Available online at www.sciencedirect.com

ScienceDirect

Pattern Recognition Letters 27 (2006) 1673–1684

Pattern Recognition Letters

www.elsevier.com/locate/patrec

Learning spatial relations in object recognition

Thang V. Pham *, Arnold W.M. Smeulders

ISIS, Informatics Institute, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands

Received 14 July 2004; received in revised form 7 March 2006
Available online 10 July 2006

Communicated by T. Breuel

Abstract

This paper studies two types of spatial relationships that can be learned from training examples for object recognition. The first one employs deformable relationships between object parts with a Gaussian model, while the second one describes pairwise relationships between pixel intensity values using Bayesian networks. We perform experiments on a human face dataset and a horse dataset, imposing the same amount of annotation of training data, which can be seen as sending knowledge to the learning algorithms. The result indicates that the Bayesian network method compares favorably to the deformable model, as it can capture long-distance stable relations in the object appearance. We also conclude that both methods are superior to strictly spatial matching by template and strictly non-spatial classifiers.

Keywords: Articulated object; Bayesian network; Deformable model; Part-based approach; Shape; Spatial relation

1. Introduction

Object is an abstract concept for which just few aspects are valid in general, for all types of objects. In the recognition of object classes, objects are usually characterized by distinct appearances (signs), textures (tiger), features (coins) or parts (eyes plus mouth for a face). In this paper, we concentrate on the other common aspect of object: that they share some spatial order. We focus on learning spatial relationships among in the object appearance or among object parts for the purpose of object recognition.

In the study of spatial relationships one may take a syntactical approach. Tagare et al. (1995) derives a metric between two Voronoi diagrams to describe spatial relationships of similar patches. This is similar to the approach in (Egenhofer, 1997) where they admit a relaxation of the spatial constraint. At any rate, such approaches require a very robust part-labeling scheme. They are too brittle for the effects of partial occlusion and undetected parts that the method is not effective for object recognition.

Graph matching (Bunke, 2000) provides another, more frequently explored path to describe the spatial relations of object parts. The construction of object graphs is not obvious. The method in (Rocha and Pavlidis, 1994), for example, resorts to elaborate feature extractors together with manual interactive design. A separate class is the shock graphs capturing the object topological structure (Sebastian et al., 2004; Siddiqi et al., 1999). This is an important class because the extraction and subsequent matching can be achieved efficiently. Nevertheless, the shock graph as well as graph representation in general lack a statistical model that can capture the variation of a collection of objects (or graphs). As a consequence, these approaches are not yet suitable for learning and recognition of object classes.

In this paper we contrast two types of spatial relationships that can be learned from a set of examples for object recognition. The first uses deformable relationships between named entities, for examples the approaches in

* Corresponding author. Tel.: +31 20 525 7553; fax: +31 20 525 7490.
E-mail address: vietp@science.uva.nl (T.V. Pham).

0167-8655/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2006.03.016

(Burl and Perona, 1996; Weber et al., 2000; Fergus et al., 2003; Fischler and Elschlager, 1973; Felzenszwalb and Huttenlocher, 2000). The named entities are derived by specific part detectors, as opposed to Harris detectors and spot detectors generating nameless points of interest. Using parts is advised when one wishes to extend the class of objects with a fixed spatial layout to include objects with articulated structures. Examples are windmills and animals with their limbs. One critical issue not fully addressed in these approaches is that of missing parts and image clutter. In this study, we choose a method using a translation invariant shape configuration (Pham and Smeulders, 2004) where the problem of missing parts and image clutter is handled efficiently.

The second type of spatial relations we study are between pairs of intensity values on the face of the object. For that purpose, the spatial relation of a pair of pixels is combined with pixel intensities at either end. Such is the topic of "graphical models", which utilize results in graph theory and probability theory into a single framework (Pearl, 1988; Jordan, 1999). Unlike the aforementioned graph matching approach where each object is associated with a separate graph, in this approach a graph describes a class of objects. Since the class of an object can be derived easily by holding it against the graphical model, this approach is suited for the object class recognition problem. Graphical models also differ from other statistical approaches that have been applied to the object recognition problem, such as (Sung and Poggio, 1998; LeCun et al., 1998; Cortes and Vapnik, 1995), in that the spatial relationships among input pixels, are described explicitly. One critical issue in this approach is how to learn the structure from examples. Here we consider the graphical model in (Pham et al., 2002) that looks for the co-occurrence of pixel values over distance. This model is learned efficiently and shows superior performance (Pham et al., 2002) in comparison with methods employing simpler structures (Schneiderman and Kanade, 2000; Colmenarez and Huang, 1997).

We contrast the recognition performance of a model for coarse-grain spatial relationship between parts (Pham and Smeulders, 2004) with two models for fine-grain spatial relation between pixels (Pham et al., 2002). We do so on two data sets, one being human faces in frontal view as the best example of the class of objects with a fixed spatial layout, and the other horses faced to the right with large variations in action and hence in the relative positioning of the limbs. The last data set is taken as the best example of the articulated object class. We do not aim to reach for optimal performance but rather to understand the behavior of spatial relation classifiers.

A proper comparison requires careful consideration of the amount of a priori knowledge inserted in each method. Ideally, the a priori information should be equal for all different methods. We consider annotation as sending information to the learning algorithm. One mouse click in the annotation is one message to the learning algorithm. From that point of view, we take care that the amount clicks is (almost) identical for all methods. For the learning phases, object examples are extracted from images by one mouse click per object roughly in the center of its frame. In addition, there is a few mouse clicks on one image to initialize the part learning.

The paper is organized as follows. After the review the methods in Section 2, we describe the datasets used for evaluation in Section 3. The experimental results are presented in Section 4. Finally, we draw conclusions in Section 5.

2. Methods

In this section, we discuss part-based learning, Bayesian network learning and Bayesian network learning with object masking. In the first method, the object of interest is decomposed into parts. The learning phase consists of learning the part detectors and the spatial relationships among them. The recognition phase consists of the detection of object parts and searching for the spatial configuration optimal for a target object. In the second method, object appearance is learned by finding optimal pairwise relations among the pixels as well as modeling the dependences of the pixel values. The runtime process includes scanning the input image to find the best object location according to the learned statistical model. The last method is a variant of the second one where an object mask is employed to reduce the effect of background pixels.

