Semantic Description of Humans in Images

Gaurav Sharma

UNIVERSITÉ DE CAEN BASSE NORMANDIE

U.F.R. de Sciences
ÉCOLE DOCTORALE SIMEM

T H È S E

Présentée par

M. Gaurav SHARMA

soutenue le

17 Décembre 2012

en vue de l'obtention du

DOCTORAT de l'UNIVERSITÉ de CAEN

Spécialité : Informatique et applications

Arrêté du 07 août 2006

Titre :

Description Sémantique des Humains Présents dans des Images Vidéo

(Semantic Description of Humans in Images)

The work presented in this thesis was carried out at
GREYC - University of Caen and LEAR – INRIA Grenoble

Jury

M. Patrick PEREZ	Directeur de Recherche	INRIA/Technicolor, Rennes	*Rapporteur*
M. Florent PERRONNIN	Principal Scientist	Xerox RCE, Grenoble	*Rapporteur*
M. Jean PONCE	Professeur des Universités	ENS, Paris	*Examinateur*
Mme. Cordelia SCHMID	Directrice de Recherche	INRIA, Grenoble	*Directrice de thèse*
M. Frédéric JURIE	Professeur des Universités	Université de Caen	*Directeur de thèse*

A picture is a poem without words
– Horace

Abstract

In the present thesis we are interested in *semantic description of humans in images*. We propose to describe humans with the help of (i) semantic attributes *e.g.* male or female, wearing a tee-shirt, (ii) actions *e.g.* riding a horse, running and (iii) facial expressions *e.g.* smiling, angry.

First, we propose a new image representation to better exploit the class specific spatial information. The standard representation *i.e.* spatial pyramids, has two shortcomings. It assumes that the distribution of spatial information (i) is uniform and (ii) is same for all tasks. We address these shortcomings by learning the discriminative spatial information for a specific task. Further, we propose a model that adapts the spatial information for each image for a given task. This lends more flexibility to the model and allows for misalignments of discriminative regions *e.g.* the legs may be at different positions, in different images for running class. Finally, we propose a new descriptor for facial expression analysis. We work in the space of intensity differences of local pixel neighborhoods and propose to learn the quantization of the space and use higher order statistics of the difference vector to obtain more expressive descriptors.

We introduce a challenging dataset of human attributes containing 9344 human images, sourced from the internet, with annotations for 27 semantic attributes based on sex, pose, age and appearance/clothing. We validate the proposed methods on our dataset of human attributes as well as on publicly available datasets of human actions, fine grained classification involving human actions and facial expressions. We also report results on related computer vision datasets, for scene recognition, object image classification and texture categorization, to highlight the generality of our contributions.

Keywords

Computer vision • Image understanding • Semantic attributes • Human attributes • Facial analysis • Image classification • Face verification • Expression classification • Discriminative saliency

Contents

Chapter 1

Introduction

Contents

1.1 Motivation

Internet has become a gigantic source of rich multimedia content. The emergence of cheap and easy to use digital cameras and websites, hosting practically unlimited amount of user generated videos and images, has led to an explosion in the amount of digital content online. To give an idea of the scale, about 48 hours of video are uploaded every minute to Youtube, which makes about 8 years of video per day[1], and Facebook had almost 90 billion photos, with 200 million more uploaded every day, in early 2011[2].

Another major source of digital video is surveillance. Numerous digital cameras, estimated in millions for some countries[3], are working round the clock in many different public as well as private places *e.g.* streets, airports, supermarkets. Which means that millions of digital images are generated every second as a result of surveillance.

[1] `http://www.youtube.com/faq`, accessed July 23, 2012

[2] `http://www.quora.com/How-many-photos-are-uploaded-to-Facebook-each-day`, data provided by a Facebook Photos engineer on Jan 25, 2011, accessed on July 23, 2012

[3] `http://en.wikipedia.org/wiki/Mass_surveillance`, accessed on July 23, 2012

Many of the internet videos and images feature important events and activities in the lives of the users (*e.g.* vacations, weddings, graduations) and as people are the main subject of surveillance, a large amount of this digital visual data is human focused.

The presence of huge amount of human focused images (and videos) underlines the need of methods to automatically analyze these images. Such methods could be applied to numerous tasks *e.g.* organizing images, inferring about images, and retrieving images based on descriptions. In particular inferring high level semantic knowledge about the person in a still image is of wide interest. In web scale databases, this would allow for indexing and searching of images and video with high level queries *e.g.* you could search for a 'young woman wearing shorts and running' or 'kid riding a bike and smiling'. In surveillance scenarios, such capability can prove very useful as generally people can only describe the suspects approximately. So if the database has been indexed for the semantic content of the frames the system can quickly retrieve a 'middle aged man wearing a jacket'. Figure 1.1 illustrates the point with example images.

1.2 Problem addressed

In the present thesis we address the task of *semantic* description of humans in images. If we parse short sentences which describe humans *e.g.* Figure 1.1, we are very likely to have the humans as the *subjects*, actions as the *verbs e.g.* 'running', 'sitting', and attributes and emotions as *complements e.g.* 'is a girl', 'is wearing shorts', 'is happy', 'is angry'. Hence, we propose to semantically describe humans with the help of high level concepts based on actions, attributes and face expression. We pose the problems as supervised classification (see next section) where we have images annotated with their actions/attributes/expressions and we predict similar properties for new test images. In the following we first give an overview of the computer vision technology to analyze images, then discuss briefly the works related to the present thesis, to set the context. Finally, we proceed to present the main contributions made in this thesis.

1.3 Automatic image understanding

Automatic image understanding is a major research problem in computer vision. The goal is to analyze an image and be able to make inference based on it *e.g.* given an image of a person, infer what is she doing (*e.g.* [19, 20, 21, 35, 40, 41, 42, 74, 106]). Earlier most of such inference was based on text analysis methods [80] and relied on the (noisy) annotations that came with the image. These annotations could be the tags added by the user or the caption for the image or just the text surrounding the image on a web page. This is changing fast and computer vision technologies that analyze the *content*

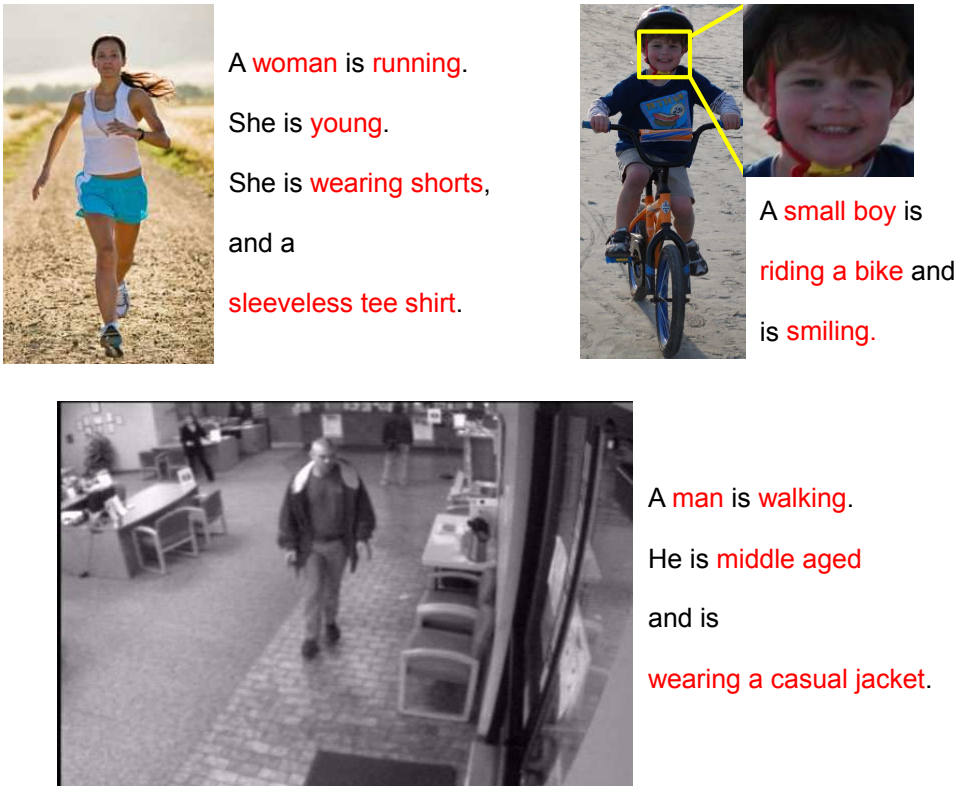

*Figure 1.1: Semantic description of humans in still images can be based on various aspects such as overall appearance (*e.g. *wearing shorts, jacket, tee-shirt), sex (male or female) or based on the action the person is performing (*e.g. *running, riding a bike) or based on expressions inferred from just the face of the person (*e.g. *smiling, angry)*

of the image, instead of the peripheral noisy text, and make inferences are replacing or augmenting the past systems (*e.g.* [4, 5, 16, 18, 25, 27, 28, 31, 37, 45, 51, 50, 58, 68, 86]). Systems are now capable of analyzing images and retrieve information about them *e.g.* Google Goggles[4] can process an image of a landmark like Eiffel tower and retrieve information about it such as its historical and cultural significance.

1.3.1 Supervised learning

The problem of automatic image understanding is usually posed as a supervised learning problem [7, 22, 79] *i.e.* the system is given *training* images with annotations about their properties and, when presented with a *test* image that it has never seen before, it has to predict the presence or absence of similar properties. Supervised *classification* is one type of supervised learning problems; here the task is to classify the images into one of the classes *e.g.* given images of persons with annotations about their gender (*i.e.* 'male' or 'female' classes), the system classifies new images into one of these two classes. In general, there can be more than just two classes *e.g.* human action recognition, is the

[4] `http://www.google.com/mobile/goggles/`

person 'riding a horse' or 'running' or 'sitting' and the images can belong to more than one class simultaneously. Formally, the goal of supervised classification is to learn a mapping, colloquially called a classifier, which takes the numerical representation of an image as input and maps it to a class (represented as a numeric value).

1.3.2 Image representation

The capability of a classification system depends on those of its two main components *i.e.* the image representation and the classifier. On one hand, the image representation should lead to similar representations for images of the same class, despite of the intra-class variations, and dissimilar representations for those of different classes, despite of inter-class similarity. While on the other, the classifier should be strong enough to perform well, even when the representation scheme is only able to capture the (dis)similarities relatively weakly.

Digital systems, like computers and digital cameras, represent and store images as two dimensional matrices of pixels where each pixel is a vector of numeric values (usually integers). If the image is grayscale, the vector is one dimensional with the only value indicating the intensity of the pixel (between a fixed minimum value, usually 0, for black and a fixed maximum value, usually 255, for white pixel). If the image has colors, assuming the pixels are coded in Red-Green-Blue (RGB[5]), the vector is three dimensional with each value similarly indicating the intensity of the red, green and blue colors respectively. The final color of the pixel is obtained by mixing the RGB colors with those intensities.

Many vision systems, such as those doing face recognition[6], work directly with the pixel representation of the images (*e.g.* [3, 89]). They take the whole image as the input vector (pixels stacked sequentially) to their learning system. This works in controlled scenarios, where generally a cooperative subject is assumed and the face images are frontal and well aligned. With such constraints, the images of the same person are numerically quite similar and there is no (or easy to eliminate) background clutter to distract the classifier.

However, in the more general case of classification of images from uncontrolled environments *e.g.* images from internet taken in arbitrary conditions, the direct representation with pixels leads to poor performance. In this case, the image description has to be designed to be *invariant* to common variations in similar images *e.g.* different illumination conditions resulting from images, of same subject, taken at different times of the day, while being *covariant w.r.t.* the differences in images from different classes.

Towards achieving such invariances, image representation could be designed to capture the *global* properties of images *e.g.* distribution of colors/edges/filter responses [92] or

[5]The pixels can be encoded in different color spaces, see `http://en.wikipedia.org/wiki/Color_space`

[6]Face recognition is the problem where given a face image the system has to identify the person from among the database of known persons

could be designed to capture aspects localized in space *e.g.* histogram of oriented gradients (HOG) computed over smalls blocks of pixels [18], derived from description of shapes [4] or local gradients [2, 58]. Noting that many such diverse image representations, both global and local [57, 62, 112], have been used in computer vision systems, we now describe the representations which have been quite successful and widely adopted and are most relevant to the present work.

Bag-of-Features

Current state-of-the-art image classification systems follow methods inspired from text document analysis and represent images with the so called *Bag of Features* (BoF) representation [16, 86]. In the BoF approach, first, multiple local patches are extracted from the image (either by random sampling or on a regular grid) and are represented numerically, as vectors, by non-linear transformations of their pixels [2, 4, 58]. Such transformations (*e.g.* Scale Invariant Feature Transform, SIFT [58]) are designed to be invariant to common variations *e.g.* those caused due to changes in illumination and/or affine transformations. The local patches from all the training images are then vector quantized into a codebook of typical patches or the *visual words*, using some clustering algorithm *e.g.* k-means. Finally, the image is represented as the histogram of assignments of all its local patches over the visual words, analogous to the Bag-of-Words histograms in text analysis.

Spatial pyramid

An important relaxation of the BoF representation is that it ignores the spatial information as it considers the local patches without considering their spatial position in the image. This is counter intuitive as the spatial information is expected to be important for visual tasks, specially for human attributes. Many different methods have been proposed to incorporate the spatial information into the BoF representation (we discuss them in appropriate detail in the relevant chapters) but the most adopted method of doing so is the Spatial Pyramid Representation (SPR) [50]. SPR works by dividing the image spatially with uniform grids at multiple resolutions and then concatenate the BoF histograms of the spatial cells with appropriate normalizations. Doing this encodes the rough spatial layout into the overall representation. This simple way of incorporating spatial information has proved to be very effective and efficient. SPR not only beats the simple BoF by a large margin but also performs competitively against much elaborate methods.

Local patterns

Another successful image representation is that using Local Binary Patterns [68] (and extension to Local Ternary Patterns [87]). The representation considers patterns of pixel neighborhoods as small as 3×3 pixel patches. As the focus is on capturing the texture, the local pixel patches are first made invariant to monotonic changes in intensities by subtracting the center pixel from the rest. They are then represented as binary vectors by thresholding the values at zero. The image is, finally, represented as a histogram over *uniform* patterns *i.e.* patterns which have at most one 0-1 and at most one 1-0 transitions in bits when viewed as a circular bit string, while all the non-uniform patterns are discarded. The use of uniform pattern is based on the empirical observation that uniform patterns are dominant among all the binary patterns. Despite of being very simple the representation achieves very good performance for texture classification and has been used successfully on facial analysis and object localization as well.

1.3.3 Overview of related computer vision literature

To set the general context in which the present thesis is developed, we now discuss a representative sampling of computer vision tasks and methods.

Many interrelated computer vision problems like

- *Categorization:* Predict the classes *e.g.* 'indoors', 'street', 'forest', 'beach', to which the images belongs,
- *Semantic annotation:* Suggest tags for images based on the semantics of the image *e.g.* 'sunny', 'christmas', 'police',
- *Segmentation:* Label image regions which represent distinct components *e.g.* 'sky', 'grass', 'building',
- *Object localization:* Predict the positions and scales at which given objects, *e.g.* 'dog', 'bottle', 'person', appear in the images,
- *Pose/viewpoint estimation:* Predict the viewpoint or the pose of the object present in the image,

etc. have lately attracted much research attention. All these tasks are important towards achieving automatic understanding of digital images for indexing, searching, retrieval, inferring semantic knowledge from the actual content of the images *etc.* Such capabilities are, in turn, useful in many applications like forensics (*e.g.* face recognition, automatic video surveillance), robotics (*e.g.* terrain identification for autonomous exploration) and many other consumer applications (*e.g.* searching images in personal and public image collections, duplicate image filtering).

Recent works on object categorization and localization can be broadly divided into two groups:

- The first group of methods describe the objects as a sparse collection of local features. Fergus *et al.* [31] describe the objects with probabilistic generative models, the *Constellation of Parts models*, and learn their parameters by maximizing the likelihood of the training images. In similar spirit, Leibe *et al.* [52] represent objects with *Implicit Shape models* which are probabilistic extension of generalized Hough transforms. They aim to learn the prototypical local appearances of object 'parts' along with their spatial arrangements.

- The second group of methods derive object representations based on statistics of densely sampled local features. Different works propose different combinations of local features and statistics. Local features aim to capture one of the many visual properties *e.g.* appearance [2, 58], shape [4, 18] and texture [68, 87]. The statistics used to describe the distribution of features vary from histograms [16, 86] to higher-order Fisher scores [71, 72].

In practice, systems have to combine various features, capturing complimentary information, to achieve state-of-the-art results [25, 90].

Many works report methods for pose estimation of articulated objects, especially humans. They handle the task by learning a model over the different 'parts' of the object, encoding their different aspects *e.g.* appearance and spatial arrangements. *Pictorial structures model*, originally proposed by Fischler and Elschlanger [32], has motivated many successful recent models [28, 29, 77, 76]. The main idea in these works is to model the object as a star model over its different parts. In particular, *Deformable Part models (DPM)* [28] have been very successful in object localization and have become a part of almost all state-of-the-art systems [25]. DPM models objects' parts and allow then to move (deform) to accommodate the variability in deformable objects. This gives them the generating capabilities to account for unseen poses of articulated objects. They are formulated as latent Support Vector Machine (SVM) [28, 79] problem and are discriminatively trained in a supervised manner. Extensions and adapted versions of DPM have also been proposed for other tasks *e.g.* scene recognition [69], pose estimation [107], face detection and landmark localization [116].

In the closely related task of human action recognition from images, works have reported that using the traditional bag-of-features representations gives good results while the DPM do not perform well [19]. Matching human pose for recognizing actions has also been explored [106, 109]. Other works have exploited relations between the person and objects [20, 74, 108, 110] with varying success.

We postpone further discussion of related methods, specific to our work, till the respective chapters, and present our contributions in the following section.

1.4 Contributions

The work presented in this thesis is mainly concerned with the image representation stage of the vision system architecture.

Human actions and attributes can have highly localized discriminative characteristics *e.g.* for wearing shorts we need to focus on the legs while ignoring the upper body[7]. To leverage such locality, Chapters 2 and 3 concentrate on the improving the incorporation of spatial distribution of discriminative information.

In spatial pyramid representation (SPR) the spatial partitioning is taken as uniform and same for all classes. In Chapter 2 we propose to learn the spatial partitioning for a given classification task. This addresses two limitations of SPR *i.e.*

(i) The learnt grids are able to better exploit the locality of the discriminative information and

(ii) The grids can vary and adapt for the different tasks for which they are learnt.

Learning such spatial representations turn out to be favorable specially at lower vector lengths.

Learning grids per task is interesting as it allows per class adaptation of spatial partitions for capturing discriminative information. However, such information may vary slightly among different images of the same class *e.g.* when looking for 'bent arms' two persons may have arms bent differently in two different images. To address this issue, in Chapter 3, we propose to learn discriminative saliency per image for a given classification task. Our saliency modeling does both per class and per image adaptation of discriminative spatial information and hence, is more flexible and powerful.

Studying generic attributes for humans is a relatively recent topic in computer vision and hence there are no standard datasets for benchmarking the methods. To fill this gap, as another contribution, we propose a new challenging database of Human Attributes (HAT). HAT database has more than nine thousand human images taken from unconstrained images downloaded automatically from the internet. It contains annotations for twenty seven human attributes based on sex, pose, age and appearance of the humans. We present the dataset in detail in Section 2.3 of Chapter 2.

As the final contribution, in Chapter 4, we propose a new image representation, which we call Local Higher-Order Statistics (LHS). LHS improves over local binary/ternary patterns (LBP/LTP) in two ways,

(i) LBP/LTP perform a fixed quantization of the local pattern space by choosing to quantize each coordinate into two/three bins by thresholding, and do a heuristic

[7]Relatively speaking, as there might be some correlation between shorts and certain types of upper body clothes

pruning of the space by discarding non uniform patters, while LHS learns the quantization of the space from the data and

(ii) LBP/LTP use only low order statistics of the data *i.e.* histograms, while LHS uses higher order statistics.

Thus, LHS leads to a more expressive representation which improves performance.

Chapter 2

Discriminative Spatial Representation

Contents

2.1 Introduction

In visual classification tasks, the spatial information is important *e.g.* for predicting if a person is 'wearing a sleeveless t-shirt' we should be focusing on the upper part of the image, which is likely to contain the shoulders, instead of the lower part (see Figure 2.1). In this chapter, we address the problem of incorporating spatial layout information relevant to a given classification task [83].

Figure 2.1: In visual classification task the spatial information is important. Eg. for 'coastal' scene category the sky, beach/sea layout is similar across images, for 'car' object category the cars are expected to appear in similar locations and scales and for 'wearing a sleeveless T-shirt' attribute we need to look only at the upper part of the image.

Image representation is a fundamental problem in computer vision. Recent works have established, somewhat surprisingly, the bag-of-features (BoF) representation [16] as being an effective representation for various computer vision tasks *e.g.* object recognition, object detection, scene classification etc. Briefly, in the BoF approach local patches are first described as feature vectors [58, 2], then they are vector quantized and the image is represented as the histogram, of all its local patch features, over the quantization codebook. This is parallel to the bag-of-words approach in text analysis and hence the name.

The main drawback of the BoF approach is the loss of spatial information in the coding. As one would expect, the spatial information is important for visual classification tasks and it has been recently shown [50, 55, 64, 75, 78, 113] that adding spatial information to the standard BoF improves performance. Among these one of the most popular representation incorporating spatial information is the Spatial Pyramid Representation (SPR) by Lazebnik *et al.* [50]. SPR incorporates spatial information of the features by dividing the image into uniform grids at different scales and then concatenating the BoF features from the different grid cells with appropriate normalizations. Coupled with discriminative maximum margin based classifiers [79], it has become the standard representation

and has been shown to perform competitively with more complex representations and models for many tasks [8, 72, 105] including human action recognition [19].

However, the choice of spatial partitioning *i.e.* uniform grids (at different scales *i.e.* 2×2, 4×4), does not have any particular theoretical or empirical motivation *i.e.* there have been no systematic exploration of the space of partitions and the grids have been derived out of practitioners' experience. The choice of partitioning is expected to be important for the task *e.g.* a partition with prominently horizontal cells for 'coastal scene' (with beach, sea and sky) and one with prominently vertical cells for 'tall buildings' (both of these classes are part of the public benchmark Scene-15 dataset). Also, for cases where the discriminative information is localized, the grids could be relatively finer in the important regions. We would expect it to be specially important for the recognition of human attributes in human centered images *e.g.* in case of the 'wearing shorts' attribute, the partitioning in the middle part of the human is expected to be discriminant.

