P2) = 0 \ \rightarrow \ P1 = P2 \tag{5.6}$$

$$K : [0; 1] \tag{5.7}$$

5.3.1.3 Theoretical discussion on our dissimilarity measure

Two questions are addressed in this part, the first one concerns the unity of PMD when facing heterogeneous documents (different scales and orientations) and the second point is devoted to the proof of PMD as being a metric. This point is crucial to demonstrate the ability of providing a rank information which is representative of the error level of a given document with respect to the entire collection.

The cost function behavior: One condition imposes by the Hungarian method is that the cost function has to be strictly positive or zero, this assumption is respected (see equation 5.5). In addition, the normalization between zero and one (see equation 5.4, equation 5.7) confers some interesting aspects to the distance.

Without normalization, the distance is highly dependent on the polygon surfaces. A higher importance would be given to large polygons and they could highly impact the final score, while small polygons would be not treated with significance.

Hence, this measure considers with equity whether the concerned polygons are small or not. It means that no bias will be introduced when facing large polygons. Thereby, a highly over segmented vectorization with many small areas will be roughly as bad as an under segmented vectorization with few large polygons. Finally, the normalization leads to a lower and an upper bounds (equation 5.7) of our distance which is useful to compare a document collection with different scales. In this way, PMD is dependent to scale, translation and rotation variations. Nevertheless, these are desired properties for a distance which wants to represent the exactness between two polygonized drawings.

The Polygon Matching Distance is a metric (PMD):

Proof. To show that our measure of similarity between documents is a metric, we have to prove four properties for this similarity measure.

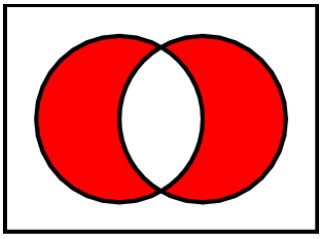


Figure 5.12: $A \Delta B$

- $PMD(D_1, D_2) \geq 0$
  The polygon matching distance between two documents is the sum of the cost for each polygon matching. As the cost function is non-negative, any sum of cost values is also non-negative.

- $PMD(D_1, D_2) = PMD(D_2, D_1)$
  The minimum-weight maximal matching in a bipartite graph is symmetric, if the edges in the bipartite graph are undirected. This is equivalent to the cost function being symmetric. As the cost function is a metric, the cost for matching two polygons is symmetric. Therefore, the polygon matching distance is symmetric.

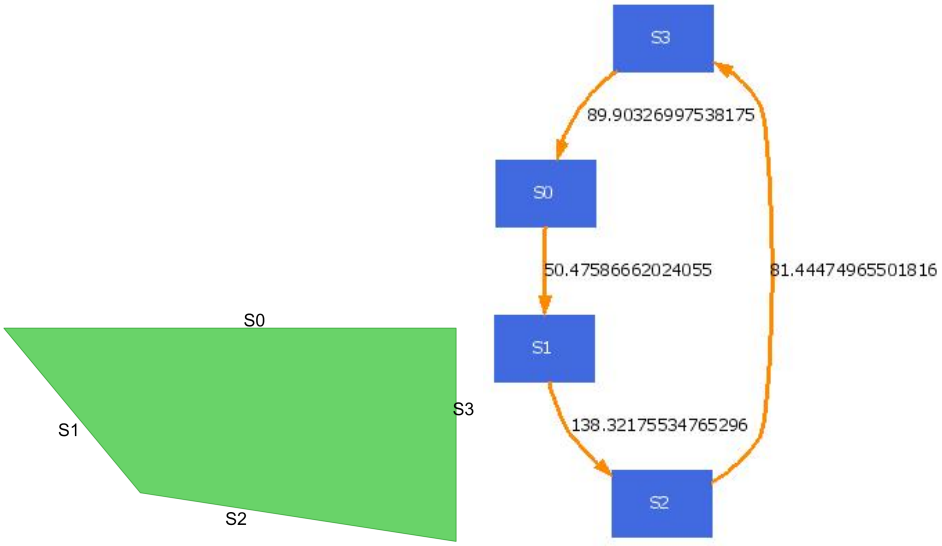
- $PMD(D_1, D_3) \leq PMD(D_1, D_2) + PMD(D_2, D_3)$
  As the cost function is a metric, the triangle inequality holds for each triple of documents in D_1, D_2 and D_3 and for those polygons that are mapped to an empty polygon. The polygon matching distance is the sum of the cost of the matching of individual polygons. Therefore, the triangle inequality also holds for the polygon matching distance.

- $PMD(D_1, D_2) = 0 \Rightarrow D_1 = D_2$
  If one of the polygon of D_1 cannot be matched exactly with a polygon of D_2 then $PMD(D_1, D_2) > 0$. A straightforward interpretation of this fact leads to the uniqueness property. Where all D_1' polygons are matched with a cost of zero to the polygons of M_2, it implies $D_1 = D_2$.

□

5.3.2 Matched edit distance for polygon comparison

The Hungarian method provides a formal framework to perform a one to one mapping between polygons. Each mapped pair of polygons minimizes its symmetric difference providing a topological information. However, this measure does not take into account the labor work that has to be done to change a polygon from the CG to a correct polygon from the GT. In order to compensate this weakness, we decide to include an additional measure which reveals how many edit operations have to be done to change a polygon into another according to some basics operations. That's



(a) Polygon (b) Cycle Graph

Figure 5.13: From polygon to cycle graph

why we present an edit distance for polygon comparison. From definition 19 and the figure 5.13, a clear link exists between a polygon and its representation by a cycle graph. The next part defines an Cycle Graph Edit Distance (CGED) for polygon comparison, this latter starts from the string matching theory to end with the graph matching problem applied cycle graph.

The guide lines illustrated in figure 5.14 will drive our discussion. We first define string matching theory which is the basement of our work. Then step by step, we explain why string matching and cyclic string matching cannot completely address our problem. Thirdly, we define a graph-based tool for polygon comparison. Finally, we provide a concrete interpretation of the cost functions for the Cycle Graph Edit Distance.

5.3.2.1 String Matching Theory and Algorithms

String edit distances were first defined by Wagner and Fischer in [Wagner 1974] to find out the minimum cost edit sequence to convert the string A into the string B using edit operations. Although the origin of the algorithm is spelling correction, it has been used for different purposes, and particularly as an approach to the problem of recognizing and classifying polygons. The problem is to define dissimilarity measures between polygons, and to find algorithms that compute these measures fast enough. The string matching-based approaches should be independent of the scale, translation and rotation of the polygons under analysis. Let us review the string matching theory and algorithms and subsequently provide the details about our particular cost functions to match polygons.

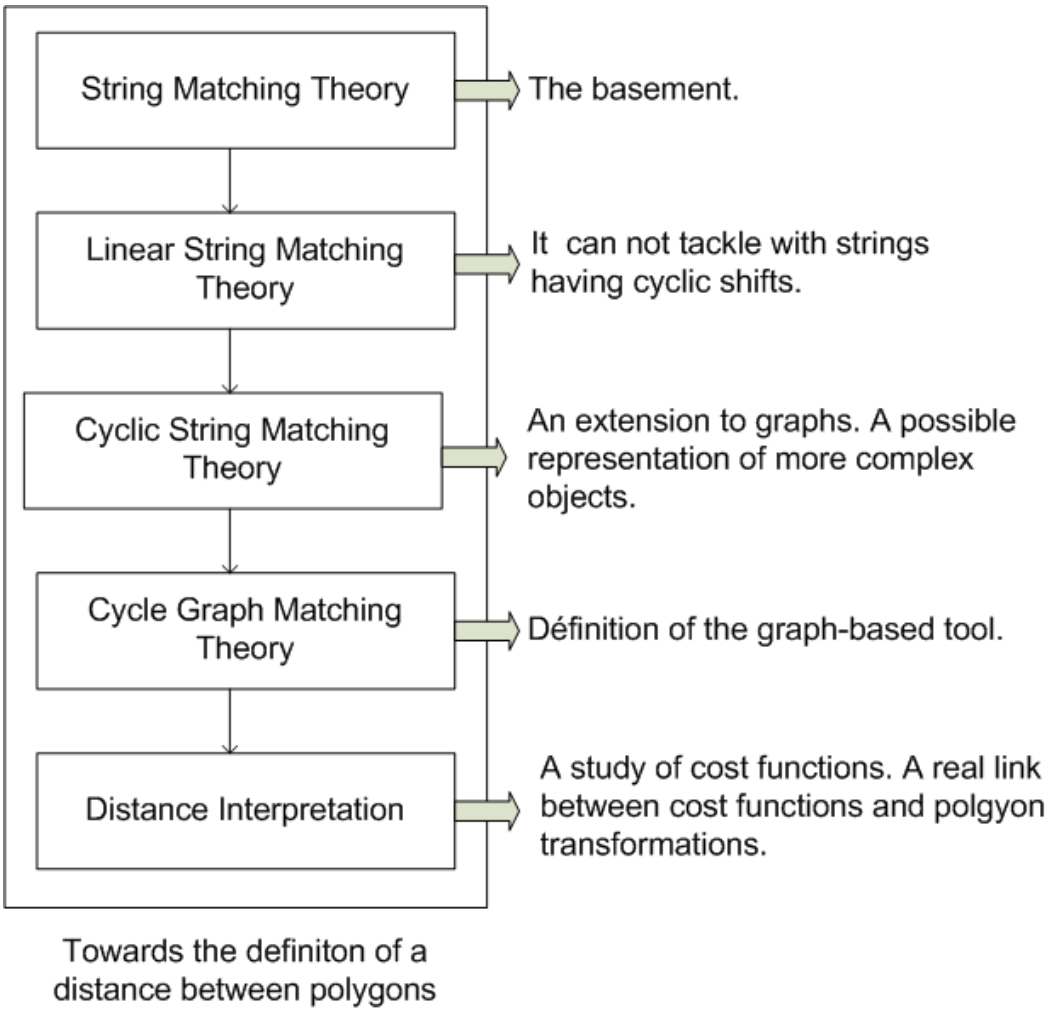


Figure 5.14: Cycle Graph Matching for Polygon Comparison

Definitions: Let us first introduce some basic notations and definitions of the basic string matching algorithm first proposed by Wagner and Fischer in [Wagner 1974].

Definition 21. *Let P be a polygon and let A denote the string over P constituted of all points of P. The length $|A|$ of a string $A \in P$ is the number of points in A. And let Λ denote the null string which has length 0.*

Definition 22. *For a string $A = a_1a_2...a_n \in P$, a cyclic shift is a mapping $\sigma : A \rightarrow A$ defined by $\sigma(a_1a_2...a_n) = a_2a_3...a_na_1$. For all $k \in \mathbb{N}$, let σ^k denote the composition of k cyclic shifts. Two strings A and $\overline{A}$ will be called equivalent if $A = \sigma^k(\overline{A})$.*

Definition 23. *An edit operation is an ordered pair $(a, b) \neq (\Lambda\ ,\Lambda)$ of strings, each of a length less than or equal to 1, denoted by $a \rightarrow b$. An edit operation $a \rightarrow b$ will be called an insert if $a = \Lambda$, a delete operation if $b = \Lambda$, and a substitution operation otherwise.*

Definition 24. *A string B results from a string A by the edit operation $s = (a \rightarrow b)$, denoted by $A \rightarrow B$ via s, if there are strings C and D such that $A = CaD$ and $B = CbD$. An edit sequence $S = s_1s_2...s_k$ is a sequence of edit operations. We say that S takes A to B if there are strings $A_0, A_1, ..., A_k$ such that $A_0 = A, A_k = B$ and $A_{i-1} \rightarrow A_i$ via s_i for all $i \in \{1, 2, ..., k\}$.*

Definition 25. *Let γ be a cost function that assigns a non-negative real number $\gamma(s)$ to each edit operation. For an edit sequence S, we define the cost $\gamma(S)$ as*

$$\gamma(S) = \sum_{i=1}^{k} \gamma(s_i)$$

The edit distance $\delta(A, B)$ from string A to string B is then defined as

$$\gamma(A, B) = min\,\{\gamma(S)\}$$

And the edit distance $\delta([A]\,,[B])$ of two cyclic strings [A] and [B] is given by

$$\delta([A], [B]) = min\left\{\delta(\sigma^k(A), \sigma^l(B))\right\} \text{ with } k, l \in \mathbb{N}$$

Linear String Matching: Let A and B be two strings over $\sum$ of length n and m respectively. The Wagner and Fischer [Wagner 1974] algorithm takes $O(nm)$ time to find $\delta(A, B)$ by determining a minimum weighted path in a weighted directed graph. Let $D(i, j)$ denote the cost of a minimum weighted path from the vertex $v(0, 0)$ to the vertex $v(i, j)$, so $D(n, m) = \delta(A, B)$.

Cyclic String Matching: Linear string matching can not tackle with strings having cyclic shifts since the computed path starts always from a given initial symbol. A cyclic string matching procedure is needed in the case of cyclic strings. Given two finite strings A and B, the cyclic string matching problem is the problem of determining $\delta([A],[B])$ and an edit sequence realizing this cost. Let $BB = b_1b_2...b_mb_1b_2...b_m$ be the concatenation of B with itself. For all $l \in \{1,2,...,m\}$, we can find a minimum cost edit sequence from A to $\sigma^l(B)$ by determining a minimum weighted path from $v(0,l)$ to $v(n,m+l)$. Although the computation of only one path takes $O(nm)$ time, the computation of all these paths can be done in $O(nm\,log\,m)$ time, since all the paths can be chosen such that two different paths never cross.

5.3.2.2 A Cycle Graph Matching Distance for Polygon Comparison

Visually, two chains of segments are similar if the length attributes and angles between consecutive segments can be aligned. In the literature on polygonal shape recognition, most approaches base the distance definition between two polygonal shapes on length and angle differences. For example, Arkin et al. used in [Arkin 1991] the turning function which gives the angle between the counterclockwise tangent and the x-axis as a function of the arc length. Their results are in accordance with the intuitive notion of shape similarity. More recently, in [Lladós 2001], Lladós et al. represented regions by polylines and string matching techniques are used to measure their similarity. The algorithm follows a branch and bound approach driven by the RAG edit operations. This formulation allows matching computing under distorted inputs. The algorithm has been used for recognizing symbols in hand drawn diagrams.

Polygonal shapes require to characterize the segments (their length) but also their relationships by the angle information.

The graph-based representation was preferred to string representation. In fact, the protocol is designed for polygons but may also be extended to other line shapes, for instance this could be made by completing the graph representation to connected vectors instead of searching for cyclic polygons. In this way, the graph-based viewpoint could be the container of a wider range of entities. It leaves the door open to a more global paradigm, the object matching question.

The concept of edit distance has been extended from strings to trees and to graphs [Bunke 1983], [Sanfeliu 1983]. Similarly to string edit distance, the key idea of graph edit distance is to define the dissimilarity, or distance, of graphs by the minimum amount of distortion that is needed to transform one graph into another. Compared to other approaches to graph matching, graph edit distance is known to be very flexible since it can handle arbitrary graphs and any type of node and edge labels. Furthermore, by defining costs for edit operations, the concept of edit distance can be tailored to specific applications. A standard set of distortion operations is given by insertions, deletions, and substitutions of both nodes and edges. We denote the substitution of two nodes u and v by $(u \rightarrow v)$, the deletion

of node u by$(u \rightarrow \Lambda)$, and the insertion of node v by $(\Lambda \rightarrow v)$. For edges we use a similar notation.

Given two graphs, the source graph G_1 and the target graph G_2, the idea of graph edit distance is to delete some nodes and edges from G_1, relabel (substitute) some of the remaining nodes and edges, and insert some nodes and edges in G_2, such that G_1 is finally transformed into G_2.

A sequence of edit operations $e_1; ...; e_k$ that transforms G_1 completely into G_2 is called an edit path between $G_1$and G_2. Obviously, for every pair of graphs $(G_1; G_2)$, there exist a number of different edit paths transforming G_1 into G_2. To find the most suitable edit path, one introduces a cost for each edit operation, measuring the strength of the corresponding operation. The idea of such a cost function is to define whether or not an edit operation represents a strong modification of the graph. Obviously, the cost function is defined with respect to the underlying node or edge labels. Clearly, between two similar graphs, there should exist an inexpensive edit path, representing low cost operations, while for dissimilar graphs an edit path with high costs is needed. Consequently, the edit distance of two graphs is defined by the minimum cost edit path between two graphs. The computation of the edit distance is carried out by means of a tree search algorithm which explores the space of all possible mappings of the nodes and edges of the first graph to the nodes and edges of the second graph.

Definition 26. *(Cycle Graph Matching)*

In this work, the problem which is considered concerns the matching of cycle directed labeled graphs. Such graphs can be defined as follows: Let L_V and L_E denote the set of node and edge labels, respectively. A labeled graph G is a 4-tuple $G = (V, E, \mu, \xi)$, where

- *V is the set of nodes,*
- *$E \subseteq V \times V$ is the set of edges*
- *$\mu : V \rightarrow L_V$ is a function assigning labels to the nodes, and*
- *$\xi : E \rightarrow L_E$ is a function assigning labels to the edges.*
- *$|V|$=$|E|$*

Now, let us define a function CGED based on the Cycle Graph Edit Distance that induces a mapping of G_1 and G_2 into $\mathbb{R}$ ($CGED : G_1 \times G_2 \mapsto \mathbb{R}$) :

$$CGED(G_1, G_2) = \min_{(e_1,..,e_k) \in \gamma(G_1,G_2)} \sum_{i=1}^{k} (edit(e_i))$$

Where $\gamma(G_1, G_2)$ denotes the set of edit paths transforming G_1 into G_2, and edit denotes the cost function measuring the strength $edit(e_i)$ of edit operation e_i

.

Table 5.1: Edit costs

	Node	Edge
Label Substitution	$\gamma((l_i^A) \rightarrow (l_j^B)) = \left\lvert \frac{l_i^A}{\lvert A \rvert} - \frac{l_j^B}{\lvert B \rvert} \right\rvert$	$\gamma((\Phi_i^A) \rightarrow (\Phi_j^B)) = \frac{\lvert \Phi_i^A - \Phi_j^B \rvert}{360}$
Addition	$\gamma(\lambda \rightarrow (l_j^B)) = \frac{l_j^B}{\lvert B \rvert}$	$\gamma(\lambda \rightarrow (\Phi_j^B)) = \frac{\lvert \Phi_j^B \rvert}{360}$
Deletion	$\gamma((l_i^A) \rightarrow \lambda) = \frac{l_i^A}{\lvert A \rvert}$	$\gamma((\Phi_i^A) \rightarrow \lambda) = \frac{\lvert \Phi_i^A \rvert}{360}$

A Cycle Graph is a cycle which visits each vertex exactly once and also returns to the starting vertex. As a consequence, the set of all edit paths is considerably reduced and the Cycle Graph matching can be solved in $O(nm\,log\,m)$ time.

In order to use cycle graph matching for polygon accuracy evaluation, we use an attributed graph representation. Starting from a polygonal approximation of the shape, a graph is built. We use the segments as primitives, encoding them with a set of nodes. Each node is labeled with a real number l_i, where l_i denotes the length of the segment s_i. Then, edges are built using the following rule: two nodes are linked with a directed and attributed edge if two constitutive segments share a common point. Each edge is labeled with a real number Φ_i that denotes the angle between s_i and s_{i-1} in the counterclockwise direction. We can appreciate an example of how these attributes are computed for a sample shape in figure 5.13.

Let A and B be two chains of adjacent segments, represented as cycle graphs, with total lengths $|A|$ = n and $|B|$ = m and with respectively attributed cycle graph representations:

$$G_A = (V^A, E^A) = (l_i^A ... l_n^A), (\Phi_i^A ... \Phi_n^A)$$

and,

$$G_B = (V^B, E^B) = (l_i^B ... l_m^B), (\Phi_i^B ... \Phi_m^B)$$

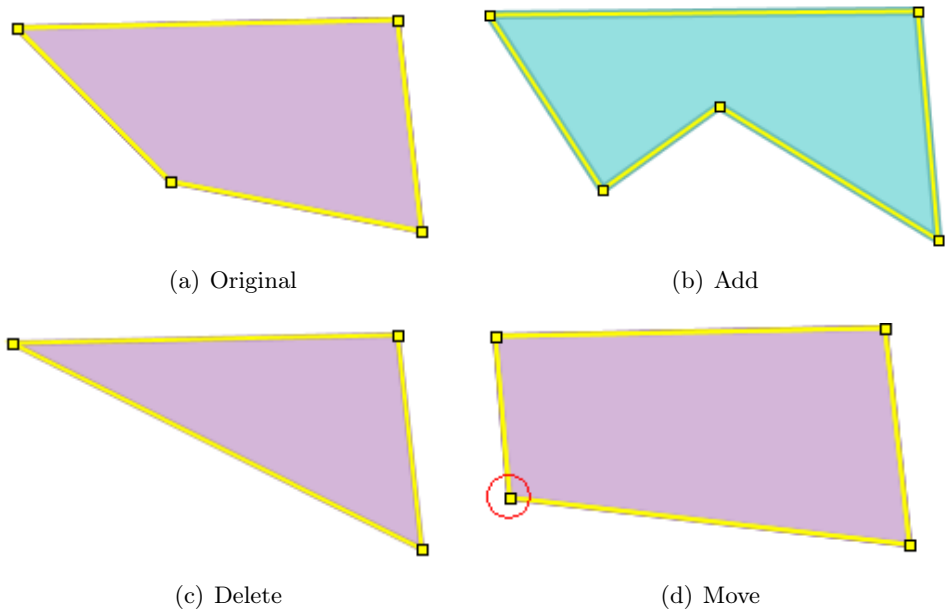
The cost functions for attributed cycle graph matching are reported in table 5.1

5.3.2.3 Interpretation of the edit operations

There are the proposed cost functions inspired by the ones proposed by Tsay and Tsai in [Tsay 1989] where they use string matching for shape recognition.

The operation attributes decrease the edit costs for primitives undergoing noisy transformations, as the inherent segment fragmentation from the raster-to-vector process. And it aims to compare polygons with different number of segments making the system tolerant to segment cardinality.

Furthermore, our edition functions can describe real transformations applied to polygons. When editing a vectorization basic operations are: Remove, Add or Move a segment; a visual illustration of theses operations is given in figure 5.15. Through the linear combination of the cost functions, it is possible to recreate the usages of a person modifying a vectorization, and the definitions below present the combinations



(a) Original (b) Add

(c) Delete (d) Move

Figure 5.15: Basic edit operations applied to a polygon.

to obtain different polygon transformations. Conceptually, we are here very close to the ideas proposed by Chhabra [Chhabra 1998].

Definition 27. *Segment deletion transformation*

$$\gamma(s_i \rightarrow \lambda) = \gamma((l_i^A) \rightarrow \lambda) + \gamma((\Phi_i^A) \rightarrow \lambda) + \gamma((\Phi_i^A) \rightarrow (\Phi_j^B))$$

$$\gamma(s_i \rightarrow \lambda) = \frac{l_i^A}{|A|} + \frac{\Phi_i^A}{360} + \frac{|\Phi_i^A - \Phi_j^B|}{360}$$

Definition 28. *Segment addition transformation*

$$\gamma(\lambda \rightarrow s_j) = \gamma(\lambda \rightarrow (l_j^A)) + \gamma(\lambda \rightarrow (\Phi_j^A)) + \gamma((\Phi_i^A) \rightarrow (\Phi_j^B))$$

$$\gamma(\lambda \rightarrow s_j) = \frac{l_j^A}{|A|} + \frac{\Phi_j^A}{360} + \frac{|\Phi_i^A - \Phi_j^B|}{360}$$

Definition 29. *Segment move transformation*

$$\gamma(s_i) \rightarrow (s_j) = \gamma((l_i^A) \rightarrow (l_j^B)) + \gamma((\Phi_i^A) \rightarrow (\Phi_j^B))$$

$$\gamma(s_i) \rightarrow (s_j) = \left| \frac{l_i^A}{|A|} - \frac{l_j^B}{|B|} \right| + \frac{|\Phi_i^A - \Phi_j^B|}{360}$$

5.3.2.4 Matched Edit Distance

To complete the process, the Cycle Graph Matching Distance ($CGED$) has to be performed on every pair of mapped polygons found by the Hungarian methods when it is based on the symmetric difference. Note that if one polygon is associated to an empty dummy item then the cycle graph matching distance will be only composed of addition operations. The Matched Edit Distance(MED) is then composed of the sum of all $CGED(G_1, G_2)$ computed on every pair of graphs extracted from the polygons.

$$MED(D_1, D_2) = \frac{\sum_{i=1}^{\max(|(D_1|,|D_2|)} CGED(G_i^{D_1}, G_i^{D_2})}{\max(|(D_1|, |D_2|)}$$

5.3.3 Type of errors and notations

Here, we sum up a set of two criteria which will help us to evaluate a given raster to vector conversion. Each measure is a viewpoint on the vectorization process. However, every criterion can still be divided into two categories according to the nature of the error it expresses. Hence, the next part defines the different kind of errors than can occur when dealing with object retrieval systems.

5.3.3.1 Type of errors

- Type I error, also known as an "error of the first kind", a false alarm or a "false positive": the error of rejecting a null hypothesis when it is actually true. Plainly speaking, it occurs when we are detecting a polygon when in truth there is none, thus indicating a test of poor specificity. An example of this would be if an application retrieves a polygon when in reality there is not. Type I error can be viewed as the error of excessive credulity.
- Type II error, also known as an "error of the second kind", a "false negative": the error of failing to reject a null hypothesis when it is in fact not true. In other words, this is the error of failing to detect a polygon when in truth there is one, thus indicating a test of poor sensitivity. An example of this would be if a test shows that there is not a polygon when in reality she is. Type II error can be viewed as the error of excessive skepticism.

5.3.3.2 Notations

For the comprehension of theses tests, we first introduce notations that will make the reading much simpler. A dissimilarity measure between vectorized documents is a function :

$$d : X \times X \to \mathbb{R}$$

Notation	Method	Type of error	Distance
PMD_{tp}	PMD	True Positive	symmetric difference
PMD_{fn}	PMD	False Negative	symmetric difference
PMD_{fp}	PMD	False Positive	symmetric difference
PMD_{md}	PMD	Miss Detection (fn+fp)	symmetric difference
PMD_{all}	PMD	(tp+fn+fp)	symmetric difference
MED_{tp}	MED	True Positive	Cycle Graph Matching
MED_{fn}	MED	False Negative	Cycle Graph Matching
MED_{fp}	MED	False Positive	Cycle Graph Matching
MED_{md}	MED	Miss Detection (fn+fp)	Cycle Graph Matching
MED_{all}	MED	(tp+fn+fp)	Cycle Graph Matching
η_{tp}	-	True Positive	♯ of well-detected polygons
η_{fp}	-	False Positive	♯ of over-detected polygons
η_{fn}	-	False Negative	♯ of under-detected polygons

Table 5.2: Distance between vectorized documents.

where X is a vectorized document. We report in table 5.2, the notations derived from this general form.

X_{tp} puts forward the cost that occurs when matching a pair of mapped polygons. This is a synonym of accuracy, it denotes how well suited is the detected polygon from the D_{CG}. X_{fp} takes the stock of the over detections issued from the raster to vector conversion step. On the other hand, X_{fn} represents the miss-detections, it occurs when the software used to vectorized has a strict policy of rejection which leads to an under detection of objects. For clarity reason, when no precision is specified, X refers to X_{all}. Finally, a desirable information is the number of false alarms, false negative and true positive polygons retrieved by the system of retro-conversion. These values are normalized as follow to obtain a comparable rate between documents.

$$\eta_{fn} = \frac{\sharp \text{ of miss-detected polygons}}{\max\left(|(D_1|, |D_2|\right)}$$

$$\eta_{fp} = \frac{\sharp \text{ of over-detected polygons}}{\max\left(|(D_1|, |D_2|\right)}$$

5.4 Experiments

This section is devoted to the experimental evaluation of the proposed approach. Firstly, we describe databases that are used to benchmark our measures. Then the protocol of our experiments is defined by enumerating the kind of assessments we performed. The two first tests are dedicated to graphical symbols from GREC contests. On this basement, we aim at illustrating the ability of Polygon Matching Distance (PMD) and Matched Edit Distance (MED) of being representative of polygon deformations (shape variation and polygonal approximation modification, respectively). The last evaluation concerns the cadastral map subject, we show results on a large collection of maps. We provide guidelines to understand the meaning of our set of indices, in this pedagogic objective, a visualization of detection errors is proposed. In this practical work, methods were implemented in Java 1.5 and run on a 2.14GHz computer with 2G RAM. Both databases and performance evaluation tools are freely available on this web site:

http://alpage-l3i.univ-lr.fr/

Both datasets and the experimental protocol are firstly described before investigating and discussing the merits of the proposed approach.

5.4.1 Databases in use

In recent years the subject of performance evaluation has gained popularity in pattern recognition and machine learning. In the graphics recognition community, a huge amount of efforts was made to elaborate standard and publicly available data sets. Especially, E. Valveny [Valveny 2004],[Valveny 2007] and M. Delalandre [Delalandre 2010] published on-line symbol datasets for a symbol classification purpose. In this section, we describe two databases derived from [Valveny 2007] and [Delalandre 2010] and we also present a cadastral map collection. The content of each database is summarized in tables 5.3, 5.4.

Base A: Shape distortion. The paper presented in [Delalandre 2010] gave birth to a publicly available database of symbols[2]. From this setting, we removed all polygon-less symbols to fit our purpose which was to evaluate polygon detection methods. Hence, we selected 70 symbols from the GREC'05 contest [Dosch 2006] and a sample is presented in figure 5.16.

On perfect symbols, a vectorial noise is applied to generate a collection of degraded elements. We could not afford to use real data because of the difficulty of collecting images with all kinds of transformations and noise. Besides, it is not easy to quantify the degree of noise in a real image. Then, it is not possible to define a

[2] *http : / / mathieu. delalandre.free.fr /projects /sesyd /sketches.html*

Table 5.3: Characteristics of the cadastral map collection: Base C

	♯ of polygons	♯ of vectors
$\|GT\|$	2335	654017
$\|CG\|$	2626	850667
mean $\|GT\|$	23.35	64.75
mean $\|CG\|$	26.26	85.06
max $\|GT\|$	83	101
max $\|CG\|$	69	100

Table 5.4: Characteristics of the symbols data sets: Base A, B

	Base A	Base B
Number of classes (N)	70	53
$\mid Base \mid$	360	371
Noise type	Vectorial	Binary
Noise levels	4	6
Assessment purpose	Shape distortion	Digital curve approximation

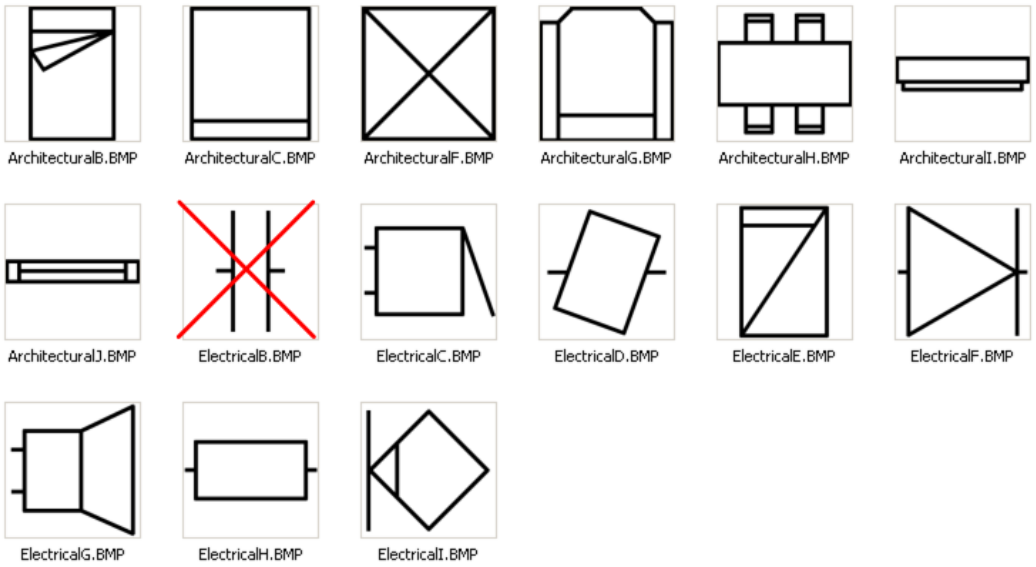


Figure 5.16: A sample among the seventy symbols used in our ranking test. Polygonless symbols were removed.

ranking of difficulty of images according to the degree of noise. In our experiments, we have re-used methods for the generation of shape transformation (based on active shape models [Valveny 2004]).

Vectorial Distortion: The goal of vectorial distortion is to deform the ideal shape of a symbol in order to simulate the shape variability produced by hand-drawing. The method for the generation of vectorial distortions of a symbol is based on the Active Shape Models [Cootes 1995]. This model aims to build a model of the shape, statistically capturing the variability of a set of annotated training samples. In order to be able to apply this method, we need to generate a good set of training samples. This is not a straightforward task due to the statistical nature of the method. The number of samples must be high enough, and the samples must reflect the usual kind of variations produced by hand-drawing. However, it is difficult to have a great number of hand-drawn samples of each symbol. To be really significant, these samples should be drawn by many different people. Thus, the decision of generating automatically the set of samples has arisen. Based on the generation of deformed samples through the random modification of a different number of vertices of the symbol each time [GHOSH 1999].

Each sample is represented using the model described in [Valveny 2003], which permits easy generation of deformed shapes. Each symbol is described as a set of straight lines, and each line is defined by four parameters: coordinates of mid-point, orientation and length. Thus, each deformed sample can be seen as a point x_i in a $4n$ dimensional space, where n is the number of lines of the symbol. Then, principal component analysis (PCA) can be used to capture the variability in the sample set. Given a set of samples of a symbol, we can compute the mean $\overline{x}$ and the covariance matrix S. The main modes of variation are described by the first eigenvectors p_k of the covariance matrix S. The variance explained by each eigenvector is equal to its corresponding eigenvalue. Thus, each shape in the training set can be approximated using the mean shape and a weighted sum of the eigenvectors:

$$x = \overline{x} + Pb$$

where $P = p_1, ..., p_m$ is the matrix of the first m eigenvectors and b is a vector of weights. This way, new images of a symbol can be generated by randomly selecting a vector of weights b. Increasing values of b_i will result in increasing levels of distortion (see figure 5.17).

The model of vectorial distortion described in the former paragraph has been applied with four increasing levels of distortion to generate 280 (70*4) images of symbols. The variance was tuned from 0.00025 to 0.00100 by step of 0.00025. This way of changing the variance is coherent with the protocol presented in [Delalandre 2010]. The entire database is then made up of 360 elements, 280 degraded symbols and 70 models. The shape distortion generator, 3gT system was provided by M. Delalandre[3]. 3gT "generation of graphical ground Truth" is a system to generate random

[3]http://mathieu.delalandre.free.fr/projects/3gT.html

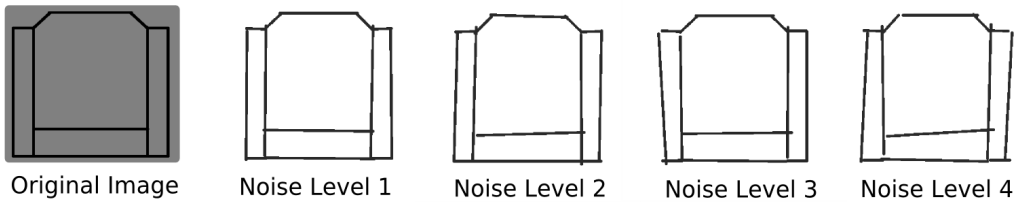


Figure 5.17: Examples of increasing levels of vectorial distortion

graphical documents (exported into SVG) of low level primitives (vectors, circles, ...) with their corresponding ground truth. Base A is reliable source to evaluate the shape distortion sensitivity of our polygon location measure. Numerical details concerning this data set are presented in table 5.4.

Base B: Binary degradation. First of all, we decided to use the data set provided by the GREC'03 contest. Mainly two application domains, architecture and electronics were adopted as a representative sample of a wide range of shapes. GREC'03 database is originally constituted of 59 symbols from which we removed symbols without polygons. This pruning step led us to a database of 53 symbols. From the 9 noise levels available, we only focused on the 6^{th} firsts. Consequently, database B is made of 318 (6*53) binary degraded symbols plus 53 ideal model. A total of 371 polygonized elements according to the process explained in the next paragraph.

Binary Degradation: Kanungo et al. have proposed a method to introduce some noise on bitmap images [Haralickt 2009]. The purpose of this method is to modelize noises obtained by operations like printing, photocopying, or scanning processes. The problem is approached from a statistical point of view. The core principle of this method is to flip black and white pixels by considering, for each candidate pixel, the distance between it and the closest inverse region. The degradation method is validated using a statistical methodology. Its flexibility in the choice of the parameters requires some adaptations. Indeed, a large set of degradations can be obtained. The method itself accepts no less than 6 parameters, allowing to tune the strength of white and black noise, the size of the influence area of these noises, a global noise (which do not depend of the presence of white/black pixels), and a post-processing closing based on well-known morphological operators. Of course, these 6 parameters may generate a large number of combinations, and thus, of models of degradation. So, if the core method used for the degradation tests is formal and validated for its correctness, the determination of the set of parameters used for the contest is more empirical. This framework was applied to the organization of the GREC'03 contest on symbol recognition. In [Valveny 2007], authors attempted to reproduce a set of degradation representing some realistic artifacts (to simulate noise produced when printing, photocopying and scanning). Six levels of degradation (see figure 5.18) were determined by [Valveny 2007]. They took care to

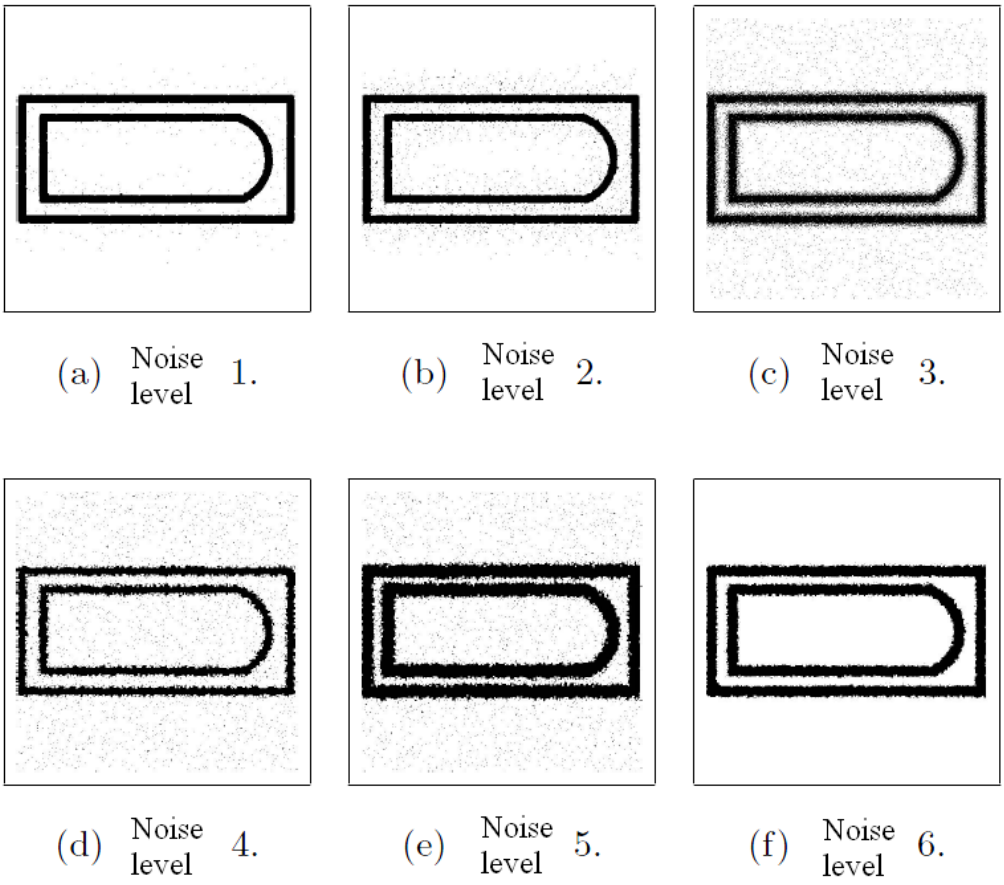


Figure 5.18: Samples of some degraded images generated using the Kanungo method for each level of degradation used.

represent some "standard" noises: local, global troubles.

Binary degradation impacts on polygonal approximations: The higher is the noise level the higher are the distortions on the polygonal approximation. The noise level has a direct influence on the vectorization algorithm. On this experiment, we used a standard data flow process to polygonize the symbols: (i) Cleaning[4]; (ii) Skeletonization; (iii) Polygonal approximation and (iv) Polygonizer. Arbitrary, we adopted the well-knowns di Baja's skeletonizer [di Baja 1996] and the Wall and Danielsson's vectorization [Wall 1984]. Then a polygonizer was performed to transform the set of segments into polygons. Theses steps are summed in figure 5.19. A piece of polygon is zoomed-in to show the perturbation applied on the polygonal approximation when the noise increases. The method only needs a single threshold i.e., the ratio between the algebraic surface and the length of the segments which makes this linear time algorithm fast and efficient. This parameter is set to 60 pixels

[4] A simple 5x5 median filter

for all the experiments. We did not want to assess the impact of the approximation threshold but more likely, the noise impact on the polygonization when the threshold is frozen.

More information concerning that data is detailed in table 5.4.

Base C: Cadastral map collection. In the context of a project called "ALPAGE", a closer look is given to ancient French cadastral maps related to the Parisian urban space during the 19th century. Hence, the map collection is made up of 1100 images issued from the digitalization of Atlas books. On each map a domain-objects called Parcels are drawn by using color to distinguish them. From a computer science point of view, the challenge consists in the extraction of information from color documents in the objective of providing a vector layer to be inserted in a GIS (Geographical Information System).

Automatic polygon detection: In this project, a bottom-up strategy is adopted. In bottom-up strategies, algorithms are performed in a fixed sequence, usually starting "low-level" analysis of the gray level or black and white image, in which primitives are extracted. From this starting point, the four stages for extracting parcels from a cadastral map are put forward. (i) At first, a color gradient is performed to locate objects within the image. (ii) Then, a text/graphic segmentation is run on the gradient image to preserve only graphic elements [Raveaux 2008a], [Raveaux 2008b]. (iii) Thirdly, a digital curve approximation is performed to transform pixels into vectors [Locteau 2006]. (iv) finally, vectors are gathered to form polygons using a polygonizer algorithm [Jr 2003]. This parcel extractor is fully described in chapter 4. An evaluation of our method is proposed thanks to the corresponding ground-truthed maps which were manually vectorized.