2.1. Part-based learning

Let the object consist of p parts. The training object examples are labeled for the part positions. That is, for each training object example j, there are p labeled points in a two-dimensional plane $(x_{\alpha,j}, y_{\alpha,j}) \in R^2$, $\alpha = 1,\ldots,p$. For each object part α, a detector d_α is learned using the labeled examples extracted from the corresponding labeled points. We use the very simple template matching with normalized cross correlation (Pratt, 1991) as a part learner. Recall that the normalized cross correlation measure ρ between part α template t_α and a template u is

$$\rho(t_\alpha, u) = \frac{\sum_\ell (t_\alpha(\ell) - \bar{t}_\alpha)(u(\ell) - \bar{u})}{(\sum_\ell (t_\alpha(\ell) - \bar{t}_\alpha)^2 \sum_\ell (u(\ell) - \bar{u})^2)^{1/2}} \tag{1}$$

where ℓ is a two-dimensional index vector, $t_\alpha(\ell)$ and $u(\ell)$ denote pixel values of the templates at image location ℓ, and $\bar{t}_\alpha$ and $\bar{u}$ denote the average pixel values of the two patches. For each detector d_α we estimate its performance on the training set in terms of the detection rate γ_α and the false alarm rate β_α. The fact that the part detector is simple and could be substituted by better ones is not of prime relevance here as we study the virtue of spatial relations among parts as detected.

The parts of one object are spatially related as modeled by a Gaussian distribution after translation normalization.

From the labeled points in each object example j, we can constructed a vector $\mathbf{z}_j$ in a $(2p - 2)$ dimensional space by mapping $(x_{1,j},y_{1,j})$ to the origin

$$\mathbf{z}_j = (x_{2,j} - x_{1,j}, \ldots, x_{p,j} - x_{1,j}, y_{2,j} - y_{1,j}, \ldots, y_{p,j} - y_{1,j})' \quad (2)$$

The normalized configuration vector $\mathbf{z}$ is modeled by a Gaussian $\mathcal{N}(\mathbf{z}; \boldsymbol{\mu}, \Sigma)$. The estimation of the parameters of the Gaussian distribution is straightforward using the maximum likelihood estimator (Srivastava and Khatri, 1979).

The object localization problem is treated as a graph search problem with an objective function integrating the performance of part detectors and the object configuration. Let $\mathcal{O} = \{(h_\alpha, x_\alpha, y_\alpha) | \alpha = 1, \ldots, p\}$ be an object hypothesis obtained from an input image, where h_α is an indicator equal 0 if part α is not detected and 1 otherwise, and (x_α, y_α) is the spatial location of part α. Note that here only translation is considered. The score for $f(\mathcal{O})$ is

$$\begin{aligned} f(\mathcal{O}) = &\sum_{1 \leqslant \alpha \leqslant p} h_\alpha \log\left(\frac{\gamma_\alpha(1 - \beta_\alpha)}{(1 - \gamma_\alpha)\beta_\alpha}\right) \\ &+ \left[-\frac{1}{2}(\mathbf{z} - \boldsymbol{\mu})'\Sigma^{-1}(\mathbf{z} - \boldsymbol{\mu})\right] \end{aligned} \quad (3)$$

For the object localization task, we search for the solution of

$$\mathcal{O}^* = \arg\max_{\mathcal{O}} f(\mathcal{O}) \quad (4)$$

An A^* search algorithm is used to obtain the optimal solution of (4), exploiting two special properties of a Gaussian distribution to handle the missing part problem (see Pham and Smeulders (2004) for detail).

For the image classification task, we find the best object hypothesis $\mathcal{O}^*$ in the input image. Consequently, the value $f(\mathcal{O}^*)$ is used as a confidence value. If $f(\mathcal{O}^*)$ is greater than a threshold value, the input image is classified as containing an object instance. Otherwise, it is declared as background.

2.1.1. Part alignment in learning phase

Box 1: The part alignment algorithm.

```
Alignment algorithm

(1) Initialize $p_{\alpha,j}$ by extracting the image patch at $\ell_\alpha$
    in each training example $j$, $j = 1, \ldots, N$.
(2) Repeat $T$ times
    (a) update part $\alpha$ template $t_\alpha$ as the mean of $p_{\alpha,j}$.
            $t_\alpha = \frac{1}{N}\sum_{j=1}^{N} p_{\alpha,j}$        (5)
    (b) update part $\alpha$ in image $j$, $p_{\alpha,j}$
            $p_{\alpha,j} = \arg\min_{u \in \mathcal{N}(j,\ell_\alpha,m)} \rho(t_\alpha, u)$        (6)
        where $\mathcal{N}(j, \ell_\alpha, m)$ denotes the set of patches in
        the $m \times m$ neighborhood of $\ell_\alpha$ of image $j$.
```

One issue with the part-based approach is that it requires annotated training data of object parts. When the training data are collected using one mouse click as in this study, one has to perform automatic part annotation.

In order to label object parts in all training examples, we use the first example to create a map of relative part positions. Let ℓ_α denote the center position of part α. The image patch for part α in each training example j, denoted by $p_{\alpha,j}$, can fluctuate within a predefined neighborhood of size $m \times m$ of ℓ_α. Let $\mathcal{N}(j, \ell_\alpha, m)$ denote the set of all image patches in this neighborhood.

Box 1 presents an iterative algorithm for part alignment. In each loop, the algorithm updates the template t_α for each part α and subsequently finds in the neighborhood $\mathcal{N}(j, \ell_\alpha, m)$ in each training example j the closest image patch to t_α in terms of normalized cross correlation. The result serves as the updated annotation in each example j.

2.2. Bayesian network learning

The Bayesian network method (Pham et al., 2002) treats an image example as a feature vector $\boldsymbol{v} = [v_1, v_2, \ldots, v_{n_1 \times n_2}]'$ for a resolution of $n_1 \times n_2$. Furthermore, each vector $\boldsymbol{v}$ is considered as an instantiation of a random variable $\boldsymbol{V} = [V_1, V_2, \ldots, V_{n_1 \times n_2}]'$. The joint distribution for each class $P_c(\boldsymbol{v})$, where $c \in \{-1, +1\}$ with -1 denoting the background class and $+1$ the object class, is estimated using a forest structured Bayesian network

$$P_c(\boldsymbol{v}) = \prod_{i=1}^{n_1 \times n_2} P_c(V_i = v_i | \Pi_i = \pi_i) \quad (7)$$

where Π_i denote the parent of V_i in the network structure. The lower case notations v_i and π_i denote specific values of the corresponding random variables in upper case. Each pair (V_i, Π_i) forms a directed edge of the network. This model clearly ignores much of the dependency among nearby pixels. Nevertheless, it is necessary for computational reason.

The special characteristic of the method is that the network structure is learned from training data rather than fixed a priori. This is achieved by maximizing the Kullback–Leibler divergence between the two distributions $P_c(\boldsymbol{v})$. This optimization problem is equivalent to the problem of finding a maximum branching in a weighted graph (Pham et al., 2002) where the weight $W_{(V_i,\Pi_i)}$ for each directed edge (V_i, Π_i) is

$$W_{(V_i,\Pi_i)} = \sum_{\pi_i} \sum_{v_i} P_{+1}(v_i, \pi_i) \log \frac{P_{+1}(v_i|\pi_i)}{P_{-1}(v_i|\pi_i)} \quad (8)$$

The maximum branching problem is a graph problem that can be solved in polynomial time (Tarjan, 1977). Thus, learning the structure is efficient. The weight $W_{(\Pi_i,V_i)}$ indicates the discriminatory power of the relation. In other words, a larger weight contributes more to the classification of object versus background.