In this chapter we propose to learn the spatial partitioning for a given classification task. We define the space of grids (Section 2.2.1) as the set containing grids generated by recursive splitting of grid cells by axis aligned cuts (starting with the full image as the only cell). We then formulate the classification problem (Section 2.2.2) in the maximum margin framework and perform optimization over both the weight vector *and* the grid parameters. We propose an efficient approximate algorithm (Section 2.2.3, Alg. 2.1) to perform the optimization and show experimentally (Section 2.4) that the learnt grids perform better than the standard SPR while leading to vectors smaller (as much as half) in length to the SPR. We also introduce a challenging dataset (Section 2.3) of human attributes (based on age, sex, appearance and pose) containing real world images collected from image sharing site Flickr.com. We demonstrate the relevance of learning the grids on such cases where the discriminating information is spatially localized.

2.1.1 Related works on spatial representations for BoF

The current state-of-the-art methods for object/scene recognition are built upon the bag-of-features (BoF) representation of Csurka *et al.* [16]. The representation works by extracting local features (*e.g.* SIFT [58], SURF [2]) from the images, vector quantizing them (*e.g.* using k-means clustering) and then representing images as histograms over the quantization codebook or the *visual words*. Thus, in the BoF representation the spatial layout is completely discarded as the local features only describe appearances of the local patches and have no spatial information.

Various methods have been proposed to incorporate spatial layout in the BoF representation. These methods can be grouped into roughly two classes. First, the methods which encode position of local features relative to other local features and, second, those which encode the absolute positions of the local features.

Among the first class of methods, Savarese *et al.* [78] proposed to form a bag-of-word representation over spatially close image regions, Liu *et al.* [55] used a feature selection method based on boosting which progressively mines higher-order spatial features, while Morioka *et al.* [64] proposed joint feature space clustering to build a compact local pairwise codebook and in another work [65] incorporated the spatial orders of local features. Quack *et al.* [75] suggested finding distinctive spatial configurations of visual words using data mining techniques.

In addition to pairwise relationships, images often have global spatial biases *i.e.* the composition of the images of particular object or scene category typically share common layout properties. This is especially true for the recognition of attributes in human centered images (see Figure 2.1).

A pioneering works in the direction of exploiting absolute spatial layout of features was the Spatial Pyramid Representation (SPR) by Lazebnik *et al.* [50]. In SPR, the image is divided into uniform grids at different scales *i.e.* 2×2, 4×4, and the features are concatenated over all cells with appropriate normalization. SPR, working at spatial level rather than feature level, improved the BoF performance by a significant margin.

More recently, Yang *et al.* [104] showed that incorporating sparse coding into the SPR improves performance. Cao *et al.* [11] projected local features of an image to different directions or points to generate a series of ordered bag-of-features. Zhou *et al.* [113] modeled region appearances with a mixture of Gaussian (MoG) density and used the posterior over visual words for the image regions to generate so called 'Gaussian maps', encoded by SPR. Very recently, Harada *et al.* [36] divided images into a regular grid, and learned weight maps for the grid cells.

In practice, many state-of-the-art classification methods are based on the SPR [24, 25], but the different parameters involved, the number of pyramid levels and the structure of the grid at each level, are *empirically* adapted to the situation *e.g.* [50, 104] use up to 4 pyramid levels with uniform grids of 1×1, 2×2, 4×4 and 8×8, while the winner of Pascal VOC 2007 competition, Marszalek *et al.* [60] followed by many others such as [114, 105], use three pyramid levels with grids of 1×1, 2×2 and 3×1. The SPR parameters are chosen in an ad-hoc manner and no work reports systematic construction of the representation.

The method proposed in this chapter addresses this issue and learns a representation where the parameters are learnt for the given task.

2.2 Discriminative Spatial Representation

As discussed above, while the Spatial Pyramid Representation (SPR) [50] has been very successful in the task of visual classification, the spatial grids are fixed and are the same for all the classes, which is a significant limitation of the approach. We propose to learn

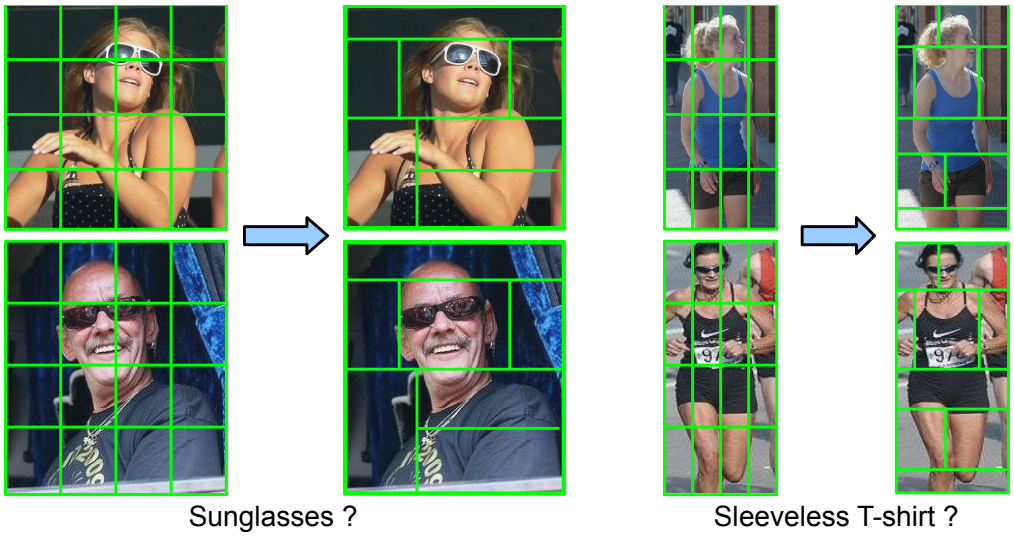

Figure 2.2: In Spatial Pyramid Representation (SPR) [50] the spatial grids (i) are uniform and (ii) are same for all tasks. We propose to learn grids adapted to the task, better capturing the regions in a way that benefits the classifier performance.

the spatial partitioning for the given classification task, by defining a space of spatial grids and learning the best grid over this space for the task. Learning such grids is expected to be beneficial as they can help the classifier focus better on the regions which are relatively more discriminant *e.g.* in Figure 2.2, the uniform grids treat both attributes, 'wearing sunglasses' and 'wearing a sleeveless T-shirt', similarly and spatially uniformly while the grids could be adapted, as illustrated, to focus on the important regions for the respective tasks.

We use an image representation similar to the original Spatial Pyramid Representation [50]. We represent images by quantizing SIFT [58] features using a dictionary learnt with the k-means algorithm. The SIFT descriptors themselves are computed on patches sampled densely, with overlap, over uniform grids at multiple scales. Given a spatial partitioning of the image (e.g. uniform grid of 2×2) we construct spatial histogram for the image and use it as the image representation.

We define a grid as a mapping $g : \mathcal{I} \rightarrow \mathcal{R}^d$, where $\mathcal{I}$ is the set of all images and $\mathcal{G}$ is the set of all possible spatial grids (we define this space in the next Section 2.2.1). Any grid g thus maps an image $I \in \mathcal{I}$ to its spatial bag-of-features histogram $g(I) \in \mathcal{R}^d$. We denote the dot product between two vectors a and b as $a \cdot b$.

We can write the scoring function, *w.r.t.* a linear hyperplane classifier with normal vector w, as

$$f(I) = w \cdot \hat{g}(I) + b, \tag{2.1}$$

where $\hat{g}(I) \in \mathcal{R}^d$ is the histogram feature obtained by applying the best grid $\hat{g} \in \mathcal{G}$, learned for the task, to the image $I \in \mathcal{I}$. Unlike spatial pyramid, we use only the fi-

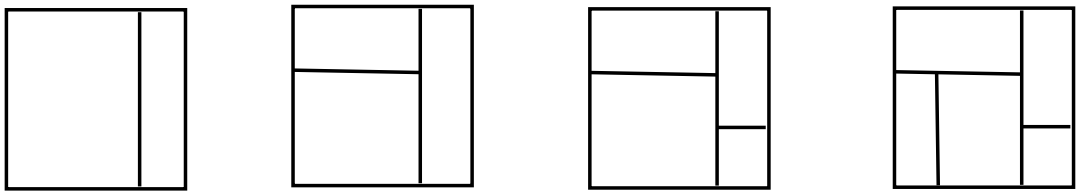

Figure 2.3: The formation of the spatial grid by successive splitting of cells; grid with depth 1 (left) to depth 4 (right)

nal grid obtained after optimization and not a pyramid as we expect the grid to adjust its resolution depending on the spatial distribution of discriminative information for the class.

2.2.1 The space of grids

Although, to incorporate spatial information, any partition of the image space with arbitrary number and type of curves could be considered, we work in a restricted but sufficiently general space of grids. We consider all grids containing cells obtained with only vertical and/or horizontal straight lines.

Formally, we define the space of grids $\mathcal{G}$ by construction. Starting with the full image as the grid with one cell, we recursively split the cells further, into two parts each, with axis aligned straight lines (Figure 2.3). Thus, the usual spatial pyramid grids of 2×2, 4×4 and 8×8 partitions are members of the considered space of grids $\mathcal{G}$, and so is the finest grid possible for a digital image *i.e.* one which isolates every pixel in the image.

We split the space of grids into disjoint parts as $\mathcal{G} = \cup_k\{\mathcal{G}_k\}$, with each subset $\mathcal{G}_k$ containing the grids obtained with exactly k successive splits. We call the number of splits taken to obtain the grid as the *depth* of the grid and represent the grid as a set of cells $g = (g_1, g_2, \ldots, g_{k+1})$ with $g_i = (x_1^i, y_1^i, x_2^i, y_2^i) \in \mathcal{R}^4$ representing the i^{th} cell in the grid. Here $x, y \in [0, 1]$ are fractional multiples of the image width and height respectively. We write g^k for a grid g where we want to make explicit the depth k (note that a grid with depth k has $k + 1$ cells, the full image is obtained with a grid of depth 0). In theory the splits can occur continuously anywhere between 0 to 1 (in fractional multiple of image height/width), while in practice they are quantized.

2.2.2 Learning the grids

We formulate the learning problem in a maximum margin framework, with slack variables,

$$\min_{w,g} \frac{1}{2}||w||^2 + C\sum \xi_i \tag{2.2}$$

$$\text{s.t.} \quad y_i(w \cdot g(I_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0,$$

where i indexes over the set of training images. Here, along with optimization on the separating hyperplane (normal) $w \in \mathcal{R}^d$, we are optimizing on the grid $g \in \mathcal{G}$. While the w vector lies in the familiar d dimensional real space $\mathcal{R}^d$, the space of grids $\mathcal{G}$ is new and the issue now is how do we explore this space. To this end, we propose an efficient approximate solution to the problem using block coordinate descent like iterations with greedy forward selection. The block coordinate descent step helps us separate the optimizations over w and g and the greedy forward selection, along with the definition of our grid space, helps us optimize over the space of grids efficiently albeit approximately.

2.2.3 Dual form

The dual of the optimization problem (2.2) with Lagrange's multipliers, $\alpha = \{\alpha_i\}$ for separation constraints and $\mu = \{\mu_i\}$ for constraints on non-negativity of slack variables $\{\xi_i\}$, is a saddle point problem given by

$$\min_{w,g} \max_{\alpha,\mu} \frac{1}{2}||w||^2 + C\sum \xi_i - \sum_i \alpha_i[y_i(w \cdot g(x_i) + b) - 1 + \xi_i] - \sum_i \mu_i \xi_i \qquad (2.3)$$

$$\text{s.t.} \quad \alpha_i \geq 0, \quad \mu_i \geq 0$$

The dual formulation allows us to propose an efficient approximate optimization strategy (Alg. 2.1) based on two popular methods, block coordinate descent like iterations and greedy forward selection. We treat the SVM parameters α and the grid parameters g as two sets of variable on which we do block coordinate descent like iterations to find the best grid for a fixed depth. To increase the depth of the grid we resort to greedy forward selection, computing the next best split using gradient based optimization. The numerical gradient is computed efficiently using integral histograms and matrix dot products.

In the block coordinate descent iterations, when the optimization is on the SVM parameters and the grid is fixed, we recover the usual SVM dual optimization,

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \;\; g(I_i) \cdot g(I_j) \qquad (2.4)$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq C \text{ and } \sum_i \alpha_i y_i = 0,$$

where we have used standard algebraic manipulations *e.g.* Section 3.5 in [10].

2.2.4 Gradient based optimization for learning grid

Now we consider the block coordinate descent step where the SVM parameters are kept constant. When we keep α constant, we use numerical gradient to optimize efficiently

Algorithm 2.1 Computing the grid $g^K \in \mathcal{G}^K$ for the given classification task

```
Initialize the grid g^0 ∈ G^0 to be the full image.
for k=1 ... K do
    for all grid cells do
        Initialize {s, α} by choosing s randomly and optimizing (2.4)
        while convergence or maxiters do
            Optimize (2.5) w.r.t. s (keeping α fixed) using the gradient in (2.7)
            Optimize w.r.t. α (using efficient linear SVM solvers) keeping s fixed
        end while
    end for
    g^{k+1} ← (g^k, s*), s* being the best grid cell split
end for
```

for the gird at the given depth. The coordinate descent step becomes an unconstrained optimization problem with the objective

$$F(g) = -\frac{1}{2}\sum_{i,j} t_{ij} \;\; g(I_i) \cdot g(I_j), \tag{2.5}$$

where $t_{ij} = \alpha_i \alpha_j y_i y_j$ are constants depending on the current α and the training labels y and we have omitted terms not depending on g. When a further split is made the new histogram differs only in the part of the image which was split (Figure 2.4). The gradient of the objective function depends on the gradient of the linear kernel function parametrized by the split parameter s. Considering two images I_i and I_j with histograms $g(I_i) = x$ and $g(I_j) = y$, we can write the kernel function between them as

$$k_s(x, y) = \begin{pmatrix} x_s \\ x_o \end{pmatrix} \cdot \begin{pmatrix} y_s \\ y_o \end{pmatrix} = x_s \cdot y_s + x_o \cdot y_o, \tag{2.6}$$

where for histogram $x = (x_s x_o)^T$, x_s is the part of the histogram which is affected by the split while x_o is the part which isn't. Thus, we calculate the numerical gradient for the kernel function, when we take a step from s to s' (Figure 2.4), as

$$\begin{aligned}
\Delta_s k_s(x, y) &= x_{s'} \cdot y_{s'} - x_s \cdot y_s \\
&= \frac{1}{N_x}\begin{pmatrix} c_1^x - c_\Delta^x \\ c_2^x + c_\Delta^x \end{pmatrix} \cdot \frac{1}{N_y}\begin{pmatrix} c_1^y - c_\Delta^y \\ c_2^y + c_\Delta^y \end{pmatrix} - \frac{1}{N_x}\begin{pmatrix} c_1^x \\ c_2^x \end{pmatrix} \cdot \frac{1}{N_y}\begin{pmatrix} c_1^y \\ c_2^y \end{pmatrix} \\
&\propto (c_1^x - c_\Delta^x) \cdot (c_1^y - c_\Delta^y) + (c_2^x + c_\Delta^x) \cdot (c_2^y + c_\Delta^y) - (c_1^x \cdot c_1^y + c_2^x \cdot c_2^y) \\
&\propto (c_1^x \cdot c_1^y - c_1^x \cdot c_\Delta^y - c_\Delta^x \cdot c_1^y + c_\Delta^x \cdot c_\Delta^y) + \\
&\quad (c_2^x \cdot c_2^y + c_2^x \cdot c_\Delta^y + c_\Delta^x \cdot c_2^y + c_\Delta^x \cdot c_\Delta^y) - (c_1^x \cdot c_1^y + c_2^x \cdot c_2^y) \\
&\propto 2\, c_\Delta^x \cdot c_\Delta^y - c_1^x \cdot c_\Delta^y - c_\Delta^x \cdot c_1^y + c_2^x \cdot c_\Delta^y + c_\Delta^x \cdot c_2^y
\end{aligned} \tag{2.7}$$

where, c_1 c_2 and c_Δ are the histograms (un-normalized raw counts) of different parts involving the split as shown in Figure 2.4. The gradient for objective function in (2.5) is

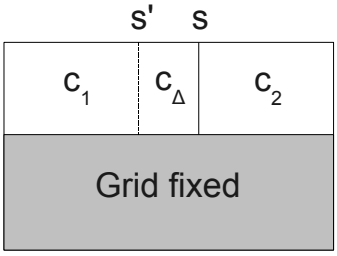

Figure 2.4: The histograms (raw counts) when a step is taken from the current split s to a new split s'. c_1 and c_2 are the histograms for the two parts generated by split s and c_Δ is the histogram for the part between split s and s'.

the sum of these gradients, for all pairs, weighted by t_{ij}.

The first cost associated in computing the gradient is the calculation of the histogram c_Δ for the changing part of the grid. The step sizes are quantized and hence the calculation of the count histogram is accelerated using integral histograms for the grid induced by the quantized step sizes. The second is the computation of the dot products of matrices, which is also performed efficiently using optimized matrix algebra libraries.

The split can occur in any of the cells of the grid *i.e.* a depth d grid has $d+1$ cells and the further split can occur in any of them. Since, we can't consider the splits in the different grid cells as being instances of a single variable, we run the one dimensional optimization separately for every grid cell and take the cell split increasing the objective function the most.

The other coordinate descent step involves training linear SVM which, owing to recent progress, is also achieved efficiently. Hence, both the steps are fast and the overall optimization is quite efficient in practice.

2.3 Database of Human Attributes (HAT)

We propose a new database of Human Attributes (HAT) for learning semantic human attributes. The database is publicly available on the internet[1]. Our database contains 9344 images and has annotations for 27 attributes.

The images in the database were collected from the internet. To obtain a large number of images we used an automatic program to query and download the top ranked result images, from the popular image sharing site Flickr.com, with manually specified queries. We used about 320 queries, chosen to retrieve predominantly images containing people (e.g. 'soccer kid' cf. 'sunset'). Appendix B gives the list of the queries used. We then ran state-of-the-art person detector [28] to obtain the human images and removed the few false positives manually.

[1] http://sharma.users.greyc.fr/hatdb/

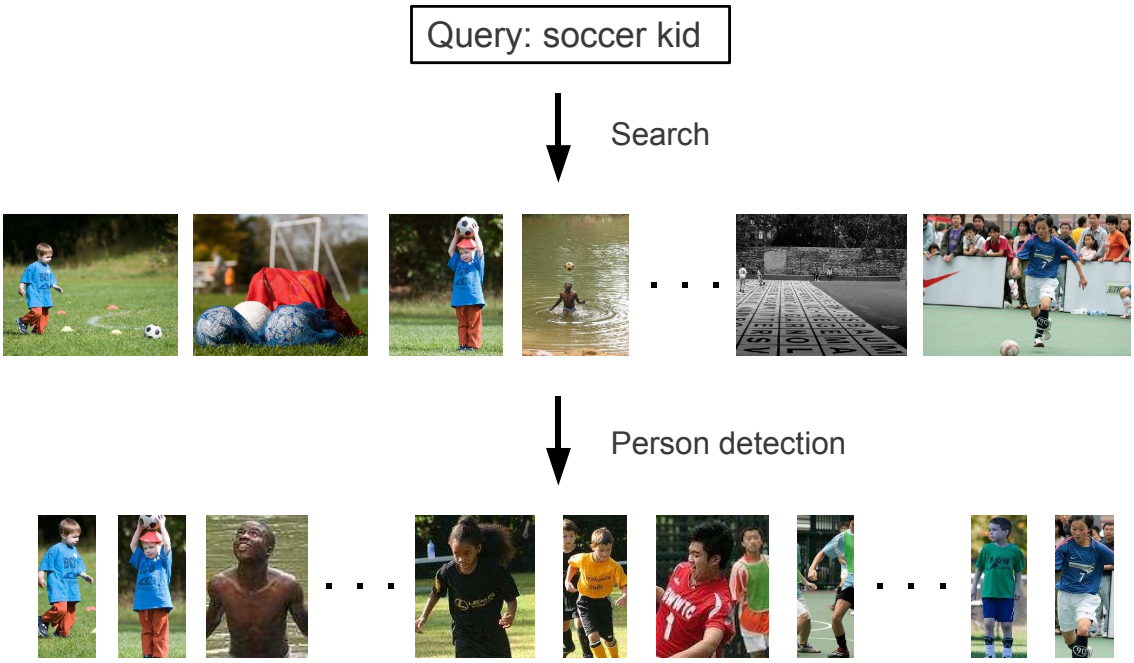

Figure 2.5: Illustration of the database creation process. A query is specified for searching on Flickr and the top result images are saved. The images are then passed to the person detection module and the person detections are then kept. The images here have been scaled to the same height for better visualization.

The database contains a wide variety of human images in different poses (standing, sitting, running, turned back etc.), of different ages (baby, teen, young, middle aged, elderly etc.), wearing different clothes (tee-shirt, suits, beachwear, shorts etc.) and accessories (sunglasses, bag etc.) and is, thus, rich in semantic attributes for humans. It also has high variation in scale (only upper body to the full person) and size of the images. The high variation makes it a challenging database. Figure 2.6 shows some example images from our dataset and Figures 2.7, 2.8 and 2.9 show some example images for some of the attributes (the images in the figures are scaled to the same height for visualization). Table 2.1 lists the various attributes present in our database along with the number of positive and negative annotated images for each attribute.

The database has been divided into `train`, `val` and `test` sets. The models are learnt with the `train` and `val` sets while the average precision for each attribute, on the `test` set, is reported as the performance measure. The overall performance is given by the mean average precision over the set of attributes.

