A system for object class detection

Daniela Hall

INRIA Rhône-Alpes, 655, ave de l'Europe,
38320 St. Ismier, France
Daniela.Hall@inrialpes.fr

Abstract. A successful detection and classification system must have two properties: it should be general enough to compensate for intra-class variability and it should be specific enough to reject false positives. We describe a method to learn class-specific feature detectors that are robust to intra-class variability. These feature detectors enable a representation that can be used to drive a subsequent process for verification. Instances of object classes are detected by a module that verifies the spatial relations of the detected features. We extend the verification algorithm in order to make it invariant to changes in scale. Because the method employs scale invariant feature detectors, objects can be detected and classified independently of the scale of observation. Our method has low computational complexity and can easily be trained for robust detection and classification of different object classes.

1 Introduction

Object detection is fundamental to vision. For most real world applications, object detection must be fast and robust to variations in imaging conditions such as changes in illumination, scale and view-point. It is also generally desirable that a detection system be easily trained, and be usable with a large variety of object classes. In this paper we show how to learn and use class specific features to detect objects under variations in scale and intra-class variability.

Our approach is similar to the work of Agarwal [1] who proposes a detection algorithm based on a sparse object representation. While her system is robust to occlusions, it can not deal with scale changes. She demonstrates her system on side views of cars. We extend Agarwal's idea to a larger set of object classes. We automatically construct class specific feature detectors that are robust to intra-class variability by learning the variations from a large data set and propose a representation for geometry verification with low computational complexity.

Fergus [4] has described a method to classify objects based on a probabilistic classifier that takes into account appearance, shape and scale of a small number of detected parts. His approach is robust to changes in scale, but is limited in the number of candidate parts that can be considered (≈ 30 maximum). The approaches of both Fergus and Agarwal depend on reliable interest point detectors with a small false positive rate. Our approach is independent of such interest point detectors, and not affected by a large number of detections. Furthermore,

our feature detectors are scale invariant, and thus provide object class detection under scale changes.

The article is organised as follows. Section 2 discusses the design of a detection and classification system. The components of this design are described in Section 3 and 4. The performance of the proposed system is demonstrated in the experimental Section 5.

2 Architecture of a detection and classification system

A successful detection and classification system must have two properties: it must be general enough to correctly assign instances of the same class despite large intra-class variability and it must be specific enough to reject instances that are not part of the class. Features robust to intra-class variability can be constructed by learning from examples. The result is a feature or part detector that can generalise from a small number of examples to new examples. Such a detector can provide a hypothesis about the presence of a class instance, but it is in general not specific enough for reliable detection and classification.

The relative position of distinct object features is important for classification and needs to be modeled as for example in the approaches of Fergus and Agarwal. In these approaches, the verification is computationally expensive, because the relations of all candidate parts need to be verified. A geometry verification module can provide the required specificity of the system. The flexibility of feature extraction and the specificity of spatial relations can be implemented in an elegant way by an architecture with two components (see Figure 1): a feature extraction module that provides features invariant to intra-class variability and a geometry verification module that introduces specificity, increases the reliability of the detection and rejects false positives.

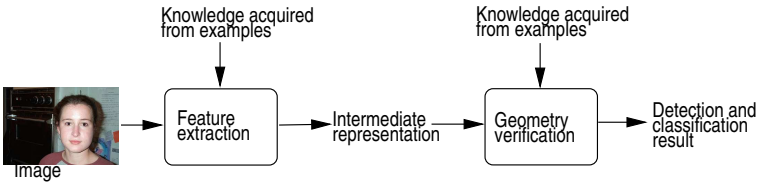

Fig. 1. System architecture consisting of low level feature extraction and higher level geometry verification.

3 Low level feature extraction

Our low level measurements are local image features with limited spatial extent. Local features are commonly described by neighborhood operators [7] that function as convolution masks and measure the responses to classes of local image

patterns. A set of neigborhood operators provides a feature vector that measures several aspects (appearance) of a local neighborhood. Many different families of local neighborhood operators can be used. For example grey-scale invariants [17], gabor filters [19], or Gaussian derivatives [5,11,15].