To classify an input pattern $\boldsymbol{v}$, the log likelihood is computed

$$\log\frac{P_{-1}(\boldsymbol{v})}{P_{+1}(\boldsymbol{v})}=\sum_{i=1}^{n_1\times n_2}\log\frac{P_{-1}(v_i|\pi_i)}{P_{+1}(v_i|\pi_i)} \tag{9}$$

This step is involved with $n_1\times n_2$ additions, which is equal to the number of input pixels. The values $\log\{P_{-1}(v_i|\pi_i)/P_{+1}(v_i|\pi_i)\}$ are stored in memory for efficient computation.

For the object localization problem, we search for the best candidate

$$\boldsymbol{v}^*=\arg\min_{\boldsymbol{v}}\log\frac{P_{-1}(\boldsymbol{v})}{P_{+1}(\boldsymbol{v})} \tag{10}$$

In this case, the Bayesian network classifier is treated as an object filter operating over the image.

Again, for the image classification task we find the optimal object hypothesis $\boldsymbol{v}^*$ for the input image. Subsequently, a threshold value is employed on the likelihood value of the optimal solution $\log\{P_{-1}(\boldsymbol{v}^*)/P_{+1}(\boldsymbol{v}^*)\}$ to determine whether or not the input image contains an instance of the target object.

2.3. Bayesian network learning with object masking

In the Bayesian network method we make no distinction between object and background. In effect, pixels and their intensities in the background area will be uncorrelated with the foreground area and hence deteriorate the result by the random noise they produce.

Instead of using a full image patch as a feature vector, an object mask is employed to better specify the object example. This extension is rather simple since we only need to rearrange the pixels under the mask into a feature vector. Specifically, the mask can be seen as a projection matrix M as follows. Let $\boldsymbol{v}$ be an $n_1\times n_2$ vector as in the previous section. The matrix M is binary having r rows and $n_1\times n_2$ columns where r is the number of elements to be selected. Each row in M has one non-zero element indicating the corresponding selected component of $\boldsymbol{v}$. The new feature vector $\boldsymbol{v}_M$ is

$$\boldsymbol{v}_M=M\boldsymbol{v} \tag{11}$$

The process of learning and classification is left unchanged from the standard Bayesian network method, except $\boldsymbol{v}_M$ is used in place of $\boldsymbol{v}$.

The mask M is generated once manually for the whole dataset such that the interior of the object silhouette is sufficiently covered, masking out the remaining pixels most of which will be background pixels.

As the pixels are features input to the classifiers, masking can be seen as feature selection. In this respect, Bayesian network with object masking is closer to part-based learning in terms of a priori knowledge than standard Bayesian network learning. This is because part selection in part-based learning is also a step to include interesting features for recognition.

3. Datasets and annotation

We use two datasets in the experiments: human faces and horses. The former is from (Weber et al., 2000), while the latter is collected from the Internet and in part from the Corel dataset. Note that the evaluation of object localization performance requires images containing an object instance. Whereas for the evaluation of image classification performance, apart from a set of object images, another set of background images containing no object instance is required.

3.1. The face dataset

This dataset contains 438 face images of 28 people (after removing nine images that are clearly different in scale from the rest of the dataset). The original images are converted to grayscale and resized by half.

The face training set consists of 216 images of 14 people in the set of face images. We manually annotate the nose of each human face in the training set. This step requires one mouse click in each image. The face examples are extracted from a frame of size 128 × 128 centered at the labeled points. See Fig. 1(a) for examples.

We also extract 10,000 background patches of the same size by sampling uniformly over the training images. See Fig. 1(b) for examples.

The face test set consists of the remaining 222 images of 14 people. Fig. 1(c) shows six examples of this set.

For the image classification task, we use an additional set of background images also provided by the authors of (Weber et al., 2000). The images are assorted scenes around the Caltech campus and in their Vision laboratory. There are 451 images in total. Fig. 1(d) show five background images of this test set.

3.2. The horse dataset

The second dataset consists of 472 horse images. All the horses are faced to the right by flipping the images left right when needed. The training set consists of 269 images. We annotate the center of the horse in each image of the training set. Again, this step requires only one click in each image. Subsequently, training horse examples of size 160 × 128 are extracted. See Fig. 2(a) for examples.

In addition, we extract 10,000 background patches of the same size by sampling uniformly over the training images. See Fig. 2(b) for examples.

The horse test set consists of the remaining 203 images. Fig. 2(c) shows examples of this set.

For the image classification task we use 865 background scenery images. Fig. 2(d) shows three background images in this test set.