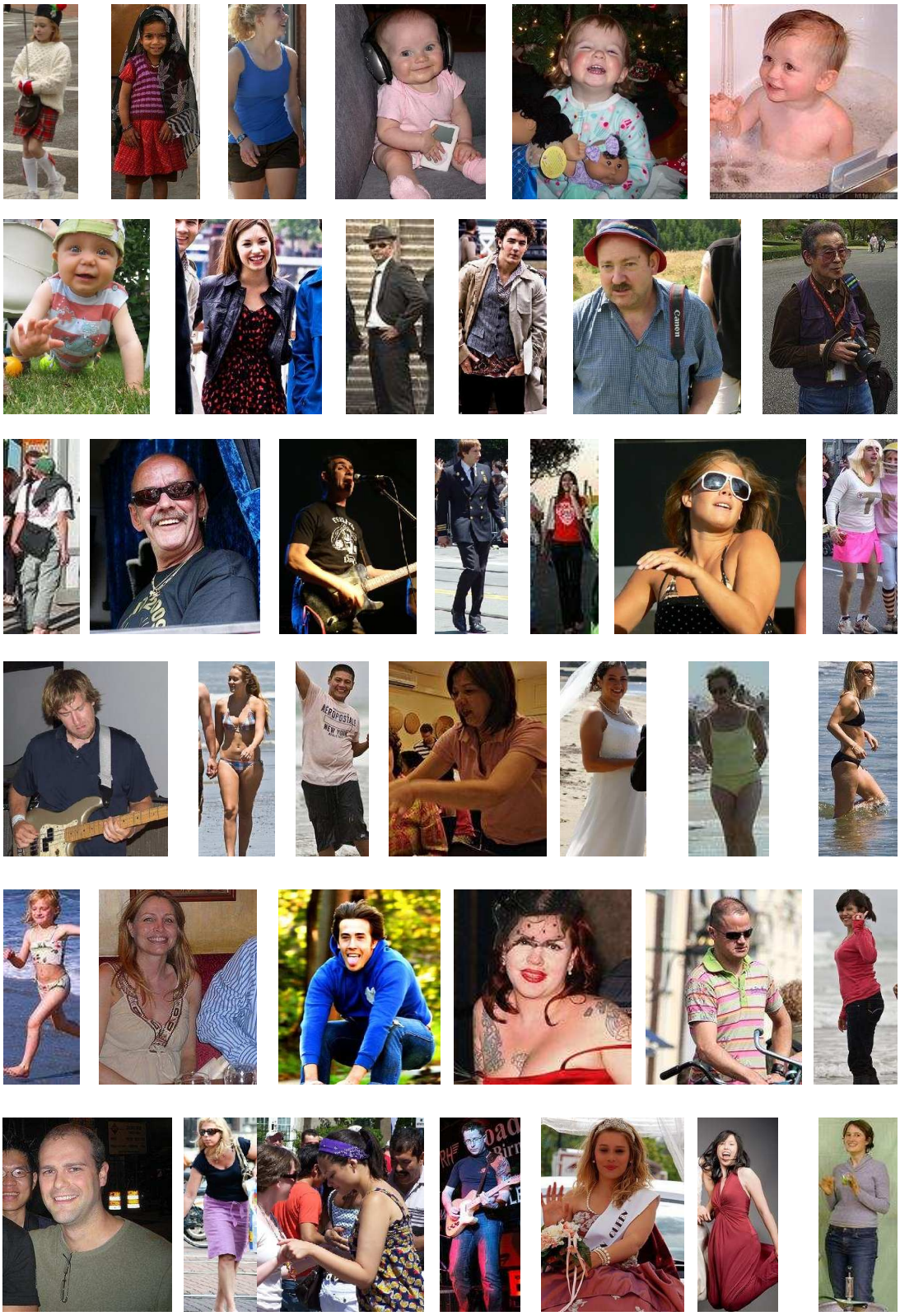

Figure 2.6: Example images from our database. The images are scaled to the same height for better visualization.

Figure 2.7: Example images for age based attributes from our database. The images are scaled to the same height for better visualization.

2.4 Experiments and results

The motivation of these experiments is twofold: first, we show that optimizing the spatial representation lead consistently to better results that the standard SPR, as demonstrated on two popular databases (Scene 15 and Pascal VOC 2007). Second, we show that the proposed representation is especially suited for the recognition of human attributes, problem for which we introduce a new database (presented above in Section 2.3).

2.4.1 Implementation details

We use the standard bag of features (BoF) with spatial pyramid representation (SPR) [50] as the baseline. We use multiscale SIFT features extracted at 8 scales separated by a factor of 1.2 and a step size of 8 pixels. We randomly sample 200,000 SIFT vectors from the training images and learn a quantization codebook using k-means with 1000 clusters. Finally, we use nonlinear SVM with histogram intersection kernel in a one-vs-all setting to perform the classification. Note that we use formulation (2.4) to learn the grids which is equivalent to using a linear kernel but we use nonlinear histogram intersection kernel for training the final SVM to compare fairly with the SPR baseline. We now introduce the other datasets we used and then proceed to give the results.

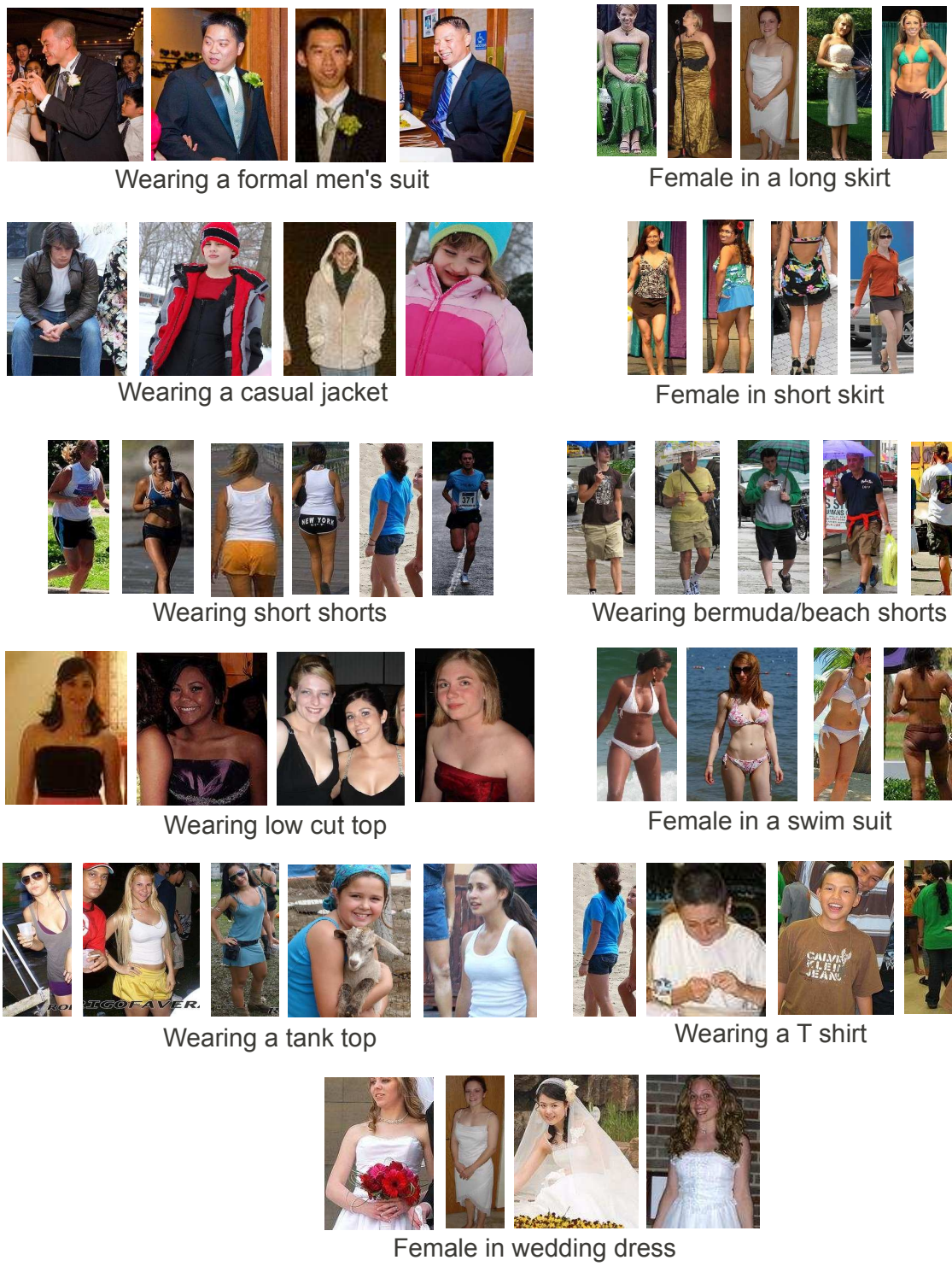

Figure 2.8: Example images for appearance/clothes based attributes from our database. The images are scaled to the same height for better visualization.

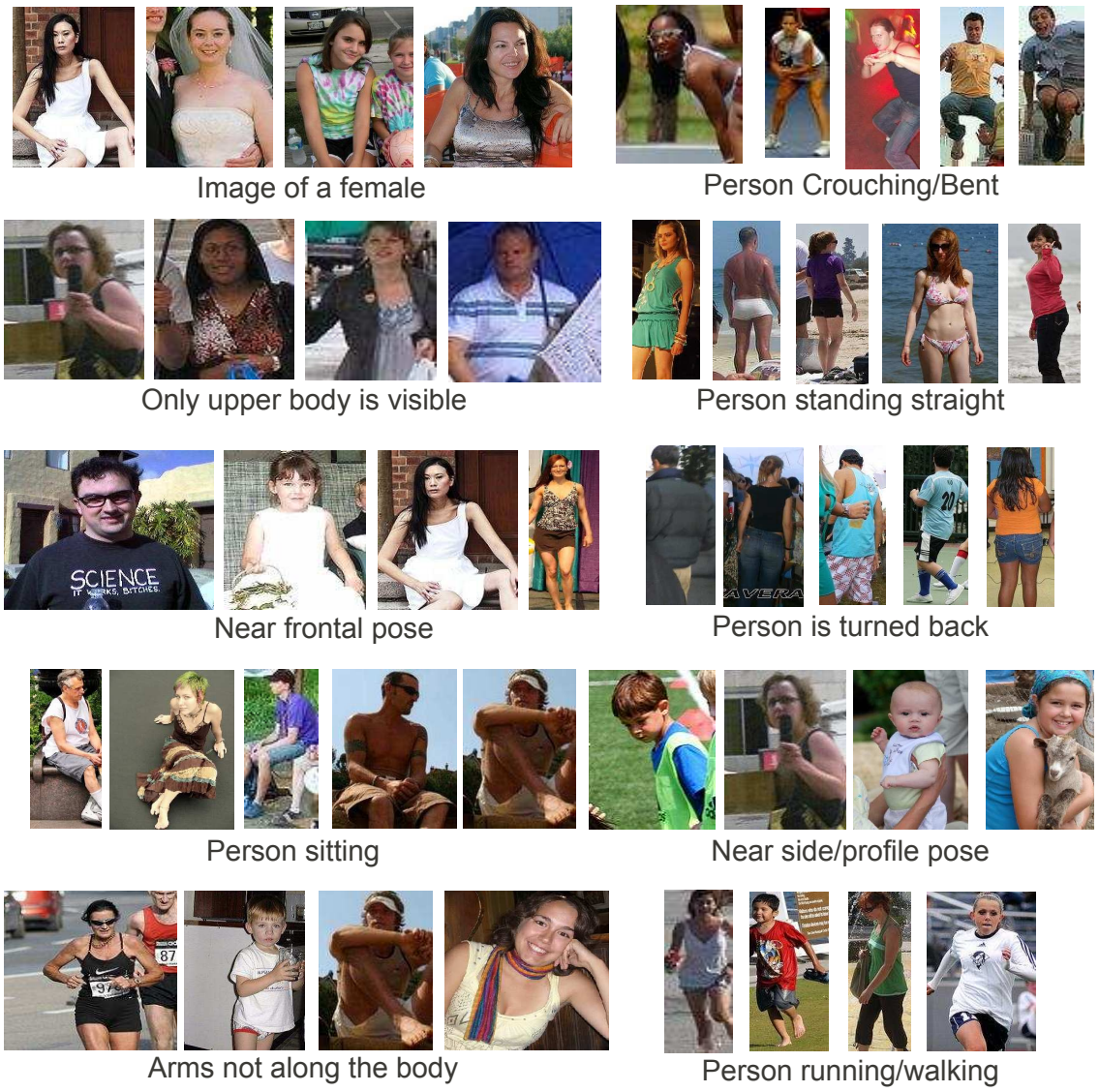

Figure 2.9: Example images for sex and pose based attributes from our database. The images are scaled to the same height for better visualization.

2.4.2 Datasets

Scene 15 database[2] contains 15 classes of different scenes e.g. *kitchen, coast, highway*. Figure 2.10 shows some example images from the dataset. Each class has 260 to 410 images and the database has a total of 4492 grayscale images. The problem is of multiclass categorization and, like previous works, we train on 100 random images per class and test on the rest. We do so 10 times and report the mean and standard deviation of the mean class accuracy.

[2]http://www.featurespace.org/data.htm

Table 2.1: The various attributes along with the number of positive and negative images for them in our database of Human Attributes (HAT)

No.	Attribute	Positive	Negative
1	Image of a female	4282	4664
2	Near frontal pose	7048	2219
3	Near side/profile pose	2119	7042
4	Person is turned back	922	8354
5	Only upper body visible	3752	5295
6	Person standing straight	4607	1367
7	Person running/walking	1264	7533
8	Person crouching/bent	177	6446
9	Person sitting	787	5750
10	Arms not along the body	5543	1788
11	Elderly person	546	7958
12	Middle aged person	4671	4521
13	Young college person	3364	5402
14	Teen aged kid	1541	7480
15	Small kid	1274	8057
16	Small baby	223	9121
17	Wearing a tank top	854	8470
18	Wearing a T shirt	3004	5870
19	Wearing casual jacket	766	8571
20	Wearing formal men's suit	618	8696
21	Female in long skirt	436	7048
22	Female in short skirt	423	7029
23	Wearing short shorts	422	5203
24	Wearing a low cut top	1590	7572
25	Female in swim suit	234	9012
26	Female in wedding dress	121	9107
27	Wearing bermuda/beach shorts	672	4221

Pascal VOC 2007 database[3] [23] has 20 object categories. It is a challenging database of images downloaded from internet, containing 9963 images split into train, val and test sets. Figure 2.11 shows some example images from the dataset. We use the train and val sets to learn our models and report the mean average precision for the 20 classes on the test set as the performance measure, following the standard protocol for this database.

2.4.3 Results

Comparison with standard SPR on Scene 15 and Pascal VOC 2007

Figure 2.12 shows the performance of the learnt grid and the uniform spatial pyramid representation on the Scene 15 database. With our implementation, the spatial pyramid

[3]http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007/

Table 2.2: Table showing the classwise average precision for the human attributes with the learnt grids at depths 0 and 4

No.	Attribute	Depth 0	Depth 4
1	Female	72.5	82.0
2	Frontal pose	90.1	91.3
3	Side pose	48.8	61.0
4	Turned back	49.5	67.4
5	Upper body	83.1	92.4
6	Standing straight	95.6	96.0
7	Running/walking	61.3	67.6
8	Crouching/bent	21.6	20.7
9	Sitting	52.2	54.6
10	Arms bent/crossed	92.0	91.9
11	Elderly	21.9	29.3
12	Middle aged	63.2	66.3
13	Young (college)	59.0	59.4
14	Teen aged	25.2	29.1
15	Small kid	33.5	43.7
16	Small baby	12.6	12.2
17	Wearing tank top	29.2	33.2
18	Wearing tee shirt	54.8	59.1
19	Wearing casual jacket	31.3	35.3
20	Formal men's suit	44.4	48.2
21	Female long skirt	23.1	49.9
22	Female short skirt	27.3	33.7
23	Wearing short shorts	38.6	42.7
24	Low cut top	47.8	55.6
25	Female in swim suit	29.0	28.2
26	Female wedding dress	51.3	62.1
27	Bermuda/beach shorts	31.6	39.3
	mAP	47.8	53.8

representation achieves a mean class accuracy of 73.7 ± 0.7 at pyramid level 0 (full image *i.e.* 1×1), 78.5 ± 0.4 at level 1 (1×1 and 2×2) and 79.6 ± 0.6 at level 2 (1×1, 2×2 and 4×4). The performance decreases for SPR if we go higher than this level. As shown in Figure 2.12 the learnt grids achieve higher performance with comparable vector sizes and outperform the SPR at depth as low as 4 (80.1 ± 0.6). The performance of the learnt grids increases quickly with the depth and saturates at around a depth of 8 which is 0.4 times the length of best SPR (depth 21). Note that the vectors are computed similarly for both representations and hence have similar sparsities *i.e.* the difference in vector sizes translates directly into computational savings.

On the more challenging VOC 2007 database where objects appear at diverse scales, locations and poses, the learnt grids again outperform SPR at lower grid depths and perform comparably at higher grid depths. The performance of most of the classes, and on a av-

Figure 2.10: Some example images for the Scene 15 database [50]

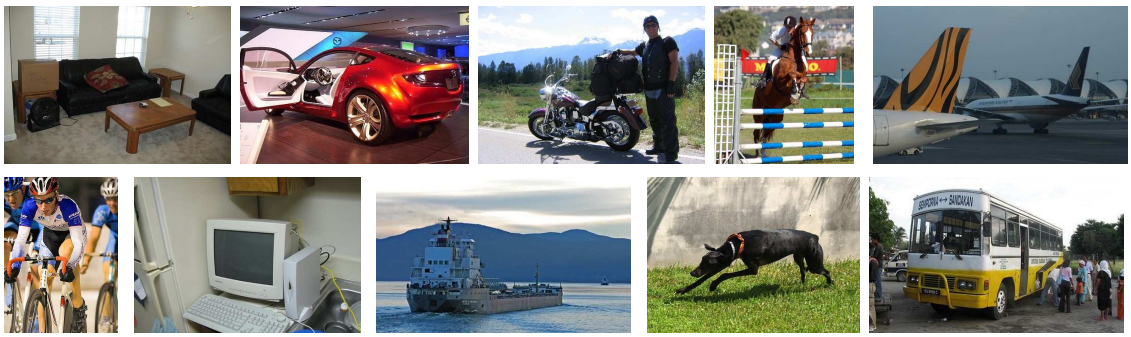

Figure 2.11: Examples images from the Pascal VOC 2007 database [23]

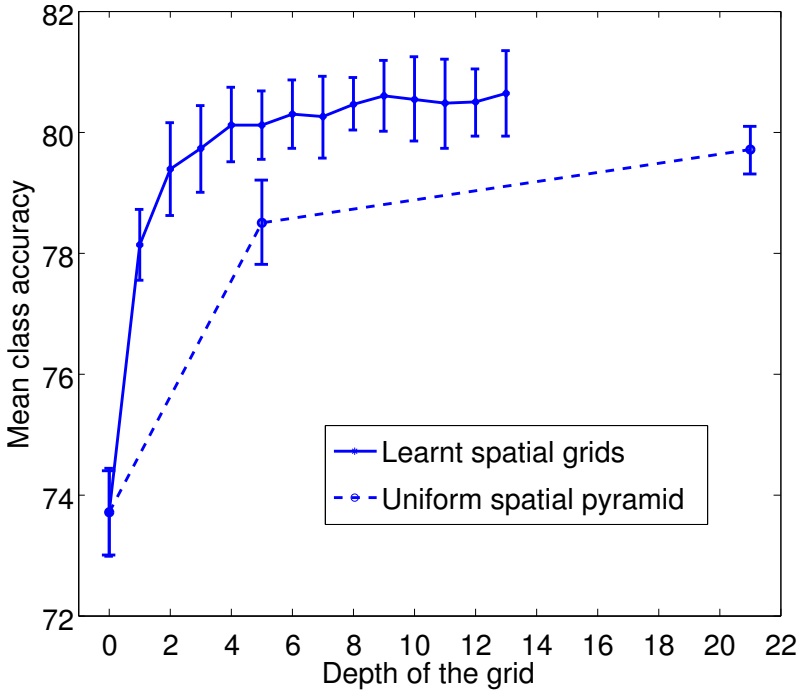

Figure 2.12: The performances of SPR and learnt grid at comparable vector lengths for Scene 15 database

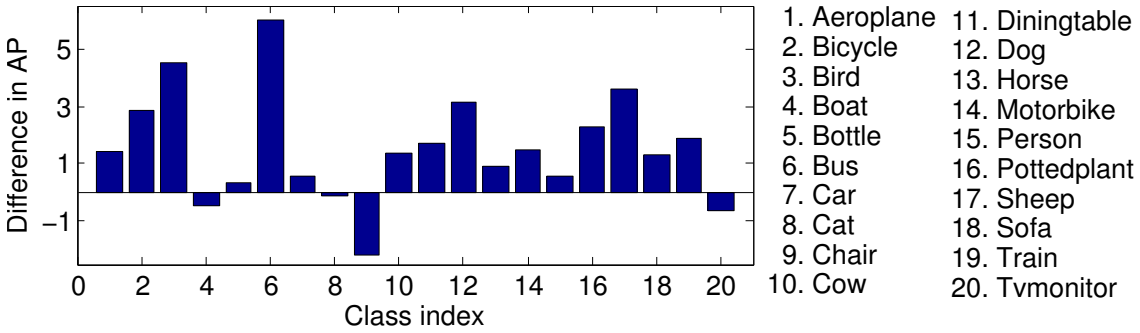

Figure 2.13: The difference in AP for all the classes of the VOC 2007 database at a grid depth of 4 with the learnt grid and the uniform spatial pyramid

erage is higher (50.8 vs. 49.5) for the learnt grids at depth 4 (Figure 2.13). Owing to the unconstrained nature of the images in the database, the structural information is limited in this database. Also, the metric used is average precision while we are optimizing on accuracy in the maximum margin formulation. It would be interesting to formulate the problem with average precision being maximized directly. We hope to pursue this further.