Ground-Truthing: With the help of experts in several fields of sciences such as Historians, Archaeologists and Geographers, a campaign of handmade vectorization was carried out. This work was intensively laboured consuming yet necessary. It was the only way to give us the opportunity to fully evaluate the accuracy of our work. The main goal was to build a reference database to investigate the merit of our parcel retrieval scheme. Manually, 100 raster maps were carefully and precisely vectorized to constitute a reliable collection of 2335 parcels of lands. These units are encoded as polygons according to the definition 19. Thereby, a real link does exist between a parcel and its polygon representation. This labor intensive procedure represents a real reference to measure up the accuracy and the validity of our automatic vectorization [Raveaux 2008b], [Raveaux 2007a]. The content of the database is summarized in table 5.3. In average, there are 25 parcels per map and this represents about 75 line segments per parcel. The ground-truth was manually made according to simple rules. Each parcel had to be described by a polygon. The median line was favored in the line tracking. The precision question was solved by imposing to fit at best the parcel contour and consequently, in each polygon, a vertex corresponds to a significant direction change. For each image of document,

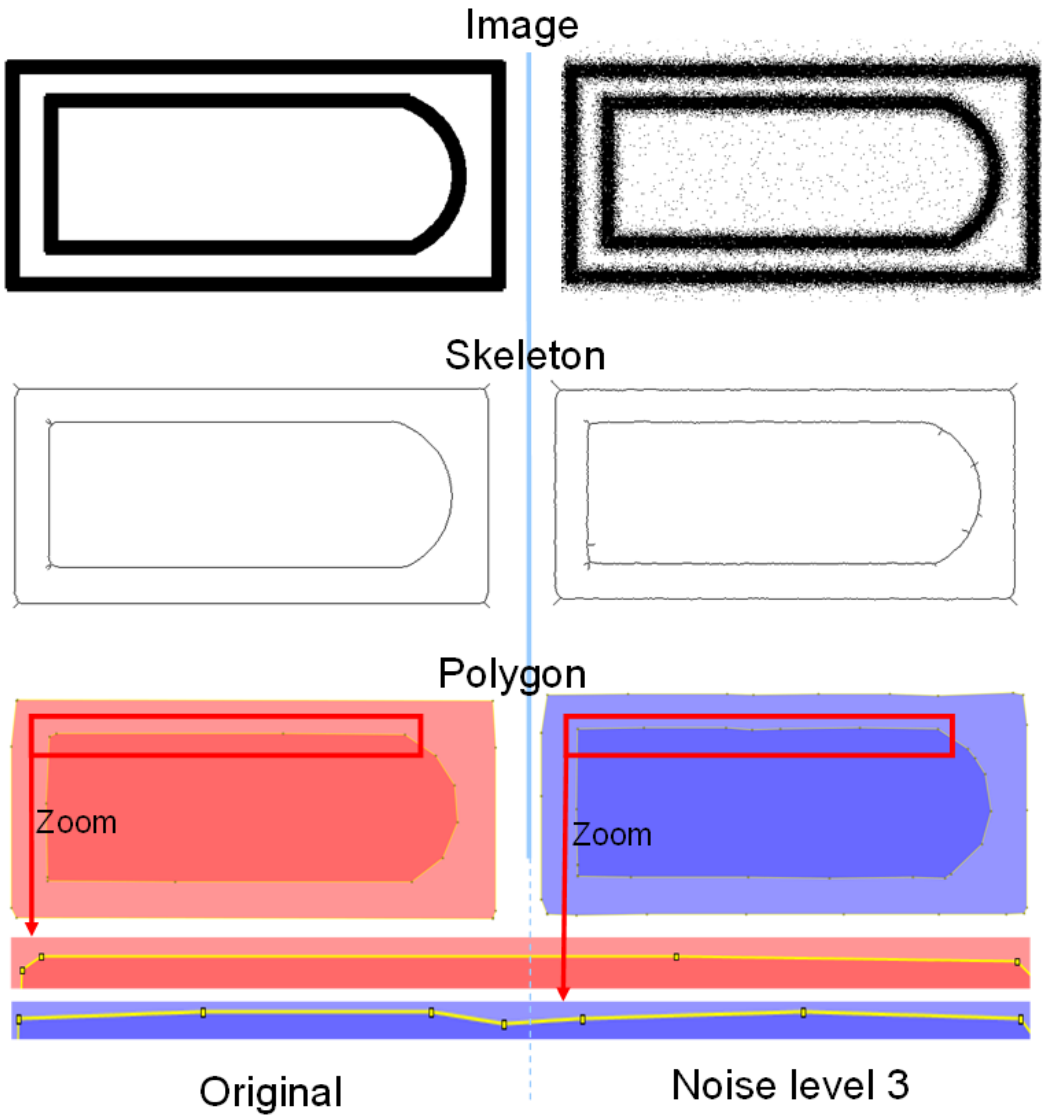


Figure 5.19: Example of the polygonal approximation when increasing the noise level.

there exists exactly one pair of vectorized maps, one map called Ground Truth and one map named Computer Generated, respectively $< D_{GT}; D_{CG} >$. An example of a pair of vectorization to be matched is displayed in figure 5.20. Further details on this data set are presented in table 5.3. Note that this database is made up of real data.

A synthesis about this database is reported in table 5.3 while the content is publicly available at

http://alpage-l3i.univ-lr.fr/PE/alpagedb.zip.

5.4.2 Protocol

Three different ways of evaluating our indices are proposed.

Polygon Matching Distance evaluation: To assess the ability of the polygon mapping distance to increase when documents get badly reconstituted, we focus on Base A. Base A is representative of different shape distortions and consequently, polygon shapes are affected by this noise. On Base A, we performed a ranking test using PMD as a dissimilarity measure. A visual explanation of how ranks are obtained is brought to view in figure 5.21. Then, ranks are compared thanks to a statistical method called a Kendall's test and defined as follows:

Definition 30. *Kendall's test*

We assess the correlation concerning the responses to k-NN queries when using PMD as dissimilarity measures. The setting is the following: in a given Base X, we select a number N of symbols, that are used to query by similarity the rest of the dataset. Top k responses to each query obtained using PMD are compared with the ground-truth. The ground-truth ranks are obtained thanks to the control of noise level. The similarity of the PMD ranks and the ground-truth ranks are measured using Kendall correlation coefficient. We consider a null hypothesis of independence(H0) between the two responses and then, we compute, by means of a two-sided statistical hypothesis test, the probability (p-value) of getting a value of the statistic as extreme or more extreme than observed by chance alone, if H0 is true. The Kendall's rank correlation measures the strength of monotonic association between the vectors x and y (x and y may represent ranks or ordered categorical variables). Kendall's rank correlation coefficient τ may be expressed as

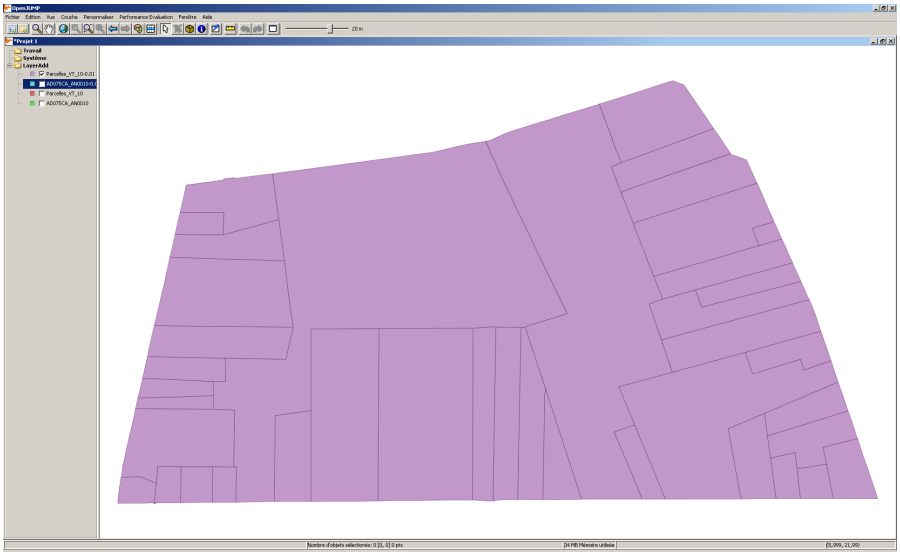
$$\tau = \frac{S}{D}$$

Where,

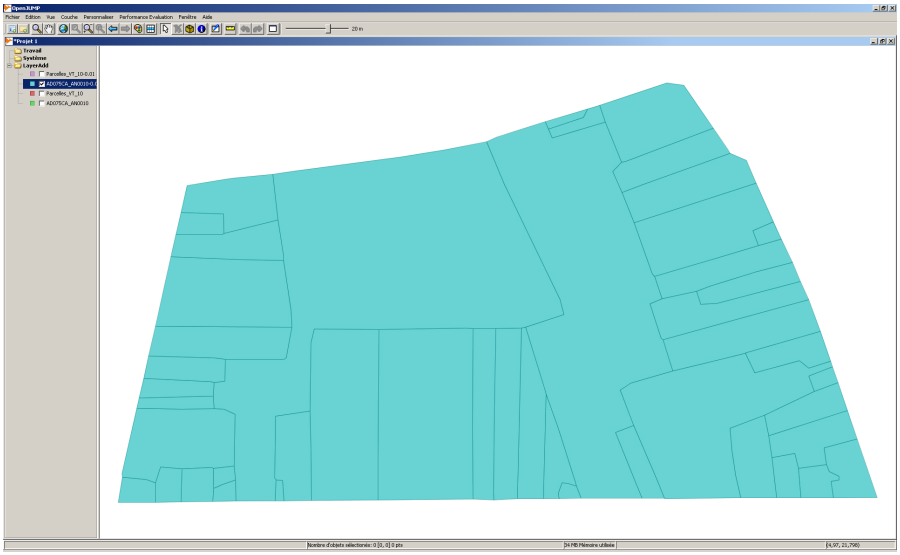
$$S = \sum_{i<j} (sign(x[j] - y[i]).sign(y[i] - x[j])) \tag{5.8}$$

And,

$$D = \frac{k(k-1)}{2} \tag{5.9}$$



(a) GT



(b) CG

Figure 5.20: Two vectorizations to be mapped ($|D_{CG}| = 46$ $|D_{GT}| = 40$).

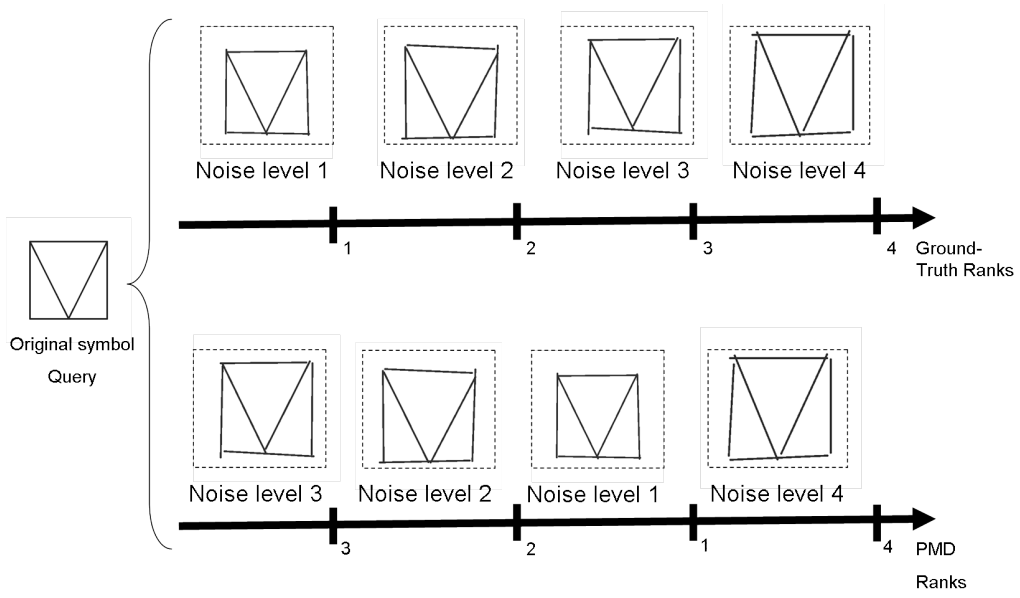


Figure 5.21: Ranking explanation. Ranks 3 and 1 were swapped by PMD

Polygonal approximation sensitivity: In this second experiment, we aim at assessing the capacity of the Matched Edit Distance (MED) to increase when the polygonal approximation gets badly reconstituted by the retrieval systems. Base B is involved in this test. Base B is a binary degraded set of symbols and the higher is the noise level on symbols and the more disturbed is the polygonal approximation from the original one. In this way, we control the distortion level of the digital curve approximation and consequently, we obtain a ground-truth order from an ideal symbol by controlling the noise level, figure 5.19. Finally, the ranks returned by MED and the ground-truth are compared according to the Kendall's test described in Def.30. using a Kendall test.

Application to the evaluation of parcel detection: Our last experiments lie on real data that composed Base C. At first, we performed the PMD distance on a single pair of given maps and this in order to highlight dissimilarities issued from the raster to vector conversion. Then, an experiment is dedicated to the evaluation of the entire collection of cadastral maps. We provide an interpretation of the results through the viewpoints of our set of indices. A statistical framework is described to point out correlation between the different indices. Finally, MMBD is assessed by comparing this unsupervised method with our ground-truth based indices. The basic idea is to determine if both behave the in same way, to reveal if a relation does exist between them.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
τ	0.0000	0.6000	0.8000	0.7029	0.8000	1.0000

Table 5.5: Summary of Kendall correlation (τ). PMD *vs* ground-truth

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
τ	0.3333	0.6190	0.7143	0.7107	0.8095	1.0000

Table 5.6: Summary of Kendall correlation (τ). MED *vs* ground-truth

5.4.3 Polygon Matching Distance evaluation

Using $N = 70$, $k = 4$ equal to the number of noise levels available in Base A, we present in figure 5.22 and table 5.5, the results obtained in terms of τ values. From the 70 tests, only 9 have a p-value greater than 0.05, so we can say that the hypothesis $H0$ of independence can be rejected in 87.4% cases, with a risk of 5%. The observed correlation between the responses to k-NN queries when using the ground-truth and Polygon Matching Distance (PMD) tends to reveal a rank relation between both (median value of $\tau = 0.800$). By stress testing a given system, we aim at demonstrating that our protocol can reveal strengths and weaknesses of a system. The PMD index increases when image degradation increases.

5.4.4 Polygonal approximation sensitivity

Using $N = 53$, $k = 6$ equal to the number of noise levels in Base A, we present in figure 5.23 and table 5.6, the results obtained in terms of τ values. From these results, we reject the null hypothesis of mutual independence between MED and the ground-truth rankings for the students. With a two sided test we are considering the possibility of concordance or discordance (akin to positive or negative correlation). A one sided test would have been restricted to either discordance or concordance, this would be an unusual assumption. In our experiment, we can conclude that there is a statistically significant lack of independence between MED and the ground-truth rankings of the symbols by MED. MED tended to rank symbols with apparently greater noise as more farther to the ideal symbol than those with apparently less noise and vice versa.

5.4.5 Application to the evaluation of parcel detection

	X_{all}	X_{tp}	X_{fp}
PMD	0.1856	0.0552	0.1304
MED	0.5068	0.3764	0.1304
η	1	0.8695	0.1304

Table 5.7: Measures of performance.

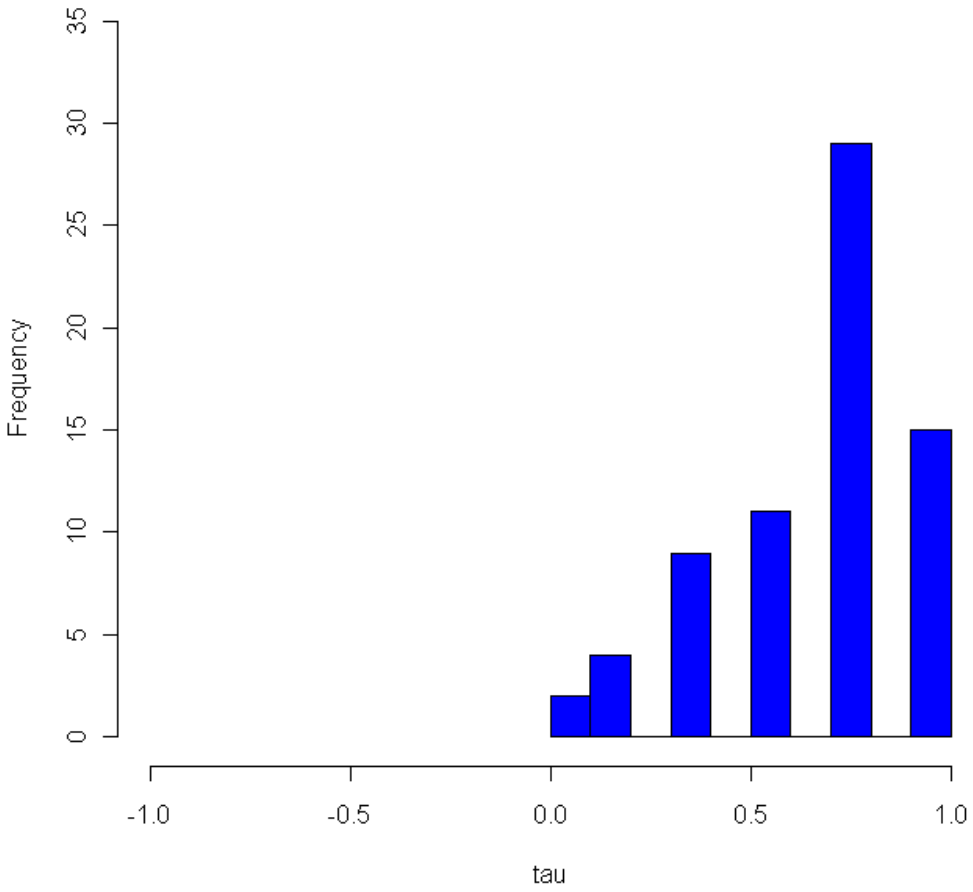


Figure 5.22: Base A: Kendal correlation. Histogram of τ values obtained comparing ground-truth and PMD ranks.

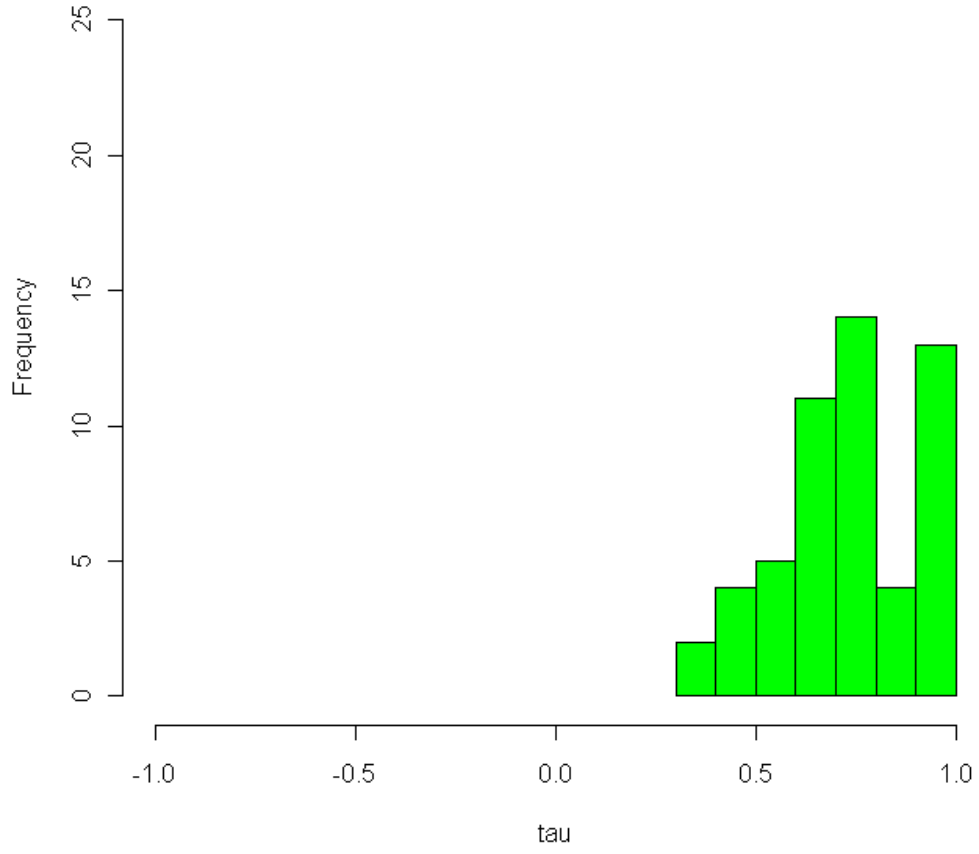


Figure 5.23: Base B: Kendal correlation. Histogram of τ values obtained comparing ground-truth and MED ranks.

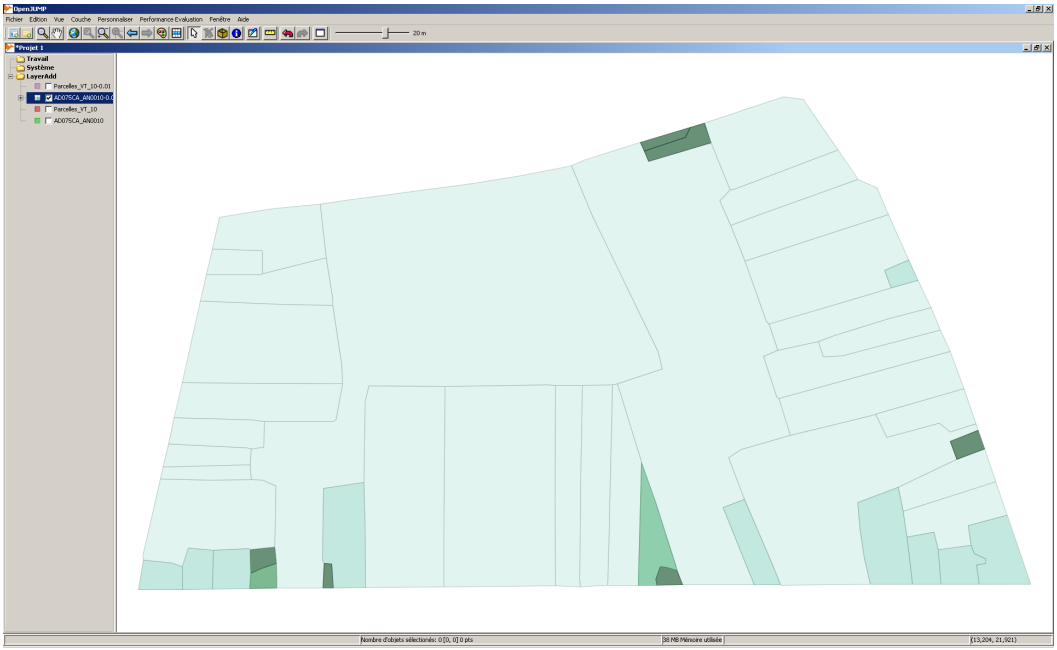


Figure 5.24: Local dissimilarities between the two maps. The lighter the better. It means the darker is a parcel, the worst is its assignment. The overall cost=0.1856 can be decomposed into a miss-detection cost, PMD_{fp}=0.1304 and a true positive cost PMD_{tp}=0.0552

A visual dissimilarity measure of local anomalies: In this part, we focus on comparing maps two by two. This difficult task needs a good observation of the local differences between the compared documents. On a randomly picked pair $< D_{GT}; D_{CG} >$, we computed the Polygon Matching Distance (PMD_{all}). A bi-dimensional representation of the costs to assign each element from the D_{CG} to the D_{GT} is displayed in figure 5.24, whereas values of the different measures are reported in table 5.7. Figure 5.24 provides a visual understanding of where the anomalies are located. Firstly, it facilitates the spotting of errors and others aberrations and especially, this framework can help domain experts understanding the limits and advantages of a vectorization software. Figure 5.24 is worth a thousand words, it makes easier the communication and the implementation of mutualized working tools for both Information and Communication Technologies (ICT) - Humanities and Social Sciences (HSS) communities.

To conclude, it can help users to spot where the mistakes are and so saving a lot of times (time saver). It can help software designers to locate easily where the R2V conversion failed and consequently, this local visualization at a polygon level facilitates the categorization of detection errors.

Evaluation of a collection of maps: From the data set of vectorized maps, we attempted to evaluate the quality of the overall conversion process through the viewpoints offered by the two main criteria that we have described, PMD, MED.

Over the map collection, we observed in figure 5.25 an over-detection tendency. In average, 31% of the retrieved polygons are misleading. 71% of these wrong polygons are implied by an over-detection behavior ($\overline{\eta_{fp}} = 0.22$).

Now, we want to figure out the nature of the mistakes, if these over-detected polygons are just some tiny polygons due to noise into the raster or if they represent a major information altered during the process of conversion. In this objective, we pay attention to the figure 5.26. The figure 5.26 shows that only 36% of the overall cost PMD_{all} is due to the well-detected polygons, hence, most of the information is accurately retrieved from the rasters and the retrieved polygons do fit precisely the Ground-Truth. At the opposite 64% of the mistakes are implied by the wrongly detected polygons. Figure 5.26 strengthens the idea that anomalies are caused by the over-acceptation policy of the automatic application.

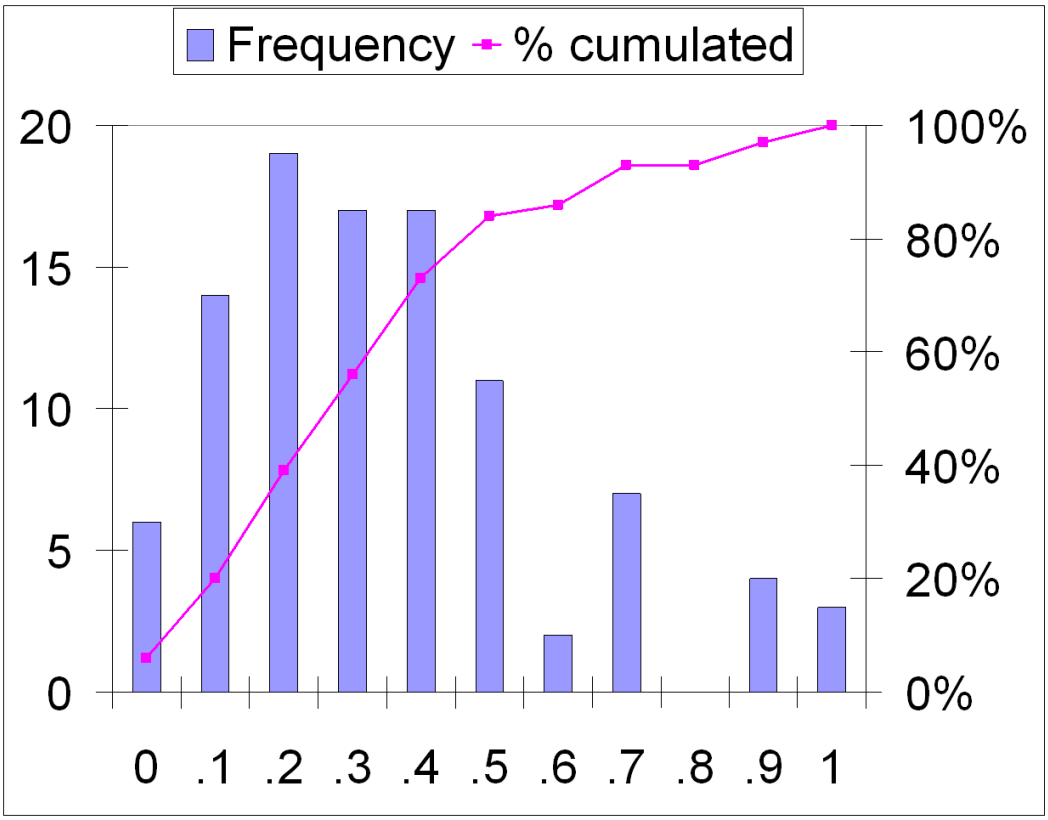
In another step, we aim at assessing how much labor work has to be made to correct the automatically vectorized polygons. A fact observed from the figure 5.27 is that 54% of the MED_{all} mistakes are engendered by the operations to be made when correcting the polygons MED_{tp}. A non-negligible part of the errors are caused by the corrections to be made to fit in the ground truth. An explanation could be a fragmentation phenomenon; many noisy strokes are broken into small pieces during the polygonal approximation process.

The rest of the errors, what is to say the MED_{md} values, is mainly due to an intensive use of the deletion operator in order to remove the over-detected polygons.

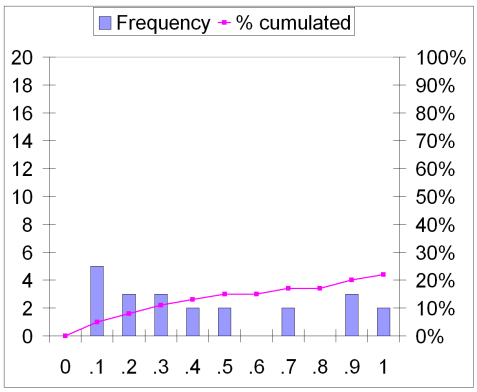
Finally, based on a common work with the historians Helene Noizet and Laurent Costa[5], an algorithm with a combined index (PMD_{all} + MED_{all}) of 0.70 or less may be considered good with respect to human vision evaluation. However, more work should employ this protocol on a series of algorithms and degraded drawings to obtain an objective assessment on commonly accepted criteria.

Correlation inter-indices: A correlation matrix is built from the data series of indices (illustrated in figure 5.28(c)), the Pearson correlation is 1 in the case of an increasing linear relationship, -1 in the case of a decreasing linear relationship, and some value between -1 and 1 in all other cases, indicating the degree of linear dependence between the variables. The closer the coefficient is to either -1 or 1, the stronger the correlation between the variables. This matrix aims to compare the different quality measures between them. A matrix is not expressive enough, so, a 256 shades of grey image is generated to express its substantial meaning in a 2D representation, called image of correlations (figure 5.28(b)). In addition, the matrix of scatterplots between the different measures of quality are given (figure 5.28(a)). From these data representations, a straightforward remark deals with the

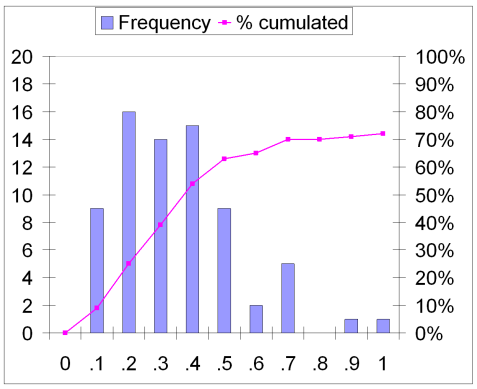
[5]Members of LAboratoire de Médiévistique Occidentale de Paris (LAMOP). UMR 8589 CNRS / UNIVERSITÉ PARIS 1 Panthéon Sorbonne



(a) η_{md}

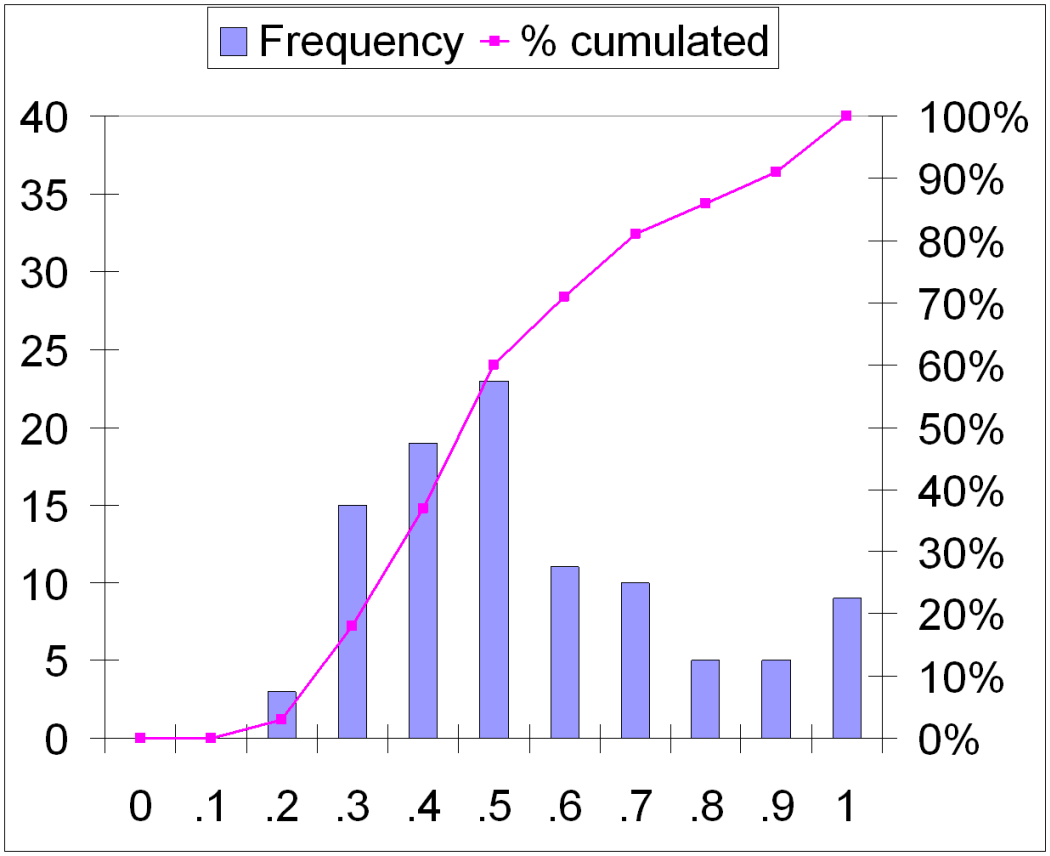


(b) η_{fn}

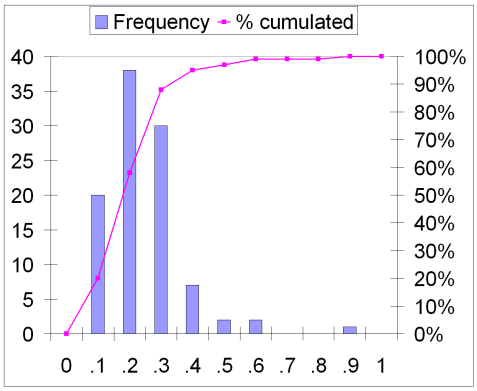


(c) η_{fp}

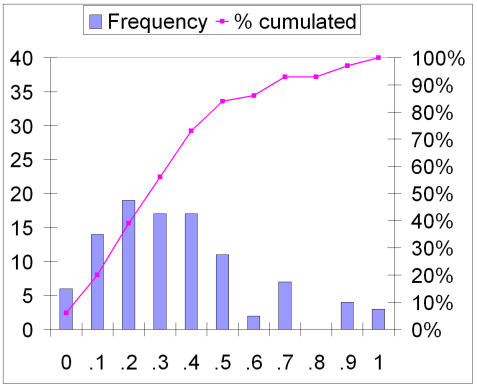
Figure 5.25: Histogram of η. The mean value $\overline{\eta_{md}} = 0.31$ and it can be decomposed in two parts, $\overline{\eta_{fp}} = 0.22$ and $\overline{\eta_{fn}} = 0.09$.



(a) PMD_{all}

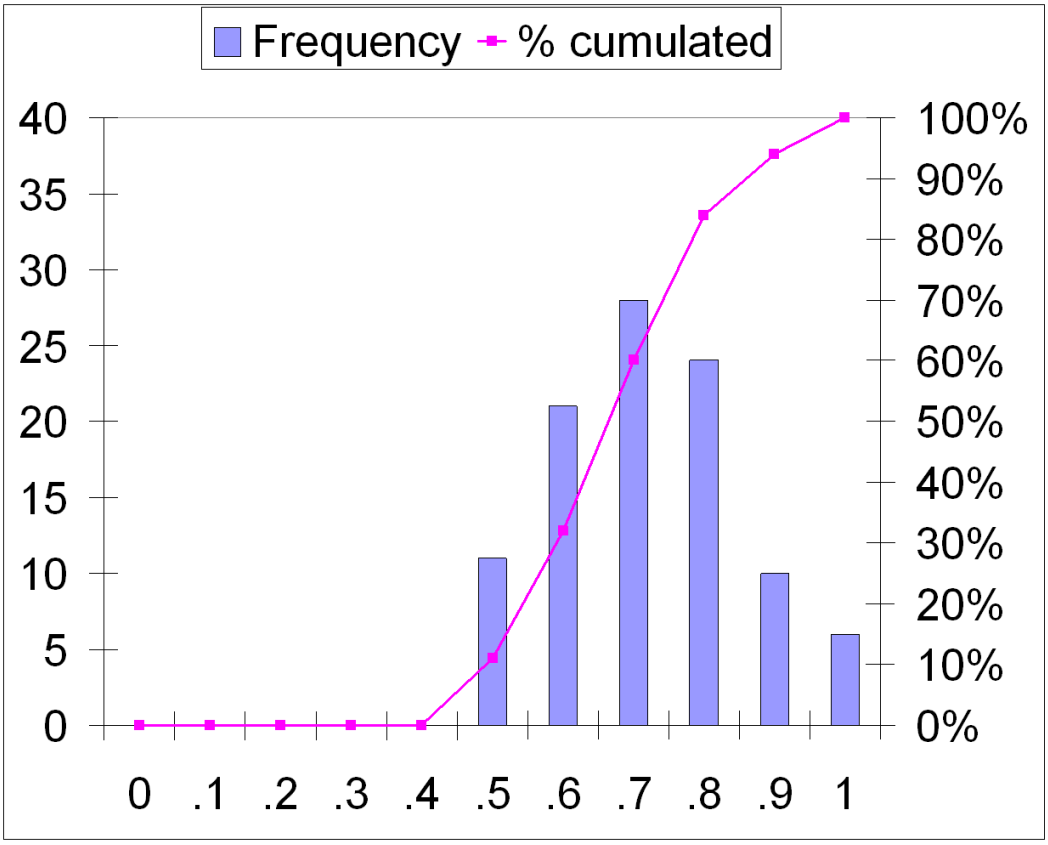


(b) PMD_{tp}

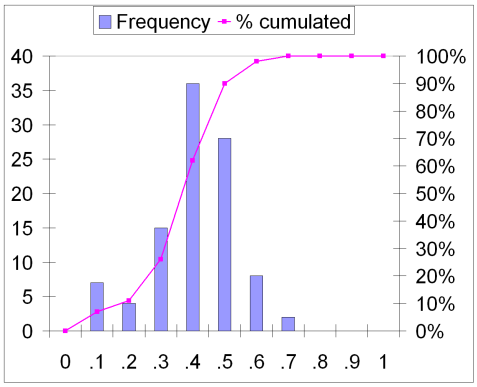


(c) PMD_{md}

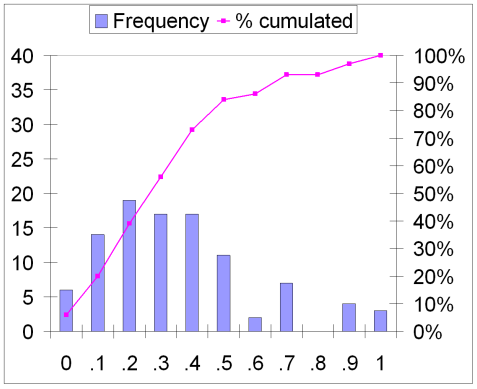
Figure 5.26: Histogram of PMD. The mean value $\overline{PMD_{all}} = 0.5$ and it can be decomposed in two parts, $\overline{PMD_{tp}} = 0.18$ and $\overline{PMD_{md}} = 0.32$.



(a) MED_{all}

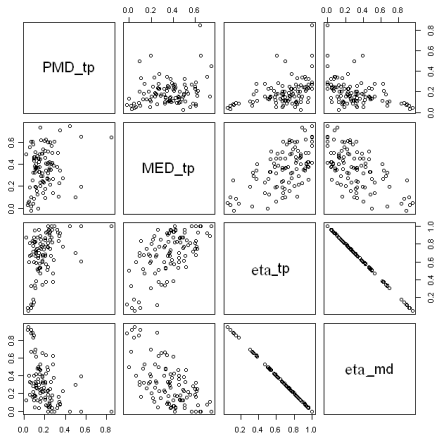


(b) MED_{tp}

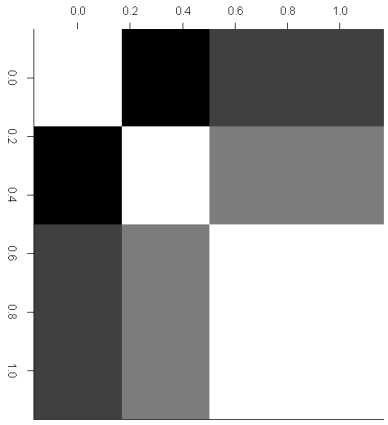


(c) MED_{md}

Figure 5.27: Histogram of MED. The mean value $\overline{MED_{all}} = 0.66$ and it can be decomposed in two parts, $\overline{MED_{tp}} = 0.36$ and $\overline{MED_{md}} = 0.30$.



(a) Scatterplots



(b) Image of the correlation matrix

	PMD_{tp}	MED_{tp}	η_{tp}	η_{md}
PMD_{tp}	1.0000000	0.2411222	0.4311698	-0.4311698
MED_{tp}	0.2411222	1.0000000	0.6088518	-0.6088518
η_{tp}	0.4311698	0.6088518	1.0000000	-1.0000000
η_{md}	-0.4311698	-0.6088518	-1.0000000	1.0000000

(c) Correlation matrix

Figure 5.28: (a) Scatterplots of the proposed indices; (b) Image the correlation matrix inter-indices, the lighter is the shade of grey, the higher is the correlation coefficient; (c) Correlation matrix.

proportional behavior of the η_{tp} and η_{md}, they are closely coupled and share the same information. On the other hand, there is no evident relation between MED and the η measures, the Pearson correlation coefficient between these two series is not indicative enough, nevertheless, the coefficient is low enough (0.60) to indicate no significant redundancy of information. Finally, a clear tendency appears between PMD and MED, it reveals a low correlation (0.24) between PMD and MED. A situation of independence between the two series can be accepted. These variables really express two different kinds of information. They represent original viewpoints on the underlying problem.

Correlation between knowledge and ground-truth method In this part, meta-model based distance (MMBD) is assessed with a ground-truth measure of performance. In this way, we want to confront a ground-truth based approach with our knowledge inspired method. The underlying question is to figure out if MMBD is relevant, if it behaves like a performance evaluation tool. As mentioned in section 5.4.1, the ground-truth elaboration is time consuming and requires a huge amount of labor work, straightforwardly, there are plenty of rooms for alternatives which could avoid an intense use of ground-truth. This is why unsupervised quality measures are of first interest and so NMBD comes up. What NMBD does and doest

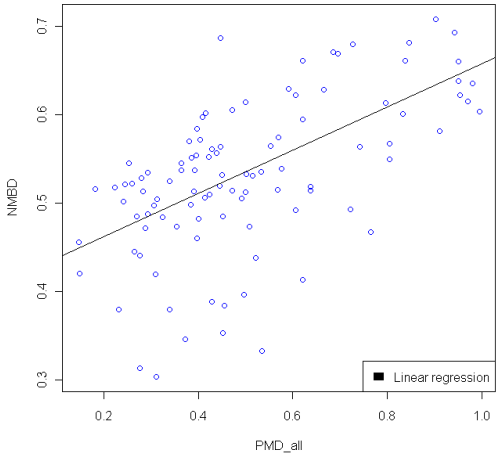
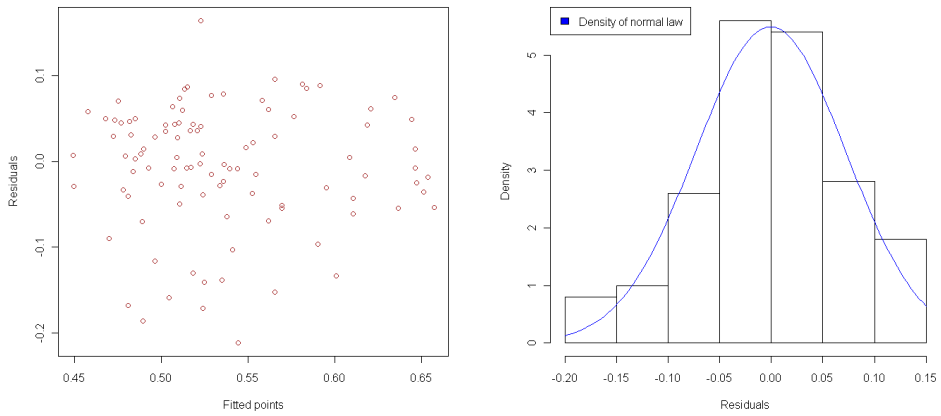


Figure 5.29: Scatter plot between PMD_{all} and NMBD

not represent? It deals with object retrieval accuracy in terms of their external consistency; objects become more reliable as they are correctly laid-out with others through the construction of the superior hierarchical level. The flip-side of the coin is the non-consideration of internal consistencies and consequently, the question of vectorization is not addressed by NMBD. For this last reason, the index which is the most likely to be related to NMBD is PMD_{all}; NMBD is compared with PMD_{all} in figure 5.29. Correlation coefficient (0.3396936) is relatively low and we cannot state any clear tendencies. However, there is still the merit of making an attempt of comparison; our unsupervised evaluation measure behavior is studied and preliminary conclusions are drawn about it. Furthermore, PMD aims at representing the wellness of the parcel alignment while NMBD looks after the organization of the overall objects in the maps (not only the parcels, but also the streets, the frame, the quarters, ...). Obviously, there is a concrete link between both measures, parcels are more likely to be well reconstituted if prior objects are correctly detected in the image processing chain. That is why we expected a clearer link between NMBD and PMD. A linear relationship cannot be validated, elements of variation unexplained by fitted model generate departures indicating an inadequate model. Residuals are estimates of experimental error obtained by subtracting the observed responses from the predicted responses. Figure 5.30 illustrates the distribution of residuals produced by the linear model for PMD *vs* NMBD. We have superimposed a normal density function on the histogram. The overall pattern of the residuals is similar to the bell-shaped pattern observed when plotting a histogram of normally distributed data. Departures from these assumptions mean that the residuals contain structure that is not accounted for in the model.