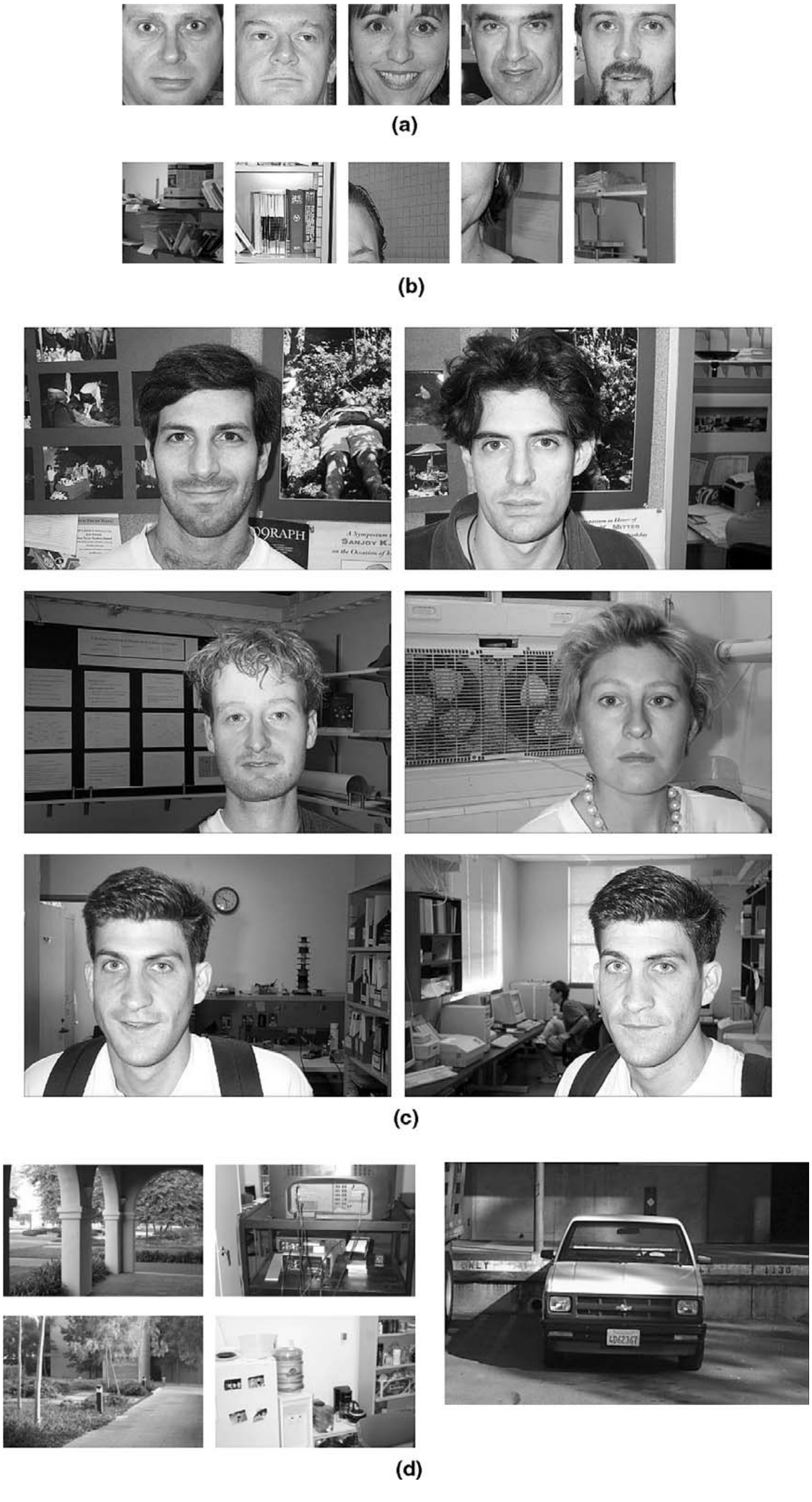


Fig. 1. Examples of the face dataset. (a) Training faces; (b) training background patches; (c) test face images; (d) test background images.

It can be seen that the recognition of horses on this dataset is hard. The images are taken outdoor with different cameras, purposes and under various lighting conditions. Although the horses all face to the right, there is still large variation in pose, background, skin, rider and pace. As a consequence, the horse recognition seems hopeless for the Bayesian network methods because the structure of the horse is non-fixed. It also seems hopeless for the part-based method because we have a very simple part learner while the dataset contains considerable variation in the object parts. However, our main interest is not in high recognition results but in understanding the role of spatial relations for object classification.

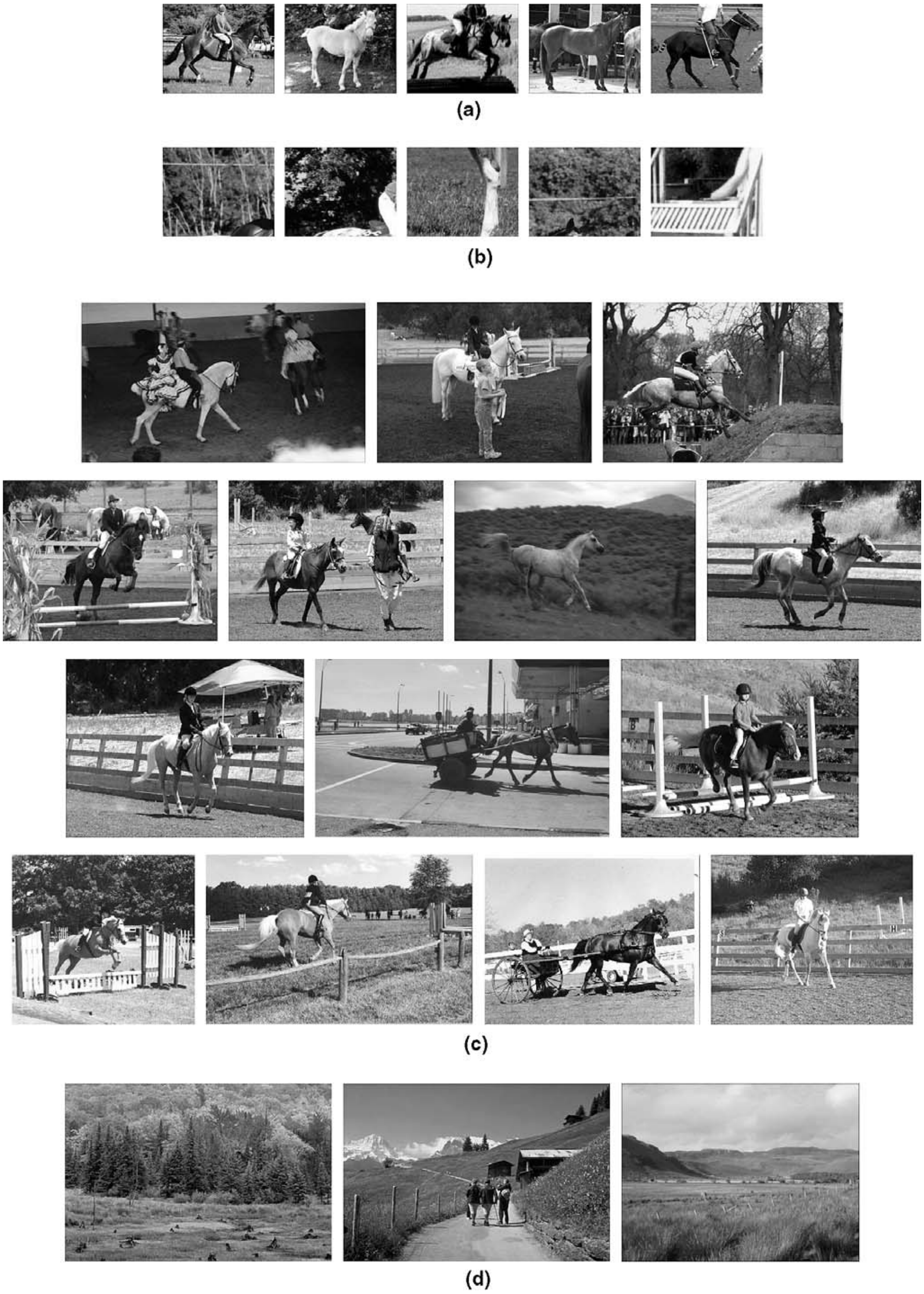


Fig. 2. Examples of the horse dataset. (a) Training horses; (b) training background patches; (c) test horse images; (d) test background images.

4. Experimental results

4.1. Object localization

4.1.1. Performance of part-based learning

For the face dataset, we learn eight object parts including the two eyes, the two cheeks, the two mouth corners, the nose, and the forehead. For the horse dataset, we learn seven parts including the hip, neck (lower part and upper part), front leg (lower part and upper part) and back leg (lower part and upper part). The size of the part templates is 32×32 in both cases. Each part α is allowed to move within 15 pixels in each direction from its center position ℓ_α for the face dataset and 20 pixels for the horse dataset. Finally, we let the annotation algorithm run for 20 iterations ($T = 20$). The parameters are derived based on the computational time of the part-learning and the A^* search algorithm in Section 2.1.