Recognizing human attributes

Table 2.2 shows the average precision of the the learnt grids for the different attributes at grids of depth 0 and 4. The learnt grids perform better than the SPR on most of the classes and also on a average. On some of the classes the improvement is quite high *e.g.*

Relative dimension → Attribute name	d=1 Depth 0	d=5 Depth 4	d=21 Full SPR
Group 1			
Running/walking	61.3	67.6	70.6
Sitting	52.2	54.6	58.4
Wearing tank top	29.2	33.2	36.8
Low cut top	47.8	55.6	59.6
Group 2			
Side pose	48.8	61.0	60.1
Female	72.5	82.0	82.0
Female long skirt	23.1	49.9	50.2
Wearing tee shirt	54.8	59.1	60.1
Group 3			
Small baby	12.6	12.2	12.4
Standing straight	95.6	96.0	96.4
Frontal pose	90.1	91.3	92.1
Arms bent/crossed	92.0	91.9	93.2

Table 2.3: We observe three groups of attributes. Group 1*: The distribution of spatial information is very peaky in these and they gain performance when the resolution of the grid increases to high levels.* Group 2*: The distribution of spatial information is relatively less peaky compared to Group 1 and they gain performance when the resolution of the grid increases to an intermediate level but do not gain performance at higher resolutions.* Group 3*: The distribution of spatial information is almost flat and they do not gain any performance upon increasing the resolution of the grids increases to high levels.*

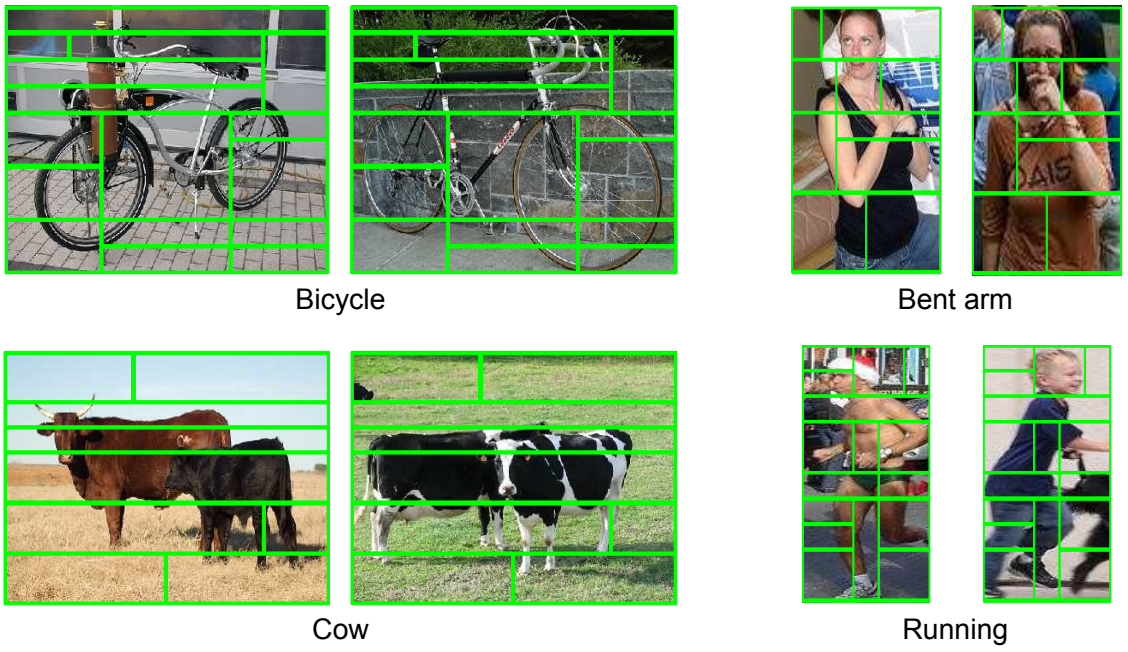

Figure 2.14: Learnt grids for VOC 2007 classes 'bicycle' and 'cow' and human attributes 'arms bent' and 'running' overlayed on representative example images.

49.9 vs. 42.7 for 'Female wearing a long skirt' and 62.1 vs. 51.9 for 'Female in wedding dress'. On an average also the learnt grids are better than the SPR (53.8 vs. 52.3).

Interestingly, we see three groups of attributes emerging from the experimental results (Table 2.3). In the first group of attributes, the distribution of spatial information is very peaky *i.e.* the discriminant regions are smaller and they gain performance when the depth of the grid increases from 0 to 4 and also when it further increases. In the second group of attributes, the distribution of spatial information is relatively flat, compared to the first group, and they gain performance when the resolution of the grid increases to an intermediate level but saturate there. In the third group of attributes, the distribution of spatial information is almost flat and they do not gain any appreciable performance upon increasing the resolution of the grids to more than one cell.

Visualizing the learnt grids

The grids learnt are interpretable in terms of spatial distribution of visual discriminant information. Figure 2.14 shows the grids from two classes of VOC 2007 and two classes of the human attributes database overlayed on representative example images from the database. The grid learnt for bicycle class seems to focus on the wheels with square cells in the middle and the bar with horizontal cells towards the top. The cells for the cow class are predominantly horizontal capturing the contour of the cow. The grids for the bent arm and running classes seem to focus on the pose of the hands and feet respectively.

2.5 Discussion and conclusion

The method proposed in this chapter builds on the Spatial Pyramid Representation of [50] and addresses fundamental limitations of this approach *i.e.* the fixed structure of the SPR and no class adaptation. We have proposed an efficient algorithm, based on a maximum margin framework, allowing to adapt the spatial partitioning to the classification tasks considered. Furthermore, we have experimentally showed that our representation significantly outperforms the standard SPR specially at lower vector length.

The experiments demonstrate that spatial information is very important for visual classification tasks. Our method is able to capture such information and adds performance to the bag-of-features (BoF) representation. However, the proposed method models the spatial information at the class level *i.e.* the grid learnt is *w.r.t.* a given classification task and is the same for all the images. Thus, the way spatial information is added does not natively allow the image parts to move in individual images. *E.g.* assuming that (a part of) the grid for 'running' attribute captures the bent legs, ideally it should adapt itself to a given image according to how the leg is positioned in that specific image. This relaxation,

of learning the grid for the class without per image adaptation, is still justified as the underlying representation for the cells is an orderless BoF histogram. Hence, given that the coarseness of the grid proportional to the spatial variability in individual discriminant regions, the representation is able to capture the informative part despite it moving for different images.

Making the grids adapt to each image is an interesting idea which could be pursued in a latent SVM [28] like framework. However, since the grids themselves are nontrivial to handle in an optimization framework, in the next chapter we propose a new method in the spirit of per image adaptation but with discriminative saliency maps instead of spatial grids.

Chapter 3

Discriminative Spatial Saliency

Contents

3.1 Introduction

In the previous chapter we proposed a method which incorporates spatial information in the bag-of-features (BoF) representation and demonstrated, experimentally, the benefits of doing so. In the previous method the spatial information was learnt per class, while in the present chapter we propose a method to incorporate spatial information, for the classification task, and let it adapt to the given image for better modeling the per-image spatial variation of similar discriminant information [84].

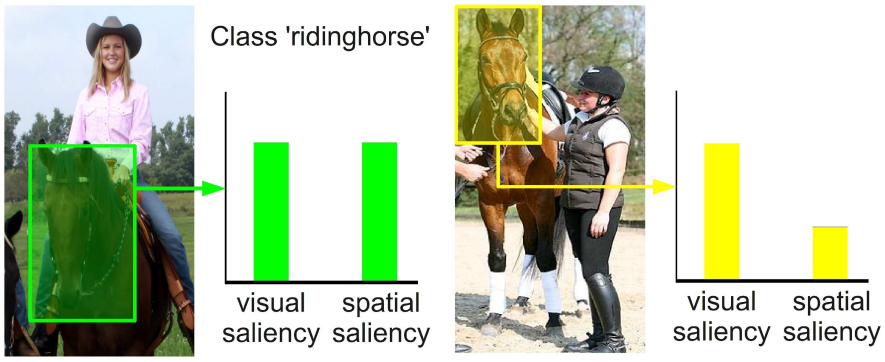

*Figure 3.1: Illustrating the importance of spatial saliency. A horse is salient for the 'ridinghorse' class. However, it is salient if it appears in the lower part of the image (*e.g. *left image), but not if it appears in some other part of the image (*e.g. *right image).*

The human visual system is capable of analyzing images quickly by rapidly changing the points of visual fixation. Estimating the distribution of such points i.e. the *visual saliency* is an important problem in computer vision [43, 47, 66, 99]. Initial works on visual saliency detection addressed generic saliency, highlighting (generally interesting) properties *e.g.* edges, contours, color, texture, building on the feature integration theory [47, 88]. For visual discrimination, generic visual saliency should be adapted to include task specific information. Many works [33, 34, 63], thus, define and compute saliency based on the *discriminative* power of local features *i.e.* how much does a feature contribute towards separating the classes. Such feature based discriminative saliency has been shown to be important in automatic visual analysis.

Furthermore, in many visual classification tasks there is a spatial bias which complements global feature saliency *e.g.* for the 'coast' class in scene classification, sky-like regions are salient, not everywhere but in the *upper part* of an image. Thus, we argue that given a class, visual saliency is attributed to different local regions based on their appearance *and* their spatial location in an image *i.e.* a task specific *spatial saliency* is associated with each image.

In the present chapter, we

(i) Extend the notion of discriminative visual saliency by including discriminative *spatial* information and

(ii) Learn it, together with the classifier, to obtain a more discriminative image representation for visual classification.

Contrary to previous works [33, 34, 43, 46, 47, 63] that use saliency of features, irrespective of their positions, we work with saliency of regions in space *i.e.* for the 'riding horse' class instead of saying 'look for horse like features' we say 'look for horse like features *in the lower part of the image*'. Figure 3.1 illustrates the point and Figure 3.2 shows saliency maps obtained by our method.

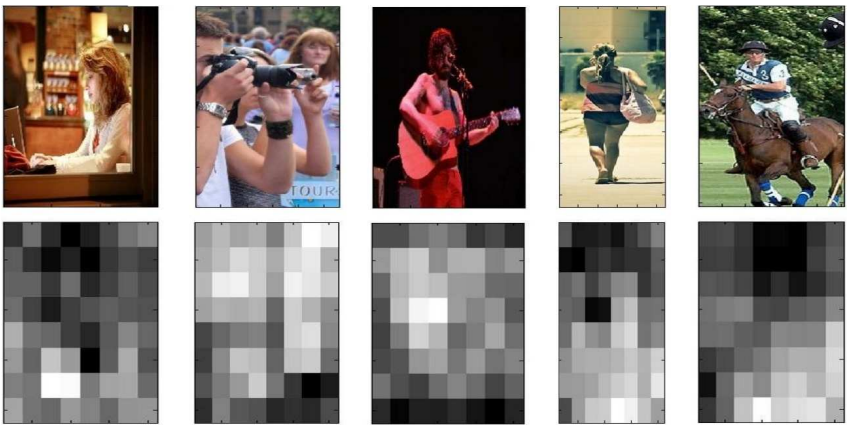

Figure 3.2: Example images and their spatial saliency maps obtained with our algorithm for 'interacting with computer', 'taking photo', 'playing music', 'walking' and 'ridinghorse' action classes (higher values are brighter).

Our definition of saliency is closely coupled with learning the classifier, unlike previous work which learn the saliency map and the classifier separately [33, 34, 70]. We learn the classifier while simultaneously modeling saliency in an integrated max margin learning framework. We formulate saliency in terms of local regions, and the learning based on a latent SVM framework adapted to incorporate the saliency model. We show that our saliency model improves results on three challenging datasets for

(i) Human action classification in still images [20]

(ii) Fined grained image classification i.e. persons playing vs. holding musical instruments [111] and

(iii) Scene classification [50].

3.1.1 Related work on visual saliency and image classification

Visual saliency has been investigated in the computer vision literature in many different ways. Salient local regions have been detected using interest points (*e.g.* [58, 61]) which can be made invariant to image transformations (*e.g.* rotation, scale, affine) and, thus, can be detected reliably and repeatably. They have been very successful for matching images under different transformations [58, 61]. Such regions were also used to sample small sets of salient patches from images for classification with bag-of-features representations [16], but dense (regular or random) sampling has been shown to perform better [67] and is currently the state-of-the-art [24].

Biologically inspired saliency, based on the feature integration theory [88], motivated another line of work. Regions were marked as salient depending on the difference with their surrounding area [43, 47], measured using low level features *e.g.* edges, texture, contours. Such generic saliency was further adapted to discriminative saliency [33, 34,

46, 63], where, given a visual classification task, saliency was defined by the capability of the features to separate the classes.

Moosmann *et al.* [63] learn saliency maps for visual search to improve object categorization. They construct the saliency maps by random sampling of image windows and classifying them as object or background. Gao and Vasconscelos [33] formulate discriminative saliency and determine it based on feature selection in the context of object detection [34]. Parikh *et al.* [70] learn saliency in an unsupervised manner based on how well a patch can predict the locations of others. Khan *et al.* [46] model color based saliency to weight features for improving object detection. Harada *et al.* [36] learn weights on regions for classification. However, they learn the weights per class *i.e.* the weights are the same for all images. Yao *et al.* [111] learn a classifier with random forests. They mine salient patches, for the decision trees, by randomly sampling patches and selecting the most discriminative ones.

We model saliency based on the contribution of regions to classification i.e. our saliency is discriminative. We do not discard features, but weight them using the saliency map, which differs from *e.g.* [34, 67, 70]. Our model incorporates saliency modeling into the learning of separating hyperplane in a max margin framework. Hence, our saliency is more tightly coupled with the visual discrimination task unlike many previous works where learning saliency and classifiers are separate steps *e.g.* [33, 34, 46, 70].

Recently, latent support vector machine (LSVM) classifiers have shown promise in many visual tasks. Felzenszwalb *et al.* [28] use LSVM for part based object detection which has become a standard component in state-of-the-art systems [24]. Bilen *et al.* [6] model the position and size of the objects using LSVM for image classification. We adapt the LSVM formulation to incorporate saliency modeling. In our model the image saliency maps are latent variables and are thus integrated with classifier learning.

3.2 Discriminative Spatial Saliency

We define image saliency as a mapping

$$s : \mathcal{G} \rightarrow \mathbb{R}, \tag{3.1}$$

where $\mathcal{G}$ is a spatial partition of the image, $c \in \mathcal{G}$ is a region of the image and $s(c)$ gives the saliency of the region. Our method is general and can work with any spatial partition of the images *e.g.* $\mathcal{G}$ can be the set of all image pixels, as in traditional saliency, or a set of user specified regions. We choose $\mathcal{G}$ to be the set of cells obtained with a spatial pyramid like uniform grid [50]. This is motivated by two reasons. First, we have a variable corresponding to every element of $\mathcal{G}$ for every image and, since contemporary

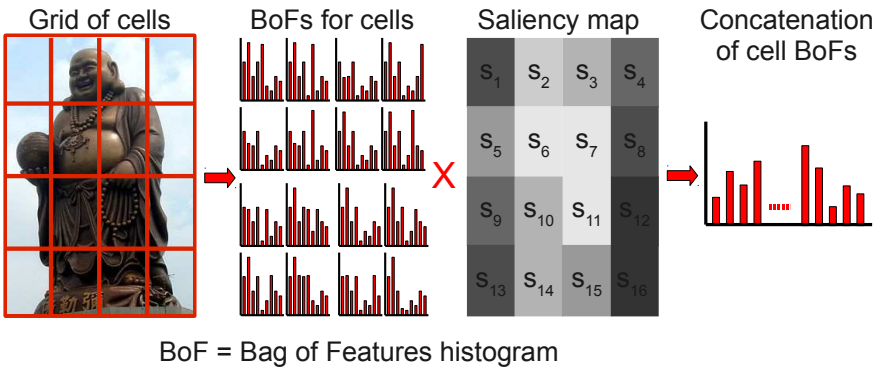

Figure 3.3: The images are represented by concatenation of cell bag-of-features weighted by the image saliency maps.

visual discrimination datasets [19, 50, 108] have limited number of training images, using very fine regions *e.g.* pixels would make the number of variables very large compared to the training data. Second, the spatial pyramid, despite of its simplicity, is competitive with methods using more complex spatial models [24]. Given our choice of $\mathcal{G}$, we can equivalently write a saliency map as an ordered list of real values *i.e.*

$$\mathbf{s} = \{s_c | c \in \mathcal{G}\} \tag{3.2}$$

where we use the row major order of the grid cells (Figure 3.3).

We work in a supervised binary classification scenario with given training images $I^i \in \mathcal{I}$ and corresponding class labels $y^i \in \{-1, 1\}$. Our model consists of three components,

(i) Separating hyperplane $\mathbf{w}$,

(ii) Image saliency maps $\{\mathbf{s}^i | I_i \in \mathcal{I}\}$ and a

(iii) Generic saliency map $\bar{\mathbf{s}}$, for regularizing the image saliency maps.

The saliency map of an image maximizes the classification score while penalizing its deviation from the generic saliency map. Our full model is obtained by solving a max-margin optimization problem with the image saliency maps as latent variables. We present our model in the following sections.

3.2.1 Maximum margin formulation

Given a saliency map $\mathbf{s}^i = \{s^i_c | c \in \mathcal{G}\}$ for the i^{th} image, we represent the image with the saliency map weighted concatenation of bag-of-features (BoF) histograms for the grid cells (Figure 3.3), *i.e.*

$$\mathbf{x}^i = [s^i_1 \mathbf{h}^i_1 \dots s^i_c \mathbf{h}^i_c \dots], \tag{3.3}$$

where $\mathbf{h}_c$ is the BoF histogram for cell $c \in \mathcal{G}$ with appropriate normalization. As noted in [97], normalization plays an important role, and we discuss it in more detail later.

We cast the problem in a maximum margin latent SVM framework with the image saliency maps $\{\mathbf{s}^i | I^i \in \mathcal{I}\}$ as latent variables. The optimization with hinge loss becomes

$$\min_{\mathbf{w}} \ \frac{1}{2}||\mathbf{w}||^2 + C \sum_i \max(0, 1 - y^i f(\mathbf{x}^i, \mathbf{w})), \tag{3.4}$$

where f is the scoring function (Sec. 3.2.2, Eq. 3.6).

Latent SVMs have been very popular recently in the computer vision community *e.g.* Felzenszwalb *et al.* [28], Bilen *et al.* [6]. They lead to a semi-convex optimization i.e. the objective function is convex if the latent variables for the positive examples are fixed (Appendix A gives a brief overview of LSVM).

3.2.2 Image score

We score a given image as, omitting superscript i for brevity,

$$f(\mathbf{x}, \mathbf{w}) = \max_{\mathbf{s}} \mathbf{w}^T \mathbf{x} \tag{3.5}$$

i.e. we allow the saliency map of the image to change to maximize its score *w.r.t.* the separating hyperplane. However, this leads to the trivial solution of selecting the highest scoring cell. To avoid this, we introduce a new variable, a generic saliency map, $\bar{\mathbf{s}}$. We penalize the score proportional to the deviation of the image saliency map from $\bar{\mathbf{s}}$. This regularizes the image saliency maps and gives smoother maps. The final score is thus obtained as

$$f(\mathbf{x}, \mathbf{w}) = \max_{\mathbf{s}} \mathbf{w}^T \mathbf{x} - \lambda(\mathbf{s} - \bar{\mathbf{s}})^T(\mathbf{s} - \bar{\mathbf{s}}), \tag{3.6}$$

where λ is the parameter controlling the trade off between maximizing the score by varying the saliency map and deviation of the image saliency map from $\bar{\mathbf{s}}$. We rewrite the first term of the score as

$$\mathbf{w}^T \mathbf{x} = \sum_{c=1}^{|\mathcal{G}|} s_c \sum_{k=1}^{K} w_{(c-1)\cdot K + k} \ h_{ck} = \mathbf{s}^T \mathbf{D}_w \mathbf{H}^T, \tag{3.7}$$

where K is the size of BoF codebook,

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_{|\mathcal{G}|} \end{bmatrix} \tag{3.8}$$

i.e. **H** is the concatenation of cell BoF histograms, with appropriate normalization, and

$$\mathbf{D}_w = \begin{bmatrix} w_1 \dots w_K & & 0 \\ & \ddots & \\ 0 & & w_{(|\mathcal{G}|-1)\cdot K+1} \cdots w_{|\mathcal{G}|\cdot K} \end{bmatrix}. \tag{3.9}$$

Normalization of the BoFs

As noted by Vedaldi et al. [97], in the context of linear classifiers, unnormalized histograms favor (assign relatively larger scores to) larger regions, L1 normalization favors smaller regions while L2 normalization is neutral and thus ideal. In our experiments, the images are of different size and the grids, specified in terms of fractional multiples of image width and height, results in different sized regions which makes normalization important. Harzallah et al. [37] had also previously noted that normalizing each cell separately instead of globally normalizing the whole descriptor gives slightly better results. Our preliminary experiments resulted in similar conclusions and in our final implementation we work with per-cell L2 normalized vectors. As a result of independent normalization of each cell, **H** is fixed for every image and the optimization problem in Eq. 3.6 takes a closed form solution involving matrix operations and is very fast to compute.

3.2.3 Regularized formulation

By introducing $\bar{\mathbf{s}}$ into the formulation we have introduced another source of scaling. Everything else fixed, by scaling the magnitude of $\bar{\mathbf{s}}$ we can change the image scores (as the saliency maps are multipliers in the score function). Thus, we can decrease the objective value without making any generalizable progress. To control such scaling we augment the objective function with a regularization term for $\bar{\mathbf{s}}$, which penalizes deviation from a uniform map (which assigns unit weight to each cell) similar to the (individual levels of) standard spatial pyramid, as

$$L(\mathbf{w}, \bar{\mathbf{s}}) = \frac{1}{2}||\mathbf{w}||^2 + \frac{\gamma}{2}||\bar{\mathbf{s}} - \mathbf{1}||^2 + C\sum_i \max(0, 1 - y^i f(\mathbf{x}^i)). \tag{3.10}$$

We now have one more parameter, $\gamma > 0$, to control the regularization of $\bar{\mathbf{s}}$. As the scales of $\bar{\mathbf{s}}$ and w are different we can not expect similar regularization *w.r.t.* loss, *i.e.* parameter C to work for both. Thus the model has three parameters for controlling different regularizations γ, C, λ.