(a) (b)

Figure 5.30: (a) Fitted points in function of residual errors; (b) Histogram of residuals.

5.5 Conclusion and perspectives

In this chapter, we defined a protocol for performance evaluation of polygon detection algorithms. A discussion between the proposed protocol and the literature is also presented. As a consequence, our protocol is positioned as an extension of prior works, an extension at polygon level. In this way, it is closer to the semantic level and closer to objects handled by humans. Former benchmarks only include synthetic images with image degradation but we completed these artificial samples by real images with manually created ground-truth. Gathering real data to test and compare graphics recognition systems is very time consuming that is why we propose our data set to the community.

Our contribution is two-fold, an object mapping algorithm to roughly locate errors within the drawing, and then a cycle graph matching distance that depicts the accuracy of the polygonal approximation. Both were theoretically defined and adapted to the performance evaluation of polygonized documents. Especially, cost functions were reconsidered, using a set distance for the polygon matching distance (PMD) and defining particular edit costs for the graph matching method.

The proposed protocol is objective and comprehensive, both detection and false alarm rates are considered. By stress testing a given system, we demonstrated that our protocol can reveal strengths and weaknesses of a system. The behavior of our set of indices was analyzed when increasing image degradation.

The results presented in figure 5.26 and 5.27 indicate that the proposed protocol reflects polygon detection and approximation performance accurately. In figure 5.28, the statistical tests demonstrated that the two proposed measures offer different kinds of information.

We have also confronted our measures of quality to a human-based evaluation.

However, more work should be done in this way to obtain an objective assessment on commonly accepted criteria.

The protocol is designed for polygons but may also be extended to other line shapes by completing the graph representation to connected vectors instead of searching for cyclic polygons. In this context, the PMD would not have to be modified at all. The MED which is representative of the manual effort to be made to correct mistakes engendered by a R2V system is envisaged through the graph matching question in terms of basic edit operations (addition, deletion, substitution). The graph formalism confers to the approach a more generic nature and opens the way to future works on more complex objects. This graph-based viewpoint could be the container of a wider range of entities. Instead of focusing on polygon items, a given element could be constituted of all connected segments to form a more complex structure than a polygon while the entire principle would remain unchanged. The graph representation is an open way to a more global paradigm, the object matching question. This could change the scope of our performance evaluation tool to the direction of object spotting.

Introducing the next chapter: Object extractors have been described and then evaluated. The question of browsing and navigating into this data has arisen. Due to the structural and geometrical aspects of the images of maps; the consideration of graph data structure to describe map contents is intuitive and seems reasonable. Graphs are frequently used in various fields of computer sciences since they constitute a universal modeling tool which allows describing structured data. The involved objects and their relations are described in a unique formalism. It is particularly the case in pattern recognition, and moreover in document indexing, since maps can be naturally described using primitives (vectors, connected components, loops...) and geometric relations between these primitives (neighbourhood, connection, parallelism...). In such a case, the map comparison problem turns into a graph classification problem. Its objective is to assign a graph describing a query map to its most similar elements using a reference database. The next chapter will present the concept of graph comparison (subsequently: graph distance and matching) while chapter 7 will discuss the questions of feature extraction and graph classification in the context of Content-Based Map Retrieval where physical and logical structure are compared thanks to the graph distance described in chapter 6.

Chapter 6

A Graph Matching Method and a Graph Matching Distance based on Subgraph Assignments

Contents

6.1 Forewords

During the last decade, the use of graph-based object representation has drastically increased. As a matter of fact, object representation by means of graphs has a number of advantages over feature vectors. As a consequence, methods to compare graphs have become of first interest. In this chapter, a graph matching method and a distance between attributed graphs are defined. Both approaches are based on subgraphs. In this context, subgraphs can be seen as structural features extracted from a given graph, their nature enables them to represent local information of

a root node. Given two graphs G_1,G_2, the univalent mapping can be expressed as the minimum-weight subgraph matching between G_1 and G_2 with respect to a cost function. This metric between subgraphs is directly derived from well-known graph distances. In experiments on four different data sets, the distance induced by our graph matching was applied to measure the accuracy of the graph matching. Finally, we demonstrate a substantial speed-up compared to conventional methods while keeping a relevant precision.

6.2 Introduction

Graphs are frequently used in various fields of computer science since they constitute a universal modeling tool which allows the description of structured data. The handled objects and their relations are described in a single and human-readable formalism. A graph G is a set of vertex (nodes) V connected by edges (links) E. Thus $G = (V, E)$. Tools for graphs supervised classification and graph mining are more and more required in many applications such as pattern recognition [Serrau 2005], case-based reasoning [Champin 2003], chemical components analysis [Ralaivola 2005] and semi-structured data retrieval [Schenker 2004]. To initiate the graph matching topic, we mention that a comprehensive survey of the technical achievements over the last 30 years is provided in [Conte 2004].

In model-based pattern recognition problems, two graphs are given, the model graph G_M and the data graph G_D. The procedure for comparing them involves to check whether they are similar or not. Generally speaking, we can state the graph matching problem as follows: Given two graphs $G_M = (V_M, E_M)$ and $G_D = (V_D, E_D)$, with $\mid V_M \mid = \mid V_D \mid$, the problem is to find a one-to-one mapping $f : V_D \rightarrow V_M$ such that $(u, v) \in E_D$ *iff* $(f(u), f(v)) \in E_M$. When such a mapping f exists, this is called an isomorphism, and G_D is said to be isomorphic to G_M. This type of problem is known as exact graph matching. On the other hand, the term "inexact" applied to graph matching problems means that it is not possible to find an isomorphism between the two graphs. This is the case when the number of vertices or the labels are different in both the model and data graphs. Therefore, in these cases no isomorphism can be expected between both graphs, and the graph matching problem does not consist in searching for the exact way of matching vertices of a graph with vertices of the other, but in finding the best matching between them. This leads to a class of problems known as inexact graph matching. In that case, the matching aims at finding a non-bijective correspondence between a data graph and a model graph [Bunke 1998a], [Tsai 1979], [Messmer 1998]. If one of the graphs involved in the matching is larger than the other, in terms of the number of nodes, then the matching is performed by a subgraph isomorphism. A subgraph isomorphism from G_M to G_D means finding a subgraph sg of G_D such that G_M and sg are isomorphic.

Two drawbacks can be stated for the use of graph matching. Firstly, the computational complexity is an inherent difficulty of the graph-matching problem. A

brute-force approach requires a computational cost of $O(n!)$ for a graph with n nodes. The subgraph isomorphism is proven to be NP-complete [Mehlhorn 1984]. However, a research effort has been made to develop computationally tractable graph-matching algorithms in particular applications [Eshera 1986], [Shapiro 1981]. Such applications use some heuristics to cut down the computational effort to a manageable size. Graph matching can even be computed in polynomial time by using approximate algorithms under particular conditions. The second drawback is dealing with noise and distortion. The encoding of an object of an image by an attributed graph may not be perfect due to noise and errors introduced in low-level stages. In such situations, the presence of noise and distortion results in distorted graphs with different attribute values, missing or added vertices and edges, etc. This fact means exact graph matching is useless in many computer vision applications. The matching must incorporate an error model able to identify the distortions which make one graph a distorted version of the other. A matching between two graphs involving an error model is referred to as inexact graph matching and is computed by an error-correcting or error-tolerant (sub)graph isomorphism [Bunke 1997], [**?**].

Several techniques have been put forward to solve the (sub)graph isomorphism problem, e.g. probabilistic relaxation [Bengoetxea 2002], [Coughlan 2002], [Christmas 1995], EM algorithm [Cross 1998], [Luo 2000], neural networks [Lee 2002], [Lee 2000], decision trees [Messmer 1999] and a genetic algorithm [Cross 1996], [Auwatanamongkol 2007]. Let us now give an overview of the main approaches and report on some of the most representative references. See reference [Lladós 1997] for further study.

Error-Tolerant Algorithms Concerning graph matching in the presence of noise and distortion, the procedural solutions to find an optimal error-tolerant subgraph isomorphism between two graphs are based on the construction of a state-space which is then searched with branch and bound techniques. A different approach to modelize the uncertainty of structural patterns was proposed by Wong and You [A.K.C. Wong 1985]. They defined random graphs as a particular type of graphs which convey a probabilistic description of the data. Seong et al. [Seong 1994] developed a branch-and-bound algorithm to find the optimal isomorphism between two random graphs in terms of an entropy minimization formulation.

Approximate Algorithms Approximate or continuous optimization algorithms for graph matching offer the advantage that they can reach a solution in polynomial time and, moreover, they can solve both the exact and the inexact graph-matching problem. However, since the similarity function which they minimize can converge in a local minimum, they may not find the optimal solution. Perhaps, the most successful of the optimization methods for graph matching use some form of probabilistic relaxation [Christmas 1995], [A.M. Finch R.C. Wilson 1997], [Gold 1996a], [Wilson 1996]. The idea is similar to the discrete relaxation methods; however, the compatibility constraints between vertex-to-vertex assignments do not have a binary

formulation, but are defined in terms of a probability function that is iteratively updated by the relaxation procedure. Another continuous optimization approach is based on neural networks [Kuner 1988], [P.N. Suganthan E.K. Teoh 1995a], [P.N. Suganthan E.K. Teoh 1995b]. The nodes of a neural network can represent vertex-to-vertex mappings and the connection weights between two network nodes represent a measure of the compatibility between the corresponding mappings. The network is programmed in order to minimize an energy (cost) function which is defined in terms of the compatibility between mappings. The problem of neural networks is that the minimization procedure is strongly dependent on the initialization of the network. Genetic algorithms is another technique used to find the best match between two graphs [A.D.J. Cross R.C. Wilson 1997], [Ford 1992], [Jiang 2000]. Vectors of genes are defined to represent mappings from model vertices to input vertices. These solution vectors are combined by genetic operators to find a solution.

Our contribution Now that we have detailed the main concepts, let's introduce our proposal. In this chapter, an error-tolerant graph matching algorithm is described. It is based on subgraph decomposition and wise use of the assignment problem. The assignment problem is one of the fundamental combinatorial optimization problems in the branch of optimization or operations research in mathematics. It consists of finding a maximum weight matching in a weighted bipartite graph.

In its proposed form, the problem is as follows:

- There are V_M number of subgraphs from G_M and V_D number of subgraphs from G_D. Any subgraph (sg_M) from G_M can be assigned to any subgraph (sg_D) of G_D, incurring some cost that may vary depending on the sg_M-sg_D assignment. It is required to map all subgraphs by assigning exactly one sg_M to each sg_D in such a way that the total cost of the assignment is minimized. This matching cost is directly linked to the cost function that measures the similarity between subgraphs.

- The adopted strategy tackles non-deterministic methods (ie. Evolutionary Algorithms) thanks to a combinatorial optimization algorithm which confers a better stability, in such a way that for a given case, every time we run the program we will obtain the same results. Moreover, this combinatorial framework cuts down the algorithmic complexity to an $O(n^3)$ upper bound, depending on the number of nodes in the largest graph. Hence, the matching can be achieved in polynomial time which tackles the computational barrier. On the other hand, the number of calls to the graph distance is highly increased. In fact, n^2 calls to the cost function are needed to complete the weighted bipartite graph. This drawback is reasonably acceptable since the comparisons are performed on rather small subgraphs. Finally, the formulation into a bipartite graph matching offers the possibility to base the cost function on any kind

of graph dissimilarity measures, making the system much more generic where the choice of the graph distance can be seen as a meta parameter.

All the later methods have as a common point the use of an optimization algorithm to best fit a graph into another. Note that in these cases, the fitness function measures the quality of the similarity. This function is designed taking into account the cost of mapping $V_D \rightarrow V_M$.

We believe that a suitable matching would lead to an accurate graph distance. According to this assumption, the performance evaluation question evolves into a graph distance problem. Furthermore, this point of view on the graph matching issue will allow a quantitative benchmark of our approach.

In the next section, a short survey is presented and graph distances used in this chapter are introduced.

The rest of the chapter is organized as follows: in section 6.4, the proposed method is theoretically defined and explained. Section 6.5 is divided into two parts: The experimental evaluation of the algorithm is described and results are examined. Finally, some discussions conclude the chapter.

6.3 Dissimilarity measures between graphs

All of the methods discussed here begin with a crisply labeled set of training data $T = \{< x_i, y_i >\}_{i=1}^{L}$. Our presumption is that T contains at least one item with class label j, $1 \leq j \leq c$. Let x be an unlabeled object that we wish to label as belonging to one of c classes. The standard nearest prototype $1-NN$ classification rule assigns x to the class of the "most similar" element in a set of labeled references. This notion of "the most similar one" is directly linked to the concept of graph distance. Hence, the graph classification problem can be stated as follows: It consists in inducing a mapping $f(x) : \chi \rightarrow C$, from given training examples, $T = \{< x_i, y_i >\}_{i=1}^{L}$, where $x_i \in \chi$ is a labeled graph and $y_i \in C$ is a class label associated with the training data.

Different approaches have been put forward over the last decade to tackle the problem of graph classification. A first one consists into transforming the initial problem in a common statistical pattern recognition problem by describing the objects with vectors in a Euclidean space. In such a context, some features (vertex degree, labels occurrence histograms,. . .) are extracted from the graph. Hence, the graph is projected in a Euclidean space and classical machine learning algorithms can be applied [Papadopoulos 1999]. Such approaches suffer from a main drawback: in order to have a satisfactory description of topological structure and graph content, the number of such features has to be very large and dimensionality issues occur.

Other approaches suggest using embeddings of the graphs in a Euclidean space of a given dimensionality using an optimization process. The aim of which is to best fit the distance matrix between each of the graphs. In such cases, a measure

allowing graph comparison has to be designed. It is the case for multidimensional scaling methods proposed in [Bonabeau 2002] and [Cox 2001].

Another family of approaches also consists in using classical machine learning algorithms. At the opposite of the approaches mentioned above, the graphs are not explicitly but implicitly projected in a Euclidean space, through the use of a similarity measure adapted to the processed data in the learning algorithm.

In such a context, many kernel-based methods such as Support Vector Machine or Kernel Principal Analysis were recently put forward [Kashima 2004], [Borgwardt 2005]. They consist in designing an appropriate graph-based kernel for computing inner products in the graph space. Many kernels have been proposed in the literature [Suard 2006], [Mahé 2004], [Mahé 2005]. In most cases, the graph is embedded in a feature space composed of label sequences through a graph traversal. According to this traversal, the kernel value is then computed by measuring the similarity between label sequences. Even if such approaches have proven to achieve high performance, they suffer from a computationally intensive cost if the dataset is large [Vapnik 1982]. This problem of computational cost is not inherent to kernel-based methods. It also occurs when using other classification algorithms like k-NN. In conclusion, the problem of classifying graphs requires the use of a fast but yet effective graph distance.

Our contribution in this chapter is two-fold; a sub-optimal inexact graph matching and a measure allowing to compare graphs with a low computational cost.

This section offers a study of the different measures used to compare graphs in the context of nearest neighbor search. Then, based on the accuracy and the performance, it justifies the choice of a measure based on subgraph assignments.

A dissimilarity measure is a function :

$$d : X \times X \rightarrow \Re$$

where X is the representation space for the object description. It has the following properties :

- non-negativity

$$d(x, y) \geq 0 \tag{6.1}$$

- uniqueness

$$d(x, y) = 0 \Rightarrow x = y \tag{6.2}$$

- symmetry

$$d(x, y) = d(y, x) \tag{6.3}$$

Measures of dissimilarity can often be transformed into measures of similarity (e.g. s(x, y)= k-d(x,y), with k being a constant). If a dissimilarity measure also respects the triangle inequality (4), it is said to be a metric.

$$d(x, y) \leq d(x, z) + d(z, y) \tag{6.4}$$

Pseudo-metrics are another kind of function which allows to compare objects. Pseudo-metrics respect the non-negativity, symmetry and triangle inequality properties, but do not respect the uniqueness property. Pseudo metrics can be obtained from dissimilarity measures, thanks to transformations that keep the order relation (e.g. $D(x, y) = \frac{d(x,y)}{1+d(x,y)} + 1$ [Gordon 1999]).

The triangle inequality property is often used to optimize similarity search in metric spaces as it is done in [Vidal 1994] or [Ciaccia 1997], with direct application to classification (k-NN) and information retrieval tasks. When the compared objects are graphs, the uniqueness condition turns into an equivalence between a null dissimilarity and graph isomorphism. Graph isomorphism search is known to be a NP-Complete problem. However, if one defines a metric which is computationally tractable, then the graph isomorphism problem is also present.

The Edit Distance (ED) is a dissimilarity measure for graphs that represents the minimum-cost sequence of basic editing operations to transform a graph into another graph by means of insertion, deletion and substitution of nodes or edges. Under certain conditions imposed to the cost associated with basic operations, the edit distance is a metric [Bunke 1998b]. In order to apply edit distance to a real world application, we have to consider that costs for basic operations are application dependent. This issue is tackled by automatic learning of cost functions [Neuhaus 2007]. But, the edit distance computation also has a worst case exponential complexity which prevents its use in the context of nearest neighbor search in large datasets.

Conditions for the edit distance being a metric The original graph to graph correction algorithm defined elementary edit operations, $(a, b) \neq (\varepsilon, \varepsilon)$, where a and b are symbols from the two graphs or the NULL symbol, ε. Thus, changing symbol x to y is denoted (x, y), inserting y is denoted (ε, y), and deleting x is denoted (x, ε). Formally, the edit distance can be expressed as the sum of the edit operations to change a graph G_1 into a subgraph G_2.

$$d_{ED}(G_1, G_2) = \min_{(e_1,..,e_k)\in\gamma(G_1,G_2)} \sum_{i=1}^{k} (edit(e_i))$$

Where $\gamma(G_1, G_2)$ denotes the set of edit paths transforming G_1 into G_2, and *edit* denotes the cost function measuring the strength $edit(e_i)$ of edit operation e_i.

From the conclusion drew in [Myers 2000], an interesting property of this quantity is that it is a metric if $edit(e_i) > 0$ for all nonidentical pairs and 0 otherwise, and if $edit(e_i)$ is selfinverse.

In order to define measures of dissimilarity between complex objects (sets, strings, graphs,...), another possibility is to base the measure on the quantity of shared terms. The simplest similarity measure between two complex objects o_1 and o_2 is the matching coefficient mc, which is based on the number of shared terms.

$$mc = \frac{o_1 \wedge o_2}{o_1 \vee o_2} \tag{6.5}$$

Where $o_1 \wedge o_2$ denotes the intersection of o_1,o_2 and $o_1 \vee o_2$ stands for the union between the two objects.
Based on this idea, dissimilarity measures which take into account the maximal common subgraph (mcs) of two graphs were put forward :

$$d(G_1, G_2) = 1 - \frac{mcs(G_1, G_2)}{max(\mid G_1 \mid, \mid G_2 \mid)} \tag{6.6}$$

Where $\mid G \mid$ denotes a combination of the number of nodes and the number of edges in G. From Eq(5), the expression $o_1 \vee o_2$ is substituted by the size of the largest graph and the intersection of two graphs ($o_1 \wedge o_2$) is represented by the maximum common subgraph.

$$d(G_1, G_2) = 1 - \frac{mcs(G_1, G_2)}{\mid G_1 \mid + \mid G_2 \mid - mcs(G_1, G_2))} \tag{6.7}$$

Where $mcs(G_1, G_2)$ is the largest subgraph common to G_1 and G_2, i.e. it cannot be extended to another common subgraph by the addition of any vertex or edge.

The edit distance (ED) and the size of mcs observe the following equation:

$$ED(G_1, G_2) = \mid G_1 \mid + \mid G_2 \mid - 2 \mid mcs(G_1, G_2) \mid \tag{6.8}$$

As long as the cost functions associated to the edit distance respect the conditions presented in [Bunke 1998b]. The way to calculate the mcs size of two graphs can be used to compute the edit distance and vice-versa. Then, both methods share the same computational complexity. Due to the difficulty in applying these metrics,

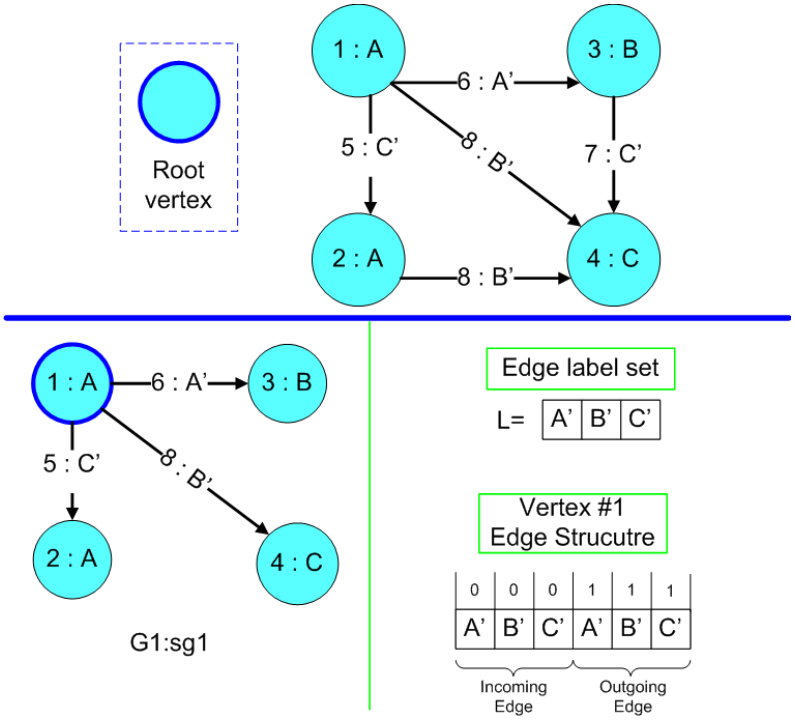


Figure 6.1: Edge Structure of a vertex in the Graph Probing context

several approaches relying on different types of approximations were proposed in [Hidovic 2004]. Three other group of techniques can be employed to evaluate graph similarity, Spectral graph theory [Robles-Kelly 2005], probabilistic methods [Myers 2000] or combinatorial optimization [Gold 1996b], [Kriegel 2003].

Among them, the node/edge matching distance (NMD) proposed in [Kriegel 2003] is a combinatorial optimization problem. It is based on the approximation of the topological conservation of isomorphism by the search of a minimum cost matching between two nodes set. The matrix cost for matching different labeled nodes serves as an input for the Hungarian algorithm. The node matching distance between two graphs G_1 and G_2 results in the cost of the minimum-weight edge matching which is given with a worst case complexity of $O(n^3)$, where n is the largest number of edges. The node cost function has to be determined taking into account a distance label matrix. The node matching distance for attributed graphs respects the non-negativity (1), symmetry (3), triangle inequality (4) properties from the metric definition as it is shown in [Kriegel 2003]. Recently, Shokoufandeh et al in [Shokoufandeh 2006] draws on spectral graph theory to derive a new algorithm for computing node correspondence. In computing a bipartite matching of nodes where their topological contexts are embedded into structural signature vectors.

A faster technique for estimating graph similarity consists in extracting a graph description as a vector of probes. This method, called graph probing proposed by [Lopresti 2003], can deal with graphs with hundreds or thousands of vertices and edges in linear time and can be applied to directed attributed graphs.

Definition 31. *Let G be a directed attributed graph and let L denote a finite set of edge labels: $\{l_1, l_2, ..., l_a\}$. Based on this notation, the edge structure of a given vertex can be described with a numerical vectors composed of a 2a-tuple of non-negative integers $\{x_1, x_2, ..., x_a, y_1, y_2, ..., y_a\}$ such that the vertex has exactly x_i incoming edges labeled l_i , and y_j outgoing edges labeled l_j.*

The figure 6.1 illustrates the principle of construction of an edge structure for a given vertex. In this context, two types of probes are defined:

- $Probe1(G)$: a vector which gathers the counts of vertices sharing the same edge structure, for all encountered edge structures.
- $Probe2(G)$: a vector which gathers the number of vertices for each vertex label.

Based on these probes and on the 1-norm L1, the graph probing distance is defined as :

$$\begin{aligned} GP(G_1, G_2) = L1(Probe1(G_1), Probe1(G_2)) \\ + L1(Probe2(G_1), Probe2(G_2)) \end{aligned}$$

The graph probing distance (GP) only respects the non-negativity, symmetry, and triangle inequality properties from the metric definition, but not the uniqueness property. In other words, GP is a pseudo-metric and two non-isomorphic graphs can have the same graph probes. However, a upper bound relation within a factor of four exists between the graph probing and the edit distance [Bunke 1998b].

$$GP(G_1, G_2) \leq 4.ED(G_1, G_2) \tag{6.9}$$

In this context, the graph topology can be partially ignored by counting the number of occurrences of a set of subgraphs (named fingerprints or probes in different contexts) from each graph and to describe the objects to be compared as vectors. Consequently, this histogram view of a graph cannot lead to an univalent mapping process.

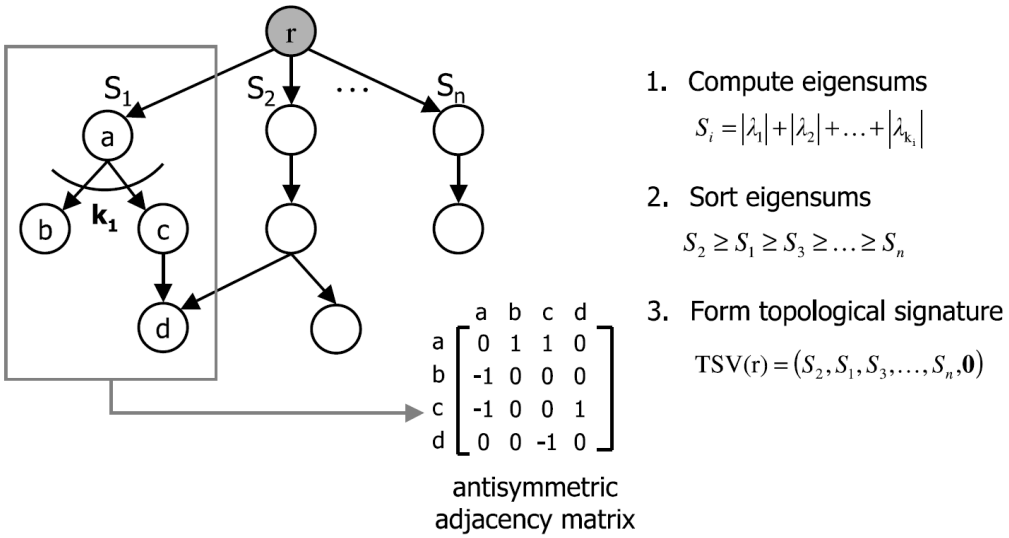


Figure 6.2: Forming the structural signature

Comparison with the related work In [Lopresti 2003], Wilfong and Lopresti proposed a graph decomposition into an histogram where histogram bins are very simple sub-structures coded as numerical vectors. This strong assumption implies sub-elements to be very simple in term of structural information while cutting off drastically the computation time. This histogram viewpoint makes the graph matching computation not feasible losing relationship between items. Instead of an histogram organization, in our case, the information is laid out in a bipartite graph, hence, a point to point mapping can be carried out.
In [Shokoufandeh 2006], a "topological signature vector" described the structural context of a node. This vector was derived from the spectral properties of the directed acyclic subgraph rooted at that node. Thereby, a bipartite graph was defined between the nodes in two graphs, and edge costs were distances between two nodes' corresponding signatures, see figure 6.2. In such a way, the structural information is partially ignored to be embedded into a numerical vector.
On the contrary, we will see that our strong point is the combination of a graph data structure encoding combined with a bipartite matching procedure to find the optimal match. This formal description gives good properties to our method. The subgraph decomposition makes different graph distances applicable, thus, a wise use of the past-work in this field of science can be done.

By now, from the original idea stated in [Kriegel 2003] and [Shokoufandeh 2006], the minimum cost matching between two element sets, the authors extended this paradigm to more complex and discriminating objects called subgraphs. Where a subgraph takes into account the vertex information and its neighborhood context.

The rest of the chapter will present a new metric that involves an univalent subgraph mapping that involves adjacent vertices into the matching process.

6.4 SubGraph Matching and Subgraph Matching Distance (SGMD)

6.4.1 Definition and Notation

6.4.1.1 Subgraph decomposition

From this definition of a given graph, the subparts for the matching problem can be expressed as follows:

Let G be an attributed graph with edges labeled from the finite set $\{l_1, l_2, ..., l_a\}$. Let SG be a set of subgraphs extracted from G. There is a subgraph sg associated to each vertex of the graph G. A subgraph (sg) is defined as a structure gathering the edges and their corresponding ending vertices from a root vertex. In such a way, the neighborhood information of a given vertex is taken into account. A subgraph represents a local information, a "star" structure from a root node. The mapping of these subparts should lead to a meaningful graph matching approximation. The subgraph extraction is done by parsing the graph which is achievable in linear time through the joint use of the adjacency matrix. The subgraph decomposition is illustrated in figure 6.3.

6.4.2 Subgraph Matching

Let $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ be two attributed graphs. Without loss of generality, we assume that $|SG_1| \geq |SG_2|$. The complete bipartite graph $G_{em}(V_{em} = SG_1 \cup SG_2 \cup \triangle, SG_1 \times (SG_2 \cup \triangle))$, where $\triangle$ represents an empty dummy subgraph, is called the subgraph matching graph of G_1 and G_2. A subgraph matching between G_1 and G_2 is defined as a maximal matching in G_{em}. We define the matching distance between G_1 and G_2, denoted by $SGMD(G_1, G_2)$, as the cost of the minimum-weight subgraph matching between G_1 and G_2 with respect to the cost function c' (i.e section 6.4.3). This optimal subgraph assignment induces an univalent vertex mapping between G_1 and G_2, such as the function $SGMD : SG_1 \times (SG_2 \cup \triangle) \rightarrow \Re_0^+$ minimized the cost of subgraph matching. If the numbers of subgraphs are not equal in both graphs, then empty "dummy" subgraphs are added until equality $\mid G_1 \mid = \mid G_2 \mid$ is reached. The cost to match an empty "dummy" subgraph is equal to the cost of inserting a whole unmapped subgraph ($c'(\emptyset, sg)$). The approximation lies in the fact that the vertex mapping is not executed on the whole structure, but more likely for subparts of it. The node matching is only constrained by the assumption of "close" neighborhood imposed by the subgraph viewpoint of a vertex. Why such a restriction? The mapping of two graphs when considering the entire structure is closely coupled with the maximum common subgraph search which is known to be a NP-Complete dilemma. More likely, this chapter adopts a "Divide

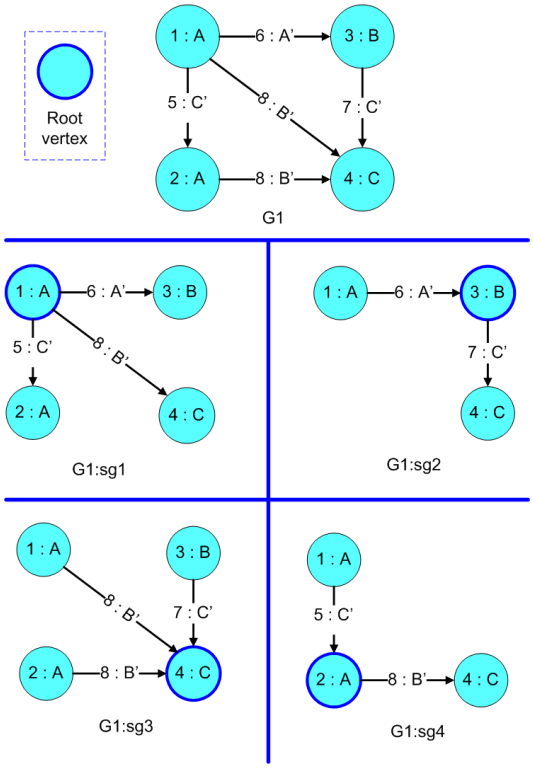


Figure 6.3: Graph decomposition into subgraph world

and Conquer strategy". An example of graph matching is proposed in figure 6.4.

6.4.3 Cost matrix construction

Definition 32. *The Assignment Problem. Let us assume there are two sets A and B together with an $n \times n$ cost matrix C of real numbers given, where $|A| = |B| = n$*

The matrix elements C_{ij} correspond to the costs of assigning the i-th element of A to the j-th element of B. The assignment problem can be stated as finding a permutation $p = p_1, p_2, ...p_n$ of the integers $1, 2, ..., n$ that minimizes $\sum_{i=1}^{n} C_{ij}$

In our approach, the cost matrix contains the distances between every pair of subgraphs from G_1 and G_2. The cost matrix C' is a $n \times n$ matrix where $n = \max(|G_1|, |G_2|) = \min(|G_1|, |G_2|) + |\Delta|$.

$$C' = \begin{vmatrix} c'_{1,1} & \dots & \dots & c'_{1,m} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ c'_{n,1} & \dots & \dots & c'_{n,m} \end{vmatrix}$$

Where $c'_{i,j}$ denotes the cost between two subgraphs. According to our formalism, a subgraph of depth "1" is defined from a root node. Hence, any graph distances

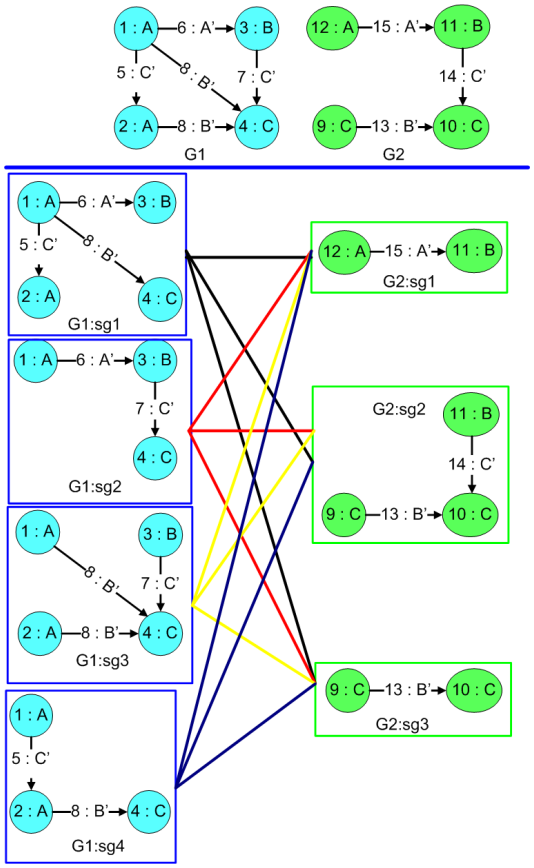


Figure 6.4: Subgraph matching : A bipartite graph

can be applied to build that cost matrix. A straightforward comment, our method does not strictly rely on the edit distance.

With the aim of highlighting this difference of paradigm, a graph distance called Graph Probing [Lopresti 2003] is also evaluated. Therefore $SGMD_{ED}$ and $SGMD_{GP}$ will respectively denote a graph matching based on edit distance or on graph probing.

6.4.4 The subgraph matching distance for attributed graphs is a pseudo metric.

Proof. To show that the Subgraph Matching Distance (SGMD) is a pseudo metric, we have to prove three properties for this similarity measure.

- $SGMD(G_1, G_2) \geq 0$
  The subgraph matching distance between two graphs is the sum of the cost for each subgraph matching. As the cost function is non-negative, any sum of cost values is also non-negative.

- $SGMD(G_1, G_2) = SGMD(G_2, G_1)$
  The minimum-weight maximal matching in a bipartite graph is symmetric, if the edges in the bipartite graph are undirected. This is equivalent to the cost function being symmetric. As the cost function is a metric, the cost for matching two subgraphs is symmetric. Therefore, the subgraph matching distance is symmetric.

- $SGMD(G_1, G_2) \leq SGMD(G_1, G_2) + SGMD(G_2, G_3)$
  As the cost function is a metric, the triangle inequality holds for each triple of subgraphs in G_1, G_2 and G_3 and for those subgraphs that are mapped to an empty subgraph. The subgraph matching distance is the sum of the cost of the matching of individual subgraphs. Therefore, the triangle inequality also holds for the subgraph matching distance.

□

The subgraph matching distance respects the non-negativity, symmetry, and triangle inequality properties from the metric definition, but not the uniqueness property. In other words, $SGMD$ is a pseudo-metric and two non-isomorphic graphs can have the same subgraphs.

6.5 Experiments

This section is devoted to the experimental evaluation of the proposed approach. All tests based on a simple idea; the more significant is the distance induced by a graph matching, the better the matching is. This assumption turns the question

Table 6.1: Characteristics of the four data sets used in our computational experiments

	Base A	Base B	Base C	Base D
Number of classes (N)	50	10	32	15
$\mid Training \mid$	14128	114	9600	5062
$\mid Validation \mid$	14101	56	3200	1688
Average number of nodes	12.03	5.56	8.84	4.7
Average number of edges	9.86	11.71	10.15	3.6
Average degree of nodes	1.63	4.21	1.15	1.3

into a graph distance comparison. Both data sets and the experimental protocol are firstly described before investigating and discussing the merits of the proposed approach. In this practical work, the exact graph Edit Distance was provided by the SUBDUE substructure discovery system [(SUBDUE)], while other methods were re-implemented by us from the literature.

6.5.1 Databases in use

In recent years the use of graph based representation has gained popularity in pattern recognition and machine learning. As a matter of fact, object representation by means of graphs has a number of advantages over feature vectors. Therefore, various algorithms for graph based machine learning have been proposed in the literature. However, in contrast with the emerging interest in graph based representation, a lack of standardized graph data sets for benchmarking can be observed. In order to overcome this difficulty, we chose to carry out our tests on four databases. The first one is composed of synthetic data allowing an evaluation in a general context on a huge dataset. The other sets are domain specific, they are related to pattern recognition subjects where graphs are meaningful. The content of each database is summarized in table 6.1.

6.5.1.1 Synthetic dataset: Base A

This data set contains over 28,000 graphs, uniformly distributed into 50 classes. The graphs are directed with edges and nodes labeled from two distinct alphabets. As the generic framework used to construct random graphs proposed in [ERDOS P. 1959] does not have the aim to depict classes, in the sense of similar graphs, we proposed a two step process to create classes of graphs. In a first step a number N (where N is the desired number of classes) of graphs are constructed using the Erdös-Rényi model [ERDOS P. 1959]. The input of this model is the number of vertices of the graph to be generated, and the probability of having an edge between two nodes. Having a low probability for edges leads to sparse graphs, that occur frequently in proximity-based graph representations found in pattern recognition (see 6.5.1.3).

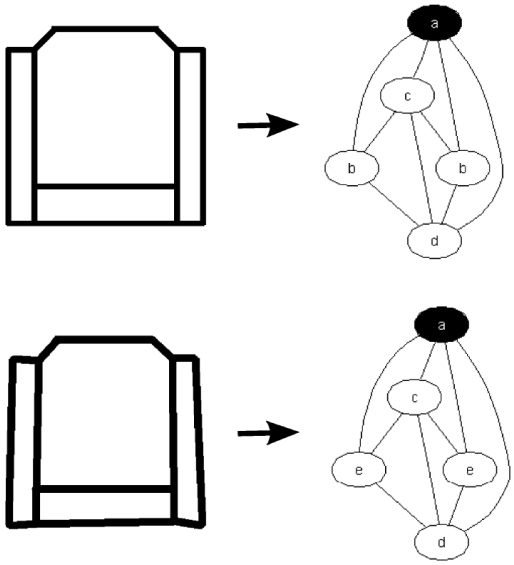


Figure 6.5: From symbols to graphs through connected component analysis

In a second step each of these graphs are modified by edge and vertex deletion or relabeling. A second stage of modifications is applied, by selecting a node from a graph and replacing it with a random subgraph. This process leads to graph classes where intra class similarity is greater than inter class similarity. Numerical details concerning this data set are presented in table 6.1. The large size of this data set is a key point for scaling up our approach.

6.5.1.2 Symbol recognition related data set: Base B

Our data is made of graphs corresponding to a corpus of 170 noisy symbol images, generated from 10 ideal models proposed in a symbol recognition contest [Valveny 2004], (GREC workshop). In a first step, considering the symbol binary image, we extract both black and white connected components. These connected components are automatically labeled with a partitional clustering algorithm [Kaufman 1990], applied on a set of features called Zernike moments [Khotanzad 1990]. Using these labeled items, a graph is built. Each connected component represents an attributed vertex in this graph. Edges are then built using the following rule: two vertices are linked with an undirected and unlabeled edge if one of the nodes is a neighbor of the other node in the corresponding image. An example of the association between two symbol images and the corresponding graphs is illustrated in figure 6.5. Further details on this data set are presented in table 6.1.

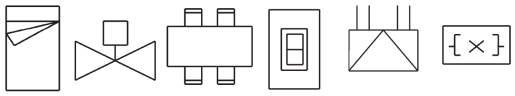


Figure 6.6: Symbol samples

6.5.1.3 Ferrer data set: Base C

In [Ferrer 2006], a structural representation is extracted from a collection of graphical symbols, 12,800 images are distributed among 32 classes. These images of symbols, without rotation and scaling changes are derived from the GREC database [Valveny 2004]. When examining symbol samples in figure 6.6, it is clear that their construction is based on straight-lines. Each segment terminates either with a terminal point or a junction point (the confluence point between two or more segments). For convenience, from now to the end of this work, we will refer to these kinds of points as TP and JP respectively.

In order to prove the robustness of the prototypes against noise, 4 different levels of distortion were introduced. Distortion is generated by moving each TP or JP randomly within a circle of radius r, given as a parameter for each level, centered at original coordinates of the point. If a JP is randomly moved, all the segments connected to it are also moved. With such distortion, gaps in line segments, missing line segments and wrong line segments are not allowed. But the number of nodes of each symbol is not changed. For each class and for each distortion 100 noisy images are created. Thus for each class we have 400 elements (100 for each distortion), straightforwardly, the amount of images is 12,800 (32x400).

In Ferrer′s case, a symbol is represented as an undirected labeled graph, where the TPs and JPs are represented as nodes. Edges correspond to the segments connecting those points. The information associated to nodes or edges are their coordinates (x,y). Due to the graph spectral theory limitation, Ferrer′s graphs are labeled using real positive or null values. Consequently, this restriction leads to the construction of two graphs for a single symbol, a graph G_x labeled with x coordinates and G_y with y coordinates. In our case, the subgraph distances impose the use of nominal labels. A 2-Dimensional mesh aims to achieve the JP and TP discretization (ie. figure 6.7). In addition, an experimental study which is not presented in this chapter has been used in order to choose mesh granularity.