Fig. 3 illustrates the state of the parts initially and as learned after 1, 5 and 20 iterations. It can be seen that

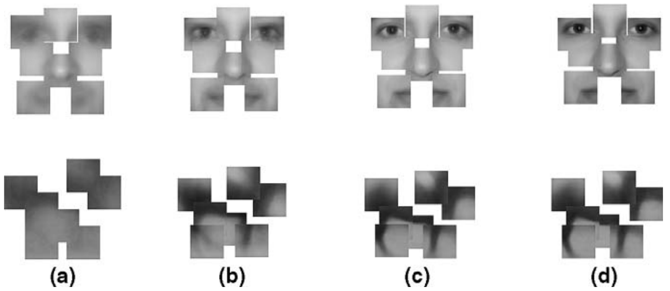
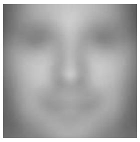
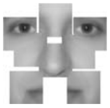
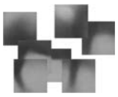


Fig. 3. The state of the part alignment algorithm (a) the initial state, (b) after one iterations, (c) after five iterations and (d) after 20 iterations.

Table 1
Accuracy of object localization

Template matching	(%)	Part-based	(%)
	88.5		96.5
	45.0		63.0

the learned parts after 20 iterations are visually much clearer than their initial states for both datasets. Fig. 4 shows the average shift of each part alignment step. The iterative algorithm achieves stable result after about five loops. The complete process takes less than 30 min on a 1 GHz CPU with a Matlab implementation. Note that one may use the average shift as a criterion to terminate the alignment procedure instead of fixing the number of iterations. However, one still needs to check the convergence condition of the average shift.

Finally, the threshold value for each detector is set such that the error rates are equal for detection and false alarm. That is, $1 - \gamma_\alpha = \beta_\alpha$ (see Eq. (3)). This is known as the equal error rate criterion.

We also performed experiments where a global object template is used for object localization. We used the same measure for matching as in our part detectors (Eq. (1)). Our purpose is to examine the performance gained by the additional complexity of the part-based method.

Table 1 presents the results of the part-based method together with the results of global template matching on the two datasets. Significant improvement in accuracy is achieved, 8% on the face dataset and 18% on the horse dataset, due to the tolerance to the variation in the object structures.

We can observe that the result on the horse dataset is low. This can be explained by the poor quality of the part detectors on this dataset, namely template matching with normalized cross correlation.

We carried out an experiment to compare the results between manual annotation versus maximal correlation annotation for the two eyes, nose and two mouth corners composition. The experimental result shows that the alignment algorithm performs well with 92% correct localization in comparison to the result of 91% for the manually annotated data (Pham and Smeulders, 2004). In this particular case, the algorithm reduces 80% of the amount of tedious annotation work and achieves an equivalent result.

Fig. 5 shows examples of the detection result of the part-based method on the two datasets. Note that the part-based method also gives the estimated position of the missing parts. The incorrect localization examples (the second picture in the second row in Fig. 5(a) and (b) are due to severe missing parts.

4.1.2. Performance of Bayesian network learning

The Bayesian networks are learned with a resolution of 32×32 for the face dataset and 40×32 for the horse dataset. Fig. 6 shows the relations and the weights of the incoming edge in the two networks learned over the two datasets. Light pixels indicate high values for the weights, hence incoming relationships important for recognition.

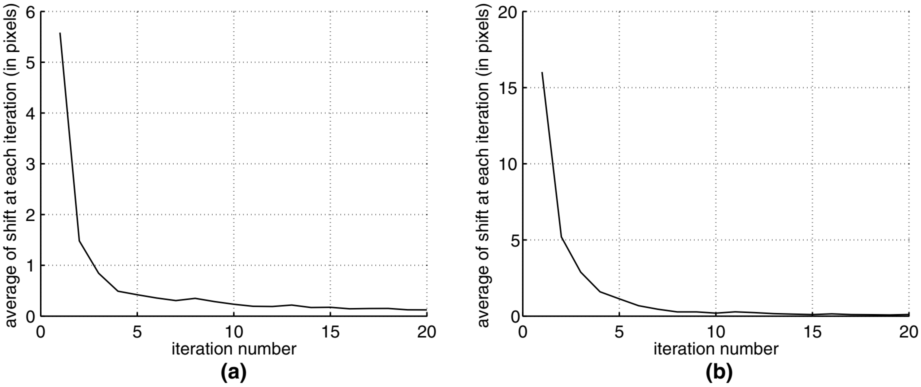


Fig. 4. Average shift at each iteration, (a) on the face dataset and (b) on the horse dataset.