The parameter C (cf. the standard SVM parameter) and γ control the relative trade-offs between constraint violation, margin maximization and regularization of $\bar{\mathbf{s}}$. The parameter λ controls the regularization of the saliency map for each image. To gain some

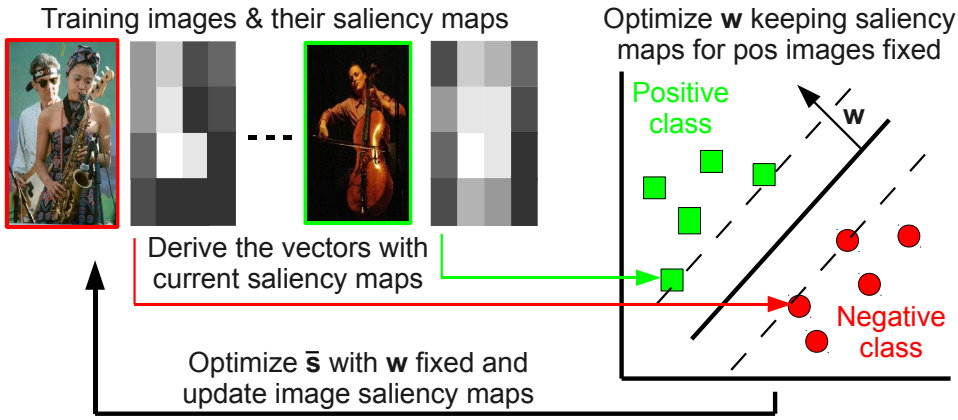

Figure 3.4: We propose to use a block coordinate descent algorithm for learning our model (Sec. 3.2.4). As in a latent SVM, we optimize in one step the hyperplane vector $\mathbf{w}$ *keeping the saliency maps of the positive images fixed and in the other step we optimize the saliency keeping* $\mathbf{w}$ *fixed.*

intuition about the parameter λ, consider the two limiting cases. In the first limiting case, when $\lambda \to \infty$, we have a highly smoothed model which forces all saliency maps to be the same as the generic saliency. In the other limiting case, when λ is zero, we have no smoothing and the saliency maps put all the weight on the best scoring cell per image.

3.2.4 Solving the optimization problem

We solve the problem with a block coordinate descent algorithm. We treat $\mathbf{w}$ and $\bar{\mathbf{s}}$ as two blocks of variables and alternately optimize on one while keeping the other fixed. Figure 3.4 illustrates the learning process. In each of the inner iterations we optimize using stochastic gradient descent as detailed in Algorithms 3.1 and 3.2, where we use (the stochastic approximations of) the sub-gradient *w.r.t.* $\mathbf{w}$,

$$\nabla_{\mathbf{w}} L = \mathbf{w} + C \sum_i g_{\mathbf{w}}(\mathbf{x}^i) \tag{3.11}$$

$$g_{\mathbf{w}}(\mathbf{x}^i) = \begin{cases} 0 & \text{if} \quad y^i f(\mathbf{x}^i) \geq 1 \\ -y^i \mathbf{x}^i & \text{otherwise,} \end{cases} \tag{3.12}$$

and sub-gradient *w.r.t.* $\bar{\mathbf{s}}$,

$$\nabla_{\bar{\mathbf{s}}} L = \gamma(\bar{\mathbf{s}} - 1) + C \sum_i g_{\bar{\mathbf{s}}}(\mathbf{x}^i) \tag{3.13}$$

$$g_{\bar{\mathbf{s}}}(\mathbf{x}^i) = \begin{cases} 0 & \text{if} \quad y^i f(\mathbf{x}^i) \geq 1 \\ 2y^i \lambda(\bar{\mathbf{s}} - \mathbf{s}^i) & \text{otherwise.} \end{cases} \tag{3.14}$$

Algorithm 3.1 Stochastic gradient descent for w ($\bar{\mathbf{s}}$ fixed)

while $t = 1 \dots T$ **do**
  Specify learning rate l_t^w for iteration t
  Choose a random training image I^i
  Calculate the saliency map s^i iff $y^i = -1$
  if $y^i f(\mathbf{x}_i, \mathbf{w}) \geq 1$ **then**
    $\mathbf{w} \leftarrow \mathbf{w} - l_t^{\mathbf{w}} \mathbf{w}$
  else
    $\mathbf{w} \leftarrow \mathbf{w} - l_t^{\mathbf{w}} (\mathbf{w} - CNy^i \mathbf{x}^i)$
  end if
end while

Algorithm 3.2 Stochastic gradient descent for $\bar{\mathbf{s}}$ (w fixed)

while $t = 1 \dots T$ **do**
  Specify learning rate $l_t^{\bar{\mathbf{s}}}$ for iteration t
  Choose a random training image I^i
  Calculate the saliency map s^i
  if $y^i f(\mathbf{x}_i, \mathbf{w}) \geq 1$ **then**
    $\bar{\mathbf{s}} \leftarrow \bar{\mathbf{s}} - l_t^{\bar{\mathbf{s}}} \gamma (\bar{\mathbf{s}} - 1)$
  else
    $\bar{\mathbf{s}} \leftarrow \bar{\mathbf{s}} - l_t^{\bar{\mathbf{s}}} (\gamma (\bar{\mathbf{s}} - 1) + 2CNy^i \lambda (\bar{\mathbf{s}} - s^i))$
  end if
end while

While keeping $\bar{\mathbf{s}}$ fixed we get a semi convex LSVM-like optimization [28] for $\mathbf{w}$. Unfortunately, that is not the case for the optimization of $\bar{\mathbf{s}}$ as, with $\mathbf{w}$ fixed, the hinge loss for each example is concave *w.r.t.* $\bar{\mathbf{s}}$ (the coefficient of $\bar{\mathbf{s}}^T \bar{\mathbf{s}}$ is $-\lambda < 0$). Thus, the total hinge loss (being the maximum over one convex *i.e.* zero function, and multiple concave functions *i.e.* per example hinge losses) is, in general, non convex and the algorithm will converge to a local minimum for $\bar{\mathbf{s}}$. To make sure that it does not end up in a very bad local minimum, we initialize $\mathbf{w}$ with a perturbed version of that learned using the baseline SVM (same optimization with all components of $\bar{\mathbf{s}}$ and $\{\mathbf{s}^i | I^i \in \mathcal{I}\}$ fixed to 1). Since we are directly minimizing the primal we can expect approximations to generalize reasonably [13]. In practice, we find that the models computed by our implementation perform well.

Parameters

We find initial learning rates $l_0^{\mathbf{w}}$ and $l_0^{\bar{\mathbf{s}}}$ by performing preliminary experiments on a subset of the full data and then we decrease the learning rates every iteration by dividing by the iteration number i.e. $l_t = l_0 / t$ (as is common with stochastic gradient methods). We fix $C = 1$ for all experiments (this gives similar results on average as with C obtained by cross validation) and select λ and γ by cross validation on the training data.

Nonlinearizing using a feature map

Recent progress in explicitly computing the feature maps [98] induced by different non linear kernels allows us to address non linearity. The approach is to apply the non linear map to compute the feature vectors explicitly, and work with linear algorithms in the feature space.

We transform the histograms by taking their element-wise square roots *i.e.*

$$\phi([h_1,\ h_2,\ \ldots,\ h_d]) = [\sqrt{h_1},\ \sqrt{h_2},\ \ldots,\ \sqrt{h_d}]. \tag{3.15}$$

It is known [98] that the product of the resulting vectors is equal to the Bhattacharyya kernel between the original histograms. Hence, using the feature map is equivalent to working with the non linear Bhattacharyya kernel, which has been shown to give better results than the linear kernel. We L1 normalize the original histograms so that the feature mapped vectors are L2 normalized.

3.3 Experimental results

We evaluate our method on three challenging datasets for

(i) Human action classification in still images [20],

(ii) Fine grained classification, of humans playing musical instruments vs. holding them [111], and

(iii) Scene classification [50].

We first give the details of our implementation and baselines and then proceed to present and discuss the results on the three datasets.

Bag-of-features implementation details

Like previous works [19, 111] we densely sample grayscale SIFT features at multiple scales. We use a fixed step size of 4 pixels and use square patch sizes at 7 scales ranging from 8 to 40 pixels. We learn a vocabulary of size 1000 using k-means and assign the SIFT features to the nearest codebook vector (hard assignment). We use the VLFeat library [96] for SIFT and k-means computation.

Spatial pyramid (SP and overlapping SP)

We use a four level spatial pyramid but instead of the usual non overlapping cells with uniform grids we expand the cells by 50% and let them overlap *i.e.* 2×2 cells are $3/4$ of the height (width) instead of $1/2$. We found that doing so provides better statistics (less sparse histograms) for finer cells and improves performance. This is inspired by the idea of 'non sparsification' of vectors [72]. We discuss this more in Sec. 3.3.4. Our initial experiments gave similar results with classifiers trained on the full pyramid descriptor and the weighted sum of descriptors from each level. We train classifiers for each level separately and combine levels, for the baselines as well as our method, by the weighted sum of classifier scores. The weights sum to one over all levels and are higher for finer levels at resolution, similar to previous work [50].

Baselines

We use SP and overlapping SP, as baselines, with linear SVM trained without our saliency model *i.e.* we fix all the saliency maps to be uniform in the optimization reducing it to standard linear SVM with spatial BoF. The baseline results are obtained with the liblinear [26] library.

Performance measure

The performance is evaluated based on average precision (AP) for each class and the mean average precision (mAP) over all classes.

3.3.1 Willow actions

Willow actions[1] [19] is a challenging database for action classification on unconstrained consumer images downloaded from the internet. It has 7 classes of common human actions *e.g.* 'ridingbike', 'running'. It has at least 108 images per class of which 70 images are used for training and validation and rest are used for testing. The task is to predict the action being performed given the human bounding box. Like previous work [20], we expand the given person bounding boxes by 50% to include some contextual information.

Figure 3.8a shows example images and their saliency maps obtained with our model and Table 3.1 gives quantitative results on the Willow actions dataset. Our implementation of the baseline spatial pyramid [50] achieves an mAP of 62.6% while that of a spatial pyramid with overlapping cells improves by 2%. Our model obtains 65.9% which is the state-of-the art result on this dataset. To compare with previous works, Delaitre et al. [20]

[1] http://www.di.ens.fr/willow/research/stillactions/

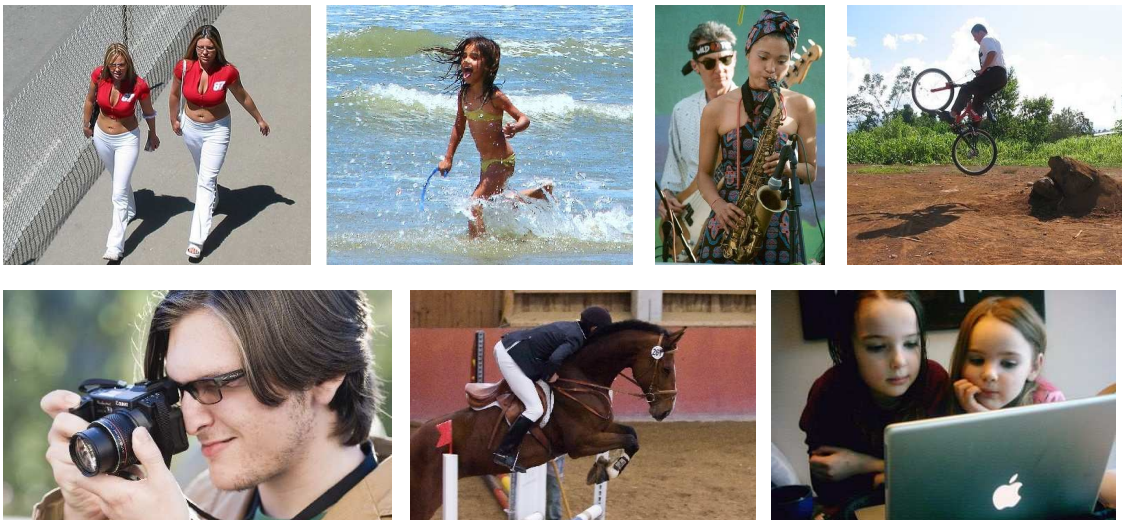

Figure 3.5: Example images from the Willow Actions dataset [19].

Table 3.1: Results (AP) on actions dataset (Sec. 3.3.1)

	Per-obj	Baselines		
	inter. [20]	SP [50]	ov. SP	Ours
inter. w/ comp.	56.6	49.4	57.8	**59.7**
photographing	37.5	41.3	39.3	**42.6**
playingmusic	72.0	74.3	73.8	**74.6**
ridingbike	**90.4**	87.8	88.4	87.8
ridinghorse	75.0	73.6	80.8	**84.2**
running	**59.7**	53.3	55.8	56.1
walking	57.6	**58.3**	56.3	56.5
mAP	64.1	62.6	64.6	**65.9**

obtain an mAP of 64.1% with a method modeling person-object interactions. Note that they model complex interactions between objects and body parts while using external data to train the several object and body part detectors.

Our method gives best results for four out of seven categories. The most significant improvement is obtained on the 'ridinghorse' class which has a strong spatial bias for horse and grass in the bottom part of the image. The saliency map modeling effectively exploits this (Figure 3.8a). The drop on 'ridingbike' class can be explained by the limitation of the method to improve performance if the classifier is able to separate the training data almost perfectly and/or if there is not enough training data (Sec. 3.3.4).

3.3.2 People playing musical instruments

People playing musical instruments (PPMI)[2] [111] is a dataset emphasizing subtle difference in interactions between humans and objects (fine grained classification). It contains

[2]http://ai.stanford.edu/∼bangpeng/ppmi.html

Figure 3.6: Example images from the PPMI dataset [108] with people playing (top row) and people holding (bottom row) the musical instruments.

Table 3.2: Results (mAP) on Task 1 of PPMI dataset (Sec. 3.3.2)

Grouplet [108]	Rn. forest [111]	Baselines SP [50]	ov. SP	Ours
36.7	47.0	45.3	46.6	**49.4**

Table 3.3: Results (mAP) on Task 2 of PPMI dataset (Sec. 3.3.2)

Grouplet [108]	Rn. forest [111]	Baselines SP [50]	ov. SP	Ours
85.1	**92.1**	89.2	90.3	91.2

classes with humans interacting with *i.e.* either playing or just holding, 12 different musical instruments. There are two tasks for this dataset

(i) ***Task 1***: 24 class classification, with the classes being the human playing and holding the 12 instruments and

(ii) ***Task 2***: 12 binary classifications, with each binary problem being that of human playing vs. holding the instruments.

Figure 3.8b shows some example images and their saliency maps and Table 3.2 and 3.3 shows our results on the PPMI datasets for *Task 1*: 24 class multi-class classification and *Task 2*: 12 binary classification problems respectively. For *Task 1*, the spatial pyramid baseline achieves 45.3% and the overlapping spatial pyramid achieves 46.6% improving by 1.3%. Our method achieves a mAP of 49.4% which is state of the art for the dataset. In comparison to previous methods, we improve by 12.7% compared to Yao et al.'s Grouplet [108] and by 2.4% compared to their Random Forest classifier [111]. For *Task 2*, the

Table 3.4: Results (mAP) on Scene 15 dataset (Sec. 3.3.3)

Pyramid level comb.	Baselines SP [50]	ov. SP	Ours
1	74.9 ± 0.5	74.9 ± 0.5	-
1+2	77.9 ± 0.4	78.8 ± 0.5	$\mathbf{85.1 \pm 1.2}$
1+2+3	81.8 ± 0.6	82.6 ± 0.4	$\mathbf{85.5 \pm 0.6}$
1+2+3+4	81.9 ± 0.5	81.9 ± 0.3	$\mathbf{84.6 \pm 0.7}$

baselines are at 89.2% and 90.3% while our method achieves 91.2% compared to 85.1% of Grouplet [108] and 92.1% of Random Forest classifier [111]. The Grouplet method uses patches at only one scale which can perhaps explain its lower performance. Note that the Random Forest classifier has a much higher complexity than our approach, as it uses 100 decision trees. At each node of the tree they evaluate a linear SVM decision thus effectively performing 100s of vector dot products, whereas our approach only has one such computation. We perform slightly worse that the state of the art in *Task 2* due to performance saturation, see Sec. 3.3.4 for a discussion.

3.3.3 Scene 15

Scene 15[3] [50] is a dataset containing 15 scene categories, *e.g.* 'beach', 'office' (see Section 2.4 for more description and example images). The task is multi-class classification with the dataset split into 100 random images per class for training and the rest for testing. Like previous works, we repeat the experiment 10 times and report the mean and standard deviation of the performance achieved in each run.

Figure 3.8c shows some example images and their saliency maps and Table 3.4 show our results on the scene 15 dataset for 15 binary one-vs-rest classification problems. Our traditional and overlapping spatial pyramid baselines achieve a performance of 81.8% and 82.6% resp. for 3 levels. Our method achieves 85.5% improving the better baseline by 2.9%. It is interesting to note that our method at a lower pyramid level of 2 already beats the best baseline, at a higher pyramid level of 3, by 2.5%, which points to the strong and coarse spatial bias in the dataset. The state-of-the-art method on this dataset [101] achieves 88.1% (mean class accuracy). However, they combine 14 different low level features. Our best result is comparable to Krapac et al. [48], who used a similar setup as ours and achieved mAP of 85.6%. Note that they quantized features using discriminatively trained decision trees outperforming k-means based quantization. In the current paper, we have used k-means and arguably our results would improve further using similar stronger quantization instead.

[3] http://www-cvr.ai.uiuc.edu/ponce_grp/data/

3.3.4 Overlapping cells and training saturation

We use overlapping cells for the spatial pyramid decomposition. As noted by Perronnin et al. [72], when sparseness of the vectors increases, the performance of linear SVM decreases. This is because the more robust distance with sparse vectors is L1 while linear SVM corresponds to the L2 distance. To decrease the effect of sparsity we take overlapping cells in the spatial pyramid partition by increasing the sizes of the cells by 50%. Figure 3.7 (top) shows the performances for different codebook sizes on the Willow actions dataset. We notice that for larger codebook sizes of 500 and 1000 the overlapping SP performs better than the non overlapping one but the difference is not significant for a codebook size of 100. As the codebook size increases, but the number of features stays the same, the sparsity of the histogram increases. Thus, pooling more features by increasing the size of the cells performs better, as the sparsity of the histograms is decreased.

We can also observe that our approach does not gain much when the training data is well separated *i.e.* the baseline SVM is saturated. This can occur when there is not enough training data or the task is relatively easy. In saturated cases the number of vectors within the margin (which effectively contribute towards refining the hyperplane), even for the baseline, are only a few (< 100) and the saliency model is not able to derive more information from so few examples. Figure 3.7 (bottom) shows the performance for the different pyramid combinations for the Willow actions dataset. We observe that as the pyramid level increases, the gap between the baselines and the proposed method decreases due to increase in training saturation. The trend is similar for increasing codebook size, Figure 3.7 (top). This also explains why we get little or no improvement for the 'ridingbike' class (Table 3.1) and the *Task 2* of PPMI dataset (Table 3.3).

3.3.5 Qualitative results

Figure 3.8 shows example images from two classes for each of the three datasets together with their saliency maps. We can observe that the saliency maps focus on those parts of the images which we expect to be discriminative. For example, in the action class 'ridinghorse' the saliency maps give high weights to the lower regions which are expected to be salient as they contain the horse and grassy texture which are highly correlated with the class. The person (in the typical riding pose) is not weighted highly, because it might be confused with 'ridingbike', stressing the discriminative nature of the maps.

Furthermore, per image adaptation can be seen in all the examples. In the 'playingmusic' class the maps follow the hands and the musical instruments and differ for every image. A similar observation holds for 'tallbuilding' class where the middle part of the buildings seems to be more discriminative probably because of predominant sky in the upper part of many images.

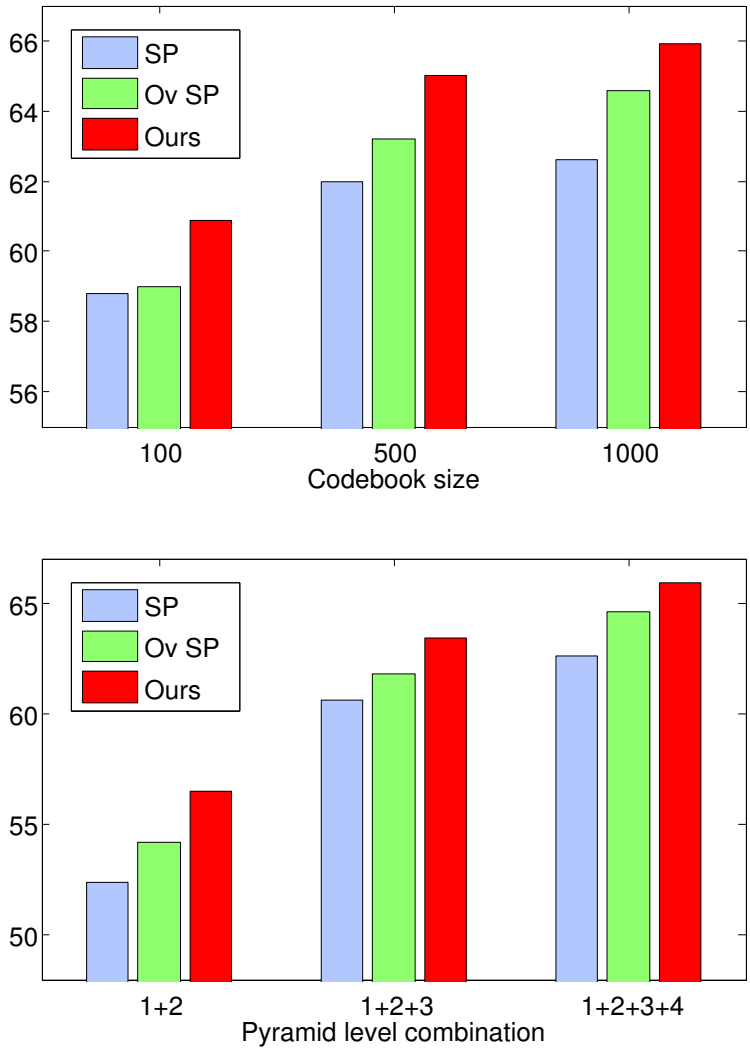

Figure 3.7: (Top) Evaluation (mAP) of the impact of the codebook size for a full pyramid representation. (Bottom) Evaluation (mAP) of the impact of the pyramid levels for a codebook size of 1000. The dataset is the Willow Actions [19].

The correlation between the locality of the task and the peaks in the maps is also clearly visible. A strong contrast is apparent between the 'playingmusic' class of the Willow actions dataset and the similar 'violin' and 'erhu' classes of PPMI dataset. In the actions dataset the discrimination is against more general actions (*e.g.* 'running', 'photographing') and hence the maps capture the instrument, the pose of the hands *etc.* and have relatively spread out maxima. In contrast, for 'violin' and 'erhu' classes the maps have sharp peaks as the task is to differentiate between holding vs. playing instruments. The maps here quite accurately focus on the region of discriminative interaction between the person and the instrument.