6.5.1.4 Letter database: Base D

The last database used in the experiments consists of graphs representing distorted letter drawings [graph database 2007]. In this experiment we consider the 15 capital letters of the Roman alphabet that consists of straight lines only (A, E, F, ...). For each class, a prototype line drawing is manually constructed. To obtain arbitrarily large sample sets of drawings with arbitrarily strong distortions, distortion operators are applied to the prototype line drawings. This results in randomly shifted,

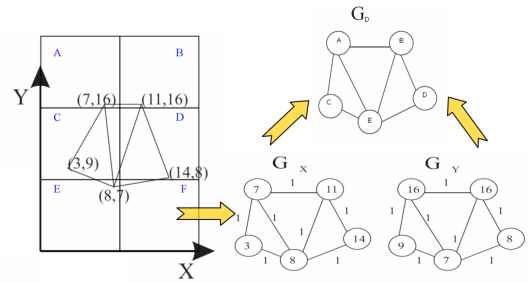


Figure 6.7: From symbols to graphs using a 2D mesh

removed, and added lines. These drawings are then converted into graphs in a simple manner by representing lines by edges and ending points of lines by nodes. Each node is labeled with a two-dimensional attribute giving its position. Since our approach only focuses on nominal attributes, a quantification is performed by the use of a mesh, as in the case of database C and more information concerning that data is detailed in table 6.1.

6.5.2 Protocol

Two ways for assessing our approach are proposed. Firstly, a statistical framework was designed to score the relation between our approach and the edit distance. Secondly, a pattern recognition stage was undertaken to measure-up the behavior in classification.

- In the first experiment, we assess the correlation concerning the responses to k-NN queries when using edit distance(ED) or subgraph matching distance(SGMD) as dissimilarity measures. The setting is the following: in a graph data set(Base D), we select a number M of graphs, that are used to query the rest of the data set by similarity. Top k responses to each query obtained in the first place using edit distance and subgraph matching distance are compared using the Kendall correlation coefficient. We consider a null hypothesis of independence($H0$) between the two responses and then, we compute, by means of a two-sided statistical hypothesis test, the probability (p-value) of getting a value of the statistic as extreme or more extreme than observed by chance alone, if $H0$ is true. The Kendall's rank correlation measures the strength of monotonic association between the vectors x and y containing k elements.(x and y may represent ranks or ordered categorical variables). Kendall's rank correlation coefficient τ may be expressed as

$$\tau = \frac{S}{D}$$

Where,

$$S = \sum_{i<j} (sign(x[j] - x[i]).sign(y[i] - y[j])) \tag{6.10}$$

And,

$$D = \frac{k(k-1)}{2} \tag{6.11}$$

In a second step, the distance matrices($M \times M$) between ED and SGMD are evaluated using the Pearson correlation. Finally, these two steps are repeated to compare the responses to k-NN queries when using edit distance(ED) or node matching distance(NMD) or Graph Probing(GP).

- The classification stage is the last experiment. It consists in a graph classification stage. Let $X = \{x_1, .., x_n\}$ a crispy labeled set of training data. Our presumption is that X contains at least one graph with class label i, $1 < i < c$. Let x be an unlabeled object that we wish to label as belonging to one of c classes. The standard nearest-neighbor $(1 - NN)$ classification rule assigns x to the class of the *most similar* prototype in a set of labeled training data (or reference set). Why use a nearest prototype classifier? Because the graph classification problem is defined in a dissimilarity space, only graph kernel based classifiers and a k-NN classifier can be used to categorize objects in such a space. The 1-NN classifier is pertinent in our context, since it is simple (parameterless) and often, pretty accurate. Hereafter, $E_{np}(X_{tr}; X_{test})$ denotes the test error committed by the 1-NN rule that uses X_{test} when applied to the training data.

6.5.3 Correlation between SGMD and edit distance

Kendall test Using $M = 1200$, $k = 30$, we present in figure 6.8, the results obtained in terms of τ values. As the differences become larger when considering elements farther and farther from the query, the fact of dealing with a huge number of significant neighbors (k) may not be relevant. The notion of order could be simply corrupted by a saturation phenomenon directly implied by very high distances. Hence, from our point of view the examination of the 30 closest neighbors is fair enough. From the 1200 tests, only 124 have a p-value greater than 0.05, so we can say that the hypothesis $H0$ of independence can be rejected in 89.67% cases, with a risk of 5%. The observed correlation between the responses to k-NN queries when using edit distance(ED) and node matching distance(NMD) tends to reveal a rank relation between ED and $SGMD_{ED}$(median value of $\tau = 0.733$). Moreover, the SubGraph Matching Distance overrides the Node Matching Distance in terms of relation with the edit distance while keeping a reasonable time complexity (figure 6.12).

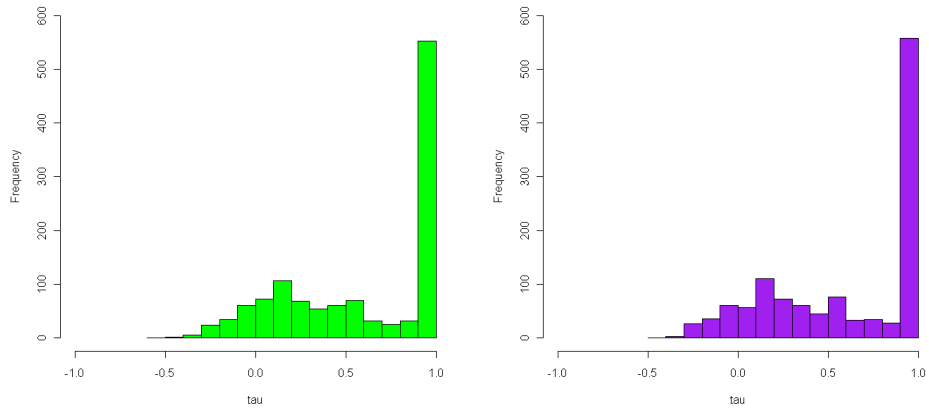
Pearson Correlation on distance matrices (1200×1200) The histograms of Pearson correlations in figure 6.9 lead to the following conclusion; the distance values between ED and SGMD are highly correlated, a linear relation does exist between ED and SGMD(median value of the Pearson correlation = 0.858). This strengthens our decision to use a faster (and simpler) dissimilarity measure than edit distance in order to perform a graph classification.

PCA on correlation matrix A correlation matrix is built from the mean values of the Kendall correlations (illustrated in table 6.2). This matrix aims to compare the different graph distances between them. A matrix is not expressive enough, barely readable, in fact and so, a principal component analysis is performed to express its substantial sense on a 2D plot, called the correlation circle. Each eigenvalue corresponds to a factor, and each factor to a one dimension. A factor is a linear combination of the initial variables, and all the factors are un-correlated (τ=0). The eigenvalues and the corresponding factors are sorted by descending order of how much of the initial variability they represent (converted to %). In our situation stated in figure 6.11, the first two factors allow us to represent 71.40% of the initial variability of the data. This is a good result, and we can be confident with the reliability of the representation of the data. The correlation circle (on axes F1 and F2) shows a projection of the initial variables in the factors space. When variables are close to the circle edge, it means that the variable is well expressed by the two factors and so an interpretation is feasible. If variables are: Close to each other, they are significantly positively correlated (τ close to 1); If they are orthogonal, they are not correlated (τ close to 0); If they are on the opposite side of the center, then they are significantly negatively correlated (τ close to -1). From these explicative guidelines, a first statement leads to conclude that the most highly correlated variables with ED are $SGMD_{ED}$ and $SGMD_{GP}$. Secondly, $SGMD_{ED}$ and $SGMD_{GP}$ are closely coupled. The simple reason being that both distances rely on the same principles to compute the matching, they only stand apart from each other by the cost function involved. Finally, NMD is the farthest from ED; this demonstrates the weakness of this method that does not take into account the edge information.

Pairwise distance scatter plot These scatter plots give us a visual representation of the accuracy of the suboptimal methods on the Letter data. We plot for each pair of graphs its exact (horizontal axis) and approximate (vertical axis) distance value. Based on the scatter plots given in figure 6.10, we express the mean and the standard deviation of the difference between the approximate and exact distances. These measurements are given after a normalization by their maxima respectively, hence the errors are comparable to each other. The residuals from the least squares method are an estimation of the fitness of the linear model between ED and other distances. Another indicator called ISE (Integral Square Errors) denotes the sum of the square errors between the linear model and the data, this estimator brings to light the amount of mistakes provoked by the linear approximation of the data. Lowest values are obtained by SGMD distances when high values for NMD tend to reveal the limits of a linear model for such sparse data. Note that all distances computed by the suboptimal methods($SGMD_{ED}$ and $SGMD_{GP}$) are equal to, or larger than, the exact distances.

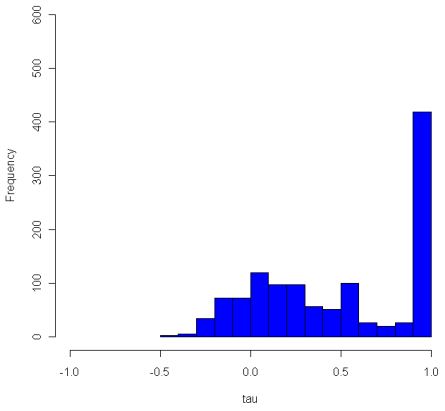
	$SGMD_{ED}$	$SGMD_{GP}$	NMD	GP	ED
$SGMD_{ED}$	1.000	0.608	0.400	0.444	0.604
$SGMD_{GP}$	0.608	1.000	0.435	0.474	0.614
NMD	0.400	0.435	1.000	0.544	0.442
GP	0.444	0.474	0.544	1.000	0.494
ED	0.604	0.614	0.442	0.494	1.000

Table 6.2: Kendall Auto-Correlation Matrix (mean values)

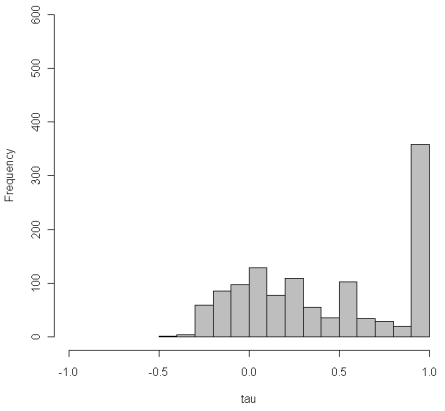


(a) tau $SGMD_{ED}$

(b) tau $SGMD_{GP}$

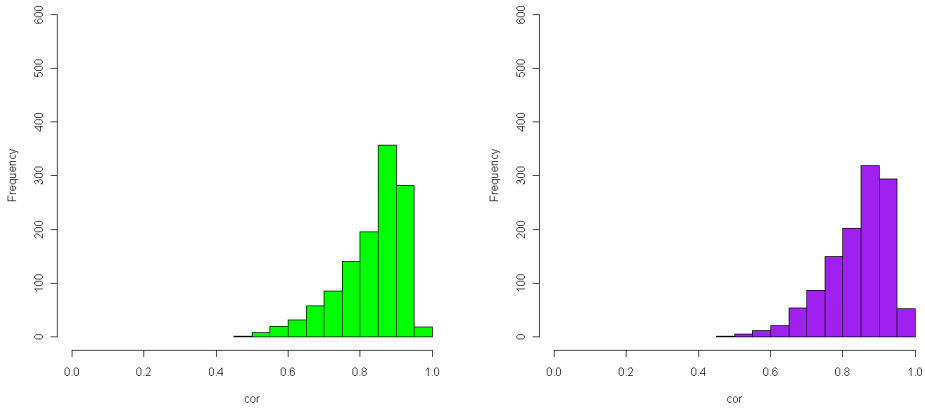


(c) tau GP



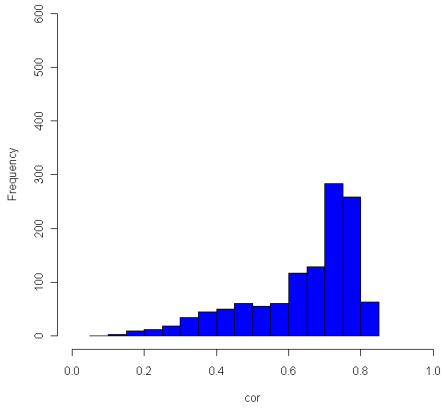
(d) tau NMD

Figure 6.8: Histogram of Kendall correlations, Rank correlation to the responses to the k-NN queries

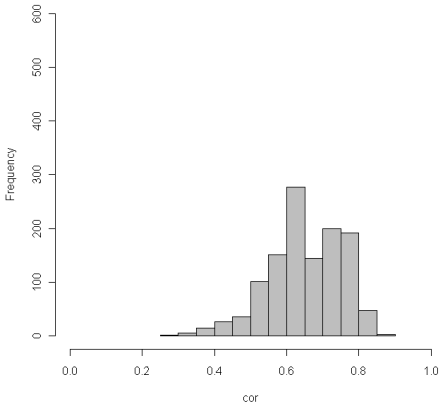


(a) cor $SGMD_{ED}$

(b) cor $SGMD_{GP}$

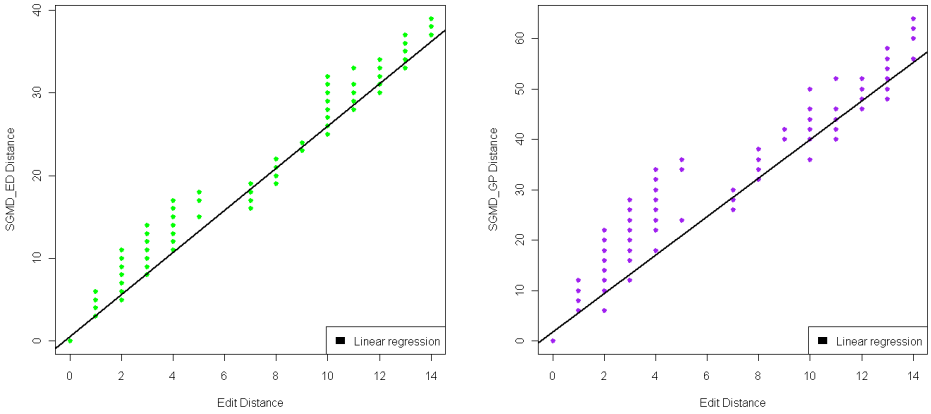


(c) cor GP

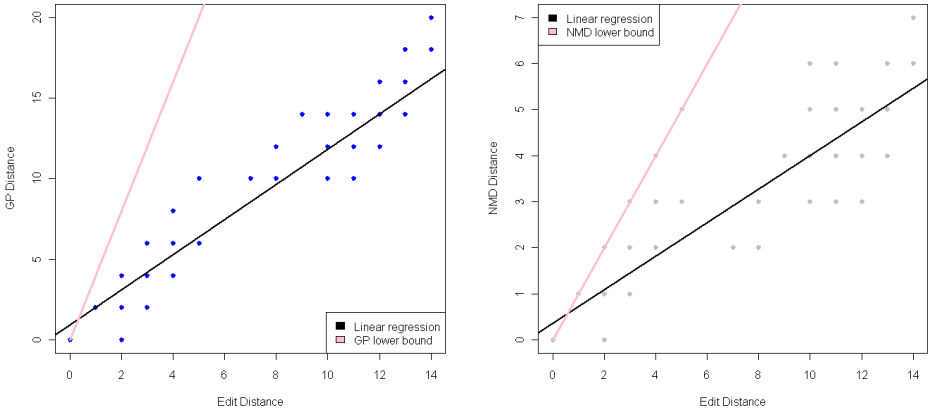


(d) cor NMD

Figure 6.9: Histogram of Pearson correlations, numeric correlations on distance matrices.



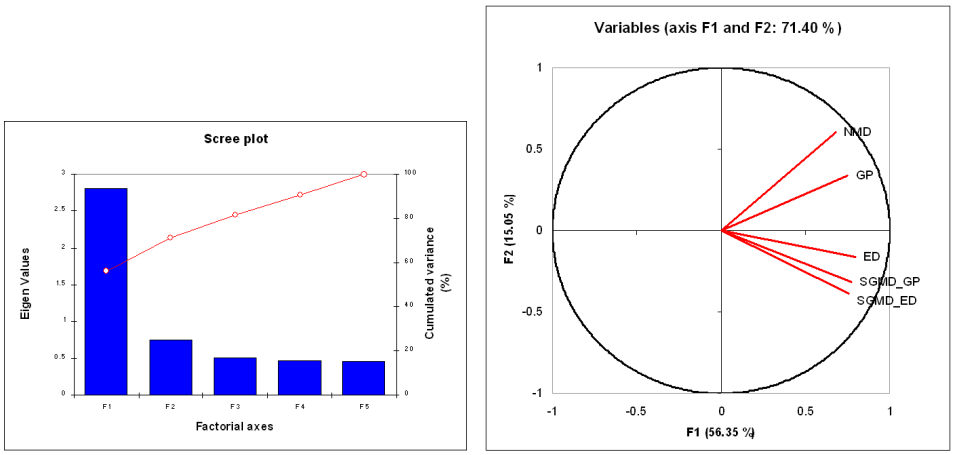
(a) $SGMD_{ED}$ (b) $SGMD_{GP}$



(c) GP (d) NMD

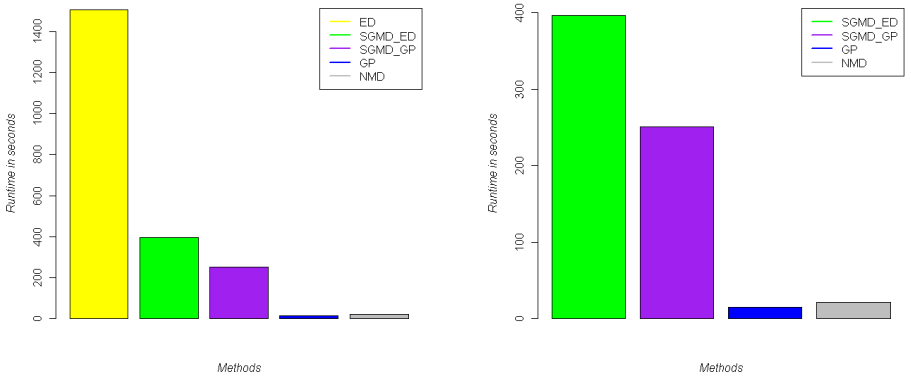
	$SGMD_{ED}$	$SGMD_{GP}$	NMD	GP
μ	0.046	0.054	0.2007	0.1381
σ	0.885	0.857	1.540	1.238
ISE	989	1246	4852	3232

Figure 6.10: Scatter plots of the the suboptimal distances (y-axis) and the exact edit distances (x-axis). The mean and the standard deviation of the difference between the approximate and exact distances are reported in the table above.



(a) PCA - eigen values (b) PCA - correlation circle

Figure 6.11: Correlation matrix representation



(a) All methods (b) Focus on the 4 fastest methods

Figure 6.12: Time complexity

6.5.4 Classification context

Back on track, we keep in mind that the final purpose is to perform a classification stage in order to measure our graph matching precision. Based on the data sets described in section 6.5.1, a 1-NN rule is applied to obtain the number of correctly classified instances (CCI) and the corresponding classification rate. These results are reported in table 6.3. Note that the classification stage using the graph edit distance could only be achieved on two datasets, the datasets B and D which are made up of relatively small graphs (ie. table 6.1). The computational complexity is exponential in the number of nodes of the involved graphs. Consequently, exact graph edit distance is feasible for graphs of rather small size only.

Over the four databases, the SubGraph Matching Distance outperforms the Node Matching Distance. This observation finds a straightforward explanation, it confirms the interests of considering subgraphs which is to say local structured information when NMD only focuses on simple node sets.

Another remark leads us to mention that SGMD provides better results than GP. Two reasons can explain this behavior. Firstly, in GP, probes are incorporated into a histogram comparison. This simplicity imposes the loss of relations between probes of two graphs and therefore, no matching can be expected from GP. On the contrary, SGMD aims to search the best subgraph to subgraph mapping according to a cost function. Hence, in GP, probes are treated independently whereas in SGMD, the mapping of two subgraphs is made considering all possibilities, meaning that the mapping of a given subgraph will impact the rest of the assignment problem. Secondly, we can underline the fact that in GP, the edge structures ($Probe2$) do not gather any node information whereas our approach does. In GP, nodes and edges are processed separately.

On one out of four data sets, the classification accuracy of a nearest-neighbor classifier improves when the exact edit distances are replaced by the suboptimal ones returned by our algorithm. This can be explained by the fact that all distances computed by the suboptimal methods are equal to, or larger than, the exact edit distances. A suboptimal graph distance will not necessarily lead to a deterioration of the classification accuracy of a distance based classifier.

Finally, the edit distance gives a better result on $Base\,D$ than the two others approaches. Objectively, this last remark denotes the loss information due to the approximation introduced by the concept of small subgraph of depth 1. In fact, the proposed algorithm considers only local, rather than global information. However, this little loss of accuracy (3%) should not discourage the use of the SGMD considering the important speed-up it provides while being quite accurate.

6.5.5 Time complexity analysis

The matching distance can be calculated in $O(n^3)$ time in the worst case. To calculate the matching distance between two attributed graphs G_1 and G_2, a minimum-weight subgraph matching between the two graphs has to be determined. This is

Table 6.3: Classification rate according to the graph distance in use

Method	Base A	Base B	Base C	Base D
$ED(\%)$	—	92.86	—	82.10
$SGMD_{ED}(\%)$	88.54	94.64	99.54	80.86
$SGMD_{GP}(\%)$	88.48	94.64	99.21	78.79
$GP(\%)$	57.01	92.86	98.33	59.89
$NMD(\%)$	29.49	89.28	88.75	36.96

equivalent to determining a minimum-weight maximal matching in the subgraph matching of G_1 and G_2. To achieve this, the method of Kuhn [Kuhn 1955] and Munkres [Munkres 1957] can be used. This algorithm, also known as the Hungarian method, has a worst case complexity of $O(n^3)$, where n is the number of subgraphs in the larger one of the two graphs.

A way to compare the computational cost of the different types of distance was to undertake an empirical study. The figure 6.12 depicts a comparison of the runtime execution according to the kind of distances. This test was performed when calculating the distance matrices on 1200 graphs taken from Base D. A first comment aims at illustrating the high time consumption of the edit distance. This over-load discourages the use of this graph measure, even in case of low dimension graphs when its computation is feasible. On the other hand, the experiments demonstrated that the GP is four times faster than $SGMD_{ED}$. Firstly, GP and SGMD do not have the same purpose, GP is fast but do not express any mapping between vertices. Secondly, the time gap is low enough to not reject SGMD as a suitable solution considering the significant accuracy gain it implies. $SGMD$ is a good trade-off between time complexity and performance.

6.6 Conclusion

In the context of graph data classification, the complexity issues linked to graph dissimilarities measures make the process of classification for a large data set an important topic. After experimentally testing the correlation between subgraph matching distance and edit distance, it came up that the subgraph matching distance was of first interest, the best trade-off between accuracy and velocity. Furthermore, we obtain better results in terms of classification accuracy, than conventional graph measures on multi-class graph classification problems. Our contribution gives the proof for the use of a rapid and simple, yet sufficient graph distance which can be processed to scale up a k-NN classification step.

This chapter pointed out the following fact, when the edit distance is not applicable, in the case of high dimension graphs, our approach is an alternative to process an accurate classification. Another strong advantage of our method relies on its deterministic computation.

In addition, graph distances often suffer from their shallow aspect. A black box concept where a single number expresses the link between two graphs. On the other hand, the proposed distance is induced by a graph matching algorithm, this observation implies a precious property. Our global distance is made up of local similarities that can be traced to find out precisely where the defects are. It provides a real explanation of how similar two graphs are.

we can conclude that in general the classification accuracy of the 1-NN classifier is not negatively affected by using the approximate rather than the exact edit distances. Our pseudo metric for graph-based representation will not necessarily lead to a deterioration of the classification accuracy of a distance based classifier.

A future promising work is under investigation. It deals with the graph decomposition into subgraphs of bigger size. In such a way, a closer look will be given to the influence of matching subgraphs of length 1,2,...,$|G|$. Our method makes such possibilities feasible.

Chapter 7

Multiple Representations in a Content Based Image Retrieval Context

Contents

7.1 Forewords

Here, we propose an automatic system to annotate and retrieve images. We assume that regions in an image can be described using a small vocabulary of blobs. Blobs are generated from image features using clustering. Images are structured into a region adjacency graph. This representation is used to perform a similarity search into an image set. Hence, the user can express his need by giving a query image, and thereafter receiving as a result all similar images. Our graph based approach is benchmarked to conventional Bag of Words methods. Results tend to reveal a good behavior in classification of our graph based solution on two publicly available databases. In addition, when facing a warehouse of natural scenes to be queried by examples; Conventional methods would just look at the system level comparing the query images to all the images within the corpus. By system level, we mean the pixel image in its self sufficient way, pixels or a gathering of pixels. When talking about images of documents, the scenario is fairly different because we are dealing with images created by humans and dedicated to humans. This makes a huge difference and allows comparisons and an exploration at higher levels. Two more stages can be drawn : (a) Image of documents can be meaningfully vectorized, and the collection can be addressed thinking at the vector level. (b) Document images have a strong semantic and a navigation using a model representation has come true.

7.2 Introduction

With the development of the Internet, and the availability of image capturing devices such as digital cameras, image scanners, the size of digital image collection is increasing rapidly. Efficient image searching, browsing and retrieval tools are required by users from various domains, including remote sensing, fashion, crime prevention, publishing, medicine, architecture, etc. For this purpose, many general purpose image retrieval systems have been developed. There are two frameworks: text-based and content-based. The text-based approach can be tracked back to 1970s. In such systems, the images are manually annotated by text descriptors, which are then used by a database management system (DBMS) to perform image retrieval. There are two disadvantages with this approach. The first is that a considerable level of human labor is required for manual annotation. The second is the annotation inaccuracy due to the subjectivity of human perception [Eakins 2001] [I.K. Sethi 2001]. To overcome the above disadvantages in text-based retrieval system, content-based image retrieval(CBIR) was introduced in the early 1980s. In CBIR, images are indexed by their visual content, such as color, texture, shapes. A pioneering work was published by Chang in 1984, in which the author presented a picture indexing and abstraction approach for pictorial database retrieval [S.K. Chang 1984]. The pictorial database consists of picture objects and picture relations. To construct picture indexes, abstraction operations are formulated to perform picture object clustering and classification. In the past decade, a few commercial products and experimental prototype

systems have been developed, such as QBIC [C. Faloutsos R. Barber 1994], Photobook [A. Pentland R.W. Picard 1996], Virage [Gupta 1997], VisualSEEK [J.R. Smith 1996], Netra [W.Y. Ma B. Manjunath 1997], SIMPLIcity [J.Z. Wang J. Li 2001]. Comprehensive surveys in CBIR can be found in Refs. [F. Long H.J. Zhang 2003] [Rui 1999].
Image retrieval has been an active research area over the last decade. There are many researches and review articles that mention the importance, requirements and applications of CBIRS [M. De Marsicoi L. Cinque 1997], [Y. Rui T. Huang 1997], [Y. Rui T. Huang 1999] and [H. Muller N. Michoux 2004]. Most researchers provide an extensive description of image archives, various indexing methods and common searching tasks, using different techniques and technologies. Currently CBIR techniques can be classified into two categories: Global approach by using global visual features to describe images and Local approach by considering images as the combination of multiple objects, keypoints or regions.

Global methods

This technique deals with image globally and tries to characterize it by using visual/statistical features calculated from the entire image. Visual features are classified into primitive features such as color or shape, logical features such as identity of objects shown and abstract features such as significance of scenes depicted [H. Muller N. Michoux 2004].

- **Color:** In domain of photograph retrieval, color has been the most effective feature and almost all systems employ colors. Although most of the images are in the red, green, blue (RGB) color space. Color histograms are used to compare images in many applications. Their advantages are efficiency, and insensitivity to small changes in camera viewpoint. However, color histograms lack spatial information, so images with very different appearances can have similar histograms.

- **Texture:** Some of the most common measures for capturing the texture of images are wavelets and Gabor filters. These texture measures try to capture the characteristics of the image or image parts with respect to changes in certain directions and the scale of the changes. This is most useful for regions or images with homogeneous texture. Again, invariances with respect to rotations of the image, shifts or scale changes can be included into the feature space. Other popular texture descriptors contain features derived from co-occurrence matrices, features based on the factors of the Fourier transform and the so-called Wold features [R. Sriram J. M. Francos 1996].

- **Shape Features:** There are many shape representation and description techniques in the literature. Marr and Nishihara [Marr 1978] and Braddy [Brady 1993] have thoroughly discussed representation and sets of criteria for the evaluation of shape. Shape description or representation is an important issue both in object recognition and classification. It has been used in

CBIR in conjunction with color and other features for indexing and retrieval. Many techniques, including chain code, polygonal approximations, curvature, Fourier descriptors and moment descriptors have been proposed and used in various applications [Pratt 2002]. The query images are represented by Fourier descriptors which serve powerful boundary-shape representation tools because of invariance property in affine transformation. Among the well-known shape descriptors, the Zernike moments have been successfully used in many shape contests [Valveny 2004].

Literature on image content indexing is very large, see for example [Datta 2006] for a survey. A common approach to model image data is to extract a vector of features from each image in the database (e.g. a color histogram) and then use the Euclidean distance between those feature vectors as similarity measure for images. But the effectiveness of this approach is highly dependent on the quality of the feature transformation. Often it is necessary to extract many features from the database objects in order to describe them sufficiently, which results in very high-dimensional feature vectors. Those extremely high-dimensional feature vectors cause many problems commonly described by the term 'curse of dimensionality'. Especially for image data, the additional problem arises how to include the structural information contained in an image into the feature vector. As the structure of an image cannot be modeled by a low-dimensional feature vector, the dimensionality problem gets even worse.

To address this topic, several solutions were proposed, involving spatial relationships between entities in images which can be symbolic objects(e.g. objects highlighted after a phase of automatic detection or recognition, localization and labeling) as well as low-level features(e.g. salient points).

Local approaches

The detection and description of local image features can help in object recognition. The Scale Invariant Feature Transform (SIFT) features are local and based on the appearance of the object at particular interest points, and are invariant to image scale and rotation. They are also robust to changes in illumination, noise, and minor changes in viewpoint. In addition to these properties, they are highly distinctive, relatively easy to extract, allow for correct object identification with low probability of mismatch and are easy to match against a (large) database of local features. Object description by set of SIFT features is also robust to partial occlusion; as few as 3 SIFT features from an object are enough to compute its location and pose. Here, we use SIFT (scale invariant feature transform) [Lowe] to lead a comparative study. The full SIFT feature set is a 128 dimensional vector that captures the spatial structure and the local orientation distribution of a region surrounding a keypoint. Recently studies have shown that SIFT is one of the best descriptors for keypoints [et al E.Nowak 2006]. On the other hand, SIFT provides subsamples of the image leading to a high number of regions of interest. This phenomenon is illustrated in figure 7.2. This subsampling effect is not suitable for a meaningful topological

arrangement. A given image is convolved with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images are taken. Keypoints are then taken as maxima/minima of the Difference of Gaussians. This keypoints detection is quite light to execute, however SIFT produces a high number areas which are most of the time involved into a Bag Of Words strategy, in the literature.

Bag Of Words (BoW) The basic idea of Bag of Words is to depict each image as an orderless collection of local features. For compact representation, a visual vocabulary is usually constructed to describe BoW through the clustering of features. With the visual vocabulary, we can describe the image as a feature vector according to the presence or count of each visual word. Under the supervised learning platform, the feature vector forms the basic visual cue for object and scene classification. In a BoW approach, the classification stage turns into a histogram based classification, although the paradigm is simple, it do not contain any geometry information.

Spatial relationships Similarity retrieval by spatial image content is done by using multiple objects and their relationships in space. The main idea of this technique is to consider an image as a group of objects or Regions Of Interest (ROI). Therefore, normally this approach requires segmentation process. Once an image is segmented to many regions, we can use both of their local features and spatial features for retrieval. In retrieval by spatial image content, not only the shape, color and texture properties of individual image regions must be similar, but also they must have the same arrangement (spatial relationships). Among the more known categories of spatial relationships, we can mention the directional [Chang 1998], [Huang 2004], [BERRETTI 2003], topological [Li 1998], geometrical [Guru 2001], and orthogonal [Chang 1986] ones. Ideally, the relationships are described with a graph as the Attributed Relational Graphs (ARGs) or Containment Trees (CT) [Petrakis 2002], [Petrakis 2001].

Our approach

A standard CBIR data flow process relies on three phases. The first one, the extraction of local information aims at finding relevant areas in the image and then to extract features in these regions. All these features are grouped into clusters using a partitional algorithm. Those clustered features forms a visual vocabulary that can be used to express the content of an image. Hence, often an image is transformed into a bag of words, and the comparison of two images turns into a distance between histograms of words. Here in this chapter, another point of view is adopted by trying to take into account the spatial relationship between regions of interest. Therefore, a given image is not longer reduced to a set of words but more likely, a graph based representation is built from the image to enrich the model.

Here, the chapter explores the possibility of adding structural information for image retrieval. The contribution of the chapter is twofold: Firstly, we propose a combination of existing techniques (for segmentation and feature description) in

order to obtain a new image description based on regions (what we call IFFS). On the other hand, we propose two different ways of representing structural relationships between these regions using trees or graphs. This new description is evaluated in two well-known datasets and compared against a reference method, such as the bag-of-words approach. The next section is dedicated to the methodology, then the chapter organization is presented.

7.3 Methodology

An important question is how can one obtain an image vocabulary. In other words, how does one represent every image in the collection using a subset of items from a finite set of items. An intuitive answer to this question is to segment the image into regions, cluster similar regions and then use the regions as a vocabulary. The hope is that this will produce semantic regions and hence a good vocabulary.

7.3.1 Blob extraction

Barnard and Forsyth [Barnard 2001] and Duygulu et al. [Duygulu 2002] used general purpose segmentation algorithms like Blobworld [Carson 1999] and Normalized-cuts [Shi 2000] to extract regions. These algorithms do not always produce good segmentations but are useful for building and testing models. For each segmented region, features such as color, texture, position and shape information are computed. Duygulu et al [Duygulu 2002] used Normalized-cuts to segment images and then extracted 33 features from the images. They ignored regions which were smaller than a threshold size. Given a set of training images, a K-means clustering algorithm is applied to cluster the regions on the basis of these features. These clusters which they call "blobs" compose the vocabulary for the set of images. Each blob is assigned a unique integer to serve as its identifier (analogous to a word's ASCII representation). All images in the training set can now be represented as a set of blobs from this vocabulary. Figure 7.1 shows the segmentation and the clustering process for some training images. The resulting blobs produced by this approach still leave a lot to be desired. However, given the complexity of images, this is a good first start. Given a new test image, it can be segmented into regions and region features can be computed. The blob which is the closest to it in the cluster space is assigned to it. In our approach, an information extraction stage called Invariant Feature From Segmentation (IFFS) is proposed. The partition into regions is based on a recent statistical region merging algorithm while standard Color, Shape and Texture features are extracted from each region to characterize them.

7.3.2 Information Organization

In content-based image retrieval the use of simple features like color, shape or texture is not sufficient. Instead, the ultimate goal is to capture the content of an image via extracting the objects of the image. Usually images contain an inherent structure

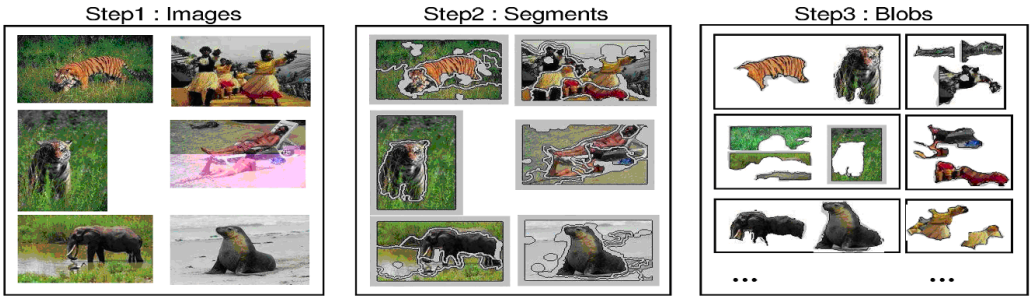
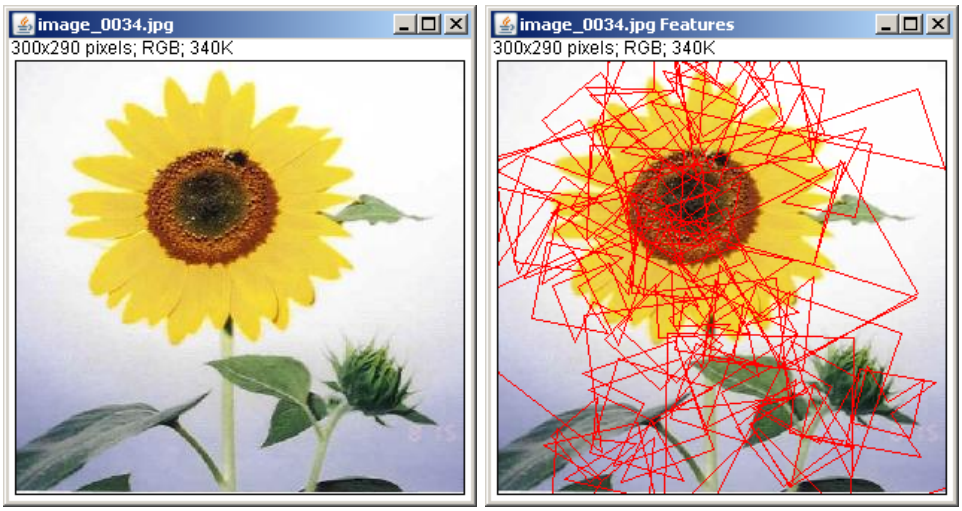


Figure 7.1: Image preprocessing: Step 2 shows the segmentation results from a typical segmentation algorithm (Blobworld) The clusters in step 3 are manually constructed to show the concept of blobs. Both the segmentation and the clustering often produce semantically inconsistent segments (breaking up the tiger) and blobs (seals and elephants in the same blob). This figure was directly taken from [Jeon 2003] since it illustrates well how to obtain blobs.



(a) (b)

Figure 7.2: Regions of Interest found by the SIFT algorithm. Processing SIFT took 407ms, 60 features were identified and processed

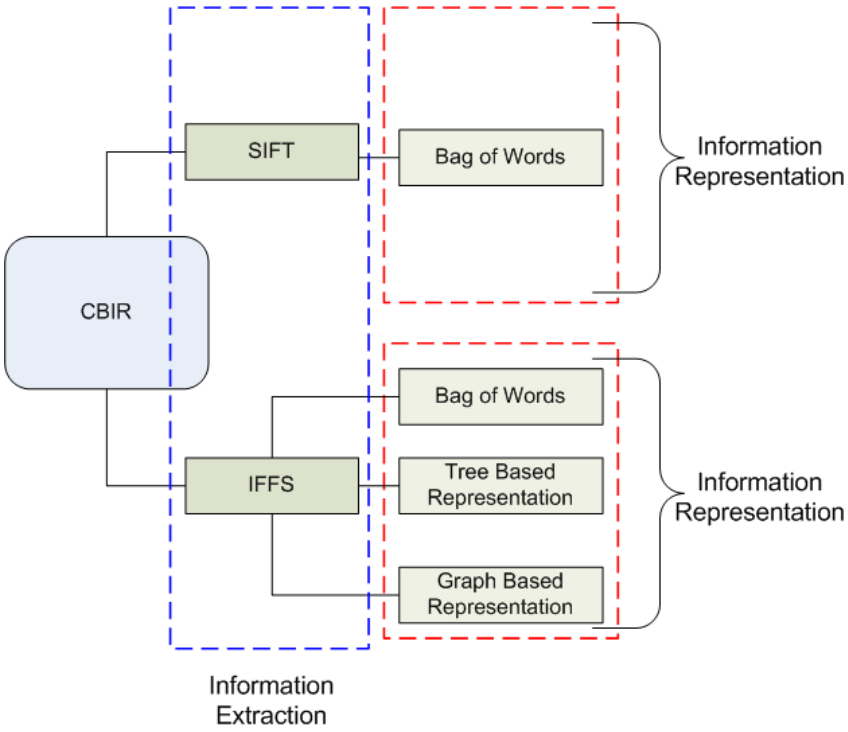


Figure 7.3: CBIR taxonomy

which may be hierarchical. Once features are extracted, there is still the question of how to organize them to perform a classification stage. We describe two models for image representation and similarity measurement, which take content features like color, texture, shape into account. A CBIR decomposition is proposed in figure 7.3.

7.3.2.1 Tree Based Representation (TBR)

One way to model images for content-based retrieval is the use of trees representing the structural and content information of the images. To utilize the inherent structure of images for content-based retrieval, we model them as so called containment trees. Containment trees (CTs) model the hierarchical containment of image regions within others. To extract the containment tree of an image we first segment the image based on the colors of the regions using a region growing algorithm. The resulting segments are attributed with their color and size relative to the complete image. In a second step, the containment hierarchy is extracted from the set of segments by determining which regions are completely contained in other regions.

7.3.2.2 Graph Based Representation (GBR)

Here, an extension of the BoW and the CT methods is proposed. Blobs are structured into an attributed related graph in order to take into account spatial relationships between blobs.

An ARG is a graph where its vertices correspond to regions and edges correspond to relationships between regions of images. Both vertices and edges are labeled by attributes corresponding to properties (features) of objects and relationships respectively.
To retrieve the similarity of images by using ARGs, it is required to perform a distance measure or a graph matching. The graph matching is a complicated process with high complexity. In this chapter, a graph distance that compromises between accuracy and time consumption is presented.

Comparison with some recent works on spatial re-ranking

In [Philbin 2007], the authors investigates two directions for improving visual object-retrieval performance.

- Improving the visual vocabulary.
  Firstly, they improve the clustering method by using an approximate k-means algorithm. In typical k-means, the vast majority of computation time is spent on calculating nearest neighbors between the points and cluster centers. Philbin et Al replace this exact computation by an approximate nearest neighbor method, and use a forest of 8 randomized k-d trees [Amit 1996] [Lepetit 2005] built over the cluster centers at the beginning of each iteration to increase speed. The algorithmic complexity of a single k-means iteration is then reduced from $O(NK)$ to $O(Nlog(K))$, where N is the number of features being clustered and K the number of clusters.

- Incorporating spatial information.
  The output from performing a query is a ranked list of images for a significant section of the corpus. In each image, features have been until now considered as a visual bag-of-words and have ignored the spatial configurations of features. Philbin et al investigates re-ranking the top-ranked results using spatial constraints. The spatial verification procedure estimates a transformation between the query region and each target image, based on how well its feature locations are predicted by the estimated transformation. They then re-rank target images based on the discriminability of the spatially verified visual words.

- Comparison
  To highlight differences between both systems, we take a closer look to the two main divergence points: The clustering methods and the way to consider spatial information. In [Philbin 2007], the authors make a wise use of an approximate k-means algorithm when our approach relies on a conventional flat k-means. A flat k-means clustering is effective but difficult to scale to large vocabularies. Flat k-means can be scaled to similarly large vocabulary sizes by the use of approximate nearest neighbor methods. As Philbin et al demonstrated, this method has a low complexity but far superior performance.

Table 7.1: Summary of the main differences between our approach and J.Philbin's paper

	J.Philbin et al	Our approach
Clustering	Approximate K-means ($O(Nlog(K))$)	flat K-means ($O(NK)$)
Segmentation	Affine-invariant Hessian Regions	Statistical Region Merging
Features	SIFT	IFFS
Spatial Information	Postponed to a re-ranking stage. A spatial verification procedure estimates a transformation between the query region and each target image. Three spatial adjustments are allowed, scale changes, x and y translations and vertical shear.	Taken into account early in the system by the use of a Graph-Based Representation. This representation is invariant to scale, rotation and translation.