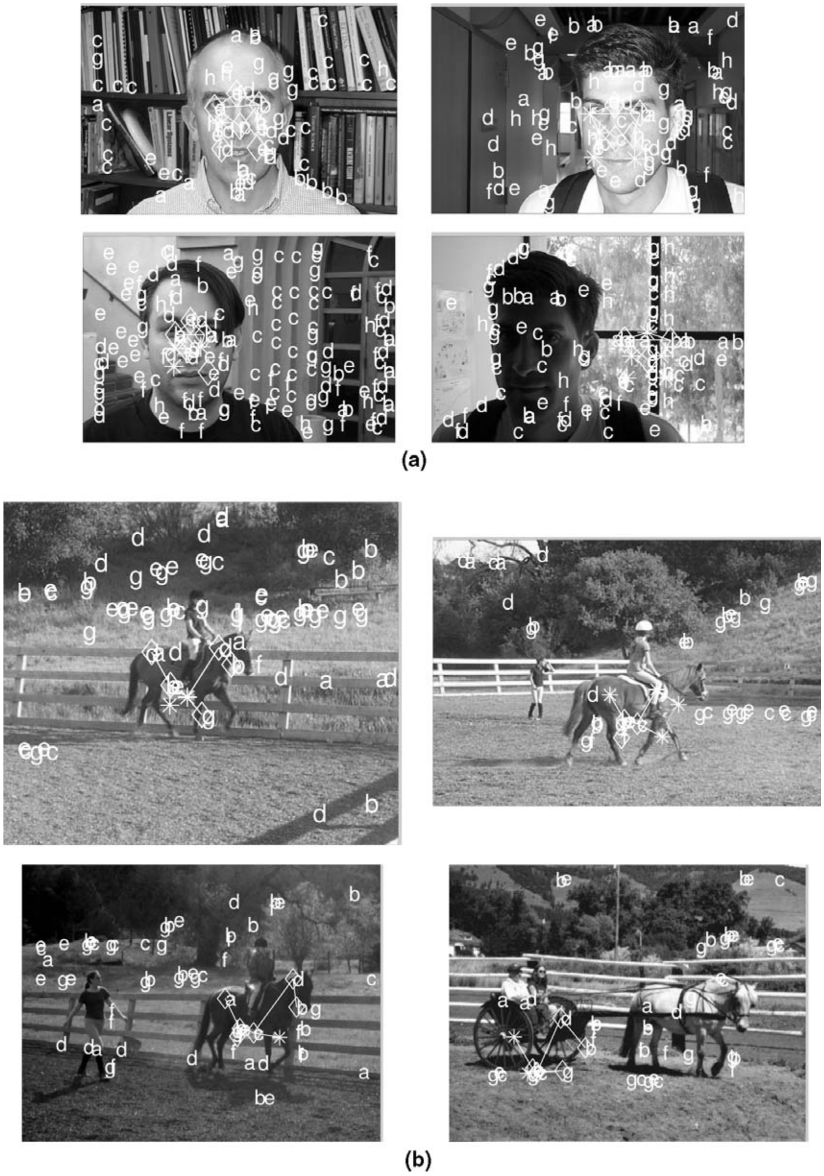


Fig. 5. Example of part-based object localization on the two test sets, (a) results on the face test set, see Fig. 1(c), and (b) results on the horse test set, see Fig. 2(c). The letters denote the detected locations of object parts. The diamonds represent detected parts of the solution while the *'s represent parts not found. (a) Faces; (b) horses.

For the face dataset, it can be seen that Bayesian network learning indicates that edges to the forehead and the cheeks have the most weights. That these edges depart from the eye and cheek regions (data not shown) indicates that the most important discriminating relations are found by long-distance co-occurrences of forehead and eyes and cheeks. Similarly, for the horse dataset the Bayesian network method is able to learn long-distance dependences among pixel values. In particular, the relations between the head region and the body region and between the body region and the front leg region are clearly shown. From Fig. 6(b) and (d) it can be learned that there is little or no contribution from the context of the object. Note that unlike a complete independence model of naive Bayes, the dependency between two variables is allowed. When long-distance dependencies are more discriminative than nearby alternatives, they will be chosen. Hence the learned relations are completely data driven.

Table 2 gives the localization result of the Bayesian network method on the two datasets. The result on the face dataset is worse than that of the part-based method with the eight-part composition, which almost completely covers the face. The result on the horse dataset is superior to that of the part-based method. This is because Bayesian network learning can focus on the stable regions and models the dependences among them, including long-distance ones.

4.1.3. Performance of Bayesian network learning with object masking

The Bayesian networks for the two object classes are depicted in Fig. 7. For the face dataset, about half of the

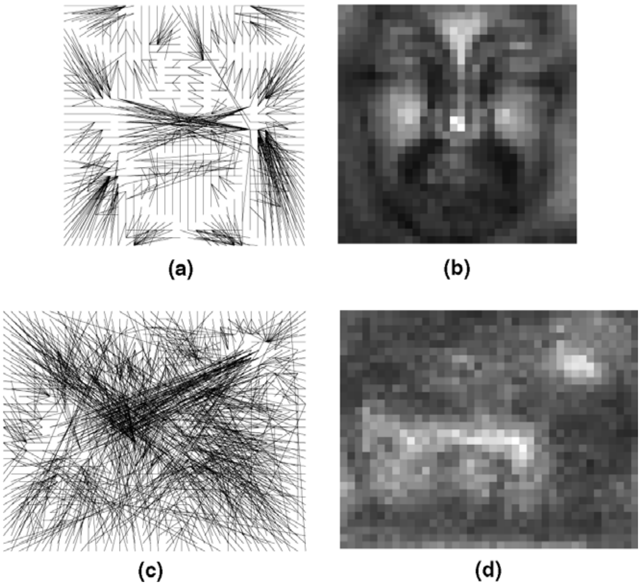


Fig. 6. Bayesian network relations and their strengths (see Eq. (8)) as learned over the training sets, (a,b) faces and (c,d) horses.

Table 2
Performance of Bayesian network learning for the object localization task on the test sets

Dataset	Localization accuracy (%)
Faces	92.5
Horses	83.0

See Figs. 1(c) and 2(c) for examples of these test sets.

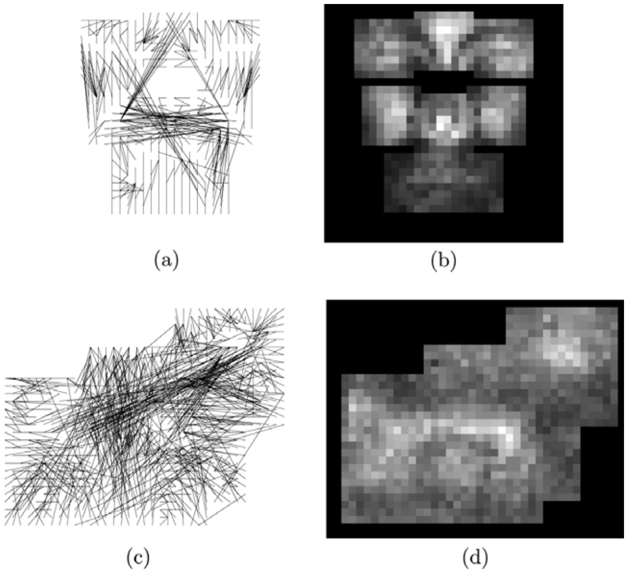


Fig. 7. Bayesian network relations and their strengths (see Eq. (8)) as learned over the training sets, (a,b) faces and (c,d) horses. Object masking is employed for both datasets.

pixels are masked out. For the horse dataset, approximately one-third of the pixels are masked out.

Table 3
Performance of Bayesian network with object masking for the object localization task on the test sets

Dataset	Localization accuracy (%)
Faces	97.0
Horses	83.0

See Figs. 1(c) and 2(c) for examples of these test sets.

Table 3 shows the result of Bayesian network with object masking on the two datasets. The removal of pixels by masking is effectively a feature selection procedure as the pixel relations and pixel values corresponding to the background are no longer in the classifier equation. For those images where the background acts as noise to the object recognition, it is done to improve the recognition rate. The improvement is clearly shown for the face dataset where the correct localization rate increases from 92.5% to 97%. This effect, however, is absent in the horse dataset, where the floor of grass and sand will be highly correlated with horse images.

Note that the computational gain is significant in both cases. As can be seen from Eq. (9), the number of operations required for each classification is equal to the number of features. Thus, the classification cost is reduced by half and one-third for the face dataset and the horse dataset, respectively.

4.2. Image classification

In the next set of experiments, we evaluate the methods on an image classification task. The aim is to determine whether or not a test image contains an instance of the target object.