3.1 Appearance description by Gaussian derivatives

Orthogonal families of neighborhood operators describe a local neighborhood with a minimum number of independent local features. Among the different neighborhood operators, several properties make the Gaussian derivative family an ideal candidate for appearance description of local neighborhoods. The family is orthonormal and complete. Scale is controlled by an explicit parameter. The low order Gaussian derivatives measure the basic geometry of a local neighborhood. Similarity of neighborhoods can be measured by defining a distance metric in feature space. The low dimensions of the feature space enables fast algorithms and avoids computational problems due to the curse of dimensionality.

Lindeberg [10] has proposed an algorithm to determine the intrinsic scale of local image features. Normalising features to the intrinsic feature scale enables a scale invariant description of local appearance. The intrinsic scale of a feature is characterised by a maximum in scale and space. Such a maximum can be found by sampling the response of a normalised Laplacian at different scales. Gaussian derivatives are applied successfully to various computer vision problems. The fast implementation of derivatives [18] and the algorithm for scale normalisation makes the Gaussian derivative local jet an ideal candidate for the scale invariant description of the appearance of local image neighborhoods.

In this article we focus on the detection of instances of object classes. The color information of images of the same class has a high variance. The variance of the texture in the luminance channel is less pronounced. The luminance channel is less affected by changes in illumination conditions. In our experiments, we use first and second derivatives computed from luminance and normalised to the intrinsic scale. The raw features are therefore points in a five dimensional scale invariant Gaussian derivative feature space. We do not normalise for orientation, because the absolute orientation of features is discriminant for particular features. If orientation invariance is required, a rotation invariant feature space can be used such as the one proposed by Schmid [16].

3.2 Features appropriate for classification

The raw Gaussian derivative features are appropriate for retrieval of corresponding matching candidates according to the distance in feature space that measures their similarity. This matching principle produces very good results in identification, image retrieval or other applications where the exact entity of the local neighborhood is searched, because in such cases the appearance variance between model and observed neighborhood is small. An image class is characterised by the co-occurrence of typical parts in a particular spatial arrangement. The typical parts can also have a large variance in their spatial relation. Using raw Gaussian

derivatives directly for detection and classification is going to fail because the intra-class variability makes matching unreliable.

Features that can compensate for intra-class variability can be found by extracting the common parts of images of a visual class and learning the variation in appearance. Fergus [4] learns a probabilistic classifier from a large number of examples. Classification is obtained by evaluating a maximum likelihood ratio on different combination hypotheses of potential parts that are indicated by a salient region detector. This detector is essentially equivalent to a scale invariant interest point detector, such as the Harris Laplacian proposed by Mikolajczyk [13], that is also applied by Schmid and Dorko in [3,16]. The approach depends on the detection of salient regions. No false negatives are tolerated and at the same time the number of potential candidates should be small, because the computational complexity is exponential.

Much effort is done to make interest point detectors stable and accurate. However, interest point detectors respond to image neighborhoods of particular appearance (corner features or salient features). This limits the approach to objects that can be modeled by this particular kind of features. Uniform objects can be missed because the interest point detector does not detect any points. In the following section we propose a method to compute class specific feature detectors that are robust to the feature variance of images of the same class and that are independent from general interest point detectors.

3.3 Computation of class-specific feature detectors

For the extraction of class-specific features, we learn the appearance of class-specific object parts from a dense, pixelwise, grid of features by clustering. Clustering of dense features is similar to Leung and Malik's approach for computation of generic features for texture classification [12]. The feature extraction is fast due to the recursive implementation of Gaussian derivatives [18]. Furthermore, the clustering produces statistically correct results, because a large amount of data points is used.

We use k-means clustering to associate close points in feature space. K-means is an iterative algorithm that is initialised with points drawn at random from the data. In each iteration, the points are associated to the closest cluster centers which are updated at the end of each cycle. An overall error is computed which converges to a minimum. The risk of returning a sub-optimal solution is reduced by running k-means several times and keeping the best solution in terms of overall error.