Concerning the spatial constraints, in [Philbin 2007] the question is postponed to a re-ranking stage. The query region features are matched to each target image according to the best fit of three affine transformations. These transformations cover situations such as a change in zoom or camera distance to the scene, foreshortening and vertical shear. In our case, the topological question is taken into account early in the system by the use of a Graph-Based Representation. Our representation is invariant to scale, rotation and translation. Therefore, photos can be taken from any views and no strong assumptions are inserted (no strong restrictions on how the photo was taken is imposed). Table 7.1 sums up the main differences between both approaches.

7.3.3 Chapter Organization

The rest of this chapter is organized as follow: In section 7.4, a description of the IFFS descriptor is given. It describes the segmentation algorithm involved and the visual features in use. Another section, 7.5, deals with the blobs layout into structured data, and the question of dissimilarity between topological arrangements is investigated. Section 7.6 introduces the browsing into map collections at three different levels, from a raw pixel level to a structured modeling. Section 7.7 is dedicated to our experimental results, comparing BoW, Tree and Graph based approaches. Finally, a conclusion is given and future works are brought in section 7.8.

7.4 Invariant Feature From Segmentation (IFFS)

7.4.1 Segmentation Algorithm

Segmentation is the process of partitioning an image into disjoint and homogeneous regions. A more formal definition can be given in the following way [Yz] : let I denote an image and let H define a certain homogeneity predicate; the segmentation of I is a partition P of I into a set of N regions $R_1, R_2, ..., R_N$, such that:

- $\bigcup_{n=1}^{N} R_n = I$ with $R_n \cap R_m = \emptyset$, $n \neq m$;
- $H(R_n) = true\ \forall n$;
- $H(R_n \cup R_m) = false\ \forall\ R_n$ and R_m *adjacent*;

Recently, thanks to the increasing speed and decreasing cost of computation, many advanced techniques have been developed for segmentation of color images. In particular we used the Statistical Region Merging [Nock 2004] algorithm that belongs to the family of region growing techniques with statistical test for region fusion. SRM is based on the follow model of image: I is an image with $|I|$ pixels each containing three values (R, G, B) belonging to the set $1, 2, ..., g$. The model considers image I as an observation of perfect unknown scene $I*$ in which pixels are represented by a family of distributions from which each color level is sampled. In particular, every color level of each pixel of I^* is described by a set of Q independent random variables with values in $[0, g/Q]$. In I^* the optimal regions satisfy the following homogeneity properties:

- inside any statistical region and for any color channel, statistical pixels have the same expectation value for this colour channel;
- The expectation value of adjacent regions is different for at least one color channel.

From this model Nielsen and Nock obtain the following merging predicate:

$$P(R, R') = \begin{cases} true & \text{if } \forall\ a \in R, G, B, \mid \overline{R'_a} - \overline{R_a} \mid \leq b(R) + b(R'); \\ false & otherwise. \end{cases} \tag{7.1}$$

$$b(R) = g\sqrt{\frac{1}{2Q \mid R \mid}(ln\frac{\mid R_{|R|} \mid}{\delta})} \tag{7.2}$$

$\overline{R_a}$ denotes the observed average for color a in region R whereas $R_{|l|}$ is the set of regions with l pixels

The order in which the tests of merging were done follows a simple invariant A:

- When any test between two true regions occurs, that means that all tests inside each region have previously occurred.

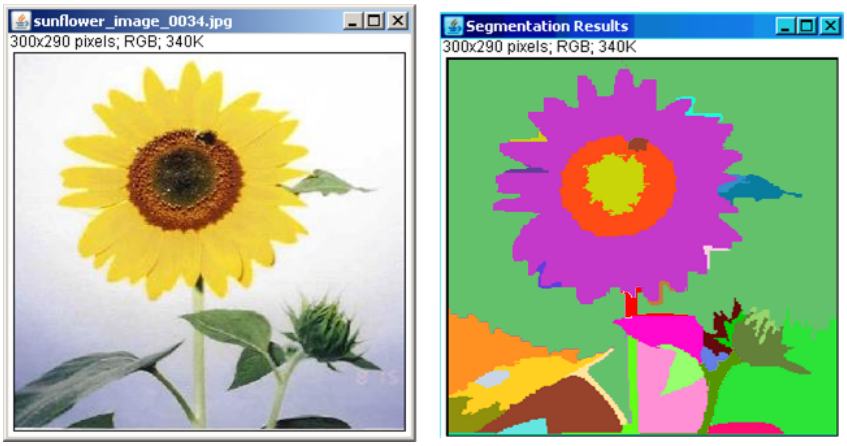


Figure 7.4: A segmentation result. Processing SRM took 1625ms and 26 features were identified and processed

In the experiments, A is approximated by a simple algorithm based on gradient of nearby pixels. In particular Nielsen and Nock consider a function f defined as follow:

$$f(p, p') = \max_{a \in R,G,B} f_a(p, p') \tag{7.3}$$

A simple choice for f_a is:

$$f_a(p, p') = \mid p_a - p'_a \mid \tag{7.4}$$

The set of the pairs of adjacent pixel (S_I) is sorted according to the value of equation 7.3. Afterwards the algorithm takes every couple of pixels (p, p') of S_I and if the regions to which they belong $(R(p)$ and $R(p'))$ were not the same and satisfactory equation 7.1, it merges the two regions. Some image examples segmented by SRM algorithm are shown in figure 7.4.

7.4.2 Features for visual classification

7.4.2.1 Color features: Color Histograms $< H >$

Without loss of generality, we will assume that all images are scaled to contain the same number of pixels M. We discretize the color space of the image such that there are n distinct (discretized) colors. A color histogram H is a vector $< h_1, h_2, ..., h_n >$, in which each bucket h_j contains the number of pixels of color j in the image. Typically images are represented in the RGB color space, and a few of the most significant bits are used from each color channel. For example, Zhang [HongJiang Zhang Atreyi Kankanhalli 1993] uses the 2 most significant bits of each color channel, for a total of $n = 64$ buckets in the histogram. For a given image I, the color histogram H_I is a compact summary of the image. A database of images can be queried to find the most similar image to I, and can return the image I' with the most similar color histogram $H_{I'}$. Typically color histograms are compared

using the sum of squared differences (L2-distance). So the most similar image to I would be the image I' minimizing :

$$\|H_I - H'_I\| = \sum_{i=1}^{n}(H_I[j] - H_{I'}[j])^2 \tag{7.5}$$

7.4.2.2 Texture features: Co-occurrence matrices $< T >$

Statistical methods use second order statistics to model the relationships between pixels within the region by constructing Spatial Gray Level Dependency (SGLD) matrices [J.F. Haddon 1993]. A SGLD matrix is the joint probability occurrence of gray levels i and j for two pixels with a defined spatial relationship in an image. The spatial relationship is defined in terms of distance d and angle θ. If the texture is coarse and distance d is small compared to the size of the texture elements, the pairs of points at distance d should have similar gray levels. Conversely, for a fine texture, if distance d is comparable to the texture size, then the gray levels of points separated by distance d should often be quite different, so that the values in the SGLD matrix should be spread out relatively uniformly. Hence, a good way to analyze texture coarseness would be, for various values of distance d, some measure of scatter of the SGLD matrix around the main diagonal. Similarly, if the texture has some direction, i.e. is coarser in one direction than another, then the degree of spread of the values about the main diagonal in the SGLD matrix should vary with the direction d. Thus texture directionality can be analyzed by comparing spread measures of SGLD matrices constructed at various distances d. From SGLD matrices, a variety of features may be extracted. The original investigation into SGLD features was pioneered by Haralick et al. [R. M. Haralick 1973]. From each matrix, 14 statistical measures are extracted including: angular second moment, contrast, correlation, variance, inverse different moment, sum average, sum variance, sum entropy, difference variance, difference entropy, information measure of correlation I, information measure of correlation II, and maximal correlation coefficient. The measurements average the feature values in all four directions and give us a vector $< T >$ of $4 \times 14 = 56$ components.

7.4.2.3 Shape features: Zernike Moments $< S >$

Let $I(i, j)$ be a discrete image function with spatial dimension $M \times N$, their moments of order n with repetition m are given by :

$$A_{nm} = \frac{n+1}{\Pi}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(I(i,j).R_{nm}(r_{ij}).e^{-i.m\theta_{ij}}) \tag{7.6}$$

Where the discrete polar coordinates :

$$r_{ij} = \sqrt{x_j^2 + y_i^2} \tag{7.7}$$

$$\theta_{ij} = \arctan(\frac{y_i}{x_j}) \tag{7.8}$$

Then the polynomial form can be expressed :

$$f(x,y) = \lim_{N\to\infty} \sum_{n=0}^{N} \sum_{m} (A_{nm}.V_{nm}(x,y)), \tag{7.9}$$

$$V_{mn}(x,y) = R_{mn}(x,y)e^{jm\tan^{-1}(y/x)} \tag{7.10}$$

where the second sum is taken over all $|m| \leq n$, such that $n - |m|$ is even.

T. Taxt in [Due 1996] using equation (7.9), $N = 1...., 13$, indicates that moments of orders up to 8-11 are needed to achieve a reasonable shape classification. According to this result, our shape feature vector $< S >$ is composed the 13^{th} first Zernike moments.

7.4.3 Super Feature Vector

The complete feature vector $< F >$ is made up of the three feature descriptors defined above. This lead us to a vector of dimension 133:

$$\mid F \mid = \mid H \mid + \mid T \mid + \mid S \mid = 64 + 56 + 13 = 133$$

$$< F >= \langle < H >, < T >, < S > \rangle, \mid F \mid = 133.$$

For comparison needs and without loss of performance, each component is normalized between $[0, 1]$ by finding its maximum over a training set.

Now, considering two feature vectors F_1 and F_2, the distance between these two super vectors can be written as the sum of the L2 distances between subvectors contained in $< F >$:

$$d(F_1, F_2) = \frac{1}{\mid H \mid}\|H_1 - H_2\| + \frac{1}{\mid T \mid}\|T_1 - T_2\| + \frac{1}{\mid S \mid}\|S_1 - S_2\|$$

This distance is weighted in such a way that every subvector is considered as equal than any others despite the feature length variation.

7.4.4 Motivation of our choices

Segmentation algorithm About the segmentation algorithm, SRM is a linear-time fast and simple (yet effective) region growing segmentation algorithm based on an adaptive statistical threshold merging predicate on color channels that does not require to maintain dynamically the region adjacency graph (RAG). It runs fast and handles nicely occlusion, noise and user-input bias.

Color features Many color features could be used, however color histograms are frequently used to compare images. Examples of their use in multimedia applications include scene break detection [Arun Hampapur Ramesh Jain 1995], [Farshid Arman Arding Hsu 1993] and querying a database of images [M. G. Brown J. T. Foote 1995], [Ogle 1995]. Their popularity stems from several factors.

- Color histograms are computationally trivial to compute.
- Small changes in camera viewpoint tend not to effect color histograms.
- Different objects often have distinctive color histograms.

Texture features On texture classification contests, the co-occurrence matrix is a popular texture method, which was assessed successfully on the publicly available Meastex database [Meastex], [Smith 1997].

Shape features The first thirteen Zernike invariant moments [Hse 2004] give us global point of view of segmented regions. They provide sufficient information of shapes which are not too specific to shape details. Zernike moments often describe pretty well shapes. Undoubtedly, they remain on top of shape descriptors, they always achieve good results in shape contests [Valveny 2004].

7.5 From Image to Topological Arrangement

This part is dedicated to image representations and complex object measurements. Two models are described a Containment Tree (CT) and a Region Adjacency Graph (RAG). These paradigms are illustrated figure 7.5. When dealing with structured objects the question of dissimilarity measure between objects arise. Here a discussion is brought about the compromise between computational complexity and accuracy.

7.5.1 From Image to structured objects

From Image to Region Adjacency Graph (RAG) Local features are automatically labeled with a partitional clustering algorithm [Kaufman 1990] applied on a set of features. Using these labeled items, a graph is built. Each region represents a vertex in this graph. Then, edges are built using the following rule: two vertices are linked with an undirected and unlabeled edge if one of the node is connected to another in the corresponding image.

Transforming an Image into a Containment Tree In this context, a region R_{in} is said to be contained in a region R_{cont} if for every point $p \in R_{in}$ and every straight line $L \ni p$ there exist two points $o_1, o_2 \in R_{cont}$ with $o_1, o_2 \in L$ and o_1, o_2 are on opposite sides of p.

7.5.2 Measuring the distance between two Containment Trees

To measure the similarity of containment trees, special similarity measures for attributed trees are necessary. A successful similarity measure for attributed trees is the edit distance. Well known from string matching [Levenshtein 1966], [Wagner R.A. 1974], the edit distance is the minimal number of edit operations

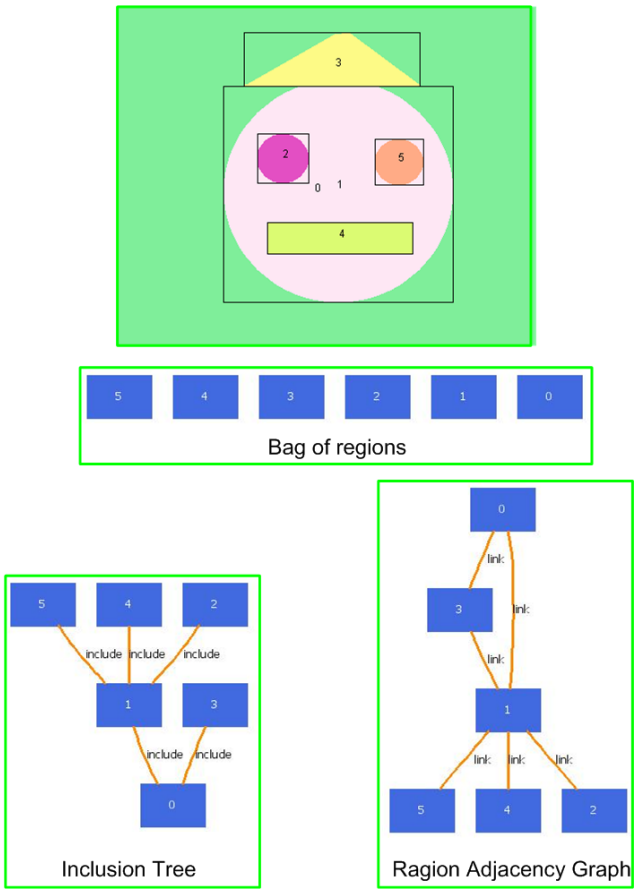


Figure 7.5: Multiple representations

necessary to transform one tree into the other. The basic form allows two edit operations, i.e. the insertion and the deletion of a node. In the case of attributed nodes the change of a node label is introduced as a third basic operation. A great advantage of using the edit distance as a similarity measure is that along with the distance value, a mapping between the nodes in the two trees is provided in terms of the edit sequence. The mapping can be visualized and can serve as an explanation of the similarity distance to the user. However, as the computation of the edit-distance is NP-complete [Zhang K. 1992], constrained edit distance like the Zhang and Shasha edit distance [Zhang 1989] has been introduced. They were successfully applied to trees for web site analysis [Wang J.T.L. 2002], structural similarity of XML documents [Nierman A. 2002] or shape recognition [Sebastian T.B. 2001].

Zhang introduced the constrained edit distance between two trees ($T1$,$T2$) denoted by δ_c, which is defined as an edit distance under the restriction that disjoint subtrees should be mapped to disjoint subtrees. Formally, $\delta_c(T1, T2)$ is defined as a minimum cost mapping $(Mc, T1, T2)$ satisfying the additional constraint, that for all $(v1, w1), (v2, w2), (v3, w3) \in M_c$.

- $(v1, v2)$ is a proper ancestor of $v3$ iff $(w1, w2)$ is a proper ancestor of $w3$.

In [Zhang 1996], Zhang presents algorithms for the computing the minimum cost constrained mappings. For the ordered case he gives an algorithm using $O(|T1|.|T2|)$ time.

7.5.3 Dissimilarity measure between graphs

The graph classification problem can be stated as follows. It consists in inducing a mapping $f(x) : \chi \rightarrow C$, from given training examples, $T = \{< x_i, y_i >\}_{i=1}^{N}$, where $x_i \in \chi$, is a labeled graph and $y_i \in C$ is a class label associated with the training data.

Different approaches have been proposed during the last decade to tackle the problem of graph classification. A first one consists in transforming the initial problem in a common statistical pattern recognition one by describing the objects with vectors in a Euclidean space. In such a context, some features (vertex degree, labels occurrence histograms,. . .) are extracted from the graph. Hence, the graph is projected in a Euclidean space and classical machine learning algorithms can be applied [Papadopoulos 1999]. Such approaches suffer from a main drawback: to have a satisfactory description of topological structure and graph content, the number of such features has to be very large and dimensionality issues occur.

Other approaches propose to use embeddings of the graphs in a Euclidean space of a given dimensionality using an optimization process the aim of which is to best fit the distance matrix between each of the graphs. In such cases, a measure allowing graph comparison has to be designed. It is the case for multidimensional scaling methods proposed in [Bonabeau 2002] and [Cox 2001].

Another family of approaches also consists in using classical machine learning algorithms. At the opposite of the approaches mentioned above, the graphs are

not explicitly but implicitly projected in a Euclidean space, through the use of a similarity measure adapted to the processed data in the learning algorithm.

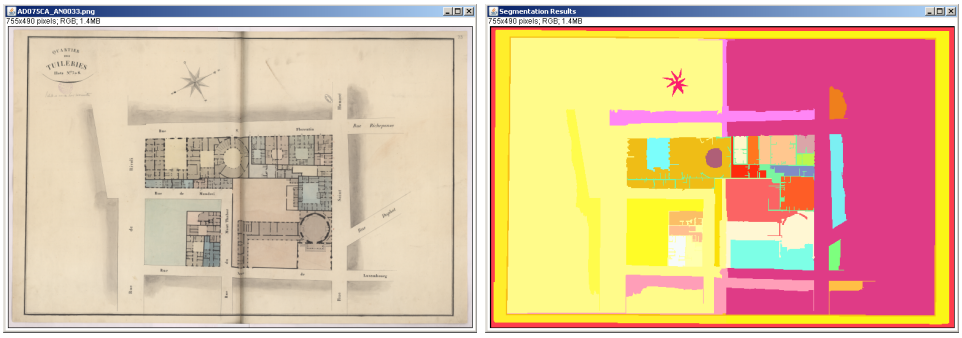
In such a context, many kernel-based methods such as Support Vector Machine or Kernel Principal Analysis were proposed recently [Kashima 2004], [Borgwardt 2005]. They consist in designing an appropriate graph-based kernel for computing inner products in the graph space. Many kernels have been proposed in the literature [Suard 2006], [Mahé 2004], [Mahé 2005]. In most cases, the graph is embedded in a feature space composed of label sequences through a graph traversal. According to this traversal, the kernel value is then computed by measuring similarity between label sequences. Even if such approaches have proven to achieve high performance, they suffer from their computationally intensive cost if the dataset is large [34]. This problem of computational cost is not inherent to kernel-based methods. It also occurs when using other classification algorithms like k-NN. In conclusion, the problem of classifying graphs requires the use of a fast but yet effective graph distance. In this objective, we used in our experiments the SubGraph Matching Distance ($SGMD_{GP}$) defined in chapter 6.

7.6 Content-Based Map Retrieval

Traditionally when facing a warehouse of natural scenes to be queried by examples; Conventional methods would just look at the system level comparing the query images to all the images within the corpus. By system level, we mean the pixel image in its self sufficient way, pixels or a gathering of pixels. When talking about images of documents, the scenario is fairly different because we are dealing with images created by humans and dedicated to humans. This makes a huge difference and allows comparisons and an exploration at higher levels. Two more stages can be drawn : (a) Image of documents can be meaningfully vectorized, and the collection can be addressed thinking at the vector level. (b) Document images have a strong semantic and an navigation using a model representation has come true.

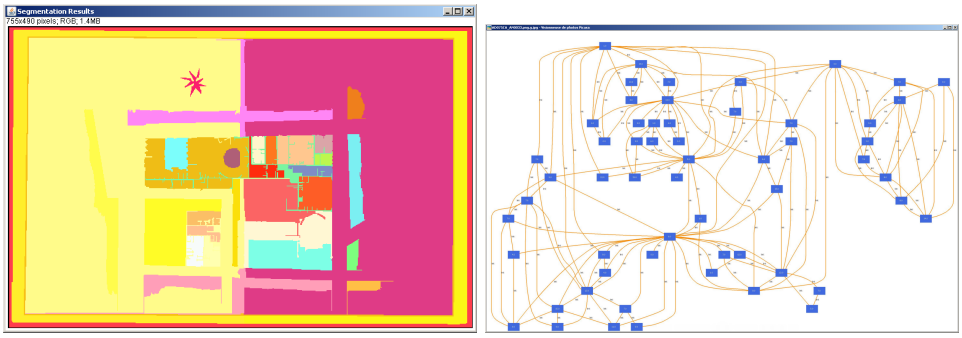
7.6.1 System level

In this step, the image warehouse of lands is queried at the system level as in a generic CBIR application. A segmentation algorithm is run on a source image. Figure 7.6 illustrates the results of segmentation when applying the Statistical Region Merging method. On top of the region partition a Region Adjacency Graph is built. Morphological features are extracted from the regions while color and textures features are left behind; the color and the texture are not comparable from map to map; for instance: color is only meaningful within a given map to distinguish parcels, it does represent a remarkable information when comparing two images. So finally, only the Zernike moments are considered and extracted for each region. As introduced earlier, features are gathered together into clusters using a K-means algorithm. The cluster IDs are set as node labels into the RAG. A segmented image and its RAG representation are displayed in figure 7.7. The Content-Based Map



(a) (b)

Figure 7.6: Cadastral Map Segmented by SMR algorithm.



(a) (b)

Figure 7.7: Region Adjacency Graph: From cadastral image to RAG.

Retrieval application is depicted in figure 7.8 where the graphical user interface is presented. The training phase proposes to the user to set some parameters such as the number of clusters (words) or the input and output directories. When the training is done on the whole set of images, the user can start the navigation process by presenting a cadastral map image as an input to the system. It would respond by comparing sequentially the query image the whole dataset using the RAG representation. The final result is an HTML page presenting the input image and the ten most similar images. An example of results is shown in figure 7.9. In this example, for the map "AN033", the 10 best responses are presented in ascending order. For each response, three information are visualized: the image, the RAG representation and the distance between the query image and an image in the database.

7.6.2 Vector level

In chapter 4, the question of vectorization issued from images of cadastral maps was discussed while in chapter 5, a polygon dissimilarity measure was defined. Combining the Map Edit Distance and the vector representation lead to a new way of browsing the collection. Presenting the vectorized map as an input to the system;

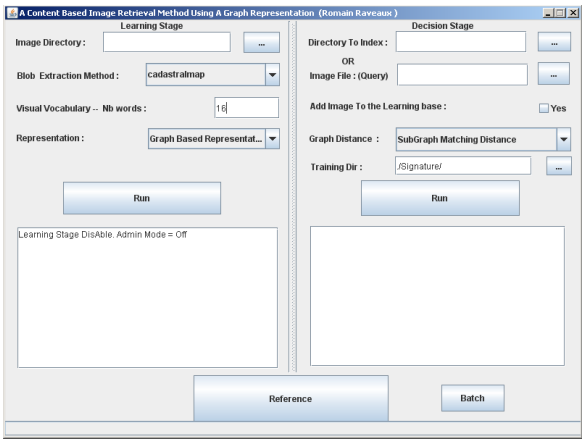


Figure 7.8: A Graphical User Interface (GUI) at system level.

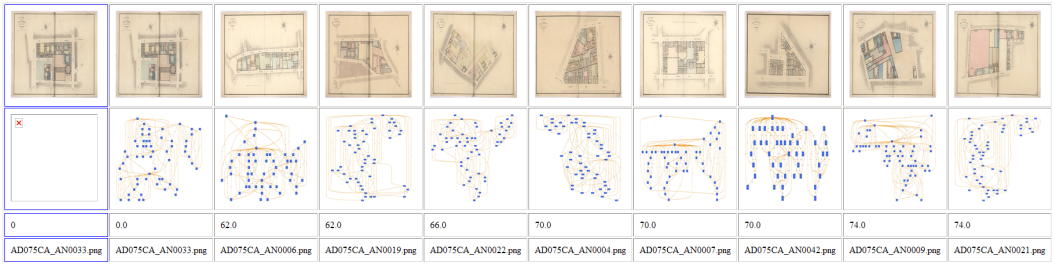
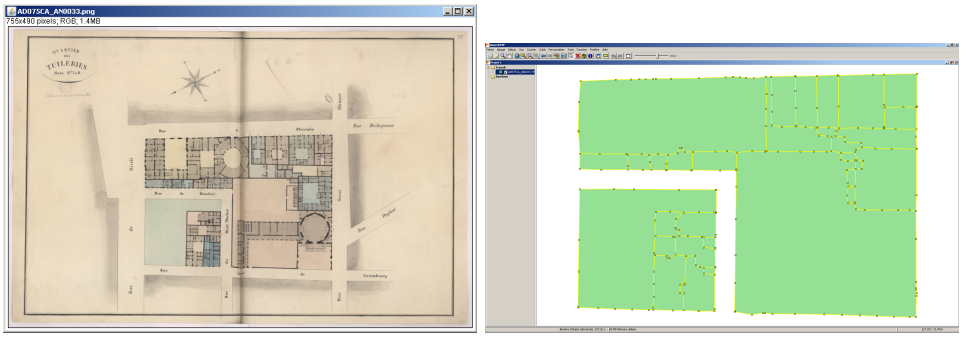


Figure 7.9: Similar map responses from a query image.



(a) (b)

Figure 7.10: Automatically Vectorized Version of the Cadastral Map.

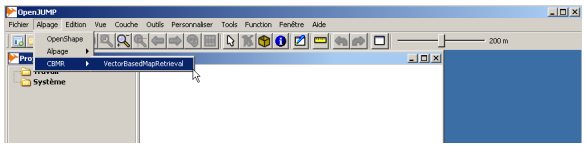
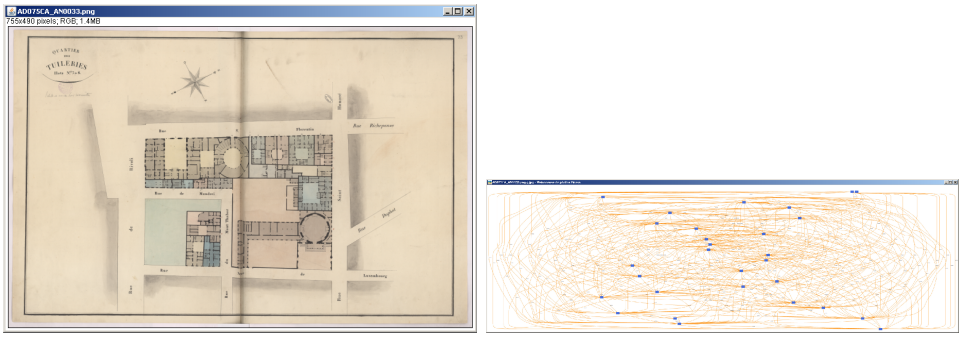


Figure 7.11: Content-Based Map Retrieval at vector level: A plug-in integration into OpenJUMP.

the application would retrieve similar documents according to the MED which relies on polygon matching and vector alignment. Figure 7.10 puts forward an association between cadastral maps and its vectorized form. The whole system is integrated in the OpenJUMP[1] framework. OpenJUMP is an open source Geographic Information System written in the Java programming language. An example of integration of our plug-in inside OpenJUMP is displayed in figure 7.11. The responses to the vector query of the "AN033" map are presented in figure 7.12.

[1] http://www.openjump.org/

Figure 7.12: Similar map responses from a query image. In the second row, the vector representation is drawn in blue.



(a) (b)

Figure 7.13: Model instance graph. Semantic graph.

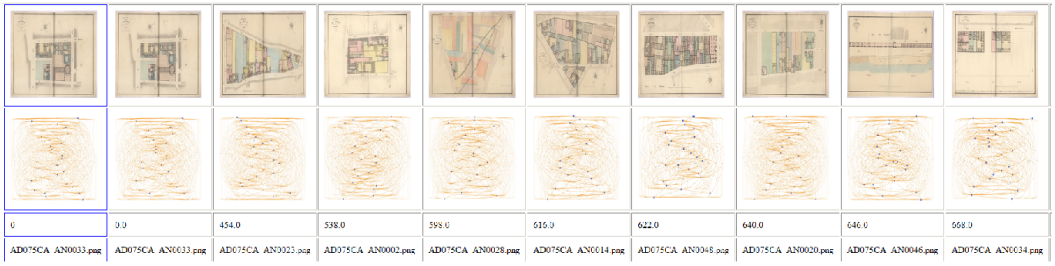


Figure 7.14: Similar map responses from a query image.

7.6.3 Semantic level

Finally, the cadastral map understanding system lies on a model architecture. This principle implies the generation of a model for each document. The model generation is ensured by the joint use of image processing algorithms and a domain ontology. The model is handled through the graph formalism. The graph construction was described in chapter 4 and consequently, the question of comparing two images turns into a graph comparison; this issue was addressed in chapter 6. The representation of a given map image under the semantic graph formalism is depicted in figure 7.13. Such a graph involves map objects (Frame, Quarters, Parcels, Streets...) and relations between objects (connected, isOutside...). The ten most similar images from a query image are given in figure 7.14. The strong semantic nature of images of documents is the basic brick of this approach. The geometry is voluntary neglected to impose the semantic in the center of the searching algorithm.

7.6.4 Discussion

A navigation into image masses at three levels of description has arisen. The ways of browsing the image collection is driven by the user. It depends on the user needs, if he/she is interested into a geometric similarity or a semantic comparison. Finally when dealing with image of documents the context is like no others and richer

representations are emerging. Due to the image nature, images made by humans to human understanding, models and semantic can be integrated into the CBIR system through the construction of semantic graphs or vector parcels; at the same time, the distance has to be re-thought to fit the user expectations. The query objectives is closely coupled to the dissimilarity measure. The semantic level can address questions like "Please find quarters from a given quarter query with the same parcel organization?" while when working with the vector representation; a question which is likely to appear is "Please find similar quarters from a query quarter in terms of sharing a maximum parallel vectors." In both cases, a specially designed distance has to be created. This phase involves a co-operation between the user and computer scientists to elaborate meaningful object comparators; for instance, in our last query, the dissimilarity measure integrates the counting of parallel vectors. In other words, a richer representation leads to a larger range of queries and it implies the design of various distances dedicated to user needs.

7.7 Experiments

In this section, our graph based approach was benchmarked and measured up to a conventional BoW methods using both $SIFT$ and $IFFS$ as information extraction systems. In a two-step mechanism, we started to analyze the vocabulary size impact choosing the best parameters and finally we compared both Bag of Words and graph based representation solutions. A pattern recognition stage was undertaken to analyze the behavior in classification. The database images are ranked in the ascending order of their distance to the query image, with the top k images returned. Two publicly available databases,Coil-100 and Caltech-101, are used to achieve our benchmark (ie. section 7.7.2).

In this practical work, the tree distance approximation was provided by Stephen Wan, Macquarie University in Australia (Reference [Implementation]) and the SIFT algorithm is an ImageJ plug-in publicly available [lightweight SIFT-implementation for Java after the paper of David Lowe (2004)] while others methods where re-implemented by us from the literature. The methods were implemented in Java 1.5 and run on a 2.14GHz computer with 2G RAM. For the comprehension of theses tests, we first introduce notations that will make the reading much simpler. A dissimilarity measure between images is a function :

$$d : X \times X \rightarrow \Re$$

where X is an image. We report in table 7.2, the notations derived from this general form.

7.7.1 Protocol

- In the first experiment, an image classification stage was carried out. Let $X_{tr} = \{x_1, .., x_n\} \ni R^P$ a crispy labeled set of training data. Our presumption is that X_{tr} contains at least one point with class label $j,\ 1 < j < C$. Let x be

Notation	Method	Representation	Distance
$SIFT_{BoW}$	SIFT	Bag of Words	Euclidean
$IFFS_{BoW}$	IFFS	Bag of Words	Euclidean
$IFFS_{Tree}$	IFFS	Containment Tree	Zhang&Shasha tree distance
$IFFS_{GBR}$	IFFS	Region Adjacency Graph	SubGraph Matching Distance

Table 7.2: Distance between images.

an unlabeled object that we wish to label as belonging to one of C classes. The standard nearest-neighbor (1-NN) classification rule assigns x to the class of the *most similar* prototype in a set of labeled training data (or reference set). Why do we use a nearest prototype classifier? Because the graph classification problem is defined in a dissimilarity space, the 1-NN classifier can be used to categorize objects in such a space, in addition, it is intuitive, simple, and often, pretty accurate. Hereafter, $E_{np}(X_{tr}; X_{test})$ denotes the test error committed by the 1-NN rule that uses X_{test} when applied to the training data. For a better understanding of the time consumption and the classification behavior, the number of classes influence is evaluated. Each data set is split up into 6 subsets containing from 5 to 100 classes (Number of classes: 5,10,20,40,80,100). These 6 folds allow us to extend our benchmark. It makes feasible, for each approach, an estimation of the generalization power over small or large data sets.

- The last experiment consists in a Content-Based Image Retrieval process. Images are ranked in the ascending order of their distance to a given query image. All these responses ($|X_{tr}|$ responses) to the query are returned to compute two measures of performance, named, Precision and Recall. Precision and recall are two widely used statistical classifications. Precision can be seen as a measure of exactness or fidelity, whereas Recall is a measure of completeness. Algorithm 5 states clearly how to obtain the values.

7.7.2 Data set descriptions

In this chapter, we consider two different labeled image databases. The well-known caltech-101 database [L. Fei-Fei 2004]. Pictures of objects belonging to 101 categories (figure 7.16). About 40 to 800 color images per category. Most categories have about 50 images. The training images were hand labeled to create a consistent ground truth. Note that we consider completely general lighting conditions, camera viewpoint, scene geometry, object pose and articulation. Our database was split randomly into roughly 75% training, 25% validation sets, while ensuring approximately proportional contributions from each class. More information about this data set is presented in table 7.3. The COIL-100 database [S. A. Nene] consists of images of 100 different objects. The objects were placed on a motorized turntable against black background. The turntable was rotated through 360° to vary object

Algorithm 5 Precision and Recall computation

Require: For the i^{th} query x_{ij} belonging to the class j from X_{test}.
Ensure: There exists exactly $|X_{tr}|$ pairs of precision and recall measures.

1: **For** $k = 1$ **To** k= $|X_{tr}|$ by **Step**=1 **Do**
2: Get the k top responses and put them into a list called O
4: Within the list O compute the precision and recall values.
5:

$$precision_{ik} = \begin{cases} \frac{|\{Relevant\ Documents\} \cap \{Retrieved\ Documents\}|}{|\{Retrieved\ Documents\}|} \\ \\ \frac{|Correctly\ Labelled\ Documents|}{k} \end{cases}$$

6:

$$recall_{ik} = \begin{cases} \frac{|\{Relevant\ Documents\} \cap \{Retrieved\ Documents\}|}{|\{Relevant\ Documents\}|} \\ \\ \frac{|Correctly\ Labelled\ Documents|}{|\ Documents\ of\ class\ j\ |} \end{cases}$$

7: **End For**

pose with respect to a fixed color camera. Images of the objects were taken at pose intervals of 5°. This corresponds to 72 poses per image. Figure 7.15 shows an example image of each class. Randomly, for each class of object, 18 images are withdrawn from the initial set to constitute a test set. This leads us to a training set of 5400 images and a test base of 1800 items.

These two sets of data are fairly different and represent an heterogeneous environment to prove the merit of our systems. The Coil-100 database is known to be relatively simple since no backgrounds are considered and images within the same class are derived from a single original object, on the contrary, the caltech-101 set is more complex, a same concept gathers different kind of images from different sources.

7.7.3 A classification context

Back on track, we keep in mind that the final purpose is to perform a classification stage in order to evaluate the relevance of the image models. Based on the data sets described in section 7.7.2, a 1-NN rule is applied to obtain the number of correctly classified instances (CCI) and the corresponding classification rate. Firstly, the number of words impact is investigated and the results are illustrated in figures 7.17 and 7.18 for the Coil-100 and Caltech-101 databases respectively. Then a comparison between the four image distances is brought considering the best number of words for each method. These results are shown in figures 7.19.

Complete results figures:

Figure 7.15: Columbia University Image Library

Table 7.3: Characteristics of the data set used in our computational experiments

	Caltech-101	Coil-100
$\mid Training \mid$	6821	5400
$\mid Test \mid$	2323	1800
IFFS: Feature Length	133	133
IFFS: Average number of nodes	31.34	14.10
IFFS: Average number of edges	72.15	25.905
SIFT: Feature Length	128	128
SIFT: Average number of interest points	121.57	40.02

Figure 7.16: Image Samples from the Caltech-101 Data set. The 101 object categories and the background clutter category. Each category contains between 45 to 400 images. Two randomly chosen samples are shown for each category. The categories were selected prior to the experimentation, collected by operators not associated with the experiment, and no category was excluded from the experiments. The last row shows examples from the background dataset. This dataset is obtained by collecting images through the Google image search engine (www.google.com). The keyword "things" is used to obtain hundreds of random images. Complete datasets can be found at http://vision.caltech.edu

- Figure 7.17 gathers four histograms, one for each method exposed to our evaluation framework ($SIFT_{BoW}$, $IFFS_{BoW}$, $IFFS_{Tree}$, $IFFS_{GBR}$). This complete test aims at underlying the influence of the vocabulary size parameter on the recognition rate. The scalability question is also addressed by increasing progressively the number of classes. In this way, the behavior of each approach is depicted as the problem becomes more and more complex. These tests were run on the Coil-100 set.

- Figure 7.18 reflects the recognition rate evolution according to the number of words and the number of classes for the Caltech-101 database.

Summary results figures:

- Figure 7.19 expresses the best results in classification obtained for the most suited number of words. It is the quintessence of the results over the two databases, hence, it makes the comparison more readable and clearer.

- Figure 7.20 presents how many words are needed for each method to provide their best accuracy level. It shows how sensitive and greedy are the methods about this question of the number of clusters.

Number of words impact

Tests on the number of words were carried out. Performance in classification according the number of words (w) are presented in figure 7.20. The question of the vocabulary size is an important issue. Here, a decision of tuning the parameter w from 4 to 1024 was taken. In this way, we expect to cover a wide range of possibilities. A first comment states that structural approaches reach their maxima with a smaller number of clusters than BoW methods. Reducing the vocabulary size put more weight on the graph data structure while a large number of words is highlighting the information carried by each regions. A compromise between the feature expressivity and the importance given to the spatial organization has to be found. As an example, too many words may turn the representation very sensitive to noises and small variations, on the other hand, if no feature is extracted from the regions then only the structure is taken into account. Those extrema are representative of how the vocabulary size can impact the classification process.

The histograms presented in figure 7.20 corroborates the following hypothesis, when the number of classes increases the vocabulary size should be extended too. Bigger is the problem more words are needed to describe it. However, our experiments pointed out that $IFFS_{GBR}$ and $IFFS_{Tree}$ needed a smaller set of words than BoW for the same configuration to reach their best performances.

Recognition rate comparison

GBR *vs* BoW

In the meantime, each database were divided into six subsets to analyze the number of classes influence. The figure 7.19 denotes a straightforward fact, a high number of classes leads to a decrease of the performances as the problem becomes more complex. Contrarily to our first thoughts, structural based representation did not overcome the *BoW* methods in terms of accuracy. Over the two databases, results of *BoW* systems outperform the structured ones. This leads us to the question: does structure really matter when indexing natural scene images? The main advantage of a description of patterns by graphs instead of vectors is that graphs allow for a more powerful representation of structural relations. However, in natural images, it appears that the structure may not be stable enough and this variability might be misleading. Nevertheless, the use of a GBR method is recent in CBIR, and we can say that they achieve reasonable results for a "new born" solution. They can obtain similar or slightly under performance than *BoW*. When at the same time, *BoW* methods are mature, they have been introduced decades ago in CBIR, they have the age benefits. Furthermore, these encouraging recognition rates reached by GBR methods can be improved, it does exist a rich literature dealing with the insertion of spatial information into graph edges. We can mention GBR methods using Bi-dimensional Allen Algebra (Ref. [Aiello 2004]) or Delaunay triangulation (Ref. [Finch 1997]). In addition, the Region Adjacency Graph could be swapped for a neighboring graph or a visibility graph for instance, but all these variations on the same theme are beyond the scope of this thesis. Here, the objective was to expose that our results are encouraging enough and it leave is plenty of rooms for progress in this direction. Finally, graphs lead to new kind of services, the graph matching problem can be used to locate sub parts of an image from a crop image as a query. All these points converge to state the worth of investigating the graph tools in a CBIR context. A comprehensive comparison is provided in table 7.4. This table sums up the information according the following metric (Eq.7.11). The mean value of the best results over the 6 subsets.

$$\overline{E_{np}} = mean\left(\sum_{i=1}^{6} \min_{w} \left(E_{np}\left(X_{tr_i}; X_{test_i}\right)\right)\right) \tag{7.11}$$

It turns out that classification accuracy can be improved by $IFFS_{GBR}$ compared to the reference system, that is to say $SIFT_{BoW}$, and this, on all number of classes levels. Note that 2 out of 3 improvements are statistically significant.

Independence inter methods

In this experiment, we aim at understanding whether the methods make the same mistakes or not; if the methods decide wrong at the same time or not. On the Caltech-101 database, we perform a χ^2 test of independence. A test of independence assesses whether paired observations on two variables, expressed in a contingency table, are independent of each other -for example, whether $IFFS_{BoW}$ differs in the

Table 7.4: Average results over the two databases according to the accuracy criterion and time consumption.

Criterion	$SIFT_{BoW}$		$IFFS_{BoW}$		$IFFS_{Tree}$		$IFFS_{GBR}$	
	Coil	Caltech	Coil	Caltech	Coil	Caltech	Coil	Caltech
Accuracy(%)	76.25	50.61	95.03•	46.86	78.86	36.68◦	90.68•	44.28
Time(s)	17604	14457	24358	19069	645567◦	162889◦	898279 ◦	250570◦

• Statistically significantly better than the reference system ($SIFT_{BoW}$) ($\alpha = 0.05$)

◦ Statistically significantly worse than the reference system ($SIFT_{BoW}$) ($\alpha = 0.05$)

	$IFFS_{BoW}$						x
$IFFS_{GBR}$		class1	class2	class3	...	class101	
	class1						
	class2						
	class3						
	...						
	class101						
y							$\lvert X_{test}\rvert$

Table 7.5: Dependence matrix for $IFFS_{BoW}$ and $IFFS_{GBR}$

decision with $IFFS_{GBR}$. The contingency table, in a context of classification, is also called confusion matrix. Each column of this matrix represents the number of occurrences of an estimated class, while each line denotes the number of occurrences of a real class. From the confusion matrix, we derive the construction of what we call a dependence matrix. This latter reflects the dependence of two classifiers based on different representations. In our case, each column of this matrix represents the number of occurrences of an estimated class by the method one, while each line denotes the number of occurrences of an estimated class by the method two. An example of this independence matrix is presented in table 7.5.

In this case, an "observation" consists of the values of two outcomes and the null hypothesis is that the occurrence of these outcomes is statistically independent. Each outcome is allocated to one cell of a two-dimensional array of cells (called a table) according to the values of the two outcomes. The "theoretical frequency" for a cell, given the hypothesis of independence, is

$$\chi^2 = \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

In our case, the observed value "O" corresponds to the value of the dependence matrix whereas the theoretical occurrence is defined by the average. In each cell, the expected value E_{ij} is equal to the sum of each element of the line i multiplied by the sum of the elements of the column j, divided by N.

$$E_{ij} = \frac{x_i \times y_j}{|X_{test}|}$$

The expected value (E) can be seen as the wanted value in case of independence.

	χ^2 test	df	$p-value$
$IFFS_{BoW}$ *vs* $IFFS_{GBR}$	9922	10000	0.209

Table 7.6: χ^2 independence test between a Graph-based method and a Bag of Words approach.