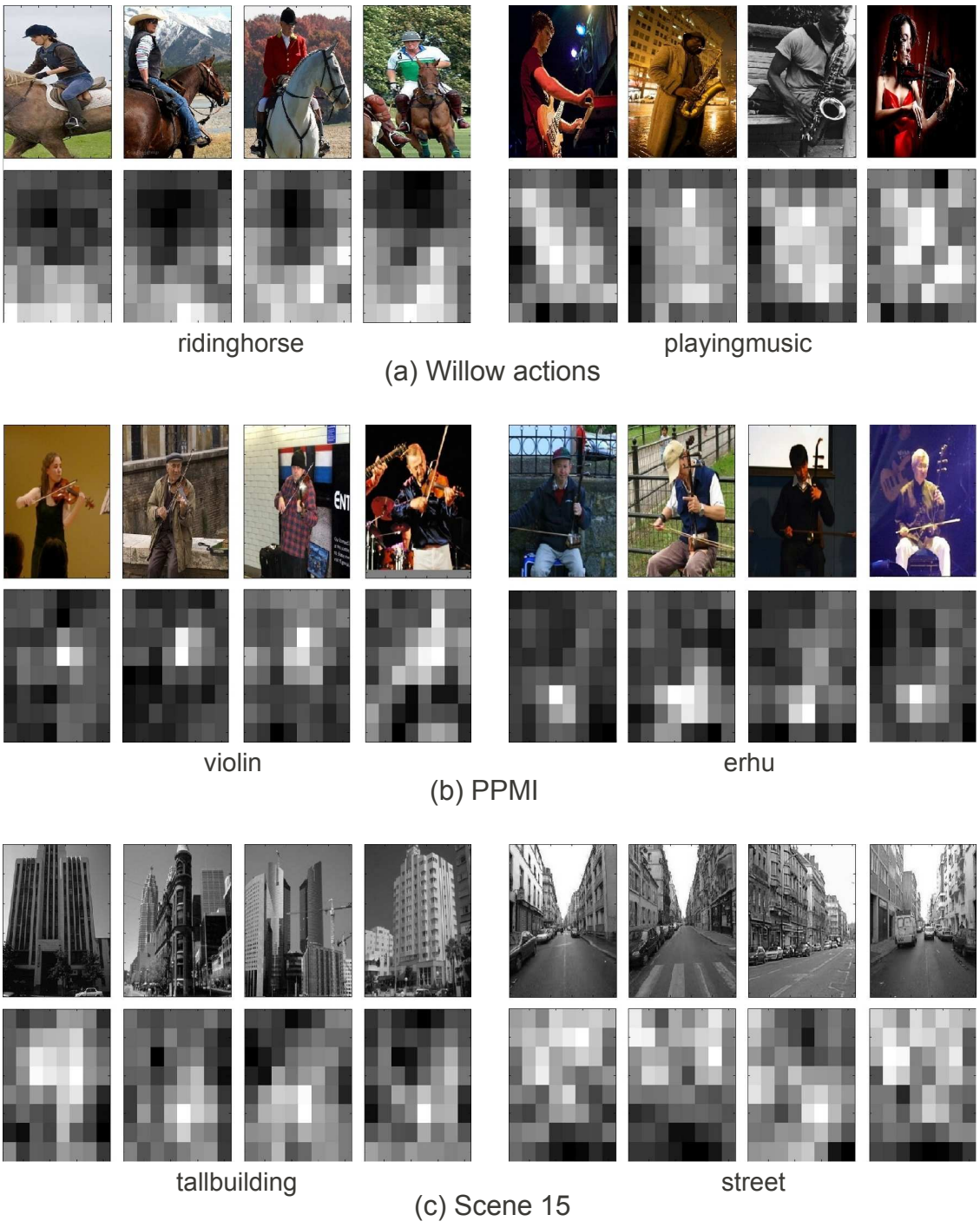

Figure 3.8: Example images and their saliency maps (8×8 resolution) for images from two classes for each of the three databases (higher values are brighter). Notice how the maps adapt to the content of the image and highlight the spatially salient regions per image.

3.4 Conclusions

In the present chapter, we presented a method for learning spatial saliency for images. The learnt spatial saliency is discriminative and task specific. We showed experimentally that the saliency modeling improves the image representation and, thus, the classification performance. The saliency map help the classifier to focus on the important regions in the image for the given classification task. Such local focus is important for visual tasks where there is a spatial bias.

The method has wide applicability as was demonstrated with experiments on three challenging datasets. It improves over the baseline without spatial saliency and achieves better or comparable results *w.r.t.* the state-of-the-art.

Chapter 4

Local Higher-Order Statistics

Contents

4.1 Introduction

Human faces convey a lot of information about a person. Many semantic attributes about humans can be inferred from the face alone *e.g.* those based on sex, facial appearance, expression. In the present chapter we propose a new descriptor for facial analysis and texture categorization [85] inspired by recent texture descriptors *e.g.* LBP/LTP.

Visual categorization under multiple sources of variations e.g. illumination, scale, pose, is a challenging open problem in computer vision. While the community has spent a lot of effort on object-category classification and object segmentation tasks [24] — leading to very powerful intermediate representation of images such as the BoW model [53, 86] — texture recognition has received relatively less attention despite its importance for several computer vision tasks. Texture recognition is beneficial for many applications such as mobile robot navigation or biomedical image processing. Texture analysis is also related to facial analysis e.g. facial expression categorization and face verification as the models

developed for texture recognition can, in general, be used successfully for face analysis. Such tasks, similarly, find important applications in human computer interaction and in security and surveillance applications. In this chapter we aim to catch up on this topic by proposing a new model providing a powerful texture and face representation.

Earlier works on texture analysis were focused on the development and application of filter banks e.g. [53, 17, 115]. They computed filter response coefficients for a number of filters or wavelets and learned their distributions. However, later works disproved the necessity of such ensembles of filters *e.g.* Ojala *et al.* [68] and Varma and Zisserman [94] showed that it is possible to discriminate between textures using pixel neighbourhoods as small as 3×3 pixels. They demonstrated that despite the global structure of the textures, very good discrimination could be performed by exploiting the distributions of such pixel neighbourhoods. More recently, exploiting such *micro-structures* in textures by representing images with distributions of local descriptors has gained much attention and has led to state-of-the art performances [73, 1, 87, 14]. However, as we discuss later, these methods suffer from several important limitations, such as the use of fixed quantization of the feature space as well as the use of heuristics to prune volumes in the feature space. In addition, they represent feature distributions with histograms and hence are restricted to the use of low order statistics.

In contrast to these previous works, we propose a model that represents images with higher order statistics of local pixel neighborhoods. We obtain a data driven partition of the feature space using parametric mixture models, to represent the distribution of the vectors, and learn the parameters from the training data. Hence, the coding of vectors is intrinsically adapted to any classification task and the computations involved remain very simple despite the strengths. The approach is validated by extensive experiments, on four challenging datasets i.e.

(i) Brodatz 32 texture dataset [93, 9],

(ii) KTH TIPS 2a materials dataset [12],

(iii) Japanese female facial expressions dataset [59], and

(iv) Labeled faces in the wild [39],

which show that using higher-order statistics gives a more expressive description and leads to state-of-the-art performance.

4.1.1 Related works on texture analysis for image classification

Most of the earlier works on texture analysis focused on the development of filter banks and on characterizing the statistical distributions of their responses e.g. [53, 17, 115], until Ojala *et al.* [68] and, more recently, Varma and Zisserman [94] showed that statistics

of small pixel neighbourhoods are capable of achieving high discrimination. Since then many methods working with local pixel neighbourhoods have been used successfully in texture and face analysis e.g. [87, 14, 56].

Local pixel pattern operators, such as Local Binary Patterns (LBP) by Ojala et al. [68], have been very successful for local pixel neighbourhood description. LBP based image representation aims to capture the joint distribution of local pixel intensities. LBP approximates the distribution by first taking the differences between the center pixel and its neighbours and then considering just the signs of the differences. The first approximation lends invariance to gray-scale shifts and the second to intensity scaling. Local Ternary Patterns (LTP) were introduced by Tan and Triggs [87] to add restistance to noise. LTP requires a parameter t, which defines a tolerance for similarity between different gray intensities, allowing for robustness to noise. Doing so lends an important strength: LTPs are capable of encoding pixel similarity information modulo noise. However, LTP (and LBP) coding is still limited due to its hard and fixed quantization. In addition, both LBP and LTP representations usually use the so-called *uniform* patterns: patterns with at most one 0-1 and at most one 1-0 transition, when seen as circular bit strings. The use of these patterns is motivated by the empirical observation that uniform patterns account for nearly 90 percent of all observed patterns in textures. While it works quite well in practice, it is a heuristic for discarding low occupancy volumes in feature space.

Most of the other recent methods, driven by the success of earlier texton based texture classification method [53] and recent advances in the field of object category classification, adopt bag-of-words models to represent textures as distributions of local textons [94, 56, 49, 112, 95, 15, 38, 102, 103]. They learn a dictionary of textons obtained by clustering vectors (*e.g.* based on either pixel intensities, sampled on local neighbourhoods, or their differences), and then represent the image as histograms over the learnt codebook vector assignments. The local vectors are derived in multiple ways, incorporating different invariances like rotation, view point *etc.* . *E.g.* [49, 112] generate an image specific texton representation from rotation and scale invariant descriptors and compare them using Earth Movers distance. While [94, 68, 56, 95] use a dictionary learned over the complete dataset to represent each image as histogram over this dictionary.

The motivations for this paper follow the conclusions that can be drawn from these related works.

(i) As shown by [68, 94], and by all the recent papers that build on these, modeling distributions of small pixel neighbourhoods (as small as 3×3 pixels) can be very effective.

(ii) Unfortunately, all the previously mentioned approaches involve coarse approximations that prevent them from getting all the benefits of an accurate representation of such small neighbourhoods, and

(iii) All these methods use low-order statistics while using high-order moments can give a more expressive representation.

Addressing these limitations by accurately describing small neighbourhoods with their higher-order statistics, without coarse approximations, is the contribution of the present paper.

4.2 The Local Higher-order Statistics (LHS) Model

As explained before, the proposed Local Higher-order Statistics (LHS) model intends to represent images by exploiting, as well as possible, the distribution of local pixel neighbourhoods. Thus, we start with small pixel neighbourhoods of 3×3 pixels and model the statistics of their local differential vectors.

Local differential vectors

We work with all possible 3×3 neighbourhoods in the image, *i.e.* $\{v^n = (v_c, v_1, \ldots, v_8)\}$ where v_c is the intensity of the center pixel and the rest are those of its 8-neighbours. We are interested in exploiting the distribution $p(v^n|I)$ of the these vectors, for a given image, to represent the image. We obtain invariance to monotonic changes in gray levels by subtracting the value of the center pixel from the rest and using the difference vector *i.e.*

$$p(v^n|I) \approx p(v|I) \text{ where, } v = (v_1 - v_c, \ldots, v_8 - v_c). \tag{4.1}$$

We call the vectors $\{v\}$ thus obtained as the differential vectors.

Higher order statistics

The key contribution of LHS is to use the statistics of the differential vectors $\{v|v \in I\}$ to characterize the images. Instead of using a hard and/or predefined quantization, we use parametric Gaussian mixture model (GMM) to derive a probabilistic representation of the differential space. Defining such soft quantization, which can equivalently be seen as a generative model on the differential vectors, allows us to use a characterization method which exploits higher order statistics. We use the *Fisher score* method (Jaakkola and Haussler [44]), where given a parametric generative model, a vector can be characterized by the gradient with respect to the parameters of the model. The Fisher score, for an observed vector v *w.r.t.* a distribution $p(v|\lambda)$, where λ is parameter vector, is given as,

$$g(\lambda, v) = \nabla_\lambda \log p(v|\lambda). \tag{4.2}$$

The Fisher score, thus, is a vector of same dimensions as the parameter vector λ. For a mixture of gaussian distribution i.e.

$$p(v|\lambda) = \sum_{c=1}^{N_k} \alpha_k \mathcal{N}(v|\mu_k, \Sigma_k) \tag{4.3}$$

$$\mathcal{N}(v|\mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_k|}} \exp\left\{-\frac{1}{2}(v-\mu_k)\Sigma_k^{-1}(v-\mu_k)\right\}, \tag{4.4}$$

the Fisher scores can be computed using the following partial derivatives (we assume diagonal Σ for decreasing the number of parameter to be learnt)

$$\frac{\partial \log p(v|\lambda)}{\partial \mu_k} = \gamma_k \Sigma_k^{-1}(v-\mu_k) \tag{4.5a}$$

$$\frac{\partial \log p(v|\lambda)}{\partial \Sigma_k^{-1}} = \frac{\gamma_k}{2}\left(\Sigma_k - (v-\mu_k)^2\right) \tag{4.5b}$$

$$\text{where,} \quad \gamma_k = \frac{\alpha_k p(v|\mu_k, \Sigma_k)}{\sum_k \alpha_k p(v|\mu_k, \Sigma_k)} \tag{4.5c}$$

where the square of a vector is element-wise square. In the derivatives above we can see that the information based on the first and second powers of the differential vectors are also coded; these are higher order statistics for the differential vectors. After obtaining the differential vectors corresponding to every pixel neighbourhood in the image, we compute the image representation as the average vector over all of them. We normalize each dimension of the image vector to zero mean and unit variance. To perform the normalization we use training vectors and compute multiplicative and additive constants to perform whitening per dimension [7]. We also incorporate two normalizations (on image vector x) [72] i.e. power normalization,

$$(x_1, \ldots, x_d) \leftarrow (\text{sign}(x_1)\sqrt{|x_1|}, \ldots, \text{sign}(x_d)\sqrt{|x_d|}), \tag{4.6}$$

and L2 normalization,

$$(x_1, \ldots, x_d) \leftarrow \left(\frac{x_1}{\sqrt{\sum x_i^2}}, \ldots, \frac{x_d}{\sqrt{\sum x_i^2}}\right). \tag{4.7}$$

The whole algorithm, which is remarkably simple, is summarized in Alg. 4.1. Finally, we use the vectors obtained as the representation of the images and employ a discriminative linear support vector machine (SVM) as the classifier in a supervised learning setup.

Relation to LBP/LTP

We can view LHS vectors as generalization of local binary/ternary patterns (LBP/LTP) [68, 87]. In LBP every pixel is coded as a binary vector of 8 bits with each bit indicat-

Algorithm 4.1 Computing Local Higher-Order Statistics (LHS)

Randomly sample 3×3 pixels differential vectors $\{v \in I | I \in \mathcal{I}_{train}\}$
Learn the GMM parameters $\{\alpha_k, \mu_k, \Sigma_k | k = 1 \dots K\}$ with EM algorithm on $\{v\}$
Compute the higher-order Fisher scores for $\{v\}$ using equations (4.5)
Compute means C_μ^d and variances C_Σ^d for each dimension d
for all images $\{I\}$ **do**
    Compute all differential vectors $v \in I$
    Compute the Fisher scores for all features $\{v\}$ using equations (4.5)
    Compute the image representation x as the average score over all features
    Normalize each dimension d as $x^d \leftarrow (x^d - C_\mu^d)/C_\Sigma^d$
    Apply normalizations, equations (4.6) and (4.7)
end for

ing whether the current pixel is of greater intensity when compared to (one of the 8) its neighbours. We can derive the LBP [68] by thresholding each coordinate of our differential vectors at zero. Hence the LBP space can be seen as a discretization of the differential space into two bins per coordinate. Similarly, we can discretize the differential space into more number of bins, with three bins per coordinate i.e. $(-\infty, -t), [-t, t], (t, -\infty)$ we arrive at the local ternary patterns [87] and so on. The use of *uniform patterns* (patterns with exactly one 0-1 and one 1-0 transitions) only, in both LBP/LTP, can be seen as an empirically derived heuristic for ignoring volumes in differential space which have low occupancies. Thus, the binary/ternary patterns are arrived at with a quantization step and rejection heuristic while in our case similar information is learnt from data.

4.3 Experimental Validation

The experimental validation is done on four challenging publicly available datasets of textures and faces. We first discuss implementation details then present the datasets and finally give the experimental results for each dataset.

As our focus is on the rich and expressive representation of local neighbourhoods, we use a standard classification framework based on linear SVM. As linear SVM works directly in the input feature space, any improvement in the performance is directly related to a better encoding of local regions, and thus helps us gauge the quality of our features.

Implementation details

We use only the intensity information of the images and convert color images, if any, to grayscale. We consider two neighbourhood sampling strategies

(i) Rectangular sampling, where the 8 neighboring pixels are used, and

(ii) Circular sampling, where, like in LBP/LTP [68, 87], we interpolate the diagonal samples to lie on a circle, of radius one, using bilinear interpolation.

We randomly sample at most 500,000 features from training images to learn Gaussian mixture model of the vectors, using the EM algorithm initialized with k-means clustering. We keep the number of components as an experimental parameter (Sec. 4.3.3). We also use these features to compute the normalization constants, by first computing their Fisher score vectors and then computing (per coordinate) mean and variance of those vectors (Alg. 4.1). We use the average of all the features from the image as the representation for the image. However, for the facial expression dataset we first compute the average vectors for non overlapping cells of 10×10 pixels and concatenate these for all cells to obtain the final image representation. Such griding helps in capturing spatial information in the image and is standard in face analysis [30, 82]. We crop the 256×256 face images to a ROI of $(66, 96, 186, 226)$, to focus on the face, before feature extraction and do not apply any other pre-processing. Finally, we use linear SVM as the classifier with the cost parameter C set using five fold cross validation on the current training set.

Baselines

We consider baselines of single scale LBP/LTP features generated using the same samplings as our LHS features. We use histogram representation over uniform LBP/LTP features. We L1 normalize the histograms and take their square roots and use them with linear SVM. It has been shown that taking square root of histograms transforms them to a space where the dot product corresponds to the non linear Bhattacharyya kernel in the original space [98]. Thus using linear SVM with square root of histograms is equivalent to SVM with non linear Bhattacharyya kernel. Hence, our baselines are strong baselines.

4.3.1 Texture categorization

Brodatz – 32 Textures dataset[1] [93, 9] is a standard dataset for texture recognition. It contains 32 texture classes *e.g.* bark, beachsand, water, with 16 images per class. Each of the image is used to generate 3 more images by

(i) Rotating,

(ii) Scaling and

(iii) Both rotating and scaling the original image.

Note that Brodatz-32 [93] is a more challenging dataset than original Brodatz and includes both rotation and scale changes. The images are 64×64 pixels histogram normalized grayscale images. Figure 4.1 shows some example images from the dataset. We use

[1] http://www.cse.oulu.fi/CMV/TextureClassification

Table 4.1: Results (avg. accuracy and std. dev.) on the different datasets.

(a) Rectangular sampling (8-pixel neighbourhood)

	Brodatz–32	KTH TIPS 2a	JAFFE E1	JAFFE E2
LBP baseline	87.2 ± 1.5	69.8 ± 6.9	86.9 ± 2.6	56.5 ± 21.0
LTP baseline	95.0 ± 0.8	69.3 ± 5.3	93.6 ± 1.8	57.2 ± 16.3
LHS (ours)	$\mathbf{99.3 \pm 0.3}$	$\mathbf{71.7 \pm 5.7}$	$\mathbf{95.6 \pm 1.7}$	$\mathbf{64.6 \pm 19.2}$

(b) Circular sampling (bilinear interpolation for diag. neighbours)

	Brodatz–32	KTH TIPS 2a	JAFFE E1	JAFFE E2
LBP baseline	87.3 ± 1.5	69.8 ± 6.7	94.3 ± 2.1	61.8 ± 24.1
LTP baseline	94.9 ± 0.8	71.3 ± 6.3	95.1 ± 1.8	60.6 ± 20.8
LHS (ours)	$\mathbf{99.5 \pm 0.2}$	$\mathbf{73.0 \pm 4.7}$	$\mathbf{96.3 \pm 1.5}$	$\mathbf{63.2 \pm 16.5}$

the standard protocol [14], of randomly splitting the dataset into two halves for training and testing, and report average performance over 10 random splits.

KTH TIPS 2a dataset[2] [12] is a dataset for material categorization. It contains 11 materials *e.g.* cork, wool, linen, with images of 4 samples for each material. The samples were photographed at 9 scales, 3 poses and 4 different illumination conditions. All these variations make it an extremely challenging dataset. Figure 4.2 shows some example images from the dataset. We use the standard protocol [14, 12] and report the average performance over the 4 runs, where every time all images of one sample are taken for test while the images of the remaining 3 samples are used for training.

Table 4.1 (col. 1 and 2) shows the results for the different methods on these texture datasets. We achieve a near perfect accuracy of 99.5% on the Brodatz dataset. Our best method outperforms the best LBP and LTP baselines by 12.2% and 4.5% respectively and demonstrates the advantage of using rich, higher-order, data-adaptive encoding of local neighbourhoods compared to fixed quantization based LBP and LTP representations. Brodatz dataset has variations in the scale and rotation of the textures and, hence, the high accuracy achieved on the dataset leads us to conclude that texture recognition can be done almost perfectly under the presence of rotation and scaling variations.

On the more challenging KTH TIPS 2a dataset, the best performance is far from saturated at 73%. The gain in accuracy over LBP and LTP is 3.2% and 1.7% respectively. The dataset has much higher variations in scale, illumination condition, pose *etc.* than the Brodatz dataset and the experiment is of texture categorization of unseen sample *i.e.* the test images are of a sample not seen on training. We again outperform LBP/LTP demonstrating the higher discriminative power and the generalization capability of our descriptor.