We assume that the data in feature space can be represented by multi-dimensional Gaussians. Non-elliptical clusters are represented by a mixture of Gaussians. Cluster C_j is caracterised by its mean $\boldsymbol{\mu}_j$ (the cluster center) and covariance Σ_j (the shape of the cluster). This allows to compute the probability that a measurement belongs to cluster C_j as:

$$p_j(\boldsymbol{m}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp(-\frac{1}{2}(\boldsymbol{m}-\boldsymbol{\mu}_j)^T \Sigma_j^{-1}(\boldsymbol{m}-\boldsymbol{\mu}_j)) \tag{1}$$

This gives rise to k probability maps where the image position (x, y) of probability map j is marked by $p_j(\boldsymbol{m_{xy}})$ of the extracted Gaussian derivative feature $\boldsymbol{m_{xy}}$. The probability maps can be condensed to a single cluster map M with k colors where the label at position (x, y) is computed as:

$$M(x, y) = \arg \max_{j=1,..,k} p_j(\boldsymbol{m_{xy}}) \tag{2}$$

Figure 2 illustrates the feature extraction process. The top right graph shows an example of the probability maps (low probabilities are black, high probabilities are light grey). We observe maps which mark uniform regions, bar like regions or more complex regions such as the eyes. The corresponding clusters are the class-specific feature detectors. Many neighboring pixels are assigned to the same cluster and form connected regions. This is natural, because the local neighborhood of close pixels have a strong overlap, with a high probability that the image neighborhoods are assigned to the same cluster.

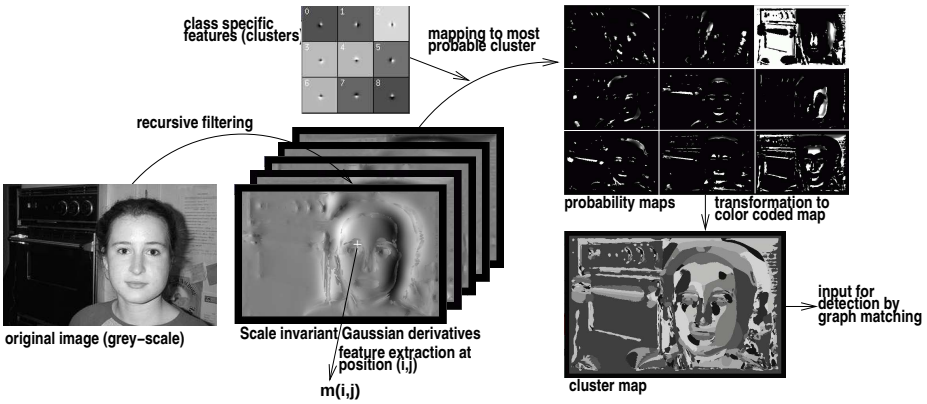

Fig. 2. Algorithm for raw feature extraction and mapping to most probable class specific features. The probability maps are condensed to a single color coded cluster map, where color k marks points that are assigned to cluster k.

The cluster map representation is an enormous data reduction, but at the same time, it preserves the information about class specific features and their location. This is the minimum information required for detection and classification. The cluster map representation is specific enough to provide detection and it is general enough to enable classification of images with a large intra-class variability. Evidence provide the experiments.

Another important point is that this cluster map representation is scale invariant. The scale invariance property of the raw features translates to the cluster prototypes and also to the mapping. In Figure 3 we show the original image at

numerically scaled resolutions and the computed cluster map. Despite the resolution changes of factor 5 (left to right) corresponding face parts have the same color label, that is $M_{\sigma_1}(\sigma_1 x, \sigma_1 y) = M_{\sigma_2}(\sigma_2 x, \sigma_2 y)$.