We computed the χ^2 for the following setting: $IFFS_{BoW}$ vs $IFFS_{GBR}$. We consider a null hypothesis of independence ($H0$) between the two methods and then, we compute, by means of a one-tailed statistical hypothesis test, the probability (p-value) of getting a value of the statistic as extreme or more extreme than observed by chance alone, if $H0$ is true. Results are presented in table 7.6. We compare the χ^2 score with the theoretical χ^2 distribution (degree of freedom (k=10000), risk level (α =0.05)), $\chi^2_{\alpha=0.05,k=10000}$=10233.8). $\chi^2 < \chi^2_{\alpha=0.05,k=10000}$, so we can say that the hypothesis $H0$ of independence can be accepted in with a risk of 5%. The calculated p-value exceeds 0.05, so the observation is consistent with the null hypothesis, the deviation from expected outcome is just small enough to be reported as being "not statistically significant at the 5% level".

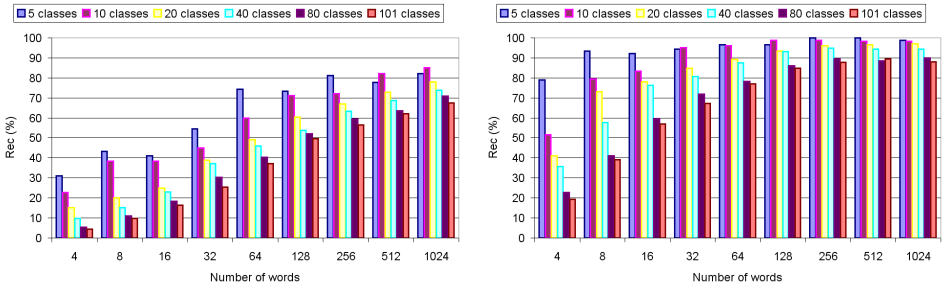
In fact, we draw the reader's attention to de-correlated methods, they are likely to be combined to perform better. Inspired from [Philbin 2007] and stimulated by these results of independence, an interesting work will come up. It would aim at speeding up the system by computing at first a BoW method and later in a second time, to process a re-ranking stage with the top k responses integrating spatial information through the use of our graph-based approach. To avoid sequential comparison of the query with all items stored in the archive.

IFFS *vs* SIFT

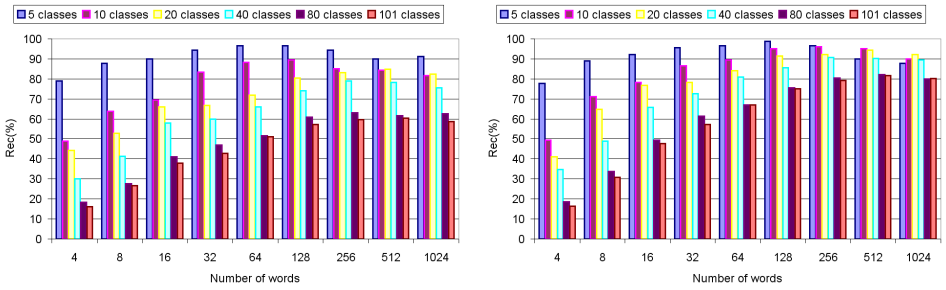
A comment on the good behavior of IFFS as an extraction information system. Hence, figure 7.19 validates the join use of an efficient segmentation algorithm (SRM) and distinctive features. On the Coil-100 database $IFFS_{BoW}$ overrides $SIFT_{BoW}$ with a significant level. Nevertheless, the power of generalization of this statement is limited by the superiority of the SIFT process on the Caltech-101 data sets.

7.7.4 In a CBIR Context

Precision is defined as the ratio of retrieved positive images to the total number retrieved. Recall is defined as the ratio of the number of retrieved positive images to the total number of positive images in the corpus. The precision and recall in a multi-class problem is defined through multi-levels (or j is greater than 1). The overall average precision and recall over all classes j can be evaluated by the macro-average, which first calculates the precision and recall on each class j followed by a calculation of the average information on the C classes.

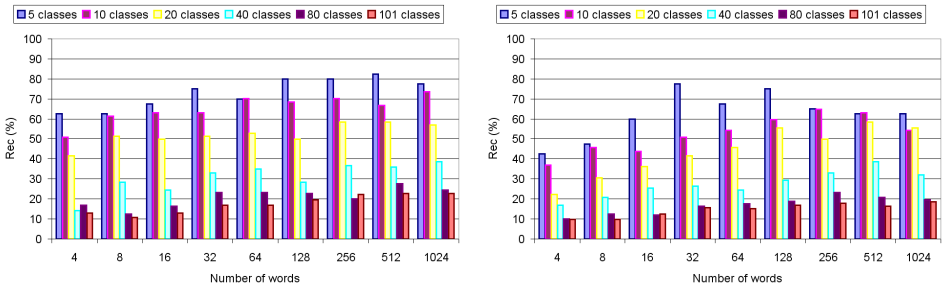


(a) $SIFT_{BoW}$ (b) $IFFS_{BoW}$

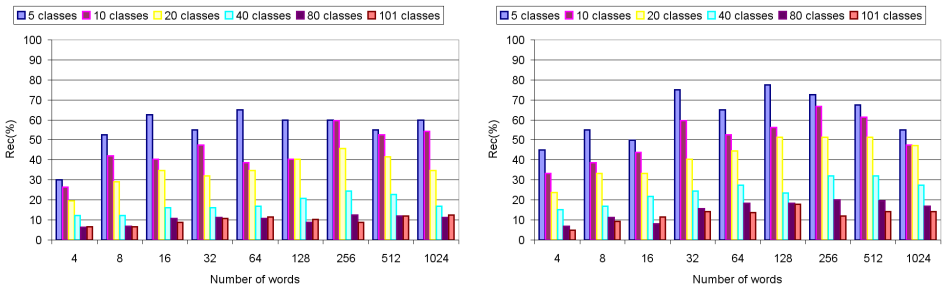


(c) $IFFS_{Tree}$ (d) $IFFS_{GBR}$

Figure 7.17: On the Coil-100 database : Recognition rate in function of the number of classes and the number of clusters.

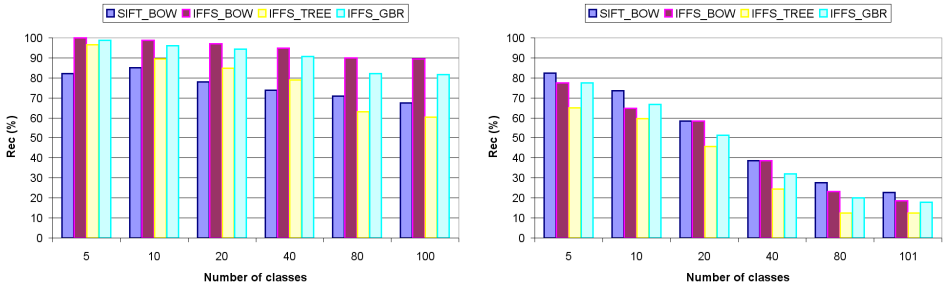


(a) $SIFT_{BoW}$ (b) $IFFS_{BoW}$



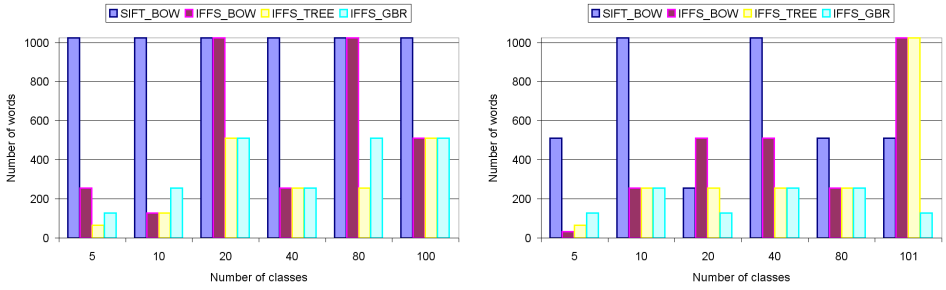
(c) $IFFS_{Tree}$ (d) $IFFS_{GBR}$

Figure 7.18: On the Caltech-101 database : Recognition rate in function of the number of classes and the number of clusters.



(a) Coil-100 (b) Caltech-101

Figure 7.19: Comparison between CBIR methods. Summary of results obtained with the best number of words for each method.



(a) Coil-100 (b) Caltech-101

Figure 7.20: Comparison between the number of words in used by the methods.

	$SIFT_{BoW}$	$IFFS_{BoW}$	$IFFS_{Tree}$	$IFFS_{GBR}$
$Coil-100$	0.0640	0.3242	0.1818	0.2288
$Caltech-101$	0.0314	0.0327	0.0279	0.0335

Table 7.7: Average Precision (AP) measure. A comparison of the performance of the four methods.

$$precision = \frac{\sum_{j=1}^{C} precision_j}{C}$$
$$recall = \frac{\sum_{j=1}^{C} recall_j}{C}$$

To evaluate the performance we use the average precision (AP) measure computed as the area under the precision-recall curve. An ideal precision-recall curve has precision 1 over all recall levels and this corresponds to an average precision of 1. The AP scores is used as a single number to evaluate the overall performance. Results are reported in table 7.7.

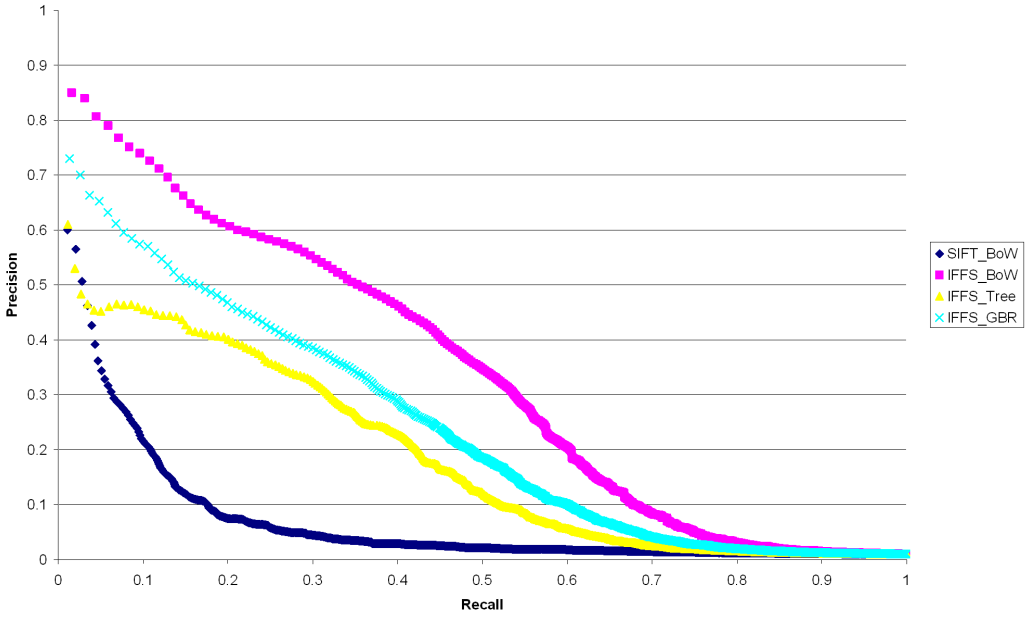
On both databases, precision and recall values are computed and displayed in the figure 7.21.

7.7.5 Analysis and discussion

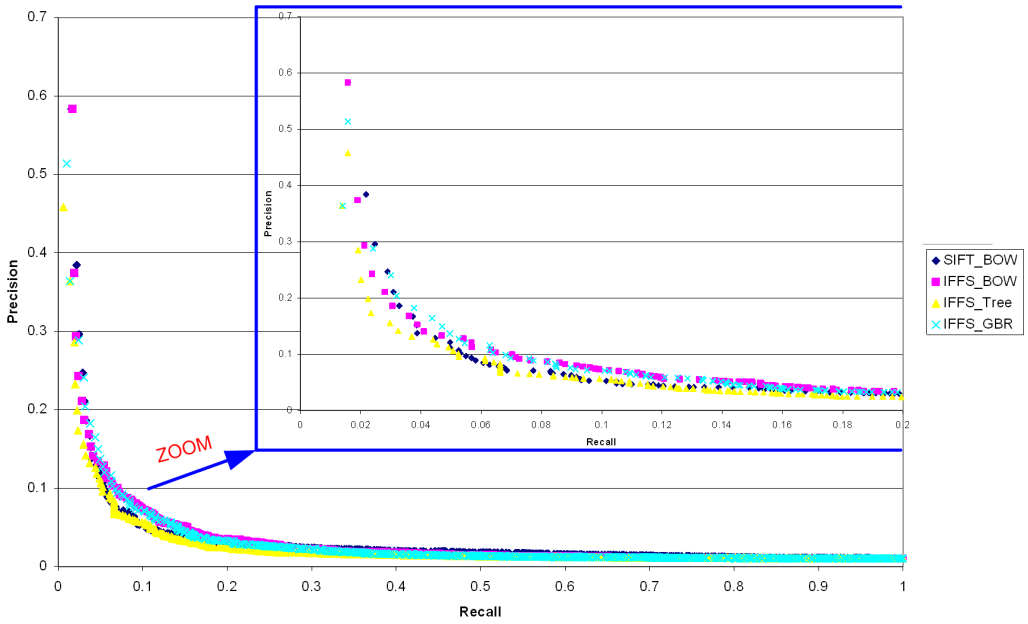
The results are somehow promising with respect to the IFFS approach. It clearly outperforms the $SIFT_{BoW}$ approach in the Coil database. This result could be expected as images are more easily segmented in this database. However, in the Caltech database, where segmentation into regions is more challenging the SIFT obtains better results. On the other hand, the figure 7.21b puts forward that IFFS does not declare forfeit and tends to get a better precision when the recall is increased. This last comment is re-enforced by measures given in table 7.7. The AP score of $IFFS_{BoW}$ ($AP_{IFFS_{BoW}}$) is slightly greater than $AP_{SIFT_{BoW}}$.

Concerning the structural representations results are somehow encouraging. They are a bit lower than the other approaches, and only in the Coil database are slightly better than BoW with SIFT, but clearly worse than BoW with IFFS, which could be the reference method in this case, as the graph representation is built on the top of IFFS.

These poor results of the structural approaches seem to refute the main initial hypothesis about the use of this type of graph representation. Nevertheless, the IFFS method is an interesting contribution since it makes possible the organization into graph or tree whereas SIFT is too versatile to be laid out into a complex structure (Too many key-points occur when running sift on an image). Taking into account, the good results on COIL-100, a discussion arises on the kind of images where IFFS method can be useful. In addition, the idea of completing



(a) Results on Coil100 database



(b) Results on a 20 classes subset from the Caltech-101

Figure 7.21: Precision and Recall curves.

this representation with structural information is also promising. There is not much work in this direction so far. Structural representations stand as a kind of alternative approach with some preliminary results, but to be further investigated. Graph-Based Representation of an image is a rich domain, relations between regions or points of interest can be modeled in many ways, among them, we can cite the representations issued from Delaunay triangulation [Finch 1997], Allen algebra [Aiello 2004] or a neighboring graph.

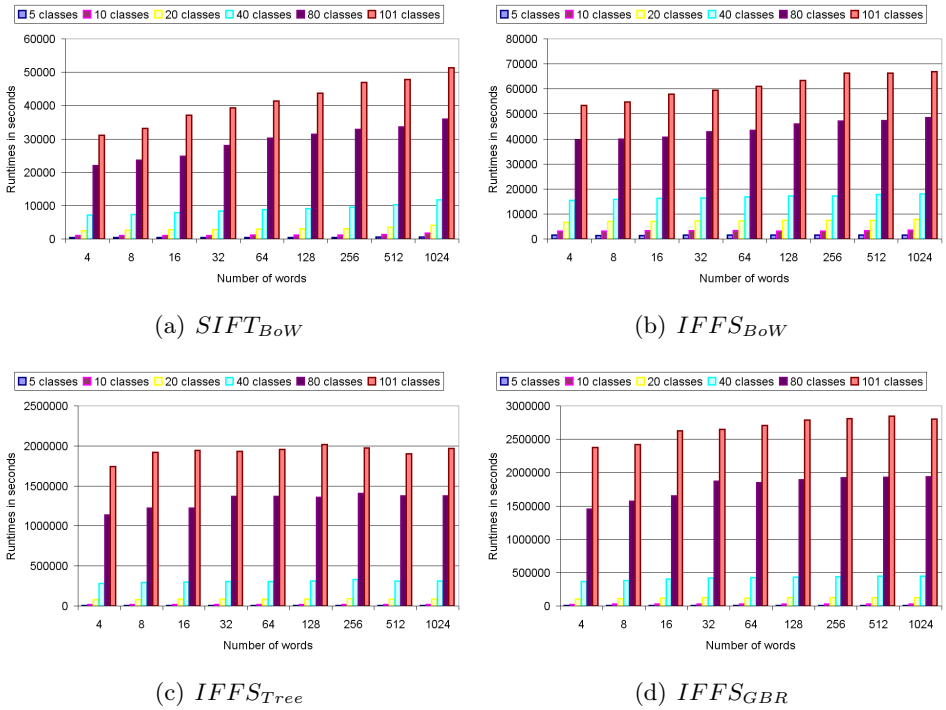
7.7.6 Time complexity

The graph matching distance ($IFFS_{GBR}$) can be calculated in $O(n^3)$ time in the worst case. To calculate the matching distance between two attributed graphs G_1 and G_2, a minimum-weight matching between the two graphs has to be determined. This is equivalent to determining a minimum-weight maximal matching in the sub-graph matching of G_1 and G_2. To achieve this, the method of Kuhn [Kuhn 1955] and Munkres [Munkres 1957] can be used. This algorithm, also known as the Hungarian method, has a worst case complexity of $O(n^3)$, where n is the number of probes in the larger one of the two graphs. On the other hand, the histogram distance (used in $IFFS_{BoW}$, $SIFT_{BoW}$) is processed in linear time in function of the number of bins that composes the histogram. A way to compare the computational cost of the different types of distance was to undertake an empirical study on the classification stage. The figure 7.22 depicts a comparison of the runtime execution according to the kind of distances. This test was performed during the classification phase on the Coil-100 database. It takes into account the computation of regions of interest (IFFS or SIFT) and the distance calculation between image representations.

A first comment aims at illustrating the high time consumption of the graph and tree distances. These techniques are computationally more intensive than others. Structural approaches may fail to face the scalability dilemma in the cases of industrial applications, although their computations remain in polynomial time. Another point illustrated by the figure 7.22 is the effect of the vocabulary size on the histogram length. Simply, higher is the number of words and larger are the histograms. In average, a linear relation exists between the number of clusters and the time complexity of histogram based methods. Finally, $SIFT_{BoW}$ runs slightly faster than $IFFS_{BoW}$ (at worst case: 51000 seconds *vs* 67000 seconds). This time gap is low enough to not reject $IFFS$ as a suitable solution considering the significant accuracy gain it can imply. This little loss of speed does not discourage the application of a color segmentation algorithm to extract blobs.

7.8 Conclusion

In this chapter, a graph based representation was proposed in a CBIR context. From a partition into regions processed by an efficient segmentation algorithm, a Region Adjacency Graph was built to consider spatial relationships between regions.



(a) $SIFT_{BoW}$ (b) $IFFS_{BoW}$

(c) $IFFS_{Tree}$ (d) $IFFS_{GBR}$

Figure 7.22: On the Coil-100 databases : Runtimes in function of the number of classes.

Each region is characterized using a set of features based on the Color, Texture and Shape. A K-means clustering algorithm is applied to cluster the regions on the basis of these features. These clusters which we call "blobs" compose the vocabulary for the set of images. Each blob is assigned to a unique integer to serve as its identifier. An efficient and yet fast dissimilarity measure between structured data was presented to compare attributed relational graphs. The whole method was compared to conventional Bag of Words strategies and to another structural approach based on Containment Trees. The Graph Based Approach overcame the Tree Based one, however it gave similar or slightly under results than BoW methods. BoW systems have been introduced a decade ago into CBIR applications while GBR are quite new in this field of science. Nevertheless, experiments showed that a structural approach requires a fewer number of words to reach its best performance.

A closer look should be given to the relation between regions. For instance, a future promising work concerns the enrichment of the graph representation by the use of a bi-dimensional Allen Algebra. This description inserted on the edge labels should provide a better representation of the region layout.

In addition, we want to express the special interest given to Graph Based Representation in CBIR context, as a final goal, GBR could offer the possibility to spot sub-parts of images from an image portion of the query image. The flip side of coin is an over-load of complexity which leads to a higher time consumption.

Inspired from [Philbin 2007] and stimulated by the results of independence be-

tween BoW and Graph methods, an interesting work will come up. It would aim at speeding up the system by computing at first a BoW method and later in a second time, to process a re-ranking stage with the top k responses integrating spatial information through the use of our graph-based approach. A sequential comparison of the query with all items stored in the archive could be avoided.

The last words comment the particularity of dealing with image of documents. This context is like no others and richer representations are available. Due to the image nature, images made by humans to human understanding, models and semantic can be integrated into the CBIR system through the construction of semantic graphs or vector parcels. The query objectives is closely coupled to the dissimilarity measure. For instance, a question which is likely to appear in order to search quarters with a similar orientation: "Find similar quarters from a query quarter in terms of numbers of common parallel vectors." In this case, the dissimilarity measure involves the counting of parallel vectors. In other words, a richer representation leads to a larger range of queries and implies the design of various distances dedicated to user needs. This last phase involves a co-operation between the user and computer scientists to elaborate meaningful object comparators.

Chapter 8

Conclusions

Contents

8.1 Foreword

In this chapter we summarize the contributions of this dissertation on the cadastral map analysis problem and in particular to the application of graph-based methods on the process of raster to vector conversion from collections of cadastral images. We also present a discussion and the limitations of the presented approaches. We finally point some possible lines of continuation on the field of technical document understanding and some improvements of the proposed methods which should be further studied.

8.2 Summary of the Contributions

In this thesis we have introduced a complete framework for ancient and color cadastral map, and in particular for a focused Geographical Information System. As explained in chapter 1, our work has been motivated by the specific problem of proposing a raster to vector conversion methodology able to locate and vectorize graphical content within a complete document image. A lot of interest is made worldwide for mass digitization of document collections and their storage in digital libraries. It results in digital repositories rich in information if they are semantically accessible.

We have identified four different levels when conceiving a raster to vector architecture for color documents.

The first level aims to take advantage of the color properties in order to better locate graphical elements that constitute a document. In the second level, these bricks describing graphical symbols are organized in a particular data structure. This data structure is carefully chosen to conserve a relevant meaning, with respect

to symbols and their relationships. After the extraction process, this semantic representation is transformed to elaborate by an inferring method a model of higher level. This meta-model represents a pivot platform which allows the comparison with *a priori* knowledge, an expert-designed model. This unsupervised checking consists in a validation stage to measure the quality of the object detection system.

The third level consists in a validation stage to determine in regard with the ground-truth whether the raster to vector conversion is efficient or not.

The last level is a querying process where different views are adopted, from pixel to semantic passing by vector elements. All the data structures are traversed and compared to find similar primitives than the queried ones.

Along this thesis, we have made some contributions in each of the four stages. Let us briefly summarize these contributions:

- **Color Segmentation in Hybrid Color Space:** The chapter 3 of the thesis has been focused on the joint use of hybrid color space and a vectorial gradient aiming to segment the graphical symbols. We have presented a hybrid color space selection framework. We brought an quantitative answer to the question of finding a color space for segmentation purpose. In addition, we have proven its superiority in regard to the standard RGB color space.

- **Unsupervised Quality Measure by Model Checking:** In chapter 4, a cadastral map modeling was introduced. This modeling gave birth to dedicated image processing. In order to auto-evaluate the quality of these operators, a meta-model comparison was presented. Object found during the detection phase were organized into a graph where the vocabulary was picked up among the expert-designed ontology. Hence, the generated model is derived from the ontology but not comparable yet with the meta-model. A higher degree has to be reached, a meta-model inference from an RDF document has been developed to achieve this task. Finally, the comparison of two meta-models is turned into a graph distance problem.

- **Graph Matching:** A distance between graphs derived from a graph matching is fully explained in chapter 6.This latter describes a SubGraph Matching Distance (SGMD) for graph comparison. A distance between two graphs is defined by solving for a max matching in a bipartite graph spanning the nodes in two graphs. A probe is computed for each node, which describes the neighborhood structure of a node out to a distance of 1 edge (along all incident edges). The cost assigned to an edge in the bipartite graph spanning two nodes is computed as the edit distance between their probes. The resulting approximation to computing the largest isomorphic subgraph is $O(n^3)$, the complexity of the bipartite matching problem.

- **Performance Evaluation Protocol:** In chapter 5, we have presented a set of measures to evaluate the performance of vectorization systems in terms of location accuracy and polygonal approximation precision. We have shown that

the proposed measures allowed to determine the weaknesses and strengths of our method. Although within the Graphics Recognition community there is an important interest in the research of the performance evaluation topic, to the best of our knowledge, no framework for evaluating the performance of polygon generator applications has been proposed in the past.

- **Content-Based Map Retrieval:** In chapter 7, a navigation into image masses at three levels of description was described. Hence, the user can express his need by giving a query image, and thereafter receiving as a result all similar images. Here, the word "similar" is voluntary not accurate enough. The user may want to give a special care to geometrical aspects between maps or on the contrary, the focus could be set on semantic, the type of objects laid into the maps. It depends on the type of model which is incorporated into the search engine, from pixel to semantic modeling. This crucial choice is left to user when throwing his query.

8.3 Discussion

In this thesis we have made some contributions about raster to vector conversion methods and the particular application of such methods for a focused retrieval system from a collection of map drawings. Herein, we want to launch an open discussion about parameters, limits and deadlocks of the proposed approaches.

- **Color processing:**
  - This chapter 3 proposes a generic framework by which the best representation color space for a computational task on a given image is selected. That impact significantly to the results of segmentation task, and somehow alleviate the embarrassing situation of giving the decision to choose color space in this task. Recognition rate is used for selecting color components in a set of basic color spaces. This is tied with a basic nearest-neighbor classification algorithm. Edge detection algorithm based on the discontinuities of the distance in the hybrid color space is then applied.
  - Why are there 25 color primaries, rather than 3 because no new information is introduced. The increase in dimension of the color space is not explicitly justified in the manuscript. However, the justification comes along with the justification for the kernel trick in classification problems. In machine learning, the kernel trick is a method for using a linear classifier algorithm to solve a non-linear problem by mapping the original non-linear observations into a higher-dimensional space, where the linear classifier is subsequently used; this makes a linear classification in the new space equivalent to non-linear classification in the original space. The underlying idea is that the new space with higher dimension better describes the problem and so makes it simple to solve.

  - The features of the hybrid color space and its adequacy to a specific class of images are not discussed. Nevertheless, we believe that standard color spaces were specifically designed for natural scenes and since the color distribution of our color documents is fairly different and holds specificities; a different color space is needed.
  - The time consumption was not a hard specification expressed by the historians. It is a commonplace in machine learning to state the case that training algorithms require much time and many computations to assimilate the data variability.
  - A low gain in performance was noticed in chapter 3. The color segmentation is closer to the user-defined ground-truth in a hybrid color space than in RGB however the improvement remains quite low. A tentative of explanation may lie on the fact that color on its own fails to describe certain objects. In our framework, the separability of the clusters found in the color space is maximized. However, clusters may not be directly linked to objects within the image. There is a gap between an object and its color representation. A given object can be represented by sparse data in the feature space, it is the case for textured objects for example, hence the colors defining a single object can be found in different clusters. This way is misleading and results in an over or under segmentation phenomena. In such a case of textured objects, a manual definition of the clusters by the user could be really a plus, it would help defining what are color classes to be discriminated.

- **Knowledge engineering:** It is quite hard to express a real and concrete problem with a computer model. A model is by definition a truncation of the real world, however this phase is essential to design image processing and to extract meaningful information from a document set. It is important to note that our segmentation methods are directly designed to respond the nature of images like ancient cadastral. The knowledge modeling provides keys to build-up dedicated image processing. Thanks to knowledge modeling step, we developed dedicated image algorithms addressing the particularity of our image set. The analysis steps consist among others, of the elaboration of two complementary information types for a given document :
  - its physical structure: it describes the document organization, in terms of objects (typographically homogeneous regions) and relationships between these objects (hierarchical decomposition, absolute and relative positions in the paper sheet) ;
  - its logical structure: it decomposes the document in information elements, characterized by the role they play in the document (frame, streets, parcels...), and specifies the relationships (syntactic and semantic) between these elements;

The formalism to describe the expert knowledge can take various form, from logic or grammars to meta-models or ontology. Our choice was guided by the huge development of the Ontology domain. Nowadays ontologies offer a common framework and standardize methods to express and to manage knowledge in computer science. Our tools allow extracting the logical structure from a RDF document and representing this logical structure as an attributed relational graph.

- **Structured object comparison:** The model comparison question can be stated as classification problem. The problem of classifying graphs requires the use of a fast but yet effective distance. We propose a novel framework called $SGMD$ for graph matching based on subgraph (probe) assignment via bipartite matchings.

  - According to our formalism, a graph is decomposed into a set of subgraphs. Each subgraph is defined from a root node and any graph distances can be applied to elaborate the bipartite graph. The two parameters of this approach are the local descriptor size and the type of sub distance. Preliminary results not developed into this manuscript tend to show that if the user of the final application is interested in retrieving the most relevant graphs from the collection, no matter the number of false alarms, a simpler description should be used. If the user is more interested in a better precision without caring the fact the system misses objects, then we should start using more and more complex and refine description techniques. However, we strongly believe that for most applications, the use of low-dimensional descriptors (a subgraph depth of 1 or 2) is enough while making the system faster. The question of the kind of the subgraph distance is more delicate, our feeling is that it depends on the velocity required by the user. In a case of large subgraphs, the use of the graph edit distance may serious slow down the $SGMD$ while using the graph probing measure the system may still react on time. A theoretical relationship would be a real plus.
  - Our contribution in chapter 6 gives the proof for the use of a rapid and simple, yet sufficient graph distance which can be processed to scale up a k-NN classification step. In this direction of taking fast decision at the classification stage, we are currently working on learning graph prototypes for shape recognition. A new approach for computing graph prototypes in the context of the design of a structural nearest prototype classifier. Four kinds of prototypes are investigated and compared : set median graphs, generalized median graphs, set discriminative graphs and generalized discriminative graphs. They differ according to (i) the graph space where they are searched for and (ii) the objective function which is used for their computation. The first criterion allows distinguishing *set prototypes* which are selected in the initial graph training set from *gen-*

eralized prototypes which are generated in an infinite set of graphs. The second criterion allows distinguishing median graphs which minimize the sum of distances to all input graphs of a given class from discriminative graphs, which are computed using classification performance as criterion, taking into account the inter-class distribution.

- Finally, a statistical relationship was found between $SGMD_{ED}$ and the graph edit distance, but a starting work, let us think that $SGMD_{ED}$ is an upper bound for the graph edit distance with factor which is still undetermined.

- **Performance evaluation:** The chapter 5 addresses the evaluation of the polygonization process (i.e. raster to polygon conversion). The main drawback is related to the definition of a benchmark protocol of the vectorization, based on the fact that this process is supposed to delivery straight lines and clustering of these lines in terms of polygons.

  The first one is related to the definition of a benchmark protocol of the vectorization, based on the fact that this process is supposed to delivery straight lines and clustering of these lines in terms of polygons. The second issue is the use of this benchmark protocol for the evaluation of a cadastral map vectorization system designed by the authors. Several other constraints were imposed either due to lack of time and resources or in order to keep the evaluation protocol simple. The primary constraints were as follows: (i) Text regions were not considered, OCR is outside the scope of this benchmark; (ii)Dashed lines and isolated segment that do not constitute polygons were ignored.

  - The method presented in the paper permits only to evaluate polygon detection and approximation.
  - We confronted our measures of quality to a human-based evaluation. However, more work should be done in this way to obtain an objective assessment on commonly accepted criteria.
  - The protocol is designed for polygons but may be easily extended to other line shapes. This is possible thanks to the graph formalism which confers to the approach a generic nature.
  - Finally, we would like to mention the importance that has the use of a performance evaluation protocol. Times where algorithms were tested with a small set of data are over. Nowadays, it is necessary the use of standard reference ground-truth and performance evaluation protocols. The Graphics Recognition community is one of the most healthy communities within the Pattern Recognition field regarding this aspect. A lot of works and efforts are centered in proposing evaluation methods which aim to track the progress in a certain specific problem. As far as we know, the works focused on polygon vectorization always have been evaluated by an ad-hoc set of measures or have been simply ignored. We

hope that the proposal of the performance evaluation protocol presented in chapter 5 can be used to evaluate other polygon vectorization methods and helps to track the progress on this topic as well as to identify the strengths and weaknesses of the proposed methods. However, one of the main problems is that we do not have any public dataset of real documents to test the proposed methods. Nowadays, the only available ground-truthed dataset which can be used to test polygons generated by raster to vector systems is our dataset. The main problem of this dataset is that it is composed only by one kind of documents which cannot reflect the entire reality, however it is the only one available and the community related to polygonization applications should start using it.

8.4 Open Challenges

Since document analysis and understanding is still an unsolved problem, we are convinced that there is still a lot of room for improvements and some open challenges.

- *User Interaction*
  - It is increasingly necessary to design analysis and recognition methods which do not work in stand-alone mode, but take into account the user's interaction, so as to be able to perform incremental learning, relevance feedback in recognition and retrieval applications, etc. But little work has been done on modeling the user, who is mostly considered as some kind of ill-defined, external entity. The fact is that there is nothing in common between a "vanilla plain" user who may be your uncle or grandma, browsing a collection of images and giving relevance feedback without really knowing anything about the application, and a highly specialized user able to input syntactical rules to represent the knowledge in a specific document analysis application. If the purpose is indeed to build a highly specialized system, this may not be a problem, but when the application is potentially very general whereas the user interaction paradigm requires the user to have a PhD in pattern recognition or to have trained for months, there is a contradiction in the whole setup which limits the applicability of the method.
- *Knowledge Driven Image Processing*
  - In our future lines of work, we think of a platform dedicated to the knowledge extraction and management for image processing applications. The aim of this platform is a knowledge-based system that adapts automatically its parameters from problem formulations given by inexperienced users. Such an platform must involve a model for the formulation of such applications. In the last fifty years, a lot of image processing applications have been developed in many fields (medicine, geography, robotic,

industrial vision, ...). We know that image processing specialists design applications by trial errors cycles. They do not enough reuse already developed solutions and design new ones nearly from scratch. The lack of application formulation modeling and formalization is a reason of this behavior. Indeed, image processing experts do not realize a complete and rigorous formulation of the applications. Only the solutions are used as their definitions. Therefore, the reusability of these applications is very poor because the limits of the solution applicability are not explicit. Moreover they often suffer from a lack of modularity and the parameters are also often tuned manually without giving explanations on the way they are set. On the contrary, in a not so far future, the user would define the problem with the terms of his/her domain by interaction with the user layer of the formulation system. This part of the system would be a human-machine interface which uses a domain ontology to handle the information dedicated to the user. It would group concepts that allow the users to formulate their processing intentions and define the image class. Then the formulation system would translates this user formulation into image processing terms taken from an image processing ontology. This translation would achieve the mapping between the phenomenological domain knowledge of the user and the image processing knowledge. The result of this translation could be an image processing request which would be sent to the managing system to modify the program that responds to this request. This cooperation would require the two sub-systems to share the image processing ontology. Then the formulation system would run the modified program on test images and would present the results to the user for evaluation purposes.

– The dream of building completely automated systems for converting drawings, maps into high-level representations seems to have vanished, as the methods we design reach their limits at a level where there is still a lot of user editing to be done. But there is still a very interesting opportunity to build combined retrieval/recognition systems, making it possible to navigate in a large document base by simple examples of what is being searched for.

Appendix A

Our publications

International Journal Under Revision (pending for results)

[1] Romain Raveaux, Sébastien Adam, Pierre Héroux and Eric Trupin. Learning Graph Prototypes for Shape Recognition. *Computer Vision and Image Understanding (CVIU)*, (Révision : 2nd round).

International Journals with Selection Committee

[2] Romain Raveaux, Jean-Marc Ogier and Jean-Christophe Burie. A Local Evaluation of Vectorized Documents by means of Polygon Assignments and Matching. *Journal: International Journal on Document Analysis and Recognition (IJDAR)*, (In press), 2010.

[3] Romain Raveaux, Jean-Christophe Burie and Jean-Marc Ogier. A graph matching method and a graph matching distance based on subgraph assignments. *Pattern Recognition Letters*, 31(5):394–406, 2010.

Proceedings of International Conferences With Selection Committee

[4] Romain Raveaux and Guillaume Hillairet. Model Driven Image Segmentation Using a Genetic Algorithm for Structured Data. In Manuel Grana Romay, Emilio Corchado, and Teresa Garcia-Sebastian, editors, *International Conference Hybrid Artificial Intelligence Systems, HAIS 2010*, pages 311–318. Springer, Lecture Notes in Computer Science, 2010.

[5] Romain Raveaux, Jean-Christophe Burie, and Jean-Marc Ogier. A colour text/graphics separation based on a graph representation. In *19th International Conference on Pattern Recognition (ICPR)*, pages 1–4, Washington, DC, USA, 2008. IEEE Computer Society.

[6] Romain Raveaux, J.-C. Burie, and J.-M. Ogier. A Colour Document Interpretation: Application to Ancient Cadastral Maps. In *ICDAR '07: Proceedings of the Ninth International Conference on Document Analysis and Recognition*, pages 1128–1132, Washington, DC, USA, 2007. IEEE Computer Society.

[7] Eugen Barbu, Romain Raveaux, Herve Locteau, Sebastien Adam, Pierre Heroux, and Eric Trupin. Graph Classification Using Genetic Algorithm and Graph Probing Application to Symbol Recognition. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 296—-299, Washington, DC, USA, 2006. IEEE Computer Society.

[8] Hervé Locteau, Romain Raveaux, Sébastien Adam, Yves Lecourtier, Pierre Héroux, and Éric Trupin. Approximation of Digital Curves using a Multi-Objective Genetic Algorithm. In *18th International Conference on Pattern Recognition (ICPR)*, pages 716–719, Washington, DC, USA, 2006. IEEE Computer Society.

International Book Chapters with Reading Committee

[9] Romain Raveaux, Jean-Christophe Burie, and Jean-Marc Ogier. *A Segmentation Scheme Based on a Multi-graph Representation: Application to Colour Cadastral Maps*, volume 5046 of *Lecture Notes in Computer Science*, pages 202–212. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[10] Herve Locteau, Romain Raveaux, Sebastien Adam, Yves Lecourtier, Pierre Heroux, and Eric Trupin. *Polygonal Approximation of Digital Curves Using a Multi-objective Genetic Algorithm*, volume 3926 of *Lecture Notes in Computer Science*, pages 300–311. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[11] Romain Raveaux, Barbu Eugen, Hervé Locteau, Sébastien Adam, Pierre Héroux, and Eric Trupin. *A Graph Classification Approach Using a Multi-objective Genetic Algorithm Application to Symbol Recognition*, volume 4538 of *Lecture Notes in Computer Science*, pages 361–370. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

International Workshops with Selection Committee

[12] Romain Raveaux, Jean-Christophe Burie, and Jean-Marc Ogier. Object Extraction from Colour Cadastral Maps. In *DAS '08: Proceedings of the 2008 The Eighth IAPR International Workshop on Document Analysis Systems*, pages 506—-514, Washington, DC, USA, 2008. IEEE Computer Society.

[13] Romain Raveaux, Jean-Christophe Burie, and Jean-Marc Ogier. A Colour Space Selection Scheme dedicated to Information Retrieval Tasks. In *Pattern Recognition in Information Systems, Proceedings of the 8th International Workshop on Pattern Recognition in Information Systems, PRIS 2008, In conjunction with ICEIS 2008, Barcelona, Spain, June 2008*, pages 123–134. INSTICC PRESS, 2008.

Our websites

- Alpage-L3i website: http://alpage-l3i.univ-lr.fr/
- My home page: http://romain.raveaux.free.fr/

Appendix B

Learning Graph Prototypes for Shape Recognition

Learning Graph Prototypes for Shape Recognition

Romain Raveaux[a], Sébastien Adam[b,*], Pierre Héroux[b], Éric Trupin[b]

[a] *Université de la Rochelle – L3I EA 2128, BP 12, 17042 La Rochelle cedex 01, FRANCE*
[b] *Université de Rouen – LITIS EA 4108, BP 12, 76801 Saint-Etienne du Rouvray, FRANCE*

Abstract

This paper presents some new approaches for computing graph prototypes in the context of the design of a structural nearest prototype classifier. Four kinds of prototypes are investigated and compared : *set median graphs*, *generalized median graphs*, *set discriminative graphs* and *generalized discriminative graphs*. They differ according to (i) the graph space where they are searched for and (ii) the objective function which is used for their computation. The first criterion allows to distinguish *set prototypes* which are selected in the initial graph training set from *generalized prototypes* which are generated in an infinite set of graphs. The second criterion allows to distinguish *median graphs* which minimize the sum of distances to all input graphs of a given class from *discriminative graphs*, which are computed using classification performance as criterion, taking into account the inter-class distribution. For each kind of prototype, the proposed approach allows to identify one or many prototypes per class, in order to manage the trade-off between the classification accuracy and the classification time.

Each graph prototype generation/selection is performed through a genetic algorithm which can be specialized to each case by setting the appropriate encoding scheme, fitness and genetic operators.

An experimental study performed on several graph databases shows the superiority of the generation approach over the selection one. On the other hand, discriminative prototypes outperform the generative ones. Moreover, we show that the classification rates are improved while the number of prototypes increases. Finally, we show that discriminative prototypes give better results than the median graph based classifier.

Keywords: Graph classification, graph prototypes, median graphs, discriminative graphs, genetic algorithm, symbol recognition

1. Introduction

Labeled graphs are powerful data structures for the representation of complex entities. In a graph-based representation, vertices and their labels describe objects (or part of objects) while labeled edges represent inter-relationships between the objects. Due to the inherent genericity of graph-based representations, and thanks to the improvement of computer capacities, structural representations have become more and more popular in many application domains such as computer vision, image understanding, biology, chemistry, text processing or pattern recognition. As a consequence of the emergence of graph-based representations, new computing issues such as graph mining (1; 2), graph clustering (3; 4) or supervised graph classification (5; 6; 7) provoked a growing interest.

This paper deals with the supervised graph classification problem. In the literature, this problem is generaly tackled using two kinds of approaches. The first one consists in using kernel based algorithms such as Support Vector Machines (SVM) or Kernel Principal Component Analysis (KPCA) (8; 9; 10; 11; 12; 13). Using such methods, the graph is embedded in a feature space composed of label sequences which are obtained through a graph traversal. The kernel values are then computed by measuring the similarity between label sequences. Such approaches have proven to achieve high performance but they are computationally expensive when the dataset is large. The second family consists in using a K-Nearest Neighbors (K-NN) rule in a dissimilarity space, using a given dissimilarity measure. This kind of approach is the most frequently chosen for

*Tel: (+33)2 32 95 52 10 Fax: (+33)2 32 95 52 10
Email addresses: Romain.Raveaux@univ-lr.fr (Romain Raveaux), Sebastien.Adam@univ-rouen.fr (Sébastien Adam), Pierre.Heroux@univ-rouen.fr (Pierre Héroux), Eric.Trupin@univ-rouen.fr (Éric Trupin)

its simplicity to implement and its good asymptotic behavior. However, it suffers from three major drawbacks: its combinatorial complexity, its large storage requirements and its sensitivity to noisy examples. A classical solution to overcome these problems consists in reducing the learning dataset through an object prototype learning procedure and to use a Nearest Prototype Classifier (NPC). Such a prototype-based strategy is not inherent to the graph classification problem. It has already been tackled for comparing shapes in computer vision application, e.g. in the approach described in (14) that learns some contour prototypes. It has also been studied for a long time in the context of statistical pattern recognition, using either prototype selection methods (see e.g (15; 16)) or prototype generation methods (see e.g. (17; 18)).