[2] http://www.nada.kth.se/cvap/datasets/kth-tips/

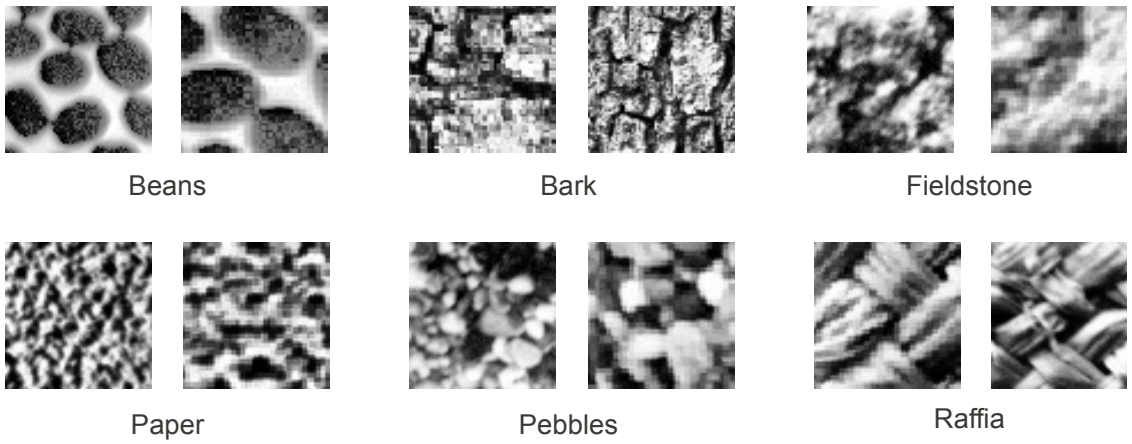

Figure 4.1: Example images from the Brodatz 32 texture dataset [93, 9]

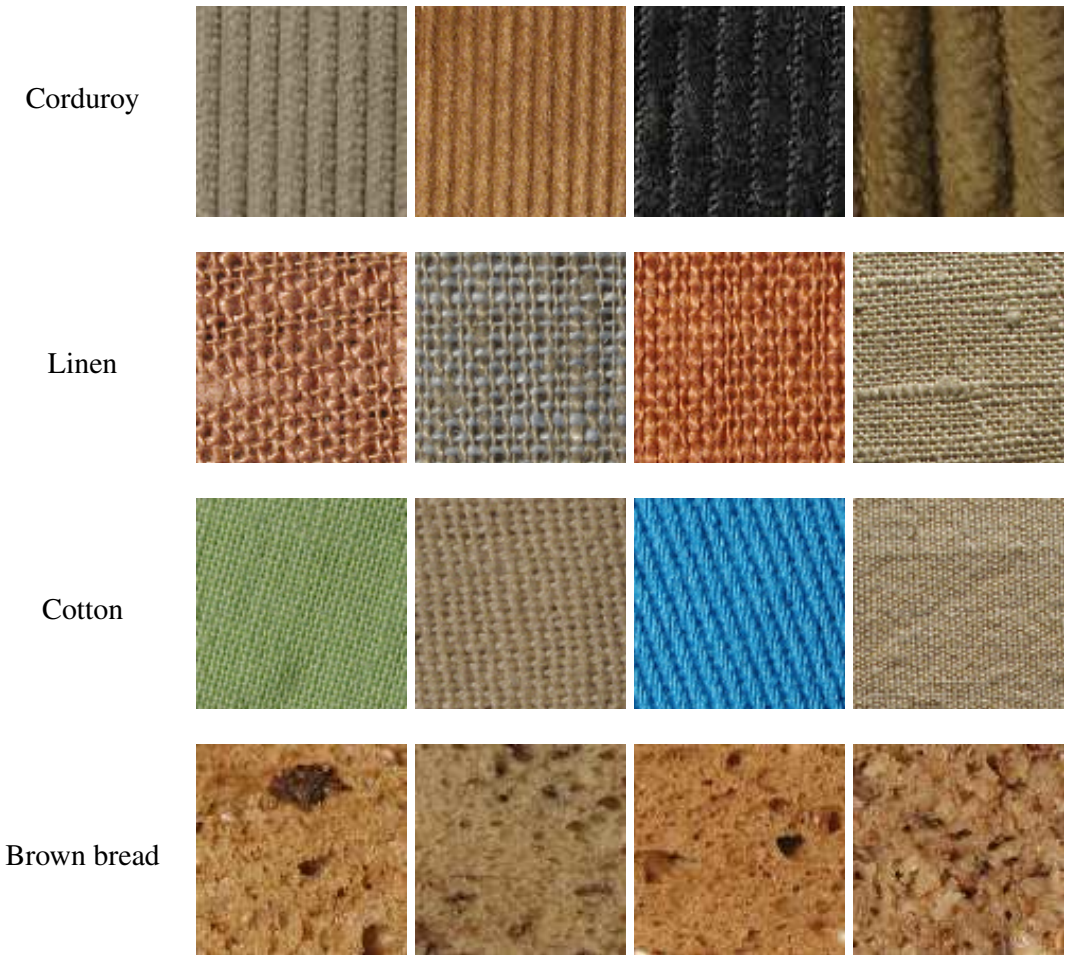

Figure 4.2: Example images from the KTH TIPS 2a texture dataset [12]

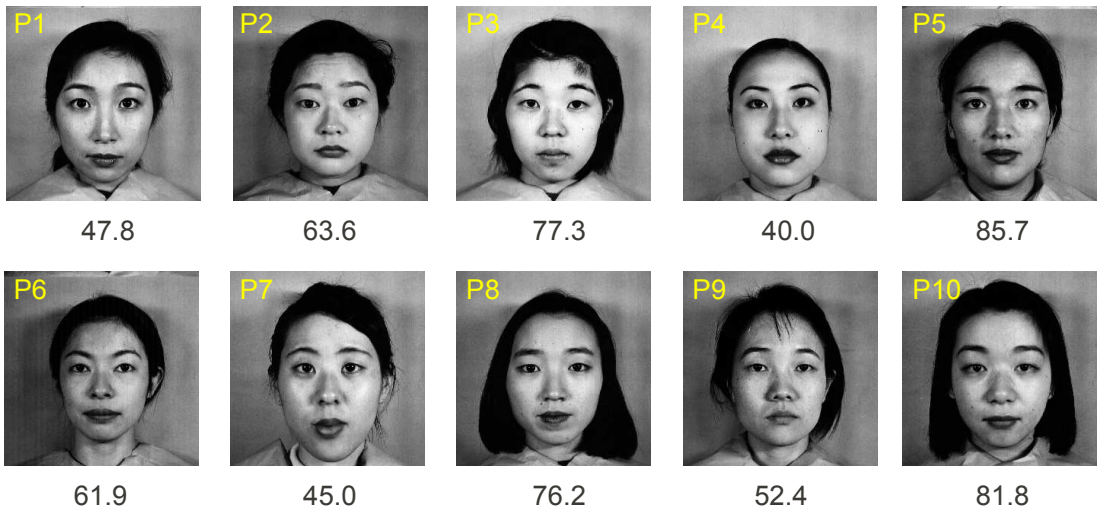

Figure 4.3: The images of the 10 persons in the neutral expression. The number below is the categorization accuracy for all 7 expressions for the person (see Sec. 4.3.2).

4.3.2 Facial analysis

Expression classification

Japanese Female Facial Expressions (JAFFE)[3] [59] is a dataset for facial expression recognition. It contains 10 different females expressing 7 different emotions *e.g.* sad, happy, angry. We perform expression recognition for both known persons, like earlier works [54], and for unknown person. In the first (experiment E1), one image per expression for each person is used for testing while remaining are used for training. Thus, the person being tested is present (different image) in training. While in the second (experiment E2), all images of one person are held out for testing while the rest are used for training. Hence, there are no images of the person being tested in the training images, making the task more challenging. In both, we report the mean and standard deviation of average accuracies of 10 runs.

Table 4.1 (col. 3 and 4) shows the performance of the different methods. On the first experiment (E1) we obtain very high accuracies as the task is of recognition of expressions, from a never seen image, of a person present in the training set. Our method again outperforms LBP and LTP based representation by 2% and 1.2% respectively. On the more challenging second experiment (E2) we see that the accuracies are much less than E1. Our best accuracy is again better than the best LBP and LTP accuracies by 2.8% and 4% respectively. Fig. 4.3 shows one image of each of the 10 persons in the dataset along with the expression recognition accuracy for that person. We can see the very high intra-person differences in this dataset, which results in very different accuracies for the different persons and hence high standard deviation, for all the methods.

[3] http://www.kasrl.org/jaffe.html

Figure 4.4: Example images (pairs of images of the same person) from the Labeled Faces in the Wild [39] (aligned version [100]) face verification dataset

Face verification

Labeled Faces in Wild (LFW) [39] is a popular dataset for face verification by unconstrained pair matching *i.e.* given two real-world face images decide whether they are of the same person or not. LFW contains 13,233 face images of 5749 different individuals of different ethnicity, gender, age, *etc.* . It is an extremely challenging dataset and contains face images with large variations in pose, lighting, clothing, hairstyles *etc.* . LFW dataset is organized into two parts: 'View 1' is used for training, validation (*e.g.* for choosing the parameters) while 'View 2' is only for final testing and benchmarking. In our setup, we follow the specified training and evaluation protocol. We use the aligned version of the faces as provided by Wolf et al. [100][4]. Figure 4.4 shows example pairs of images of the same person from the aligned version of the database.

We work in the restricted unsupervised task of the LFW dataset i.e.

(i) We use strictly the data provided without any other data from any other source and

(ii) We do not utilize class labels while obtaining the image representation.

We divide the 50×40 pixels resized images into 5×4 grid of 10×10 pixels cells. We com-

[4]http://www.openu.ac.il/home/hassner/data/lfwa/

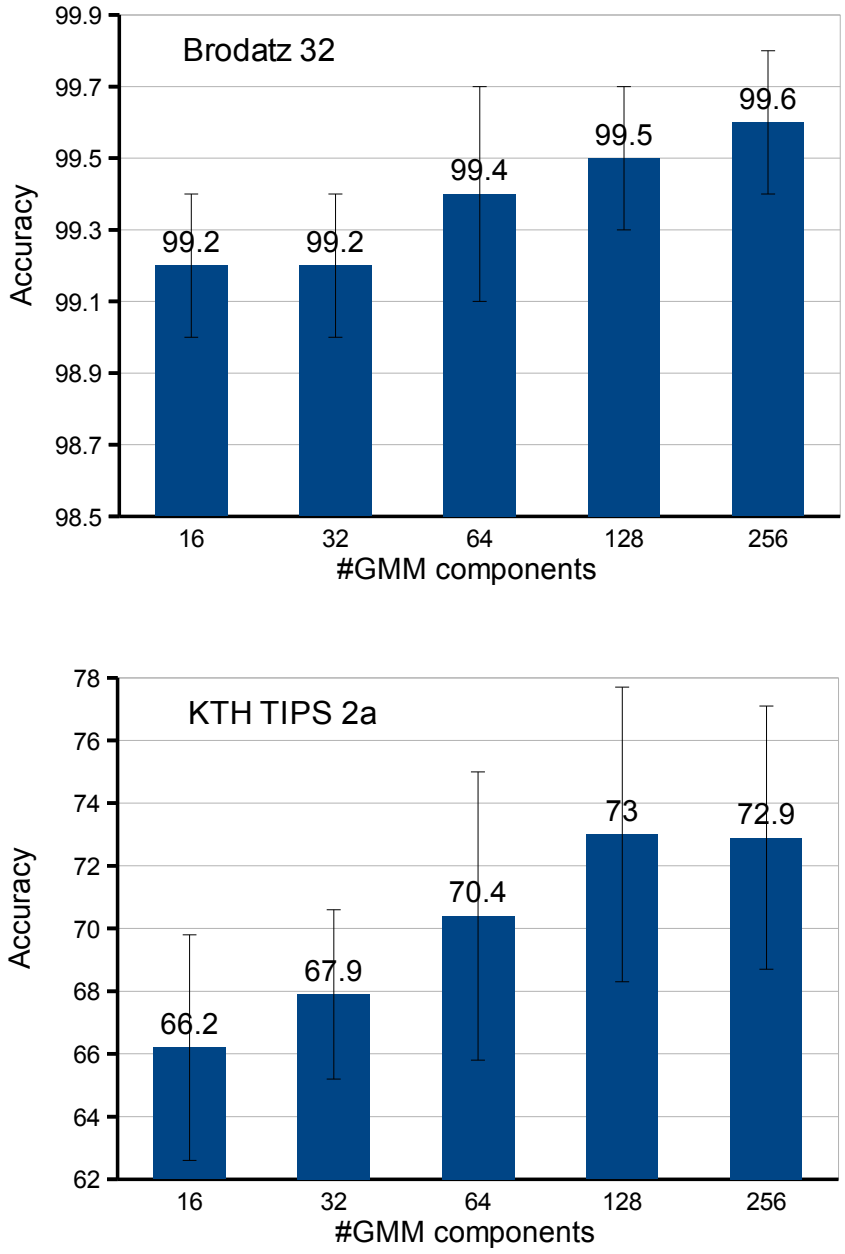

Figure 4.5: The accuracies of the method for different number of GMM components for Brodatz (left) and KTH TIPS 2a (right) dataset (see Sec. 4.3.3)

pute the LHS representations for each cell separately and compute the similarity between image pairs as the mean of L2 distances between the representations of corresponding cells. We classify image pairs into same or not same by thresholding on their similarity. We choose the testing threshold, as the threshold obtaining the best classification on the training data. We obtain an accuracy of 73.4% with a standard error on the mean of 0.4%. This is the highest performance till date in the unsupervised setting for the dataset. We compare with other approaches, including those based on LBP in Sec. 4.3.4.

4.3.3 Effect of sampling and number of components

Table 4.1 gives the results with (a) rectangular 3×3 pixel neighbourhood and (b) LBP/LTP like circular sampling of 8 neighbours, where the diagonal neighbour values are obtained by bilinear interpolation. Performance on the Brodatz dataset is similar for both the samplings while that for KTH and JAFFE datasets differ. In general, the circular sampling seems to be better for all the methods. We note that the variations and difficulty of Brodatz dataset are much less than the other two datasets and hence is possibly well represented by either of the two samplings. Thus, we conclude that, in general, circular sampling is to be preferred as it seems to generate more discriminative statistics.

Fig. 4.5 shows the performance on the two texture datasets for different number of mixture model components. As this number increases the vector length increases proportionally. While lower number of components lead to a compact representation, larger numbers lead to better quantization of the space and hence more discriminative representations. We observe that the performance, for both the datasets, increases with the number of components and seems to saturate after a value of 128. Hence, we report results for 128 components. For Brodatz dataset, we see that even with only 16 components the method is able to achieve more than 99% accuracy, highlighting the fewer variations in the dataset. For the KTH dataset we gain significantly by going from 16 to 128 components (6.8 points) which suggests that for more challenging tasks a more descriptive representation is beneficial.

4.3.4 Comparison with existing methods

Table 4.2 shows the performance of our method along with existing methods. On the Brodatz dataset we outperform all methods and to the best of our knowledge report, near perfect, state-of-the-art performance. Similarly, on the JAFFE and LFW datasets we achieve the best results reported till date.

On the KTH dataset, Chen *et al.* [14] recently proposed features based on Weber law. They report a performance of 64.7% with KNN classifier. Caputo *et al.* [12] reported 71.0% with 3-scale LBP and *non-linear* chi-squared RBF kernel based SVM classifier. Here, we

Table 4.2: Comparison with current methods with comparable experimental setup (reports accuracy, see Sec. 4.3.4).

(a) Brodatz–32

Method	Acc.
Urbach *et al.* [91]	96.5
Chen *et al.* [14]	97.5
LHS (ours)	**99.3**

(b) KTH TIPS 2a

Method	Acc.
Chen *et al.* [14]	64.7
Caputo *et al.* [12]	71.0
LHS (ours)	**73.0**

(c) JAFFE

Method	Acc.
Shan *et al.* [82]	81.0
Feng *et al.* [30]	93.8
LHS (ours)	**95.6**

(d) LFW (aligned)

Method	Acc.
Javier *et al.* [45]	69.5 ± 0.5
Seo *et al.* [81]	72.2 ± 0.5
LHS (ours)	$\mathbf{73.4 \pm 0.4}$

use linear classifiers which, are not only fast to train but also, need only a vector dot product at test time (cf. kernel computation with support vectors which are of the order of number of training features). Note that their best results were obtained with complex decision tree with non-linear classifiers at every node, with multi scale features. We expect our features to outperform the features they used with similar complex classifier.

Table 4.1 shows our performance and that of competing unsupervised methods[5]. Our method not only outperform the LBP baseline (LBP with χ^2 distance) [45] by 3.9% but also give 1.2% better performance than current state-of-the-art Locally Adaptive Regression Kernel (LARK) features of [81]. The better performance of our features compared to the LBP baseline and fairly complex LARK features on this difficult dataset once again underlines the fact that local neighborhood contains a lot of discriminative information. It also demonstrates the representational power of our features which are successful in encoding the information which is missed by other methods.

Thus the proposed method is capable of achieving state-of-the art results while being computationally simple and efficient.

4.4 Conclusions

We have presented a model, capturing higher-order statistics of small local pixel neighborhoods, which leads to a highly discriminative representation of the images. We showed, with experiments on two challenging texture datasets and two challenging facial analysis datasets, that the model codes more information than competing methods and achieves state-of-the-art results.

[5]results reproduced from webpage: `http://vis-www.cs.umass.edu/lfw/results.html`

We have shown that local neighborhoods can give very good results by themselves and combining them with more global features would be a promising direction which we will explore in future.

Chapter 5

Summary

In this thesis, we have presented methods for representing images for the tasks of human attributes classification, human action classification and human facial analysis. Image representation is a critical component of any vision system and has been an active research problem in the community. We now summarize the contributions made in this thesis and then present some directions of possible future work.

5.1 Discriminative Spatial Representation

In Chapter 2, we proposed to learn the spatial partitioning of the images to derive a more discriminative image representation for a given task. The proposed method addresses two limitations of the standard spatial pyramid representation (SPR) *i.e.*

(i) The grids learnt are adapted to the distribution of discriminative information and are not, in general, uniform and

(ii) The grids are learnt discriminatively and are, in general, different for different tasks.

We showed that the method performs better than low dimensional SPR at comparable vector lengths and achieves similar results, when compared to full SPR, with less than half the vector length. The method has general applicability to other visual tasks where the distribution of spatial information is not completely flat *e.g.* we demonstrated the method on Scene recognition and object image classification as well.

Future work. We considered only one final grid as we expected the grids to adapt to the content and hence adapt automatically to the modes in the distribution of discriminative information, even if they occur at multiple scales. However, in more complicated scenarios, similar interesting regions in different images *e.g.* attributes, objects, might themselves be clustered at multiple scales due to the aspect of the human in the image

e.g. 'bent arm' would be placed differently (absolute) in images with full human (head to toe) and those with only upper body (head to waist). In such a case it might be interesting to explore a mixture model like setting where multiple grids are learnt for the same task, each tuned to the different groups of similar layout images. The best one or a weighted combination of all of them may then be considered for the final decision.

Another extension could be a simple extension to spatio-temporal domain *i.e.* learn similar partitions in the three dimensional domain of human analysis in videos. There the 'grids' will have three dimensional cell volumes as partitions instead of the two dimensional spatial cells in the present.

5.2 Discriminative Spatial Saliency

In Chapter 3, we proposed a method to learn the discriminative saliency of images given a classification task. We argued that the (discriminative) saliency of a region depends not only on the content of the region but also on its absolute position in the image *i.e.* there is a spatial component in discriminative saliency for a given visual task. We proposed to learn such discriminative spatial saliency for classification. The proposed method addresses an important issue of adaptation of the representation per image, based on the specific distribution of discriminative information in that image *e.g.* the representation considers the fact that in two different images of persons with 'bent arms' the arms may be positioned slightly differently. We showed that the method achieves better or comparable results *w.r.t.* relatively more computationally expensive state-of-the-art methods on public datasets of human actions and fine grained classification involving humans.

Future work. The method uses the uniform spatial partition to learn the saliency as weighted combination of cell representations. Using learnt grids similar to those proposed in Chapter 2 with the saliency learning would be an interesting extension. In this approach the resolution of spatial quantization and the relative importance of different spatial regions could be optimized simultaneously along with the classifier for a given classification task. This is an interesting direction to follow.

Another extension could be, like for the previous method, having a mixture model like formulation which accounts for the clustering of aspects of humans in images *e.g.* full human images *vs.* images where only upper body of the humans is visible.

5.3 Local Higher-Order Statistics

In Chapter 4, we proposed a new image representation based on higher-order Statistics of local pixel neighborhoods. The proposed method addressed the following limitations of Local Binary and Ternary Patterns (LBP/LTP).

(i) Instead of using a fixed quantization of space, as a result of quantizing each coordinate into two/three bins in LBP/LTP, the proposed method learns the quantization of the space,

(ii) The proposed method does not need to use heuristic based pruning of volumes with low occupancy, which results from discarding the non-uniform patterns in LBP/LTP, as all this is automatically learnt and,

(iii) The proposed method uses higher-order Statistics of the pixels while LBP/LTP use only zeroth order counting statistics (as they are histograms).

We showed experimentally that the proposed method reaches state-of-the-art performance in facial analysis tasks and also on texture recognition for which the original local patterns were designed.

Future work. The work presented demonstrated that rich description of very small local pixel neighborhoods can achieve very high performance, however, the overall global structure is still missing. It would be interesting to combine this representation with those exploiting a more global layout. It would also be interesting to explore the use of these features in object detection as well, where LBP/LTP have already achieved some success.

5.4 Conclusion

In the present thesis we have mainly addressed the problem of image representation, with a stress on human focused tasks.

Human focused visual data makes up a large chunk of the total visual data on the internet and that generated by surveillance. In the near future, it will be critical to have good representations in order to be able to accurately analyze and understand images using automatic computer vision technologies. Analyzing human faces would also become an important technology owing to its numerous important applications *e.g.* surveillance, human computer interaction, medical applications etc.

Towards these goals, we have proposed image representations which extend the current state-of-the-art either by better exploiting the spatial distribution of discriminative information or by providing a rich description of highly local pixel neighborhoods. We have

demonstrated, with experiments on challenging datasets, that our proposed methods perform better or comparable to the state-of-the-art methods. We have also demonstrated the generality of the proposed methods by experiments on other computer vision datasets *e.g.* scene classification and object image classification.

In addition to study the human attributes better we have proposed a challenging dataset of human attributes with 9344 human images collected from unconstrained images automatically downloaded from the internet. The database is annotated with the presence of 27 human attributes based on sex, age, appearance/clothes and pose.

We hope our contributions have advanced the computer vision technologies towards the goal of semantic description of humans in images.