Fig. 8(a) and (b) shows the ROC curves for the three methods on the face dataset and horse dataset, respectively. One can observe that for the face dataset, the Bayesian network method with object masking performs best. The Bayesian network methods perform significantly better on the horse dataset. The poor performance of the part-based method on this dataset is because the variation in appearance of object parts is not sufficiently captured by the simple part learner.

For the face dataset, the equal error rates of part-based learning, Bayesian network and Bayesian network with object masking are 3.8%, 4.7% and 2.2%, respectively. The results are better than the error rate of 6.0% of Weber et al. (2000). The result of the part-based method is comparable to the error rate of 3.6% of Fergus et al. (2003). For state of the art results on this face dataset, the reader is referred to Gao and Vasconcelos (2005), Deselaers et al. (2005) and Fussenegger et al. (2004).

In the next experiment, we simulate object occlusion by covering a quarter of the object in back. Fig. 9(a) and (b) give examples of synthetic occlusion for each dataset.

The ROC curves for image classification under occlusion are shown in Fig. 10, together with those curves

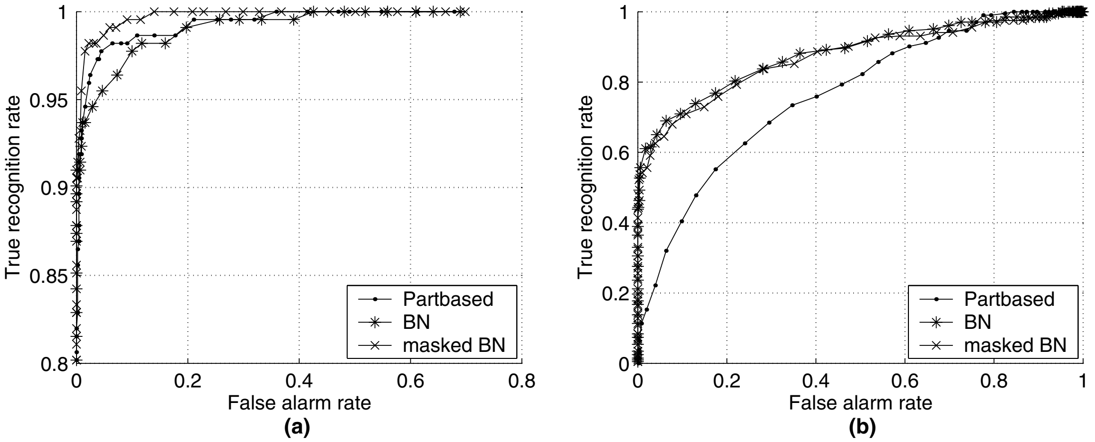


Fig. 8. ROC curves for image classification. BN indicates the Bayesian network method and masked BN indicates Bayesian network with object masking. (a) On the face test set, see Fig. 1(c). Note that the figure is magnified around the region of equal error rates. (b) On the horse test set, see Fig. 2(c).

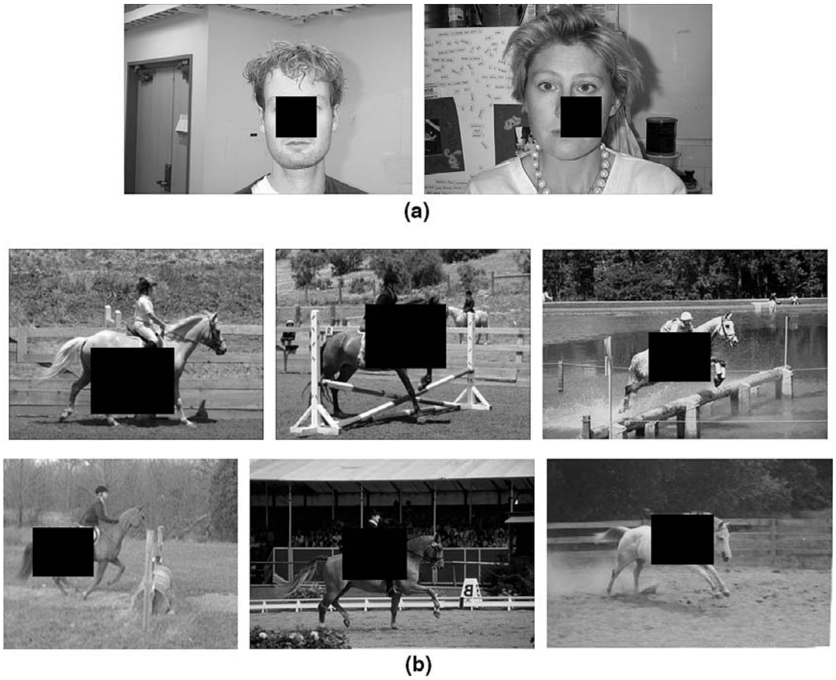


Fig. 9. Examples of the test set with randomly assigned occlusions, (a) occluded faces and (b) occluded horses.

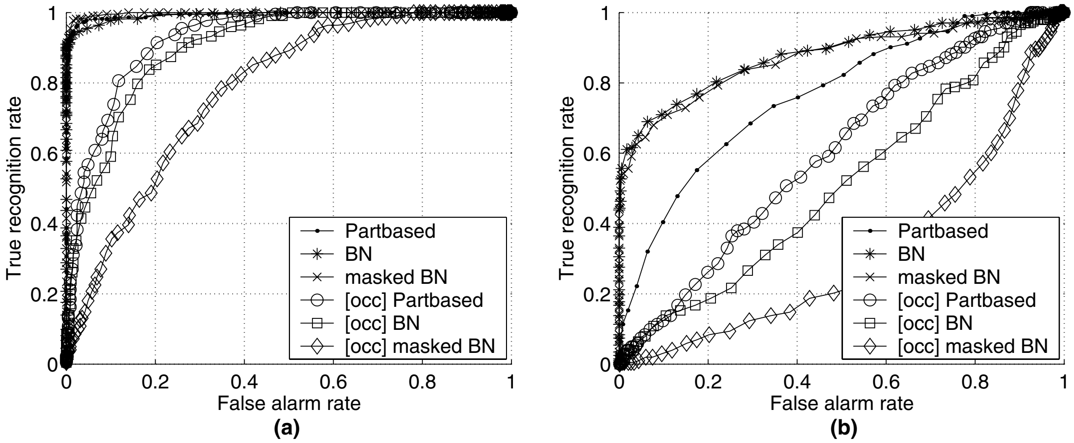


Fig. 10. ROC curves for image classification, [occ] indicates the result on the datasets with occlusion. (a) On the face test set, (b) on the horse test set.

obtained on the original datasets. The performance of all three methods drops substantially.

The part-based method appears least sensitive to the effect of occlusion. This is because the method is designed to search for the best subset of part in the desired configuration. One can also observe that the Bayesian network method with object masking is more sensitive to occlusion than without object masking. This is another sign of the feature reduction removing the redundancy by masking the background. In short, this suggests that feature selection in general does not help with the problem of occlusion.