In the field of structural pattern recognition, there also has been some recent efforts dedicated to the learning of prototypes. Among them, one can cite the pioneering approach proposed in (19) which builds prototypes by detecting subgraphs that occur in most graphs. Another approach concerning trees is proposed in (20). It consists in learning some kinds of tree prototypes through the definition of a superstructure called tree-union that captures the information about the tree training set. In the domain of graphs, the approaches proposed in (21) and (22) aim at creating super-graph representations from the available samples. One can also cite the interesting work of Marini proposed in (23) that generates some *creative prototype* by applying to a seed model a well selected set of editing operation. A last approach which is probably the most frequently used concerns median graphs (24; 25; 26; 27; 28). In a classification context, median graphs are computed independently in each class through a minimization process of the sum of distances to all input graphs. Two kinds of median graphs are proposed in the literature: the set median graphs (*smg*) and the generalized median graphs (*gmg*). The only difference between them lies in the space where the medians are searched for. In the first case, the search space is limited to the initial set of graphs (the problem is thus a graph prototype selection problem) whereas in the second case, medians are searched among an infinite set of graphs built using the labels of the initial set (the problem is thus a graph prototype generation problem). Generalized median graphs approaches have proven to keep the most important information in the classes and reject noisy examples (25). However, a drawback of median graphs when they are used as learning samples of a classification process, as for the all the approaches mentionned before, is that they do not take into account the inter-classes data distribution. In other words, median graphs are rather generative prototypes than discriminative ones.

In this paper, we overcome this drawback by using a discriminative approach while searching an optimal set of prototypes. Thus, it is the classification performance obtained on a validation dataset which is used as criterion in the prototype optimization process. Hence, we propose to use a graph based Genetic Algorithm in order to learn a set of graph prototypes, called discriminative graphs (*dg*), which minimize the error rate of a classification system. Two configurations are successively considered for extracting the discriminative graphs. In the first one, a single prototype is generated for each class of the classification problem, as in the case of median graphs. Then, this concept is extended to the extraction of multiple prototypes for each class in order to obtain a better description of the data. This extension is also considered in the case of median graphs in order to provide a suitable comparison. In both configurations, we show that discriminative graphs, and particularly multiple discriminative graphs, enable to obtain very good classification results while considerably reducing the number of dissimilarity computations in the decision stage.

Four datasets are used in the experimental protocol. The first is a huge synthetic dataset. The others are real-world datasets consisting of graphs built from a graphical symbol recognition benchmark (29) for the second and the third and from character recognition for the fourth. The classification performance obtained using discriminative graphs and median graphs are compared on these four datasets.

The paper is organized as follows. In section 2, the most important concepts and notations concerning median graphs and discriminative graphs are defined. In section 3, the proposed approach for graph prototypes extraction is detailed. Section 4 describes the experimental evaluation of the algorithm and discusses results. Finally, section 5 offers some conclusions and suggests directions for future works.

2. Definitions and notations

In this work, the problem which is considered concerns the supervised classification of directed labeled graphs. Such graphs can be defined as follows:

Definition 1. A directed labeled graph G is a *4-tuple* $G = (V, E, \mu, \xi)$ where:

- V is the set of vertices,
- $E \subseteq V \times V$ is the set of edges,

- $\mu : V \rightarrow L_V$ is a function assigning a label to a vertex,
- $\xi : E \rightarrow L_E$ is a function assigning a label to an edge.

A graph classification algorithm aims at assigning a class to an unknown graph using a mapping function f. This function is usually induced from a learning stage which can be defined as follows:

Definition 2. Let χ be the set of the labeled graphs. Given a graph learning dataset $L = \{\langle g_i, c_i \rangle\}_{i=1}^{M}$, where $g_i \in \chi$ is a labeled graph and $c_i \in C$ is the class of the graph among the N classes. The learning of a graph classifier consists in inducing from L a mapping function $f(g) : \chi \rightarrow C$ which assigns a class to an unknown graph.

In this paper, graph classification is tackled with a Nearest Prototype Classifier (NPC), *i.e.* with a NN rule applied on a reduced set of representative graph prototypes. Hence, the learning stage of the classifier consists in generating these prototypes. The objectives are (i) to overcome the well-known disadvantages of a K-NN procedure, *i.e.* the large storage requirements, the large computational effort and the sensitivity to noisy examples and (ii) to keep classification performance as high as possible.

As mentioned before, median graphs are frequently used as representative in a graph classification context. Two kinds of median graphs may be distinguished: the set median graph *smg* and the generalized median graph *gmg*. Both are based on the minimization of the sum of distances (SOD) to all input graphs. Formally, they are defined as follows:

Definition 3. Let $d(.,.)$ be a distance or a dissimilarity function that measures the dissimilarity between two graphs. Let $S = \{g_1, g_2, \ldots, g_n\}$ be a set of graphs. The set median graph (*smg*) of S is defined by:

$$smg = \arg\min_{g \in S} \sum_{i=1}^{n} d(g, g_i) \tag{1}$$

According to this definition, *smg* necessarily belongs to the set S. This definition has been extended in (25) to the generalized median graph (*gmg*) which does not necessarily belong to S:

Definition 4. Let $d(.,.)$ be a distance or a dissimilarity function that measures the dissimilarity between two graphs. Let $S = \{g_1, g_2, \ldots, g_n\}$ be a set of graphs. Let U be the infinite set of graphs that can be built using the labels of S. The generalized median graph (*gmg*) of the subset S is defined by:

$$gmg = \arg\min_{g \in U} \sum_{i=1}^{n} d(g, g_i) \tag{2}$$

Median graphs, generalized or not, have already been used as class representatives in a classification process, e.g. in (25; 26; 27). In this case, if N is the number of classes in the learning dataset L, N *smg* (resp. *gmg*) are computed independently (one for each class) and the resulting graph set constitutes the learning dataset $SMG = \{smg_i\}_{i=1}^{N}$ (resp. $GMG = \{gmg_i\}_{i=1}^{N}$) of the nearest prototype classifier. It has been shown in (25) that generalized median graphs capture the essential information of a given class. However, such prototypes do not take into account the inter-class distribution of learning samples.

In order to overcome this problem, we propose to use discriminative graphs (*dg*) as prototypes for graph classification. The main difference between median graphs and discriminative graphs lies in the criterion which is used to generate the prototypes. In the case of *dg*, rather than optimizing a sum of intra-class distances, prototypes are generated in order to minimize the classification error rate obtained on a validation dataset. Obviously, as in the case of median graphs, these prototypes can be computed in the initial set of graphs, leading to set discriminative graphs (*sdg*), or in the whole set of graphs, leading to generalized discriminative graphs (*gdg*). As a consequence, the *dg* for each class are related to each other and can not be expressed independently. The set SDG of sdg_i can be defined as follows:

Definition 5. Let N be the number of classes in the learning dataset L. Let T be a validation dataset and let $\Delta\left(T, \{g_i\}_{i=1}^{N}\right)$ be a function computing the error rate obtained by a 1-NN classifier on T using the graph prototypes $\{g_i\}_{i=1}^{N} \in L$ as learning samples. Then the set SDG composed of the sdg_i of each class is given by:

$$\begin{aligned} SDG &= \{sdg_1, sdg_2, \ldots, sdg_N\} \\ &= \arg\min_{\{g_i\}_{i=1}^{N} \subset L} \Delta(T, \{g_i\}_{i=1}^{N}) \end{aligned} \tag{3}$$

In the same way, the set GDG of *gdg* is defined as follows:

Definition 6. Let N be the number of classes in the learning dataset L. Let U be the infinite set of graphs that can be built using labels from L. Let T be a validation dataset and let $\Delta\left(T, \{g_i\}_{i=1}^{N}\right)$ be the error rate obtained by a 1-NN classifier on T using the graph prototypes $\{g_i\}_{i=1}^{N} \in U$ as learning samples. Then the set GDG composed of the *gdg* of each class is given by:

$$GDG = \{gdg_1, gdg_2, \ldots, gdg_N\} = \arg\min_{\{g_i\}_{i=1}^{N} \subset U} \Delta(T, \{g_i\}_{i=1}^{N}) \quad (4)$$

The concepts presented above involve the generation of a single prototype for each class. In some particular applications, it may be interesting to generate m prototypes for each class in order to obtain a better description of the data. In the following, we give the definition of such prototypes called m-gdg[1].

Definition 7. Let N be the number of classes in the learning dataset L. Let U be the infinite set of graphs that can be built using labels from L. Let m be the number of prototypes to be computed in each class. Let T be a validation dataset and let $\Delta\left(T, \{g_{ik}\}_{i=1,k=1}^{N,m}\right)$ be the error rate obtained by a 1-NN classifier[2] on T using the graph prototypes $\{g_{ik}\}_{i=1,k=1}^{N,m} \in U$ as learning samples. Then the set $mGDG$ composed of the m-gdg of each class is given by:

$$mGDG = \{gdg_{11}, ..., gdg_{1m}, ..., gdg_{N1}, ..., gdg_{Nm}\} = \arg\min_{\{g_{ik}\}_{i=1,k=1}^{N,m} \subset U} \Delta\left(T, \{g_{ik}\}_{i=1,k=1}^{N,m}\right) \quad (5)$$

In order to provide some fair comparisons in the experimental protocol, we also extend the median graph concept to multiple prototypes. In this case, the m-gmg (as well the m-smg) are defined independently for each class :

Definition 8. Let $d(.,.)$ be a distance or a dissimilarity function that measures the dissimilarity between two graphs. Let n be the number of samples in the considered class. Let m be the number of prototypes, gp_k be the prototypes and g_i be the graphs of the considered class. Then, the set $mGMG$ composed of the m-gmg for the considered class is given by :

$$mGMG = \{gmg_1, \ldots, gmg_m\} = \arg\min_{\{gp_k\}_{k=1}^{m} \subset U} \sum_{i=1}^{n} \min_{k \in \{1,m\}} d(gp_k, g_i) \quad (6)$$

The algorithms involved in the computation of the different kinds of representative prototypes are presented in the following section.

[1] the definition of m-sdg is easily obtained through the change of the search space from U to S.

[2] In this case, a k-NN procedure with $k > 1$ will be considered in future works, for example to allow the system to reject some patterns

3. Genetic algorithms for Graph Prototypes Generation

In section 2, the graph prototype search problem has been defined as an optimization process. Two kinds of prototypes have been distinguished: (i) set prototypes and (ii) generalized prototypes.

(i) The set prototype search problem consists in selecting the m prototypes per class which optimize an objective function. A combinatorial exploration of the solution space would result in evaluating the criterion for each of the potential solutions. If we consider that each of the N classes contains n_i elements, there are

$$\begin{pmatrix} m \\ n_1 \end{pmatrix} \times \begin{pmatrix} m \\ n_2 \end{pmatrix} \times \cdots \times \begin{pmatrix} m \\ n_N \end{pmatrix} \quad (7)$$

combinations for selecting m prototypes to represent each class. For a quite simple problem with 2 classes and 100 graphs in each class, the search for 5 prototypes per class would result in more than 75×10^6 evaluations of the criterion. Hence, a complete exploration of the solution space rapidly becomes intractable. Many heuristic methods such as multistart, genetic algorithms or tabu search (18) have been used to tackle the problem of set prototype search when dealing with vectorial data. Among them, genetic based methods have shown good performance (31; 18).

(ii) The generalized prototype search problem can also be stated as an optimization problem. However, it cannot be solved with a combinatorial approach since the set U in which the solutions are searched for is unbounded (only a subset S of U is known). In (24), the authors use genetic algorithms to approximate the generalized median graph of a set of graphs. In the context of computing a single generative prototype, they report that the solution reached by a genetic approach is often the optimal solution. In this paper, we also propose to use genetic algorithms but to solve both the set/generalized median/discriminative prototype extraction problem. The next subsections precisely describe our approach.

3.1. Genetic Algorithm

Genetic Algorithms (GA) are evolutionary optimization techniques with a wide scope of applications (32). They have been used to solve many combinatorial problems (33). An individual of a GA corresponds to a possible solution of an optimization problem. The relationship between this individual and the corresponding solution is given by an appropriate encoding. The quality of each individual is evaluated thanks to a score function

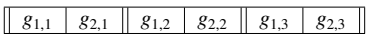


Figure 1: General encoding scheme for the m prototypes problem. Each individual contains $m \times N$ genes. Each one corresponds to a graph prototype.

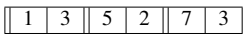


Figure 2: Set prototype encoding scheme for the m prototypes problem. Each individual contains $m \times N$ genes. Each gene is the index of the graph in the learning dataset.MODIFS A FAIRE

which enables to quantify the quality of the corresponding solution. In order to converge to the optimal solution, individuals from a size-limited population are randomly selected at each generation according to a fitness value which is computed using the scores of all the individuals of the population. New individuals are then generated from those selected individuals thanks to genetic operators such as crossover or mutation. From a general point of view, the crossover operator aims at promoting the exchange of *good* genetic material between individuals of the previous generation. The mutation operator is used to promote genetic diversity and to explore the solution space. Given these general principles, solving a specific optimization problem using GA requires the definition of :

- an appropriate encoding of the solutions;
- a function which evaluates the score of the individual;
- a selection strategy ;
- some dedicated genetic operators (mutation and crossover operators)

The following paragraphs tackle each of these points for both graph prototype selection and generation, and describe the proposed genetic algorithm.

3.2. Individual encoding

The encoding aims at giving a one-to-one relationship between the individuals manipulated by the GA and the solutions of the optimization problem. As defined before, the prototype selection/generation problem aims at providing m prototypes for each of the N classes. So, we adopt a general scheme where an individual contains $m \times N$ genes, and each gene encode a graph prototype. An example is given in Fig. 1. In this example, the individual encodes 2 prototypes for each class in a 3 classes problem and $g_{i,j}$ is the i^{th} graph prototype describing class j. Obviously, this encoding is specialized for each problem.

3.2.1. Set prototype problem encoding

As stated in section 2, the possible solutions of a set prototype problem are the combinations of m elements selected from each class in the initial graph set. For this kind of problem, an individual can be defined by a list of $N \times m$ integers which is structured as a sequence of N m-sets. Each m-set describes one of the N classes and contains the m indices of the elements from the initial set which are selected as prototype. The exemple in Fig. 2 presents the encoding of an individual for a 3-class problem where 2 prototypes are selected to describe each class. This individual indicates that class 1 is described with elements 1 and 3 of a learning subset composed of the graphs of the first class, that class 2 is described with elements 5 and 2 of the class, and that class 3 is described with graphs the indices of which are 7 and 3 in the third class subset.

3.2.2. Generalized prototype problem encoding

The index model used in the set prototype problem can not be used for the solution encoding of the generalized prototype problem since the definition of generalized (median and discriminative) graphs implies that prototypes may be outside of the initial set of graphs. As a consequence, each gene of an individual can not be a *simple* index and has to encode all the information contained in the corresponding graph. We have chosen to represent each graph with its adjacency matrix. Hence, an individual can be defined by a list of $N \times m$ adjacency matrices, structured as a sequence of N m-sets. Fig. 3 illustrates such an encoding where only one of the 6 genes is represented.

3.3. Fitness function

A fitness function aims at evaluating how the solution encoded by an individual is good for the optimization problem with respect to the entire population. The computation of a fitness value relies on two steps. First, the score of the individual has to be evaluated. It corresponds to the value of the objective function to be optimized. Then, this value is normalized with respect to the scores of all the individuals of the population. As mentioned in section 2, objectives are different for the median prototype problem and for the discriminative prototype problem. As a consequence, score functions differ for each problem.

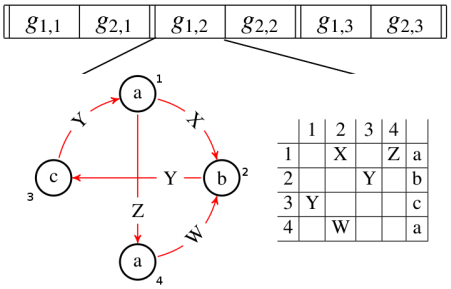


Figure 3: Generalized prototype encoding scheme for the m prototypes problem. Each individual contains $m \times N$ genes. Each gene is an adjacency matrix describing the corresponding graph. Only $g_{1,2}$ is represented here. In the adjacency matrix, the digits state for vertex identifiers. a, b, and c are vertices labels, they appear in the last column of the matrix. W, X and Y are edge labels, they appear in the adjacency matrix at the line (resp. column) corresponding to the source (resp. target) vertex.

3.3.1. Score function for median prototypes

As defined in section 2, the score function in the median prototype problem is given by :

$$S_\alpha = \sum_{i=1}^{N} \left(\sum_{j=1}^{n_i} \min_{k \in [1,m]} d\left(L_{ij}, smg_{ik}\right) \right) \tag{8}$$

where N is the number of classes, n_i is the number of elements of class i in the learning dataset, m is the number of prototypes per class, L_{ij} is the j^{th} sample of class i, and smg_{ik} is the k^{th} prototype of class i in the individual α.

3.3.2. Score function for discriminative prototypes

The score value of an individual in the discriminative prototype problem is a function which is directly linked to the error rate of the Nearest Prototype Classifier evaluated on a validation dataset T using the prototypes encoded in the individual. It is given by :

$$S_\alpha = \Delta\left(T, \{g_{ik}\}_{i=1,k=1}^{N,m}\right) \tag{9}$$

where T is the validation dataset, N is the number of classes, m is the number of prototypes per class, g_{ik} is the k^{th} prototype of class i in the individual and $\Delta\left(T, \{g_{ik}\}_{i=1,k=1}^{N,m}\right)$ is the error rate obtained by a 1-NN classifier on T using the graph prototypes $\{g_{ik}\}_{i=1,k=1}^{N,m}$ as learning samples.

The computation of both the Δ value of eq. 9 and the S_α value of eq. 8 make use of graph distance computation. The following paragraph discusses our choice for this distance definition.

3.3.3. Distance computation

Any kind of distance can be used in the proposed framework (graph edit distance (34; 35) or its approximations (36), distance based on the maximum common subgraph (37), distance based on graph union (38)...). In the experiments proposed in section 4, the graph comparison computation is performed using a dissimilarity measure proposed by Lopresti and Wilfong (39). This measure is based on graph probing which has been proved to be a lower bound for the reference graph edit distance within a factor of 4.

Let g be a directed attributed graph with edges labeled from a finite set $L_E = \{l_1, \ldots, l_a\}$. A given vertex of g can be represented with its edge structure as a $2a$-tuple of non-negative integers $\{x_1, \ldots, x_a, y_1, \ldots, y_a\}$ such that the vertex has exactly x_i incoming edges labeled l_i and y_j outgoing edges labeled l_j.

In this context, two types of probes are defined in (39):

- $P_1(g)$: a vector which gathers the counts of vertices sharing the same edge structure for all encountered edge structures ;
- $P_2(g)$: a vector which gathers the number of vertices for each vertex label.

Based on these probes and on the L_1-norm, the graph probing distance between two graphs g_1 and g_2 is given by :

$$\begin{aligned} gpd\left(g_1, g_2\right) &= L_1\left(P_1\left(g_1\right), P_1\left(g_2\right)\right) \\ &\quad + L_1\left(P_2\left(g_1\right), P_2\left(g_2\right)\right) \end{aligned} \tag{10}$$

The graph probing distance respects the non-negativity, symmetry, and triangle inequality properties of a metric, but it does not respect the uniqueness property. In other words, gpd is a pseudo-metric and two non-isomorphic graphs can have the same probes.

However, the main advantage of graph probing in this study is its low computational cost (linear function of the vertex number). Due to the intensive use of distance computations during the genetic algorithm, this property makes the graph probing distance a good candidate. Nevertheless, it is important to note that any kind of dissimilarity measure may be used in the proposed framework.

3.3.4. Fitness computation

Once the score value of an individual has been computed, a second step of individual evaluation consists in computing its fitness, through a normalization of the

$g_{1,1}$	$g_{2,1}$	$g_{1,2}$	$g_{2,2}$	$g_{1,3}$	$g_{2,3}$
$g'_{1,1}$	$g'_{2,1}$	$g'_{1,2}$	$g'_{2,2}$	$g'_{1,3}$	$g'_{2,3}$

(a) Pair of individuals selected for the crossover operation : the parents

$\mathbf{g'_{1,1}}$	$g_{2,1}$	$\mathbf{g'_{1,2}}$	$\mathbf{g'_{2,2}}$	$g_{1,3}$	$g_{2,3}$
$\mathbf{g_{1,1}}$	$g'_{2,1}$	$\mathbf{g_{1,2}}$	$\mathbf{g_{2,2}}$	$g'_{1,3}$	$g'_{2,3}$

(b) Pair of children generated by the crossover operation.

Figure 4: Illustration of the crossover operator : two selected parents (a) generate two offsprings (b). Genes 1,3 and 4 have been swapped during the operation

score value with respect to all the individuals of the population. We use the following classical fitness assignment procedure in this scope:

$$F_\alpha = \frac{S_\alpha}{\sum_{i=1}^{\rho} S_i} \tag{11}$$

3.4. Selection strategy

The selection operator aims at selecting a proportion of the existing population to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by the fitness function defined in eq. 11) are typically more likely to be selected. We use the well known roulette wheel strategy (32) in which the probability of an individual to be selected is proportional to its fitness value. In the whole reproduction process, an elitism mechanism is coupled with this selection strategy such that the μ best individuals from the previous generation are ensured to be in the next generation.

3.5. Crossover

As mentioned before, the crossover operator is designed to generate offsprings from selected individuals. The exchange of genetic material aims at generating offsprings sharing *good* genes from their parents.

In our case, the crossover is performed by a random exchange of prototypes between the parent for each class. Fig. 4 illustrates the crossover operation. The operation is the same for all the kinds of prototypes. In the case of set prototypes, where the graphs prototypes are designated by indices, only indices are permuted whereas the complete graph descriptions are exchanged when dealing with the generalized prototype problem.

3.6. Mutation

Mutations are used to promote genetic diversity and allow the exploration of regions of the solution space which can not be reached only with crossover. As the solution space is different for set prototype and generalized prototype problems, the mutation operator has to be specialized for each case.

1	3	5	2	7	3

(a) individual selected for mutation

1	**4**	**6**	2	7	**5**

(b) individual resulting from the mutation operation

Figure 5: Illustration of the mutation operator for set prototypes : genes 2,3 and 6 have mutated

3.6.1. Mutation for set prototype problem

In the set prototype problem, the solution space is defined by the combinations allowing the selection of m prototypes for each class. An elementary modification of an individual would consist in replacing a prototype by an element from the same class that is not already selected in the individual. Hence, considering the index model used to represent graphs, a simple way to perform a mutation is to arbitrarily substitute an index values by a random integer. Fig. 5 illustrates the mutation process. In this example, we can observe that element 3 has been replaced by element 4 in the mutated version of the description of class 1. In the same way, instance 5 has been replaced by instance 6 in the description of class 2. Finally, the mutated version describes class 3 using the element 5 instead of element 3.

3.6.2. Mutation for the generalized prototype problem

In the generalized prototype problem, the solution space is not restricted to the combinations of elements selected in L. Graphs that are not element of L can be generated as prototypes. As a consequence, the mutation operation can not be restricted to an index modification. It must be able to produce new graphs. To do this, a random edit operation is applied to the graph prototypes that are included in the individual. For each graph of a given individual, a first random choice according to a mutation probability enables to decide if a mutation is applied or not. Then, one of the six following possible operations illustrated on Fig. 6 is chosen randomly :

- Vertex deletion : delete a randomly chosen vertex and all its connected edges. This operation corresponds to the deletion of a row and a column in the adjacency matrix (see Fig. 6(b)).
- Edge deletion : delete a randomly chosen edge. This operation corresponds to the deletion of an

edge value in the adjacency matrix (see Fig. 6(c)).

- Vertex insertion : insert a new vertex in the graph with a randomly chosen label among the vertex label dictionary. This operation corresponds to the addition of a new row and a new column in the adjacency matrix. The label column is also updated using the randomly chosen label (see Fig. 6(d)).
- Edge insertion : insert a new edge between two random vertices with a randomly chosen label among the edge label dictionary. This operation corresponds to the addition of a randomly labeled edge in the adjacency matrix (see Fig. 6(e)).
- Vertex substitution : substitue the label of a randomly chosen vertex using the vertex label dictionary. This operation corresponds to the modification of the label column for the randomly chosen vertex(see Fig. 6(f)).
- Edge substitution : substitue the label of a randomly chosen edge using the edge label dictionary. This operation corresponds to the modification of the label for the randomly chosen edge (see Fig. 6(g)).

3.7. Proposed algorithm

Alg. 1 gives the generic structure of the GA used for the graph prototype generation/selection problems. This algorithm complies with the principles defined in section 3.1 and is specialized by setting the adapted encoding, fitness function and genetic operators presented previously.

First, an initialization procedure aims at building the initial population where each individual corresponds to a possible solution of the optimization problem. In the case of set prototypes, distinct indices are randomly chosen for each individual in order to represent the N classes with $N \times m$ graphs. For generalized prototypes, we have chosen to initialize the individuals with randomly chosen graphs from the learning dataset, since it has been shown in (24) that it is a better solution than a complete random procedure.

Then, the GA iterates over the generations, building new size-limited populations from the previous ones. Each new generation is composed of:

- the μ best individuals from the previous one. Such an elitist strategy ensures the convergence of the algorithm.
- mutated or crossed version of individuals that have been selected from the previous generation.

Finally, the algorithm provides the best individual from the last generation as the best solution of the optimization procedure.

Algorithm 1 Genetic algorithm

Require: L: the training set
Require: T: the validation set
Require: m: number of prototypes per class
Require: populationSize
Require: generationNumber
Require: mutationRate
Require: μ: elitism value
Ensure: A set of $N \times m$ prototypes
  Pop[0][] ← popInit(L,T,m,populationSize) [1]
  popEval(Pop[0],L,T)
  fitnessEval(Pop[0])
  for i = 1 to generationNumber **do**
    Pop[i][1 : μ] ← μ best individuals in Pop[$i-1$]
    $j \leftarrow \mu + 1$
    while $j \leq populationSize$ **do**
      op ← choice between mutation and crossover [2]
      if op = mutation **then**
        ind ← select an individual in Pop[$i-1$] [3]
        Pop[i][j] ← mutation(ind)
        $j \leftarrow j + 1$
      else
        ind_1 ← select an individual in Pop[$i-1$] [3]
        ind_2 ← select an individual in Pop[$i-1$] [3]
        ($newInd_1, newInd_2$) ← crossover(ind_1, ind_2)
        Pop[i][j] ← ind_1
        Pop[i][$j+1$] ← ind_2
        $j \leftarrow j + 2$
      end if
      popEval(Pop[i],L,T)
      fitnessEval(Pop[i])
    end while
  end for
  return the best individual from the last generation

[1] T is not used for the initialization in the case of discriminative graphs
[2] This choice is made according to *mutationRate*
[3] Selection is done using a roulette wheel according to fitness values

4. Experimental results and analysis

This section is devoted to the experimental evaluation of the proposed approach. First, both the datasets and the experimental protocol are described before investigating and discussing the merits of the proposed approach.

Appendix C

Progos Ontology Generator

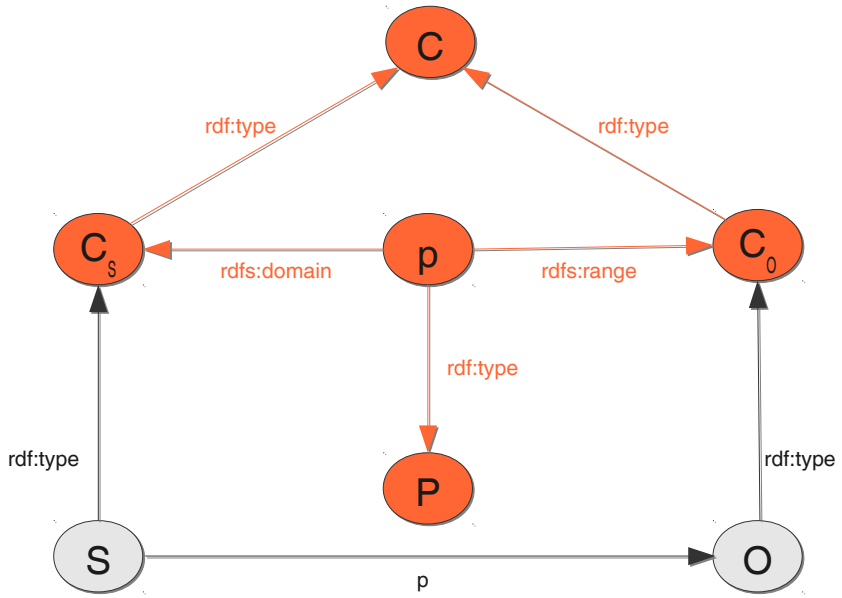
Progos Ontology Generator

Overview

The Ontology Generator generates ontology statements we can infer from an RDF document. It generates RDFS, OWL and DAML ontologies.

The anatomy of a statement

The following picture illustrates a statement, the optional type information of its subject and object and the ontology facts we can infer from it. The document space is drawn in black, the ontology space is in orange.



Suppose the document contains the following statement:

S p O (s_1)

where S is the subject, p is the predicate and O is the object of the statement. The

Progos Ontology Generator

document can contain optional information about the types of the resources we talk about:

$$S\ \text{rdf:type}\ C_s \qquad (s_2)$$

$$O\ \text{rdf:type}\ C_o \qquad (s_3)$$

Let's see what can we infer from these statements. From (s_1) we infer that p is a Property:

$$p\ \text{rdf:type}\ \text{rdf:Property} \qquad (o_1)$$

If the document contains (s_2), we also know the domain of p:

$$p\ \text{rdfs:domain}\ C_s \qquad (o_2)$$

If does not, we still know that

$$p\ \text{rdfs:domain}\ \text{rdfs:Resource} \qquad (o_3)$$

If the document contains (s_3), we can infer:

$$p\ \text{rdfs:range}\ C_o \qquad (o_4)$$

If does not, we still know that

$$p\ \text{rdfs:range}\ \text{rdfs:Resource} \qquad (o_5)$$

Or if o is a literal with a given dt datatype:

$$p\ \text{rdfs:range}\ dt \qquad (o_6)$$

In case of an untyped object:

$$p\ \text{rdfs:range}\ \text{rdfs:Literal} \qquad (o_7)$$

Finally we know that C_o and C_s are classes:

$$C_s\ \text{rdfs:type}\ \text{rdfs:Class} \qquad (o_8)$$

$$C_o\ \text{rdfs:type}\ \text{rdfs:Class} \qquad (o_9)$$

Progos Ontology Generator

OWL generation

If we generate OWL ontology, we generate some additional statements. Instead of (o_1) we generate

p rdf:type owl:ObjectProperty (o_{10})

or in case of typed literal object

p rdf:type owl:DatatypeProperty (o_{11})

depending the type. In case of untyped literal object we generate (o_9) but it may be incorrect.

In place of (o_3), we say

p rdfs:domain owl:Thing (o_{12})

We know that C_o and C_s are owl classes or datatypes:

C_s rdfs:type owl:Class (o_{13})

C_o rdfs:type owl:Class (o_{14})

or

C_o rdfs:type owl:Datatype (o_{15})

DAML ontology

Since DAML ontology has the same structure as OWL and they differ only in the namespace, we do not discuss DAML generation here.

Bibliography

[A. Pentland R.W. Picard 1996] S Scaroff A. Pentland R.W. Picard. *Photobook: content-based manipulation for image databases.* International Journal on Computer Vision, vol. 18 (3), pages 233–254, 1996. 194

[A.D.J. Cross R.C. Wilson 1997] E R Hancock A.D.J. Cross R.C. Wilson. *Inexact Graph Matching Using Genetic Search.* Pattern Recognition, vol. 30, pages pp. 953–970, 1997. 167

[Aiello 2004] Marco Aiello and Arnold M W Smeulders. *Thick 2D relations for document understanding.* Inf. Sci. Inf. Comput. Sci., vol. 167, no. 1-4, pages 147–176, 2004. 220, 227

[A.K.C. Wong 1985] M You A.K.C. Wong. *Entropy and Distance of Random Graphs with Application to Structural Pattern Recognition.* IEEE Transaction in Pattern Analysis and Machine Intelligence, vol. 7, pages 599–609, 1985. 166

[A.M. Finch R.C. Wilson 1997] E R Hancock A.M. Finch R.C. Wilson. *Matching Delaunay Graphs.* Pattern Recognition, vol. 30, pages 123–140, 1997. 166

[Amit 1996] Yali Amit, Geman August and Donald Geman. *Shape Quantization and Recognition with Randomized Trees.* Neural Computation, vol. 9, pages 1545–1588, 1996. 200

[Anselm Blumer Andrzej Ehrenfeucht 1987] David Haussler Anselm Blumer Andrzej Ehrenfeucht and Manfred K.Warmuth. *Occam's razor.* Information Processing Letters, vol. 24(6), pages 377–380, 1987. 36

[Arkin 1991] E M Arkin, L P Chew, D P Huttenlocher, K Kedem and J S B Mitchell. *An Efficiently Computable Metric for Comparing Polygonal Shapes.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13(3), pages 209–216, 1991. 136

[Arun Hampapur Ramesh Jain 1995] Arun Hampapur Ramesh Jain and Terry Weymouth. *Production model based digital video segmentation.* Journal of Multimedia Tools and Applications, pages 1–38, 1995. 205

[Auwatanamongkol 2007] Surapong Auwatanamongkol. *Inexact graph matching using a genetic algorithm for image recognition.* Pattern Recognition Letters, vol. 28, no. 12, pages 1428–1437, 2007. 166

[Barnard 2001] K Barnard and D Forsyth. *Learning the semantics of words and pictures.* IEEE International Conference on Computer Vision (ICCV), vol. 2, pages 408–415, 2001. 197

[BELAID A. 1992] TOMBRE K BELAID A. *Analyse de documents : de l'image à la sémantique.* Actes de CNED, vol. 80, pages pp. 3–29., 1992. 7

[Bengoetxea 2002] Endika Bengoetxea, Pedro Larrañaga, Isabelle Bloch, Aymeric Perchant and Claudia Boeres. *Inexact graph matching by means of estimation of distribution algorithms.* Pattern Recognition, vol. 35, no. 12, pages 2867–2880, 2002. 166

[Benz U.C. 2003] Hofmann P Willhauck G Lingenfelder I Heynen M Benz U.C. *Multi-resolution, Object-oriented Fuzzy Analysis of Remote Sensing Data for GIS Ready Information.* International Workshop on Semantic Processing of Spatial Data (GEOPRO 2003), pages 110–126, 2003. 12

[BERRETTI 2003] Stefano BERRETTI, Alberto DEL BIMBO and Enrico VICARIO. *Weighted walkthroughs between extended entities for retrieval by spatial arrangement.* IEEE transactions on multimedia, vol. 5, no. 1, pages 52–70, 2003. 196

[Bézivin 2005] J Bézivin. *On the unification power of models.* Software and Systems Modeling, vol. 4, no. 2, pages 171–188, 2005. 89

[Bloehdorn 2005] Stephan Bloehdorn, Kosmas Petridis, Carsten Saathoff, Nikos Simou, Yannis Avrithis, Siegfried H, Yiannis Kompatsiaris and Michael G Strintzis. *Semantic annotation of images and videos for multimedia analysis.* In Proceedings of the 2nd European Semantic Web Conference, ESWC 2005, pages 592–607, 2005. 55

[Boatto 1992] L Boatto. *An interpretation system for land register maps.* IEEE Computer, vol. 25(7), pages 25–32, 1992. 17, 55, 116

[BOMBARDIER] Vincent BOMBARDIER, Pascal LHOSTE and Cyril MAZAUD. *Modélisation et intégration de connaissances métier pour l'identification de défauts par règles linguistiques floues = Expert knowledge's modeling and expert knowledge's integration for defect identification by fuzzy linguistic rules.* TS. Traitement du signal, vol. 21, no. 3, pages 227–247. 55

[Bonabeau 2002] Eric Bonabeau. *Graph multidimensional scaling with self-organizing maps.* Information Science, vol. 143, no. 1-4, pages 159–180, 2002. 169, 208

[Borgwardt 2005] Karsten M Borgwardt and Hans-Peter Kriegel. *Shortest-Path Kernels on Graphs.* IEEE International Conference on Data Mining, pages 74–81, 2005. 169, 209

[Bourgeois 1971] F Bourgeois and J Lassalle. *An extension of the Munkres algorithm for the assignment problem to rectangular matrices.* Communications of the ACM, vol. 14(12), pages 802–804, 1971. 124

[Brady 1993] M Brady. *Criteria for Representations and of shape.* Human and Machine Vision Academic, pages 39–84, 1993. 194

[Bunke 1983] H Bunke and G Allermann. *Inexact graph matching for structural pattern recognition.* Pattern Recognition Letters, vol. 1, pages 245–253, 1983. 136

[Bunke 1997] H Bunke. *On a relation between graph edit distance and maximum common subgraph.* Pattern Recognition Letters, vol. 18, no. 9, pages 689–694, 1997. 166

[Bunke 1998a] Horst Bunke. *Error-Tolerant Graph Matching: A Formal Framework and Algorithms.* In Adnan Amin, Dov Dori, Pavel Pudil and Herbert Freeman, editeurs, SSPR/SPR, pages 1–14. Springer, Lecture Notes in Computer Science, 1998. 165

[Bunke 1998b] Horst Bunke and Kim Shearer. *A graph distance metric based on the maximal common subgraph.* Pattern Recognition Letters, vol. 19, no. 3-4, pages 255–259, 1998. 170, 171, 173

[Byrnes 1997] David Byrnes. *Raster-to-vector comes of age with AutoCAD Release 14.* CADALYST, pages 48–70, 1997. 115

[C. Faloutsos R. Barber 1994] M Flickner J Hafner W Niblack D Petkovic W Equitz C. Faloutsos R. Barber. *Efficient and effective querying by image content.* Journal of Intelligence Information System, vol. 3 (3-4), pages 231–262, 1994. 194

[Carson 1999] Chad Carson, Megan Thomas, Serge Belongie, Joseph Hellerstein and Jitendra Malik. *Blobworld: a System for Region-based Image Indexing and Retrieval.* 1999. 197

[Chambah 2000] Majed Chambah and Bernard Besserer. *Digital Restoration of Faded Color Movies: A Four-Step Method.* In Color Imaging Conference, pages 161–166, 2000. 26

[Champin 2003] Pierre-antoine Champin and Christine Solnon. *Measuring the similarity of labeled graphs.* Case-Based Reasoning Research and Development, pages 80–95, 2003. 165

[Chang 1986] Shi-Kuo Chang and Eriand Jungert. *A spatial knowledge structure for image information systems using symbolic projections.* In ACM '86: Proceedings of 1986 ACM Fall joint computer conference, pages 79–86, Los Alamitos, CA, USA, 1986. IEEE Computer Society Press. 196

[Chang 1998] Chin-Chen Chang and Chin-Feng Lee. *A spatial match retrieval mechanism for symbolic pictures.* Journal of Systems and Software, vol. 44, no. 1, pages 73–83, 1998. 196

[Chhabra 1998] A Chhabra and I Phillips. *The Second International Graphics Recognition Contest - Raster to Vector Conversion: A Report.* Graphics Recognition: Algorithms and Systems, Lecture Notes in Computer Science, Springer, vol. 1389, 1998. 115, 116, 117, 118, 139

[Chhabra 2000] Atul K Chhabra and Dov Dori, editeurs. Graphics Recognition, Recent Advances, Third International Workshop, GREC'99 Jaipur, India, September 26-27, 1999, Selected Papers, volume 1941 of *Lecture Notes in Computer Science*. Springer, 2000. 116

[Chien 1996] S A Chien and H B Mortensen. *Automating image processing for scientific data analysis of a large image database.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 18, no. 8, pages 854–859, 1996. 55

[Christmas 1995] William J Christmas, Josef Kittler and Maria Petrou. *Structural Matching in Computer Vision Using Probabilistic Relaxation.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 17, no. 8, pages 749–764, 1995. 166

[Ciaccia 1997] Paolo Ciaccia, Marco Patella and Pavel Zezula. *M-tree: An Efficient Access Method for Similarity Search in Metric Spaces.* Proceedings of 23rd International Conference on Very Large Data Bases (VLDB'97), pages 426–435, 1997. 170

[Clement 1993] V Clement and M Thonnat. *A Knowledge-Based Approach to Integration of Image Processing Procedures.* CVGIP: Image Understanding, vol. 57, no. 2, pages 166–184, 1993. 55

[Clouard 1999] R Clouard, A Elmoataz, C Porquet and M Revenu. *Borg: a knowledge-based system for automatic generation of image processing programs.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 21, no. 2, pages 128–144, 1999. 55

[Conte 2004] Donatello Conte, Pasquale Foggia, Carlo Sansone and Mario Vento. *Thirty Years Of Graph Matching In Pattern Recognition.* International Journal of Pattern Recognition and Artificial Intelligence, vol. 18, no. 3, pages 265–298, 2004. 165

[Cootes 1995] T F Cootes, C J Taylor, D H Cooper and J Graham. *Active shape models—their training and application.* Comput. Vis. Image Underst., vol. 61, no. 1, pages 38–59, 1995. 144

[Cordella 1996] L P Cordella and A Marcelli. *An alternative approach to the performance evaluation of thinning algorithms for document processing applications.* Graphics recognition methods and applications (Lecture Notes in Computer Science), vol. 1072, pages 13–22, 1996. 116

[Coughlan 2002] James M Coughlan and Sabino J Ferreira. *Finding Deformable Shapes Using Loopy Belief Propagation*. Internatioanl Conference on Computer Vision (ECCV 2002), pages 453–468, 2002. 166

[Cox 2001] M F Cox and M A A Cox. *Multidimensional Scaling. Chapman and Hall*. Quantitative Applications in the Social Sciences, 2001. 169, 208

[Cross 1996] Andrew D J Cross and Edwin R Hancock. *Inexact Graph Matching with Genetic Search*. Advances in Structural and Syntactical Pattern Recognition, pages 150–159, 1996. 166

[Cross 1998] Andrew D J Cross and Edwin R Hancock. *Graph Matching With a Dual-Step EM Algorithm*. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 20, no. 11, pages 1236–1253, 1998. 166

[Dangauthier 2005] P Dangauthier. *Feature Selection For Self-Supervised Learning*. American Association for Artificial Intelligence Spring Symposium Series, 2005. 36, 37

[Datta 2006] Ritendra Datta, Dhiraj Joshi, Jia Li, James and Z Wang. *Image retrieval: Ideas, influences, and trends of the new age*. ACM Computing Surveys, vol. 39, page 2007, 2006. 195

[DBL 1998] *Graphics Recognition, Algorithms and Systems, Second International Workshop, GREC'97, Nancy, France, August 22-23, 1997, Selected Papers*. In Karl Tombre and Atul K Chhabra, editeurs, GREC, volume 1389 of *Lecture Notes in Computer Science*. Springer, 1998. 116

[Delalandre 2010] Mathieu Delalandre, Ernest Valveny, Tony Pridmore and Dimosthenis Karatzas. *Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems*. International Journal on Document Analysis and Recognition, page Online first, 2010. 142, 144

[den Hartog 1996] J E den Hartog, T K ten Kate and J J Gerbrands. *Knowledge-Based Interpretation of Utility Maps*. Computer Vision and Image Understanding, vol. 63, no. 1, pages 105–117, 1996. 17, 18

[Deseilligny 1993] M P Deseilligny, H Le Men and G Stamon. *Map understanding for GIS data capture: Algorithms for road network graph reconstruction,*. In The 2nd International Conference on Document Analysis and Recognition, pages 676–679., 1993. 17, 55

[Deseilligny 1995] Marc Pierrot Deseilligny, Hervé Le Men and Georges Stamon. *Character string recognition on maps, a rotation-invariant recognition method*. Pattern Recogn. Lett., vol. 16, no. 12, pages 1297–1310, 1995. 17