Appendix A

Latent Support Vector Machine

In this appendix, we give a brief overview of latent support vector machines (LSVM). We encourage the reader to refer Felzenszwalb *et al.* [28] for a more complete description.

Consider a case where the representation of an example depends on the current value of same latent variable z. Let us denote the representation of the example as x_z where the subscript emphasizes the dependence on z. In the following, we abuse the notation a bit and use z to denote both the latent variable and its current value.

Now consider scoring the example *w.r.t.* a linear separating hyperplane w as

$$f(x) = \max_z w^T x_z. \tag{A.1}$$

With such scoring, the model learning problem can be formulated, analogous to standard SVM with hinge loss, as

$$\min_w \ \frac{1}{2}||w||^2 + C\sum_i \max(0, 1 - y^i f(x^i)), \tag{A.2}$$

where i indexes the training examples. This problem is called the latent SVM problem.

Note that, in the case of only one possible value for the latent variable the problem comes down to the standard linear SVM. A latent SVM problem is semi-convex, in the sense that the objective function is convex (in w) for the negative examples and, in general, non-convex for the positive examples.

Latent SVMs are usually learned with a simple learning strategy similar in spirit to the block coordinate descent method. The learning consists of two alternating steps

(i) Optimize (the objective) by selecting the highest scoring latent values for the positive examples.

(ii) Optimize over w by fixing the latent variables for the positive examples.

Both the steps are proved to always improve or maintain the value of the objective function. The learning process is sensitive to the initialization of w. A bad initialization could lead to unreasonable values for latent variables which in turn would lead to a bad overall model.

Usually, first step is solved with a search over the (usually discrete) latent variables, while second step is solved by stochastic gradient descent using the, easy to compute analytically, sub gradient *w.r.t.* w [28].

Appendix B

HAT database queries

To download images for our database of Human Attributes (HAT) (see Section 2.3 on page 25), we manually designed queries as input to `www.flickr.com` and downloaded the top result images. The following are the queries we used to download the images from the internet.

party people	girl with cap	tshirt people	football practice
people laughing	curly hairs	carrying an umbrella	baseball kid
beach people	wearing glasses	umbrella girl	baseball girl
market people	guy with glasses	umbrella guy	baseball game
fest people	girl with glasses	high heel girl	baseball park game
banquet people	wearing shorts	swimsuit fashion	kids park play
prom people	wearing jacket	fashion beach	kids swing
carnival people	running kids	swimwear	wearing sweater
shopping mall people	elderly lady	long legs girl	teen sunglass
dancing people	elderly man	girls in costumes	sunglass hot
crazy people	people hanging around	costume party	glasses kid
america people	talking on cellphone	hot pants	geek glasses
europe people	talking on phone	beach shorts	geek girl
mom	running late	jog	blue eyes kid
dad	marathon run	strapless dress	blue teen
sister	atheletics	black dress	zoo trip kid
brother	political rally	pink skirt	attitude girl
aunty	student fair	dancing crazy	attitude guy
uncle	dance fest	disc party	kid making faces
family	volley game	playing disc	kid posing
happy girls	jockey horse	tennis girl	old man walk
baby	lawn party	tennis guy	old lady walk
small girl	pool dance	tennis match	grand mother
small boy	basketball game	tennis serve	grandpa
smart guy	lunch party	basketball play	grandma
smart lady	wine party	basketball slam	grandma party
couple happy	office party	cheerleading	grandpa party

wedding party
conference people
rock concert
jazz concert
piano concert
acoustic guitar people
soccer people
soccer
cricket game
baseball game
vacation couple
vacation people
nice shades
nice sunglasses
colored hair
asian people
asia people
europe people
new york people
farm people
people watching
people in jeans
formal people
walk on beach
park stroll
people watching
wearing skirt
wearing blue jeans
lipstick girl
sports fans
people on a roll
people enjoying
people on vacations
tourists
cute girls
cute guys
women
men
traditional dress people
traditional dance
seashore walk
pool party
kids playing in the yard
soccer players
helmet riding
wearing shades

dance party
park kids
thailand tourists
india people
france people
new york people
california people
singing people
walking the dog
in pyjamas
tourist girls
tourist guys
rome tourists
china tourists
singapore tourists
singapore people
indonesia people
new york police
neighbors
arguing
wearing a hat
old hat
hat lady
hat guy
baseball cap wearing
cap people
married couple
wedding dress party
wedding dress
just married
baby cute
baby boy
baby girl
one year old
second birthday
best man
best maid
girl in a dress
girl dancing
girl having fun
guys dancing
guys having fun
fat guy
skinny jeans
blue jeans people
girl posing

cheerleader girl
cute skirt
park picnic
eating icecream
surfing
enjoying the beach
girl laughing
guy laughing
people laughing
enjoying the sun
enjoying snow
girl jacket
guy jacket
snow jacket
ski people
sledging
rock climbing
climbing girl
climbing guy
climbing helmet
bike helmet girl
bike helmet guy
helmet sport
helmet kid
helmet man
firefighter man
leather jacket girl
hiking girl
hiking couple
touring couple
couple fighting
singing girl
kidding girl
kidding girl
teen girl
teen party
teen play
prom guys
prom girls
prom couple
junior prom
senior prom
prom cute
evening dress girl
sweet kid
siblings

cousin
cousin playing
baby in lap
baby suit
teen portrait
glasses cute
glasses new
joker party
theme party
college party
college prom
couple date
school kid
school boy
school girl
school teen
blonde teen
blonde guy
funny kid
backpack
backpack girl
backpack guy
kid school bag
charity teen
charity girl
charity guy
charity people
dinner teen
dinner kid
dinner family
family park
family dance
family beach
family party
family glasses
family sunglasses
friends party
friends dance
friends trip
friends walk
friends jump
friends beach
friends wed
friends marry
friends market
friends fun

football
baseball
pubbing
oktoberfest people
egypt crowd
picnic game
street play
pillow fight
bikers
guy with cap
guy posing
swimsuit girl
beach fun
beach bikini
long hairs girls
short dress girls
short skirt girls
dotted dress
checked shirt
siblings play
player girl
playing the guitar
soccer kid
school soccer
soccer friends
football kids
football girl play
football boy play
friends play
friends marathon
friends run
athletics
friends convocation
friends college
friends sleepover
friends night
friends nightout

Appendix C

Publications

The following publications were made as a result of the work carried out for the thesis:

- Gaurav Sharma and Frédéric Jurie, Learning discriminative representation for image classification, in *British Machine Vision Conference*, 2011 (**Oral presentation**)
- Gaurav Sharma, Frédéric Jurie, and Cordelia Schmid, Discriminative spatial saliency for image classification, in *Computer Vision and Pattern Recognition*, 2012
- Gaurav Sharma, Sibt ul Hussain, and Frédéric Jurie, Local higher-order statistics (LHS) for texture categorization and facial analysis, in *European Conference on Computer Vision*, 2012

List of Figures

List of Tables

List of Algorithms

Bibliography

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 28(12), 2006.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. SURF: Speeded up robust features. *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008.

[3] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 19(7):711–720, 1997.

[4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24:509–522, 2001.

[5] A. C. Berg, P. N. Belhumeur, N. Kumar, and S. K. Nayar. Attribute and simile classifiers for face verification. In *International Conference on Computer Vision (ICCV)*, 2009.

[6] H. Bilen, V. Namboodiri, and L. Van Gool. Object and action classification with latent variables. In *British Machine Vision Conference (BMVC)*, 2011.

[7] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[8] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *Conference on Image and Video Retrieval (CIVR)*, 2007.

[9] P. Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover Publications, New York, 1966.

[10] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[11] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang. Spatial bag-of-features. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[12] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *International Conference on Computer Vision (ICCV)*, 2005.

[13] O. Chapelle. Training a support vector machine in the primal. *Neural Computation*, 19(5):1155–1178, 2007.

[14] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, and W. Gao. WLD: A robust local image descriptor. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(9):1705–1720, 2010.

[15] M. Croiser and L. D. Griffin. Using basic image features for texture classification. *International Journal of Computer Vision (IJCV)*, 88:447–460, 2010.

[16] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Intl. Workshop on Stat. Learning in Comp. Vision*, 2004.

[17] O. G. Cula and K. J. Dana. Compact representation of bidirectional texture functions. In *Computer Vision and Pattern Recognition (CVPR)*, 2001.

[18] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition (CVPR)*, 2005.

[19] V. Delaitre, I. Laptev, and J. Sivic. Recognizing human actions in still images: A study of bag-of-features and part-based representations. In *British Machine Vision Conference (BMVC)*, 2010.

[20] V. Delaitre, J. Sivic, and I. Laptev. Learning person-object interactions for action recognition in still images. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

[21] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative models for static human-object interactions. In *Computer Vision and Pattern Recognition (CVPR) Workshops*, 2010.

[22] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (Second Ed.)*. Wiley-Interscience, 2001.

[23] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html, 2007.

[24] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html, 2010.

[25] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2011 (VOC2011) Results. http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html, 2011.

[26] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[27] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Computer Vision and Pattern Recognition (CVPR)*, 2005.

[28] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(9):1627–1645, 2010.

[29] P. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision (IJCV)*, 61:55–79, 2005.

[30] X. Feng, M. Pietikäinen, and T. Hadid. Facial expression recognition with local binary patterns and linear programming. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 15:546–548, 2005.

[31] R. Fergus, P. Perona, and A. Zisserman. Weakly supervised scale-invariant learning of models for visual recognition. *International Journal of Computer Vision (IJCV)*, 71(3):273–303, March 2007.

[32] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 100(1):67–92, 1973.

[33] D. Gao and N. Vasconcelos. Discriminant saliency for visual recognition form cluttered scenes. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[34] D. Gao and N. Vasconcelos. Integrated learning of saliency, complex features and object detectors from cluttered scenes. In *Computer Vision and Pattern Recognition (CVPR)*, 2005.

[35] A. Gupta, A. Kembhavi, and L. S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31:1775–1789, October 2009.

[36] T. Harada, Y. Ushiku, Y. Yamashita, and Y. Kuniyoshi. Discriminative spatial pyramid. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[37] H. Harzallah, F. Jurie, and C. Schmid. Combining efficient object localization and image classification. In *International Conference on Computer Vision (ICCV)*, 2009.

[38] E. Hayman, B. Caputo, M. Fritz, and J.-O. Eklundh. On the significance of real world conditions for material classification. In *European Conference on Computer Vision (ECCV)*, 2004.

[39] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[40] N. Ikizler, G. R. Cinbis, S. Pehlivan, and P. Duygulu. Recognizing actions in still images. In *International Conference on Pattern Recognition (ICPR)*, 2008.

[41] N. Ikizler and P. Duygulu. Histogram of oriented rectangles: A new pose descriptor for human action recognition. *Image and Vision Computing (IVC)*, 27(10):1515–1526, 2009.

[42] N. Ikizler-Cinbis, G. R. Cinbis, and S. Sclaroff. Learning actions from the web. In *International Conference on Computer Vision (ICCV)*, 2009.

[43] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 20(11):1254 – 1259, 1998.

[44] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *Advances in Neural Information Processing Systems (NIPS)*, 1998.

[45] R. S. Javier, V. Rodrigo, and C. Mauricio. Recognition of faces in unconstrained environments: a comparative study. *EURASIP Journal on Advances in Signal Processing*, 2009.

[46] F. S. Khan, J. van de Weijer, and M. Vanrell. Top-down color attention for object recognition. In *International Conference on Computer Vision (ICCV)*, 2009.

[47] C. Koch and S. Ullman. Shifts in selective visual attention: Towards underlying neural circuitry. *Human Neurobiology*, 4:219–227, 1985.

[48] J. Krapac, J. Verbeek, and F. Jurie. Learning tree-structured descriptor quantizers for image categorization. In *British Machine Vision Conference (BMVC)*, 2011.

[49] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27:1265–1278, 2005.

[50] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition (CVPR)*, 2006.

[51] A. Lehmann, B. Leibe, and L. Van Gool. Fast prism: Branch and bound hough transform for object class detection. *International Journal of Computer Vision (IJCV)*, 94(2):175–197, 2011.

[52] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision (IJCV)*, 77(1):259–289, 2008.

[53] T. J. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision (IJCV)*, 43:29–44, 2001.

[54] S. Liao, W. Fan, A. C. Chung, and D. Yan Yeung. Facial expression recognition using advanced local binary patterns, Tsallis entropies and global appearance features. In *ICIP*, 2006.

[55] D. Liu, G. Hua, P. Viola, and T. Chen. Integrated feature selection and higher-order spatial feature extraction for object categorization. In *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[56] L. Liu, P. Fieguth, and G. Kuang. Compressed sensing for robust texture classification. In *Asian Conference on Computer Vision (ACCV)*, 2010.

[57] Y. Liu, D. Zhang, G. Lu, and W. Ma. A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 40(1):262–282, 2007.

[58] D. Lowe. Distinctive image features form scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[59] M. J. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba. Coding facial expressions with gabor wavelets. In *Automatic Face and Gesture Recognition (FG)*, 1998.

[60] M. Marszalek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning object representations for visual object class recognition. In *PASCAL Visual recognition challenge workshop*, 2007.

[61] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision (IJCV)*, 60(1):63–86, 2004.

[62] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(10):1615–1630, 2005.

[63] F. Moosmann, D. Larlus, and F. Jurie. Learning saliency maps for object categorization. In *European Conference on Computer Vision (ECCV) Workshops*, 2006.

[64] N. Morioka and S. Satoh. Building compact local pairwise codebook with joint feature space clustering. In *European Conference on Computer Vision (ECCV)*, 2010.

[65] N. Morioka and S. Satoh. Learning directional local pairwise bases with sparse coding. In *British Machine Vision Conference (BMVC)*, 2010.

[66] N. Murray, M. Vanrell, X. Otazu, and C. A. Parraga. Saliency estimation using a non-parametric low level vision model. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[67] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *European Conference on Computer Vision (ECCV)*, 2006.

[68] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(7):971–987, July 2002.

[69] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *International Conference on Computer Vision (ICCV)*, 2011.

[70] D. Parikh, L. Zitnick, and T. Chen. Determining patch saliency using low-level context. In *European Conference on Computer Vision (ECCV)*, 2008.

[71] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *Computer Vision and Pattern Recognition (CVPR)*, 2007.

[72] F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *European Conference on Computer Vision (ECCV)*, 2010.

[73] M. Pietikainen, A. Hadid, G. Zhao, and T. Ahonen. *Computer Vision Using Local Binary Patterns*. Springer, 2011.

[74] A. Prest, C. Schmid, and V. Ferrari. Weakly supervised learning of interactions between humans and objects. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2011.

[75] T. Quack, V. Ferrari, B. Leibe, and L. van Gool. Efficient mining of frequent and distinctive feature configurations. In *International Conference on Computer Vision (ICCV)*, 2007.

[76] D. Ramanan. Learning to parse images of articulated objects. In *Advances in Neural Information Processing Systems (NIPS)*, 2006.

[77] R. Ronfard, C. Schmid, and B. Triggs. Learning to parse pictures of people. In *European Conference on Computer Vision (ECCV)*, 2002.

[78] S. Savarese, J. Winn, and A. Criminisi. Discriminative object class models of appearance and shape by correlatons. In *Computer Vision and Pattern Recognition (CVPR)*, 2006.

[79] B. Scholkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[80] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[81] H. J. Seo and P. Milanfar. Face verification using the LARK representation. *IEEE Transactions on Information Forensics and Security*, 6(4):1275–1286, 2011.

[82] C. Shan, S. Gong, and P. W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing (IVC)*, 27:803–816, 2009.

[83] G. Sharma and F. Jurie. Learning discriminative representation for image classification. In *British Machine Vision Conference (BMVC)*, 2011.

[84] G. Sharma, F. Jurie, and C. Schmid. Discriminative spatial saliency for image classification. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[85] G. Sharma, S. ul Hussain, and F. Jurie. Local higher-order statistics (LHS) for texture categorization and facial analysis. In *European Conference on Computer Vision (ECCV)*, 2012.

[86] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)*, 2003.

[87] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *TIP*, 19(6):1635–1650, 2010.

[88] A. M. Treisman and G. Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, 1980.

[89] M. Turk and A. Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition (CVPR)*, 1991.

[90] S. ul Hussain. *Machine Learning Methods for Visual Object Detection*. PhD thesis, Laboratoire Jean Kuntzmann, 2011.

[91] E. R. Urbach, J. B. Roerdink, and M. H. Wilkinson. Connected shape-size pattern spectra for rotation and scale-invariant classification of gray-scale images. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(2):272–285, 2007.

[92] A. Vailaya, A. Jain, and H. Zhang. On image classification: City images vs. landscapes. *Pattern Recognition*, 31(12):1921–1935, 1998.

[93] K. Valkealahti and E. Oja. Reduced multidimensional co-occurence histograms in texture classification. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 20(1):90–94, 1998.

[94] M. Varma and A. Zisserman. Texture classification: Are filter banks necessary? In *Computer Vision and Pattern Recognition (CVPR)*, 2003.

[95] M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *International Journal of Computer Vision (IJCV)*, 62:61–81, 2005.

[96] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008.

[97] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *International Conference on Computer Vision (ICCV)*, 2009.

[98] A. Vedaldi and A. Zisserman. Efficient additive kernels using explicit feature maps. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[99] M. Wang, J. Konrad, P. Ishwar, K. Jing, and H. Rowley. Image saliency: From intrinsic to extrinsic context. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[100] L. Wolf, T. Hassner, and Y. Taigman. Similarity scores based on background samples. In *Asian Conference on Computer Vision (ACCV)*, 2009.

[101] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large scale scene recognition from abbey to zoo. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[102] Y. Xu, H. Ji, and C. Fermuller. View point invariant texture description using fractal analysis. *International Journal of Computer Vision (IJCV)*, 83:85–100, 2009.

[103] Y. Xu, X. Yang, H. Ling, and H. Ji. A new texture descriptor using multifractal analysis in multi-orientation wavelet pyramid. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[104] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[105] J. Yang, K. Yu, and T. Huang. Efficient highly over-complete sparse coding using a mixture model. In *European Conference on Computer Vision (ECCV)*, 2010.

[106] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[107] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1385–1392. IEEE, 2011.

[108] B. Yao and L. Fei-Fei. Grouplet: A structured image representation for recognizing human and object interactions. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[109] B. Yao and L. Fei-Fei. Action recognition with exemplar based 2.5d graph matching. In *ECCV*, Firenze, Italy, October 2012.

[110] B. Yao and L. Fei-Fei. Recognizing human-object interactions in still images by modeling the mutual context of objects and human poses. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2012 (In Press).

[111] B. Yao, A. Khosla, and L. Fei-Fei. Combining randomization and discrimination for fine-grained image categorization. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[112] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision (IJCV)*, 73(2):213–238, 2007.

[113] X. Zhou, N. Cui, Z. Li, F. Liang, and T. Huang. Hierarchical Gaussianization for image classification. In *International Conference on Computer Vision (ICCV)*, 2009.

[114] X. Zhou, K. Yu, T. Zhang, and T. S. Huang. Image classification using super-vector coding of local image descriptors. In *European Conference on Computer Vision (ECCV)*, 2010.

[115] S. C. Zhu, Y. Wu, and D. Mumford. Filters, random-fields and maximum-entropy (FRAME): Towards a unified theory for texture modeling. *International Journal of Computer Vision (IJCV)*, 27:107–126, 1998.

[116] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2879–2886. IEEE, 2012.

Description Sémantique des Humains Présents dans des Images Vidéo

Dans cette thèse, nous nous intéressons à la description sémantique des personnes dans les images en termes (i) d'attributs sémantiques (sexe, age), (ii) d'actions (court, saute) et d'expressions faciales (sourire).
Tout d'abord, nous proposons une nouvelle représentation des images permettant d'exploiter l'information spatiale spécifique à chaque classe. La représentation standard, les pyramides spatiales, suppose que la distribution spatiale de l'information est (i) uniforme et (ii) la même pour toutes les tâches. Au contraire notre représentation se propose d'apprendre l'information spatiale discriminante pour une tâche spécifique. De plus, nous proposons un modèle qui adapte l'information spatiale à chaque image. Enfin, nous proposons un nouveau descripteur pour l'analyse des expressions faciales. Nous apprenons un partitionnement de l'espace des différences locales d'intensité à partir duquel nous calculons des statistiques d'ordre supérieur pour obtenir des descripteurs plus expressifs.
Nous proposons également une nouvelle base de données de 9344 images de personnes collectées sur l'Internet avec les annotations sur 27 attributs sémantiques relatifs au sexe, à l'age, à l'apparence et à la tenue vestimentaire des personnes. Nous validons les méthodes proposées sur notre base de données ainsi que sur des bases de données publiques pour la reconnaissance d'actions et la reconnaissance d'expressions. Nous donnons également nos résultats sur des bases de données pour la reconnaissance de scènes, le classement d'images d'objets et la reconnaissance de textures afin de montrer le caractère général de nos contributions.

Mot-clés: Vision par ordinateur; Apprentissage automatique; Illustrations, images, etc., – Interpretations; Perception des visages

Semantic Description of Humans in Images

In the present thesis we are interested in semantic description of humans in images. We propose to describe humans with the help of (i) semantic attributes e.g. female, elderly, (ii) actions e.g. running, jumping and (iii) facial expressions e.g. smiling.
First, we propose a new image representation to exploit class specific spatial information. The standard representation i.e. spatial pyramids, assumes that distribution of spatial information is (i) uniform and (ii) same for all tasks. We propose to learn the discriminative spatial information for a specific task. Further, we propose a model that adapts the spatial information for each image. Finally, we propose a new descriptor for facial expression analysis. We work in the space of intensity differences of local pixel neighborhoods and propose to learn the quantization of the space and use higher order statistics to obtain expressive descriptors.
We introduce a challenging dataset of 9344 human images, sourced from the internet, with annotations for 27 semantic attributes based on sex, pose, age and appearance/clothing. We validate the proposed methods on our dataset as well as on publicly available datasets of human actions, fine grained classification involving human and facial expressions. We also report results on related computer vision datasets for scene recognition, object image classification and texture categorization, to highlight the generality of our contributions.

Keywords: Computer vision; Machine learning; Picture interpretation; Face perception

Discipline: Informatique et applications

Laboratoire: GREYC CNRS UMR 6072, Sciences 3, Campus 2, Bd Marechal Juin, Université de Caen, 14032 Caen