5. Discussion and conclusion

In this paper, we evaluate two statistical methods to evaluate the use of spatial relationships in object recognition.

We emphasize that all methods are presented with the same set training examples with (almost) identical annotation, which is one click on the center of an object in the learning set. In this way, for the purpose of fair comparison, we equate the amount of annotation information available to the methods. Due to this constraint, part alignment is indispensable. As a by-product, we conclude that the alignment works well. On the face dataset automatic alignment and manual part labeling generate the same localization results.

For the Bayesian network methods, one can observe that the long-distance dependencies within the face of the object are most informative. This is due to the spatial coherence in images implying that nearby pixels are strongly correlated almost everywhere; typically only broken at the edges between foreground objects and the background. Hence short distance relations are commonly correlated and less informative. Strongly correlated long-distance relations are rare and typical for pixel pairs on one face of an object. This observation holds true for the faces as expected for an example of the fixed layout class as well as for the horses.

Object masking is designed in analogy to a feature selection procedure for the Bayesian network. It improves the recognition result significantly on the face dataset while maintaining the result on the horse dataset. Bayesian network with object masking outperforms the part-based method on both datasets. That is, even in the case of the flexible articulated object class, here represented by the horse dataset, the Bayesian network performed superior. The Bayesian networks are capable of capturing the stable relations in the object appearance.

It should be noted, however, that for the part-based method, the issue of part learners is critical for effective recognition. We do not use the most advanced part learners. One method to be investigated further is the use of a local Bayesian networks for that purpose. Alternatively, one may consider local descriptors invariant to scale and orientation as part detectors.

None of the methods copes well with severe random patch occlusion. The part-based method is less sensitive to occlusion than the Bayesian network methods. This is ascribed to the tolerance to missing parts of the part-based method. Between the two Bayesian network methods, object masking is more sensitive to occlusion than without object masking as expected since the masking brings the information back to the minimum set of pixels. This result means that feature selection by Bayesian masking is unlikely to solve the problem of occlusion. In applications where occlusion is expected, it should be dealt with explicitly.

Further research is conducted into the issue of automatic feature selection. This is important for generic object recognition where a priori knowledge is limited and the range of object classes is huge. One approach might be to reject regions with low weight in a learned Bayesian network in an iterative fashion. The experiments show that the background regions carry little weight in classification.

In conclusion, we have presented a study for three methods that exploit spatial relations for object recognition. The relations are learned from training examples. By imposing (almost) the same amount of annotation of training data, which can be seen as sending knowledge to the learning algorithm, we can identify the strengths and weaknesses of each method. The part-based method and Bayesian networks are capable of exploiting the spatial relations for recognition. To be sure, the part-based method outperforms global template matching on both datasets. (This is expected as the part-based method is designed to be tolerant to some variation in the object structure, making the method less brittle than global matching.) And, for the Bayesian network method, a previous study (Pham et al., 2002) has already shown its superior performance in comparison with the naive Bayes classifier, which encodes no spatial relations at all. This leads us to conclude that both part-based and especially pair-wise relations in Bayesian networks are superior to strictly spatial matching by templates and strictly non-spatial classifiers.

Acknowledgement

This research is supported by MultimediaN.

References

Bunke, H., 2000. Graph matching for visual object recognition. Spatial Vision 13, 335–340.

Burl, M., Perona, P. 1996. Recognition of planar object classes. In: Proc. CVPR'96, pp. 223–230.

Colmenarez, A., Huang, T., 1997. Face detection with information-based maximum discrimination. In: Proc. CVPR'97, pp. 782–787.

Cortes, C., Vapnik, V., 1995. Support vector networks. Machine Learn. 20, 273–297.

Deselaers, T., Keysers, D., Ney, H. 2005. Discriminative training for object recognition using image patches. In: Proc. CVPR'05, vol. II, pp. 157–162.

Egenhofer, M., 1997. Query processing in spatial-query-by-sketch. J. Visual Lang. Comput. 8 (4), 403–424.

Felzenszwalb, P., Huttenlocher, D. 2000. Efficient matching of pictorial structures. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 66–75.

Fergus, R., Perona, P., Zisserman, A., 2003. Object class recognition by unsupervised scale-invariant learning. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 264–271.

Fischler, M.A., Elschlager, R.A., 1973. The representation and matching of pictorial structure. IEEE Trans. Comput. 22 (1), 67–92.

Fussenegger, M., Opelt, A., Pinz, A., Auer, P. 2004. Object recognition using segmentation for feature detection. In: Proc. ICPR'04, vol. III, pp. 41–48.

Gao, D., Vasconcelos, N., 2005. Discriminant saliency for visual recognition from cluttered scenes. In: Advances in Neural Information Processing Systems, vol. 17, pp. 481–488.

Jordan, M.I. (Ed.), 1999. Learn. Graph. Models. MIT Press.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324.

Pearl, J., 1988. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufman.

Pham, T.V., Smeulders, A.W.M., 2004. Statistical strategy for object class recognition using part detectors. In: Leonardis, A., Bischof, H. (Eds.), Proc. Statistical Learning in Computer Vision, pp. 33–48.

Pham, T.V., Worring, M., Smeulders, A.W.M., 2002. Face detection by aggregated Bayesian network classifiers. Pattern Recognition Lett. 23 (4), 451–461.

Pratt, W.K., 1991. Digital Image Processing, second ed. Wiley.

Rocha, J., Pavlidis, T., 1994. A shape analysis model with applications to a character recognition system. IEEE Trans. Pattern Anal. Machine Intell. 16 (4), 393–404.

Schneiderman, H., Kanade, K. 2000. A statistical method for 3D object detection applied to faces and cars. In: Proc. CVPR 2000, pp. 746–751.

Sebastian, T.B., Klein, P.N., Kimia, B.B., 2004. Recognition of shapes by editing their shock graphs. IEEE Trans. Pattern Anal. Machine Intell. 26 (5), 550–571.

Siddiqi, K., Shokoufandeh, A., Dickinson, S., Zucker, S., 1999. Shock graphs and shape matching. Internat. J. Comput. Vision 30, 1–24.

Srivastava, M.S., Khatri, C.G., 1979. An Introduction to Multivariate Statistics. North Holland.

Sung, K.K., Poggio, T., 1998. Example-based learning for view-based human face detection. IEEE PAMI 20 (1), 39–51.

Tagare, H., Vos, F., Jaffe, C.C., Duncan, J.S., 1995. Arrangement: A spatial relation between parts for evaluating similarity of tomographic section. IEEE Trans. Pattern Anal. Machine Intell. 17 (9), 880–893.

Tarjan, R., 1977. Finding optimum branchings. Networks 7, 25–35.

Weber, M., Welling, M., Perona, P., 2000. Unsupervised learning of models for recognition. In: Proc. ECCV, pp. 18–32.