[di Baja 1996] Gabriella Sanniti di Baja and Edouard Thiel. *Skeletonization algorithm running on path-based distance maps*. Image and Vision Computing, vol. 14, no. 1, pages 47–57, 1996. 146

[Dizenzo 1986] S Dizenzo. *A note on the gradient of a multi-image.* Computer Vision, Graphics, and Image Processing, vol. 33, no. 1, pages 116–125, 1986. 22, 41

[Dori 1993] D Dori, Y Liang, J Dowell and I Chai. *Spare pixel recognition of primitives in engineering drawings.* Machine Vision Appl, vol. 6, pages 79–82, 1993. 116

[Dori 1995] D Dori. *Vector-based arc segmentation in the machine drawing understanding system environment.* IEEE Trans Pattern Anal Machine Intell, vol. 17, pages 959–971, 1995. 116

[Dori 1996] D Dori, L Wenyin and M Peleg. *How to win a dashed line detection contest.* Graphics Recognition Methods and Applications, Lecture Notes in Computer Science, Springer, vol. 1072, 1996. 115

[Dorin Comaniciu 1997] Peter Meer Dorin Comaniciu. *Robust Analysis of Feature Spaces: Colour Image Segmentation.* Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pages 750–755, 1997. 6

[Dosch 2006] Philippe Dosch and Ernest Valveny. *Report on the Second Symbol Recognition Contest*, 2006. 142

[Draper 1996] B A Draper, A R Hanson and E M Riseman. *Knowledge-directed vision: control, learning, and integration.* Proceedings of the IEEE, vol. 84, no. 11, pages 1625–1637, 1996. 55

[Due 1996] Due, Anil K Jain and Torfinn Taxt. *Feature extraction methods for character recognition-A survey.* Pattern Recognition, vol. 29, no. 4, pages 641–662, 1996. 205

[Duygulu 2002] P Duygulu, K Barnard, J de Freitas and D Forsyth. *Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary.* Computer Vision ECCV 2002, pages 349–354, 2002. 197

[E-Cognition] E-Cognition. *Definiens Imaging GmbH e-Cognition: Object Oriented Image Analysis. http://www.definiens-imaging.com/ecognition/.* Able Software Corp.: R2V http://www.ablesw.com/r2v/; Resident Ltd.: MapEdit 4.6 http://www.resident.ru/. 11, 12

[Eakins 2001] John Eakins. *Retrieval of Still Images by Content.* Lectures on Information Retrieval, pages 111–138, 2001. 193

[ERDOS P. 1959] RENYI A ERDOS P. *On random graphs.* Publicationes Mathematicae Debrecen, vol. 6, pages 290–297, 1959. 179

[Eshera 1986] M A Eshera and K S Fu. *An image understanding system using attributed symbolic representation and inexact graph-matching.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 8, no. 5, pages 604–618, 1986. 166

[et al E.Nowak 2006] et al E.Nowak. *Sampling strategies for bag-of-features image classification.* Computer Vision (ECCV), 2006. 195

[F. Long H.J. Zhang 2003] D D Feng F. Long H.J. Zhang. *Fundamentals of content-based image retrieval.* Multimedia Information Retrieval and Management, vol. 1390, pages Springer, Berlin, 2003. 194

[Farshid Arman Arding Hsu 1993] Farshid Arman Arding Hsu and Ming Yee Chiu. *Image processing on compressed data for large video databases.* ACM Multimedia Conference, pages .267–272, 1993. 205

[Ferrer 2006] Miquel Ferrer, Ernest Valveny and Francesc Serratosa. *Spectral Median Graphs Applied to Graphical Symbol Recognition.* Progress in Pattern Recognition, Image Analysis and Applications, pages 774–783, 2006. 181

[Filipski 1992] A J Filipski and R Flandrena. *Automated conversion of engineering drawings to CAD form.* Proc IEEE, vol. 80, pages 1195–1209, 1992. 116

[Finch 1997] Andrew M Finch, Richard C Wilson and Edwin R Hancock. *Matching delaunay graphs.* Pattern Recognition, vol. 30, no. 1, pages 123–140, 1997. 220, 227

[Fletcher L.A. 1988] Kasturi R Fletcher L.A. *A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. Vol. 10, N, pages 910–918, 1988. 17

[Floyd 1967] Robert W Floyd. *Nondeterministic Algorithms.* J. ACM, vol. 14, no. 4, pages 636–644, 1967. 120

[Ford 1992] Gary P Ford and Jun Zhang. *Structural graph-matching approach to image understanding.* Intelligent Robots and Computer Vision X: Algorithms and Techniques, vol. 1607, no. 1, pages 559–569, 1992. 167

[Gelbukh A. 2003] Han SangYong Levachkine S Gelbukh A. *Combining Sources of Evidence to Resolve Ambiguities in Toponym Recognition in Cartographic Maps.* International Workshop on Semantic Processing of Spatial Data (GEOPRO 2003), vol. 1, pages 42–51, 2003. 16

[GHOSH 1999] D. GHOSH and A. P. SHIVAPRASAD. *An analytic approach for generation of artificial hand-printed character database from given generative models.* Pattern recognition, vol. 32, no. 6, pages 907–920, 1999. 144

[Gold 1996a] S Gold and A Rangarajan. *A Graduated Assignment for Graph Matching.* IEEE Transaction in Pattern Analysis and Machine Intelligence, vol. 18, pages 377–388, 1996. 166

[Gold 1996b] Steven Gold and Anand Rangarajan. *Graph matching by graduated assignment.* IEEE transactions on pattern analysis and machine intelligence, pages 239–244, 1996. 172

[Goldberg 1989] David E Goldberg. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989. 38

[Gordon 1999] A D Gordon. *Classification*. Chapman & Hall, 2nd edition edition, 1999. 170

[graph database 2007] Http://iamwww.unibe.ch/fki/databases/iam graph database. *Public IAM Graph Database Repository*. IAM, 2007. 181

[Gupta 1997] Amarnath Gupta and Ramesh Jain. *Visual information retrieval*. Commun. ACM, vol. 40, no. 5, pages 70–79, 1997. 194

[Guru 2001] D S Guru and P Nagabhushan. *Triangular spatial relationship: a new approach for spatial knowledge representation*. Pattern Recognition Letters, vol. 22, no. 9, pages 999–1006, 2001. 196

[H. Muller N. Michoux 2004] D Bandon H. Muller N. Michoux and A Geissbuhler. *A review of content-based image retrieval systems in medical applications clinical benefits and future directions*. International Journal of Medical Informatics, vol. 73, pages 1–23, 2004. 194

[Hall 1998]] M Hall. *Correlation-based feature selection for machine learning*. Thesis in Computer Science at the University of Waikato, 1998. 36, 37, 39

[Hamada 1993] A Hamada. *A new system for the analysis of schematic diagrams*. In 2nd International Conference on Document Analysis and Recognition (ICDAR), pages 369–371, 1993. 17, 55

[Haralick 1992] R M Haralick. *Performance characterization in image analysis thinning, a case in point*. Pattern Recogn Lett, vol. 13, pages 5–12, 1992. 115, 117

[Haralickt 2009] Robert M Haralickt, Henry S Baird and David Adihan. *Document Degradation Models: Parameter Estimation and Model Validation*, June 2009. 145

[Hidovic 2004] Dzena Hidovic and Marcello Pelillo. *Metrics For Attributed Graphs Based On The Maximal Similarity Common Subgraph*. International Journal of Pattern Recognition and Artificial Intelligence, vol. 18, no. 3, pages 299–313, 2004. 172

[Holte 1993] Robert C Holte. *Very Simple Classification Rules Perform Well on Most Commonly Used Datasets*. Machine Learning, vol. 11, no. 1, pages 63–90, 1993. 40

[HongJiang Zhang Atreyi Kankanhalli 1993] HongJiang Zhang Atreyi Kankanhalli and Stephen William Smoliar. *Automatic partitioning of full-motion video*. Multimedia Systems, vol. , 1:10{28}, 1993. 203

[Hori 1996] O Hori and D S Doermann. *Quantitative measurement of the performance of raster-to-vector conversion algorithms.* Graphics recognition – methods and applications (Lecture Notes in Computer Science), vol. 1072, pages 57–68, 1996. 116, 117

[Horridge 2008] Matthew Horridge and Sean Bechhofer. *The OWL API: A Java API for Working with OWL 2 Ontologies.* In Proceedings of the 5th International Workshop on OWL: Experiences and Directions, volume 529, Chantilly, VA, United States, 2008. 100

[Horton 1987] J D Horton. *A polynomial-time algorithm to find the shortest cycle basis of a graph.* SIAM Journal on Computing, vol. 16(2), pages 358–366, 1987. 85

[Hse 2004] Heloise Hse and A Richard Newton. *Sketched Symbol Recognition using Zernike Moments.* Pattern Recognition, International Conference on, vol. 1, pages 367–370, 2004. 206

[Huang 2004] Po-Whei Huang and Chu-Hui Lee. *Image Database Design Based on 9D-SPA Representation for Spatial Relations.* IEEE Transactions on Knowledge and Data Engineering, vol. 16, pages 1486–1496, 2004. 196

[Hudelot 2003] Céline Hudelot and Monique Thonnat. *A Cognitive Vision Platform for Automatic Recognition of Natural Complex Objects.* Tools with Artificial Intelligence, IEEE International Conference on, vol. 0, page 398, 2003. 55

[I.K. Sethi 2001] I L Coman I.K. Sethi. *Mining association rules between low-level image features and high-level concepts.* Proceedings of the SPIE Data Mining and Knowledge Discovery, vol. vol. III, pages 279–290, 2001. 193

[Illumination] International Commission On Illumination. *Advancing knowledge and providing standardisation to improve the lighted environment.* http://www.cie.co.at/index.html. 33

[Implementation] Tree Edit Distance Implementation. *http://web.science.mq.edu.au/~swan/howtos/treedistance/.* 214

[J. D. Rugna P. Colantoni 2004] J. D. Rugna P. Colantoni and N Boukala. *Hybrid color spaces applied to image database,.* The Journal of Electronic Imaging (JEI), vol. 5304, pages 254–264, 2004. 33, 39, 41

[J. M. Tenenbaum T. D. Garvey 1974] S.Weyl J. M. Tenenbaum T. D. Garvey and H C.Wolf. *An interactive facility for scene analysis research.* Technical Report 87, Adapted Intelligent Center, Stanford Research Institute, Menlo Park, CA, 1974. 32

[Jaisimha 1993] M Y Jaisimha, R M Haralick and D Dori. *A methodology for the characterization of the performance of thinning algorithms.* Proceedings of

the Second International Conference on Document Analysis and Recognition, vol. 1, pages 282–286, 1993. 116

[Janssen 1997] Rik D. T. Janssen and Albert M. Vossepoel. *Adaptive vectorization of line drawing images.* Computer Vision and Image Understanding, vol. 65, no. 1, 1997. 17

[Jeon 2003] J Jeon, V Lavrenko and R Manmatha. *Automatic image annotation and retrieval using cross-media relevance models.* 2003. xix, 198

[J.F. Haddon 1993] J F Boyce J.F. Haddon. *Co-occurrence matrices for image analysis.* IEEE Electronics and Communications Engineering Journal, vol. vol. 5,, pages No 2, pp. 71–83, 1993. 204

[Jiang 2000] Xiaoyi Jiang, Andreas Münger and Horst Bunke. *Synthesis of Representative Graphical Symbols by Computing Generalized Median Graph.* Graphics Recognition:Recent Advances, A.K. Chhabra and D. Dori, eds, pages 183–192, 2000. 167

[J.L.Bentley 1979] J.L.Bentley and T.Ottmann. *Algorithms for reporting and counting geometric intersections.* IEEE Transactions on Computers, pages 643–647, 1979. 85

[Joseph 1992] S H Joseph and T P Pridmore. *Knowledge-Directed Interpretation of Mechanical Engineering Drawings.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 14, no. 9, pages 928–940, 1992. 18

[J.R. Smith 1996] S F Chang J.R. Smith. *VisualSeek: a fully automatic contentbased query system.* Proceedings of the Fourth ACM International Conference on Multimedia, pages 87–98, 1996. 194

[Jr 2003] Alfredo Ferreira Jr, Jr. Manuel, J Fonseca and Joaquim A Jorge. *Polygon Detection from a Set of Lines.* In In Proceedings of 12 o Encontro Português de Computação Gráfica (12th EPCG), pages 159–162, 2003. xvii, 85, 88, 147

[J.Z. Wang J. Li 2001] G Wiederhold SIMPLIcity J.Z. Wang J. Li. *Aemantics-sensitive integrated matching for picture libraries.* IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 23 (9), pages 947–963, 2001. 194

[Kashima 2004] Hisashi Kashima and Yuta Tsuboi. *Kernel-based discriminative learning algorithms for labeling sequences, trees, and graphs.* Proceedings of the twenty-first international conference on Machine learning, 2004. 169, 209

[Kass 1988] Michael Kass, Andrew Witkin and Demetri Terzopoulos. *Snakes: Active contour models.* International Journal of Computer Vision, vol. 1, no. 4, pages 321–331, January 1988. 78

[Kasturi 1990a] R Kasturi, S T Bow, W El-Masri, J Shah, J R Gattiker and U B Mokate. *A system for interpretation of line drawings.* IEEE Trans Pattern Anal Machine Intell, vol. 17, pages 978–992, 1990. 17, 116

[Kasturi 1990b] Rangachar Kasturi, Senthil Siva and Lawrence O'Gorman. *Techniques for Line Drawing Interpretation: An Overview (Invited).* In Proceedings of IAPR Workshop on Machine Vision Applications, MVA 1990, pages 151–160, 1990. 17

[Kasturi 1992] R Kasturi, R Raman, C Chennubhotla and and L. O'Gorman. *An overview of techniques for graphics recognition.* Structured Document Analysis, vol. (H. S. Bai, pages 285–324, 1992. 55

[Kasturi 1996a] R Kasturi and K Tombre (eds.). *Graphics Recognition: Methods and Applications.* First International Workshop, University Park, PA, USA, August 1995, Selected papers published as Lecture Notes in Computer Science, vol. 1072, 1996. 115

[Kasturi 1996b] Rangachar Kasturi and Karl Tombre, editeurs. Graphics Recognition, Methods and Applications, First International Workshop, University Park, PA, USA, August 10-11, 1995, Selected Papers, volume 1072 of *Lecture Notes in Computer Science*. Springer, 1996. 116

[Kasturi 2002] Rangachar Kasturi, Lawrence O'Gorman and Venu Govindaraju. *Document image analysis: A primer.* Sadhana, vol. 27, no. 1, pages 3–22, 2002. xvi, 2, 3

[Kaufman 1990] L Kaufman and P J Rousseeuw. *Finding Groups in Data: an introduction to cluster analysis.* Probability & Mathematical Statistics, vol. ISBN-10: 0, 1990. 67, 69, 180, 206

[Khotanzad 1990] Alireza Khotanzad and Yaw Hua Hong. *Invariant image recognition by Zernike Moments.* Pattern Analysis and Machine Intelligence (PAMI), vol. Vol 12, No, pages 489–497, 1990. 69, 180

[Koller 1996] Daphne Koller and Mehran Sahami. *Toward optimal feature selection.* International Conference on Machine Learning, pages 284–292, 1996. 36

[Kong 1996] B Kong, I T Phillips, R M Haralick, A Prasad and R Kasturi. *A benchmark: performance evaluation of dashed-line detection algorithms.* Graphics recognition – methods and applications (Lecture Notes in Computer Science), vol. 1072, pages 270–285, 1996. 115, 116

[Kriegel 2003] Hans-peter Kriegel and Stefan Schonauer. *Similarity Search in Structured Data.* Data Warehousing and Knowledge Discovery, pages 309–319, 2003. 172, 174

[Kuhn 1955] H W Kuhn. *The Hungarian method for the assignment problem.* Naval Research Logistic Quarterly, vol. 2, pages 83–97, 1955. 124, 190, 227

[Kuner 1988] P Kuner and B Ueberreiter. *Pattern Recognition by Graph Matching: Combinatorial versus Continuous Optimization.* International Journal in Pattern Recognition and Artificial Intelligence, vol. 2, pages 527–542, 1988. 167

[L. Busin N. Vandenbroucke 2004] L Macaire et J.-G. Postaire L. Busin N. Vandenbroucke. *Color space selection for unsupervised color image segmentation by histogram multithresholding.* The 11th International Conference on Image Processing (ICIP'04), pages 203–206, 2004. 33, 41

[L. Fei-Fei 2004] R Fergus L. Fei-Fei and P Perona. *Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories.* IEEE. CVPR 2004, Workshop on Generative-Model Based Vision, 2004. 215

[Lam 1993] L Lam and C Y Suen. *Evaluation of thinning algorithms from an OCR viewpoint.* Proceedings of the Second International Conference on Document Analysis and Recognition, vol. 1, pages 287–290, 1993. 116

[Lee 1991] S Lee, L Lam and C Y Suen. *Performance evaluation of skeletonization algorithms for document image processing.* Proceedings of the First International Conference on Document Analysis and Recognition, vol. 1, pages 260–271, 1991. 116

[Lee 2000] R S T Lee and J N K Liu. *Tropical cyclone identification and tracking system using integrated neural oscillatory elastic graph matching and hybrid RBF network track mining techniques.* Neural Networks, IEEE Transactions on, vol. 11, no. 3, pages 680–689, 2000. 166

[Lee 2002] Yang-Lyul Lee and Rae-Hong Park. *A surface-based approach to 3-D object recognition using a mean field annealing neural network.* Pattern Recognition, vol. 35, no. 2, pages 299–316, 2002. 166

[Lepetit 2005] V Lepetit, P Lagger and P Fua. *Randomized trees for real-time keypoint recognition.* In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 2, pages 775 – 781 vol. 2, 2005. 200

[Levachkine S. 2000] Polchkov E Levachkine S. *Integrated Technique for Automated Digitation of Raster Maps.* Revista Digital Universitaria (RDU). ((ISSN 1607-6079)). On-line: http://www.revista.unam.mx/vol.1/art4/, vol. 1, 2000. 11, 12, 14

[Levachkine 2003] S Levachkine. *System Approach to R2V Conversion for Analytical GIS.* International Workshop on Semantic Processing of Spatial Data (GEOPRO 2003), vol. ISBN 970-3, pages 22–33, 2003. 11, 12

[Levachkine 2004] Serguei Levachkine. *Raster to Vector Conversion of Color Cartographic Maps*, 2004. 12, 14

[Levenshtein 1966] V Levenshtein. *Binary codes capable of correcting deletions, insertions and reversals.* .Soviet Physics-Doklady, vol. 10, pages 707–710, 1966. 206

[Leymarie 1993] F Leymarie and MD Levine. *Tracking Deformable Objects in the Plane Using an Active Contour Model.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 15, no. 6, pages 617 – 634, 1993. 78

[Li 1998] John Z Li, M Tamer Özsu and M Tamer. *Point-Set Topological Relations Processing In Image Databases.* In In First International Forum on Multimedia and Image Processing, pages 51–54, 1998. 196

[lightweight SIFT-implementation for Java after the paper of David Lowe (2004)] A lightweight SIFT-implementation for Java after the paper of David Lowe (2004). *http://fly.mpi-cbg.de/~saalfeld/javasift.html.* 214

[Lladós J. 2003] Kwon Y B Lladós J. *Graphics Recognition, Recent Advances and Perspectives.* GREC, 2003. 7

[Lladós 1997] J Lladós. *Combining Graph Matching and Hough Transform for Hand-Drawn Graphical Document Analysis. Application to Architectural Drawings.* PhD thesis, Universitat Autonòma de Barcelona and Université Paris 8, 1997. 166

[Lladós 2001] J Lladós, E Marti and J J Villanueva. *Symbol Recognition by Error-Tolerant Subgraph Matching between Region Adjacency Graphs.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, pages 1137–1143, 2001. 136

[Locteau 2006] Hervé Locteau, Romain Raveaux, Sébastien Adam, Yves Lecourtier, Pierre Héroux and Éric Trupin. *Approximation of Digital Curves using a Multi-Objective Genetic Algorithm.* In 18th International Conference on Pattern Recognition (ICPR), pages 716–719, Washington, DC, USA, 2006. IEEE Computer Society. 81, 147

[Lopresti 2003] D Lopresti and G Wilfong. *A fast technique for comparing graph representations with applications to performance evaluation.* International Journal on Document Analysis and Recognition, vol. 6, no. 4, pages 219–229, 2003. 70, 173, 174, 178

[Lowe] D Lowe. *Distinctive image features from scale-invariant keypoints.* Journal on Computer Vision, vol. 60(2):91. 195

[Luo 2000] Bin Luo and Edwin R Hancock. *Symbolic Graph Matching Using the EM Algorithm and Singular Value Decomposition.* 15th International Conference on Pattern Recognition (ICPR'00), pages 2141–2144, 2000. 166

[M. De Marsicoi L. Cinque 1997] M. De Marsicoi L. Cinque and S Levialdi. *Indexing pictorial documents by their content: A survey of current techniques.* Image and Vision Computing, vol. 15, pages 119–141, 1997. 194

[M. G. Brown J. T. Foote 1995] G J F Jones K Sparck Jones M. G. Brown J. T. Foote and S J Young. *Automatic content-based retrieval of broadcast news.* ACM Multimedia Conference, 1995. 205

[Mahé 2004] Pierre Mahé, Nobuhisa Ueda, Tatsuya Akutsu, Jean-Luc Perret and Jean-Philippe Vert. *Extensions of marginalized graph kernels.* In Proceedings of the Twenty-First International Conference on Machine Learning, 2004. 169, 209

[Mahé 2005] Pierre Mahé, Nobuhisa Ueda, Tatsuya Akutsu, Jean-Luc Perret and Jean-Philippe Vert. *Graph Kernels for Molecular Structure-Activity Relationship Analysis with Support Vector Machines.* Journal of Chemical Information and Modeling, vol. 45, no. 4, pages 939–951, 2005. 169, 209

[Maillot] Nicolas Maillot, Monique Thonnat and Alain Boucher. *Towards ontology-based cognitive vision.* Machine Vision and Applications, vol. 16, no. 1, pages 33–40. 55

[Marr 1978] D Marr and H K Nishihara. *Representation and Recognition of the Spatial Organization of Three-Dimensional Shapes.* Proceedings of the Royal Society of London. Series B, Biological Sciences, vol. 200, pages 269–294, 1978. 194

[Martin 2001] D Martin, C Fowlkes, D Tal and J Malik. *A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics.* Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on, vol. 2, pages 416–423, 2001. 47

[Meastex] Meastex. *http:// www.cssip.elec.uq.edu.au /˜guy/ meastex / meastex.html.* 206

[Mehlhorn 1984] Kurt Mehlhorn. Graph algorithms and NP-completeness. Springer-Verlag New York, Inc., New York, NY, USA, 1984. 166

[Messmer 1998] Bruno T Messmer and Horst Bunke. *A New Algorithm for Error-Tolerant Subgraph Isomorphism Detection.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 5, pages 493–504, 1998. 165

[Messmer 1999] Bruno T Messmer and Horst Bunke. *A decision tree approach to graph and subgraph isomorphism detection.* Pattern Recognition, vol. 32, no. 12, pages 1979–1998, 1999. 166

[Munkres 1957] J Munkres. *Algorithms for the Assignment and Transportation Problems.* Journal of the Society of Industrial and Applied Mathematics, vol. 5, no. 1, pages 32–38, 1957. 124, 190, 227

[Myers 2000] Richard Myers, Richard C Wilson and Edwin R Hancock. *Bayesian Graph Edit Distance.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 6, pages 628–635, 2000. 171, 172

[N. Vandenbroucke L. Macaire 1998] N. Vandenbroucke L. Macaire and J G Postaire. *Color pixels classification in an hybrid color space.* The IEEE International Conference on Image Processing - ICIP'98, vol. 1, pages 176–180, 1998. 33

[N. Vandenbroucke L. Macaire 2003] N. Vandenbroucke L. Macaire and J G Postaire. *Color image segmentation by pixel classification in an adapted hybrid color space: application to soccer image analysis.* Computer Vision and Image Understanding, vol. 90(2), pages 190–216, 2003. 34, 41

[Nagasamy 1990] V Nagasamy and N Langrana. *Engineering drawing processing and vectorization system.* Comput Vision Graphics Image Process, vol. 49, pages 379–397, 1990. 116

[Nagy 2000] George Nagy. *Twenty Years of Document Image Analysis in PAMI.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1, pages 38–62, 2000. 2

[Neisser 1976] U Neisser. Cognition and reality : principles and implications of cognitive psychology. W. H. Freeman, San Francisco, 1976. 18, 19

[Neisser 1989] U. Neisser. *Direct Perception and Recognition as Distinct Perceptual System.* Cognitive Science Society, 1989. 19

[Neuhaus 2007] Michel Neuhaus and Horst Bunke. *Automatic learning of cost functions for graph edit distance.* Information Science, vol. 177, no. 1, pages 239–247, 2007. 170

[Nierman A. 2002] Jagadish H V Nierman A. *Evaluating structural similarity in XML documents.* 5th Int. Workshop on the Web and Databases, pages 61–66, 2002. 208

[Nock 2004] Richard Nock and Frank Nielsen. *Statistical Region Merging.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 11, pages 1452–1458, 2004. 202

[Ogier 1993] J M Ogier, J Labiche, R Mullot and Y Lecourtier. *Attributes extraction for French map interpretation.* In Document Analysis and Recognition, 1993., Proceedings of the Second International Conference on, pages 672–675, October 1993. 17

[Ogier 1998] Jean Marc Ogier, Rémy Mullot, Jacques Labiche and Yves Lecourtier. *Multilevel approach and distributed consistency for technical map interpretation: application to cadastral maps.* Comput. Vis. Image Underst., vol. 70, no. 3, pages 438–451, 1998. 18

[Ogle 1995] Virginia Ogle and Michael Stonebraker. Chabot:. *Retrieval from a relational database of images.* IEEE Computer, pages 40–48, 1995. 205

[Okazaki 1988] A Okazaki, T Kondo, K Mori, S Tsunekawa and E Kawamoto. *An Automatic Circuit Diagram Reader with Loop-Structure-Based Symbol Recognition.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, pages 331–341, 1988. 17

[Ostu 1979] Nobuyuki Ostu. *A Threshold Selection Method from Gray-Level Histograms.* Systems, Man and Cybernetics, IEEE Transactions on, vol. 9, no. 1, pages 62–66, 1979. 80

[Papadopoulos 1999] Apostolos Papadopoulos and Yannis Manolopoulos. *Structure-Based Similarity Search with Graph Histograms.* Structure-based similarity search with graph histograms, pages 174–178, 1999. 168, 208

[Petrakis 2001] Euripides G M Petrakis. *Design and Evaluation of Spatial Similarity Approaches for Image Retrieval.* Image and Vision Computing, vol. 20, pages 59–76, 2001. 196

[Petrakis 2002] Euripides G M Petrakis, Christos Faloutsos and King-Ip (David) Lin. *ImageMap: An Image Indexing Method Based on Spatial Similarity.* IEEE Transactions on Knowledge and Data Engineering, vol. 14, pages 979–987, 2002. 196

[Philbin 2007] J Philbin, O Chum, M Isard, J Sivic and A Zisserman. *Object retrieval with large vocabularies and fast spatial matching.* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2007. 200, 201, 222, 228

[Phillips 1998] I Phillips, J Liang, A Chhabra and R Haralick. *A Performance Evaluation Protocol for Graphics Recognition Systems.* Graphics Recognition: Algorithms and Systems, Lecture Notes in Computer Science, Springer, vol. 1389, 1998. 115

[Phillips 1999] I Phillips and A Chhabra. *Empirical Performance Evaluation of Graphics Recognition Systems.* IEEE Transaction of Pattern Analysis and Machine Intelligence, vol. 21, No 9, pages 849–870, 1999. 115

[P.N. Suganthan E.K. Teoh 1995a] D P Mital P.N. Suganthan E.K. Teoh. *Pattern Recognition by Graph Matching Using the Potts MFT Neural Networks.* Pattern Recognition, vol. vol. 28, n, pages pp. 997–1009, 1995. 167

[P.N. Suganthan E.K. Teoh 1995b] D P Mital P.N. Suganthan E.K. Teoh. *Pattern Recognition by Homomorphic Graph Matching Using Hopfield Neural Networks.* Image and Vision Computing, vol. vol. 13, n, pages pp. 45–60, 1995. 167

[Pratt 2002] William K Pratt. *Bibliography.* In Digital Image Processing (Third Edition), pages 717–722. 2002. 195

[R. M. Haralick 1973] K Shanmugam R. M. Haralick and I Dinstein. *Textural features for image classification.* IEEE Transactions on System, Man, Cybernetics, vol. vol. SMC-3, pages 610–621, 1973. 204

[R. Sriram J. M. Francos 1996] R. Sriram J. M. Francos and W A Pearlman. *Texture coding using a wold decomposition based model.* IEEE Transactions of Image Processing, vol. 5, pages 1382–1386, 1996. 194

[R2V] R2V. *EasyTrace: Advanced Software for Automatic R2V Conversion. http://www.easytrace. com/work/english.* TerraSpace: Center for Applied Informatics http://www.terraspace. ru/eng. 11, 12

[Ralaivola 2005] Liva Ralaivola, Sanjay Joshua Swamidass, Hiroto Saigo and Pierre Baldi. *Graph kernels for chemical informatics.* Neural Networks, vol. 18, no. 8, pages 1093–1110, 2005. 165

[Raveaux 2007a] R. Raveaux, J.-C. Burie and J.-M. Ogier. *A Colour Document Interpretation: Application to Ancient Cadastral Maps.* In ICDAR '07: Proceedings of the Ninth International Conference on Document Analysis and Recognition, pages 1128–1132, Washington, DC, USA, 2007. IEEE Computer Society. 147

[Raveaux 2007b] Romain Raveaux, Barbu Eugen, Hervé Locteau, Sébastien Adam, Pierre Héroux and Eric Trupin. A Graph Classification Approach Using a Multi-objective Genetic Algorithm Application to Symbol Recognition, volume 4538 of *Lecture Notes in Computer Science*, pages 361–370. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. 70

[Raveaux 2008a] Romain Raveaux, Jean-Christophe Burie and Jean-Marc Ogier. *A colour text/graphics separation based on a graph representation.* In 19th International Conference on Pattern Recognition (ICPR), pages 1–4, Washington, DC, USA, 2008. IEEE Computer Society. 147

[Raveaux 2008b] Romain Raveaux, Jean-Christophe Burie and Jean-Marc Ogier. *Object Extraction from Colour Cadastral Maps.* IAPR International Workshop on Document Analysis Systems, vol. 0, pages 506–514, 2008. 147

[Raveaux 2010] Romain Raveaux and Guillaume Hillairet. *Model Driven Image Segmentation Using a Genetic Algorithm for Structured Data.* In Manuel Grana

Romay, Emilio Corchado and Teresa Garcia-Sebastian, editeurs, Hybrid Artificial Intelligence Systems, 5th International Conference, HAIS 2010, pages 311–318. Springer, Lecture Notes in Computer Science, 2010. 65

[Renouf 2007] Arnaud Renouf, Régis Clouard and Marinette Revenu. *A Platform Dedicated to Knowledge Engineering for the Development of Image Processing Applications.* In ICEIS (2), pages 271–276, 2007. 55

[Riesen 2008] Kaspar Riesen and Horst Bunke. *Approximate graph edit distance computation by means of bipartite graph matching.* Image and Vision Computing, vol. In Press,, pages –, 2008.

[Robles-Kelly 2005] Antonio Robles-Kelly and Edwin R Hancock. *Graph Edit Distance from Spectral Seriation.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 3, pages 365–378, 2005. 172

[Rosenberger 2006] Christopher Rosenberger. *Adaptative evaluation of image segmentation results.* ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition, pages 399–402, 2006. 50

[Rousseeuw 1987] P.J. Rousseeuw. *Silhouettes: A graphical aid to the interpretation and validation of cluster analysis.* J. Comput. Appl. Math., vol. 20, pages 53–65, 1987. 69

[Rui 1999] Y Rui, T S Huang and S.-F. Chang. *Image retrieval: current techniques, promising directions, and open issues.* Journal of Visual Commun. Image Representation, vol. 10 (4), pages 39–62, 1999. 194

[S. A. Nene] S K Nayar S. A. Nene and H Murase. *Columbia Object Image Library (COIL-100).* Technical Report CUCS-006-96. 215

[Sanfeliu 1983] A Sanfeliu and K Fu. *A distance measure between attributed relational graphs for pattern recognition.* IEEE Transactions on Systems, Man, and Cybernetics, vol. (Part B) 1, pages 353–363, 1983. 136

[Sangwine Stephen J.; Horne 1998] Robin E N (Eds.) Sangwine Stephen J.; Horne. The Colour Image Processing Handbook. Springer-Verlag New York, Inc., 1998. 33

[Schenker 2004] Adam Schenker, Mark Last, Horst Bunke and Abraham Kandel. *Classification Of Web Documents Using Graph Matching.* International Journal of Pattern Recognition and Artificial Intelligence, vol. 18, no. 3, pages 475–496, 2004. 165

[Sebastian T.B. 2001] Klein P N Kimia B B Sebastian T.B. *Recognition of shapes by editing shock graphs.* 8th Int. Conf. on Computer Vision., vol. 1, pages 755–762, 2001. 208

[Segmentation 2002] Dataset Berkeley Segmentation. *http:// www.cs.berkeley.edu / projects / vision / grouping / segbench /.* 2002. 47

[Seong 1994] Dong Su Seong, Young Kyu Choi, Ho Sung Kim and Kyu Ho Park. *An algorithm for optimal isomorphism between two random graphs.* Pattern Recognition Letters, vol. 15, no. 4, pages 321–327, 1994. 166

[Serrau 2005] Alessandra Serrau, Gian Luca Marcialis, Horst Bunke and Fabio Roli. *An Experimental Comparison of Fingerprint Classification Methods Using Graphs.* Graph-Based Representations in Pattern Recognition, pages 281–290, 2005. 165

[Sezgin 2004] Mehmet Sezgin and Bülent Sankur. *Survey over image thresholding techniques and quantitative performance evaluation.* Journal of Electronic Imaging, vol. 13, no. 1, pages 146–168, 2004. 79

[Shapiro 1981] L Shapiro and R Haralick. *Structural descriptions and inexact matching.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 3, pages 504–519, 1981. 166

[Shi 2000] Jianbo Shi and Jitendra Malik. *Normalized Cuts and Image Segmentation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pages 888–905, 2000. 197

[Shimotsuji 1992] Shigeyoshi Shimotsuji, Osamu Hori, Mieko Asano, Kaoru Suzuki, Fumihiko Hoshino and Toshiaki Ishii. *A Robust Recognition System for a Drawing Superimposed on a Map.* Computer, vol. 25, no. 7, pages 56–59, 1992. 17

[Shokoufandeh 2006] Ali Shokoufandeh, Lars Bretzner, Diego Macrini, M Fatih Demirci, Clas Jönsson and Sven Dickinson. *The representation and matching of categorical shape.* Computer Vision and Image Understanding, vol. 103, no. 2, pages 139–154, 2006. 172, 174

[S.K. Chang 1984] S H Liu S.K. Chang. *Picture indexing and abstraction techniques for pictorial databases.* IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 6 (4), pages 475–483, 1984. 193

[Smith 1997] G Smith and I Burns. *Measuring texture classification algorithms.* Pattern Recognition Letters, vol. 18, pages 1495–1501, 1997. 206

[Suard 2006] F Suard, Alain Rakotomamonjy and Abdelaziz Bensrhair. *Object Categorization Using Kernels Combining Graphs and Histograms of Gradients.* Image Analysis and Recognition, pages 23–34, 2006. 169, 209

[(SUBDUE)] Graph Based Knowledge Discovery (SUBDUE). *http://ailab.wsu.edu/subdue/.* 179

[Suzuki 1990] Satoshi Suzuki and Toyomichi Yamada. *Maris: map recognition input system.* Pattern Recogn., vol. 23, no. 8, pages 919–933, 1990. 17

[Syslo 1981] Maciej M Syslo. *An efficient cycle vector space algorithm for listing all cycles of a planar graph.* SIAM Journal on Computing, vol. 10(4), pages 797–808, 1981. 85

[Town] Christopher Town. *Ontological inference for image and video analysis.* Machine Vision and Applications, vol. 17, no. 2, pages 94–115. 55

[Tsai 1979] Wen-Hsiang Tsai and King-Sun Fu. *Error-Correcting Isomorphisms of Attributed Relational Graphs for Pattern Analysis.* Systems, Man and Cybernetics, IEEE Transactions on, vol. 9, no. 12, pages 757–768, 1979. 165

[Tsay 1989] Y T Tsay and W H Tsai. *Model-guided Attributed String Matching by Split-and-merge for Shape Recognition.* International Journal on Pattern Recognition and Artificial Intelligence, vol. 3(2), pages 159–179, 1989. 138

[Valveny 2003] E Valveny and E Marti. *A model for image generation and symbol recognition through the deformation of lineal shapes.* Pattern Recognition Letters, vol. 24, no. 15, pages 2857–2867, 2003. 144

[Valveny 2004] Ernest Valveny and Philippe Dosch. *Symbol Recognition Contest: A Synthesis.* Graphics Recognition, pages 368–385, 2004. 142, 144, 180, 181, 195, 206

[Valveny 2007] E Valveny, P Dosch, Adam Winstanley, Yu Zhou, Su Yang, Luo Yan, Liu Wenyin, Dave Elliman, Mathieu Delalandre, Eric Trupin, Sébastien Adam and Jean-Marc Ogier. *A general framework for the evaluation of symbol recognition methods.* International Journal on Document Analysis and Recognition, vol. 9, no. 1, pages 59–74, March 2007. 142, 145

[Vapnik 1982] Vladimir Vapnik. Estimation of Dependences Based on Empirical Data. Empirical Inference Science. Series: Information Science and Statistics, 1982. 169

[Vaxiviere 1992] P Vaxiviere and K Tombre. *Celestin: CAD conversion of mechanical drawings.* IEEE Comput, vol. 25, pages 46–54, 1992. 17, 55, 116

[Vidal 1994] Enrique Vidal. *New formulation and improvements of the nearest-neighbour approximating and eliminating search algorithm (AESA).* Pattern Recognition Letters, vol. 15, no. 1, pages 1–7, 1994. 170

[Wagner R.A. 1974] Fisher M J Wagner R.A. *The string-to-string correction problem.* Journal of the ACM, pages 168–173, 1974. 206

[Wagner 1974] R A Wagner and M J Fischer. *The String-to-string Correction Problem.* Journal of the Association for Computing Machinery, vol. 21(1), pages 168–173, 1974. 133, 135

[Wall 1984] Karin Wall and Per-Erik Danielsson. *A fast sequential method for polygonal approximation of digitized curves.* Comput. Vision Graph. Image Process., vol. 28, no. 3, pages 220–227, 1984. 146

[Wang J.T.L. 2002] Zhang K Chang G Shasha D Wang J.T.L. *Finding approximate patterns in undirected acyclic graphs.* Pattern Recognition, vol. 35, pages 473–483, 2002. 208

[Wenyin 1997] Liu Wenyin and Dov Dori. *A protocol for performance evaluation of line detection algorithms.* Machine Vision and Applications, vol. 9, no. 5, pages 240–250, 1997. 115, 116, 117, 118

[Wilson 1996] R C Wilson and E R Hancock. *A Bayesian Compatibility Model for Graph Matching.* Pattern Recognition Letters, vol. vol. 17, pages pp. 263–276, 1996. 166

[W.Y. Ma B. Manjunath 1997] Netra W.Y. Ma B. Manjunath. *A toolbox for navigating large image databases.* Proceedings of the IEEE International Conference on Image Processing, pages 568–571, 1997. 194

[Y. I. Ohta T. Kanade 1980] Y. I. Ohta T. Kanade and T Sakai. *Colour information for region segmentation.* Computer Graphics and Image Processing, vol. 13, pages 222–241, 1980. 33

[Y. Rui T. Huang 1997] Y. Rui T. Huang and S Chang. *Image retrieval Past, present, and future.* In International Symposium on Multimedia Information Processing, 1997. 194

[Y. Rui T. Huang 1999] Y. Rui T. Huang and S Chang. *Image retrieval: current techniques, promising directions and open issues.* Journal of Visual Communication and Image Representation, vol. 39-62, 1999. 194

[Yang 1998] Jihoon Yang and Vasant Honavar. *Feature subset selection using a genetic algorithm.* IEEE Intelligent Systems, vol. 13, pages 44–49, 1998. 36, 38

[Yz] L Lucchese Yz and S K Mitra Y. *Color Image Segmentation: A State-of-the-Art Survey.* In Procedings of the Indian National Science Academy(INSA-A), vol. vol. 67, pages A, pp. 207,221. 202

[Zhang K. 1992] Statman R Shasha D Zhang K. *On the editing distance between unordered labeled trees.* Information Processing Letters, vol. 42, pages 133–139, 1992. 208

[Zhang 1989] Kaizhong Zhang and Dennis Shasha. *Simple fast algorithms for the editing distance between trees and related problems.* SIAM Journal on Computing, vol. 18(6), pages 1245–1262, 1989. 208

[Zhang 1996] Kaizhong Zhang. *A Constrained Edit Distance Between Unordered Labeled Trees.* Algorithmica, vol. 15, pages 205–222, 1996. 208

Fouille de graphes et classification de graphes : Application à l'analyse de plans cadastraux.

Résumé: Les travaux présentés dans ce mémoire de thèse abordent sous différents angles très intéressants, un sujet vaste et ambitieux : l'interprétation de plans cadastraux couleurs. Dans ce contexte, notre approche se trouve à la confluence de différentes thématiques de recherche telles que le traitement du signal et des images, la reconnaissance de formes, l'intelligence artificielle et l'ingénierie des connaissances. En effet, si ces domaines scientifiques diffèrent dans leurs fondements, ils sont complémentaires et leurs apports respectifs sont indispensables pour la conception d'un système d'interprétation. Le centre du travail est le traitement automatique de documents cadastraux du 19e siècle. La problématique est traitée dans le cadre d'un projet réunissant des historiens, des géomaticiens et des informaticiens. D'une part nous avons considéré le problème sous un angle systémique, s'intéressant à toutes les étapes de la chaîne de traitements mais aussi avec un souci évident de développer des méthodologies applicables dans d'autres contextes. Les documents cadastraux ont été l'objet de nombreuses études mais nous avons su faire preuve d'une originalité certaine, mettant l'accent sur l'interprétation des documents et basant notre étude sur des modèles à base de graphes. Des propositions de traitements appropriés et de méthodologies ont été formulées. Le souci de comblé le gap sémantique entre l'image et l'interprétation a reçu dans le cas des plans cadastraux étudiés une réponse.

Mots clés: Théorie et modèles pour la reconnaissance de formes en document, Analyse de plans et reconnaissance de graphiques, Extraction et structuration d'informations graphiques, Recherche/fouille d'information dans les images de documents, Classification de graphes, Évaluation de performances.

Domaine de recherche: code 22 : Graphes, combinatoire, complexité, code 92 : Vision par ordinateur, reconnaissance de formes, code 91 : Traitement et analyse d'images, de son, de signaux (codes fournis par la nomenclature thématique section 27 du CNU).

Graph Mining and Graph Classification: Application to cadastral map analysis.

Abstract: This thesis tackles the problem of technical document interpretation applied to ancient and colored cadastral maps. This subject is on the crossroad of different fields like signal or image processing, pattern recognition, artificial intelligence, man-machine interaction and knowledge engineering. Indeed, each of these different fields can contribute to build a reliable and efficient document interpretation device. This thesis points out the necessities and importance of dedicated services oriented to historical documents and a related project named ALPAGE. Subsequently, the main focus of this work: Content-Based Map Retrieval within an ancient collection of color cadastral maps is introduced.

Keywords: Graph classification, graph-based representation, graph-mining, graphics recognition, color map, cadastral map, map interpretation, contextual information modeling.