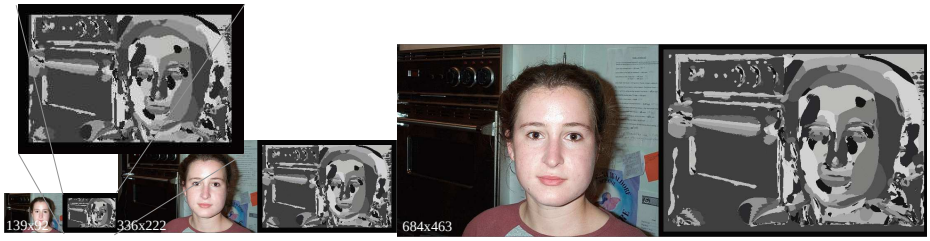

Fig. 3. Cluster maps $M_{\sigma_1}, M_{\sigma_2}, M_{\sigma_3}$ computed from images at different resolutions. The class-specific feature detectors are scale invariant.

3.4 Parameter optimisation

In this section we explain how we can judge the quality of a particular clustering and select those clusters that are useful for object description. A useful object description marks the class specific features such that the description allows to generalise from the training examples to unseen objects of the same class and the description allows to discriminate the object from non-objects such as background. A small number of clusters produce high generality, but bad discriminance. A high number of clusters has high discriminance and bad generalisation. This problem is related to finding the correct model and avoid overfitting.

We observe that neighboring image features are frequently assigned to the same cluster. This is a sign for the stability of the feature. Such stable features that have a good generalisation ability. A good set of feature descriptors therefore divides the object into several connected regions and forms a particular pattern in the cluster map representation. This pattern is exploited for detection.

The clusters should provide a segmentation into a number of connected regions. The regions should mark particular class specific features. There should be not too few regions neither too many. We tested k in the range from 5 to 40 and selected those clusters that are stable within the region of the training objects. As stability criteria we consider the average connected component size.

4 Verifying spatial relations

The complexity of identifying the best spatial configuration of a set of parts is related to a random graph matching problem. The complexity of matching a full graph with N model parts and M candidate parts is $O(M^N)$. This exponential

complexity is the reason that Fergus and Agarwal's approaches can handle only a small number of candidate parts. Labelling of graph nodes reduces the complexity to $O(\prod_{k=1}^{N} M_k)$ and M_k the number of candidate parts of model part k [9]. The complexity can be further reduced by imposing stronger constraints on the graph topology. This reduces the flexibility of the graph with the advantage of an efficient graph matching algorithm. The details of such an elastic matching of labelled graphs as proposed in [8,14,19] is explained in the next section where we also propose an alternative cost function that enables matching invariant to scale.

4.1 Elastic matching of labelled graphs

Elastic graph matching has previously been applied for grouping neurons dynamically into higher order entities [8]. These entities represent a rich structure which enables the recognition of higher level objects. Model objects are represented by sparse graphs whose vertices are labelled by a local appearance description and whose edges are labelled by geometric distance vectors. Recognition is formulated as elastic graph matching, that optimizes a matching cost function, which combines appearance similarity of the vertices and geometric graph similarity computed from the geometric information of the edges.

The matching cost function consists of two parts, C_v appearance similarity of the node labels, and C_e spatial similarity of the graph edges. A sparse graph $G = (\{x_i\}, \{\boldsymbol{\Delta_{ij}}\})$ consists of a set of vertices $\{x_i\}$ with image positions $\boldsymbol{v_i}$ and labels $\boldsymbol{m_i}$ that measure the local image appearance. The vertices are connected by edges $\boldsymbol{\Delta_{ij}} = \boldsymbol{v_i} - \boldsymbol{v_j}$ which are the distance vectors of the image position of the vertices x_i, x_j.

The spatial similarity evaluates corresponding edges of the query and the model graph by a quadratic comparison function:

$$S_e(\boldsymbol{\Delta_{ij}}^I, \boldsymbol{\Delta_{ij}}^M) = (\boldsymbol{\Delta_{ij}}^I - \boldsymbol{\Delta_{ij}}^M)^2, (i,j) \in E \tag{3}$$

where E is the set of edges in the model graph. The set E_{nn} containing the four nearest neighbors of a vertex is better suited to handle local distortions than the complete edge set [8].

The spatial similarity, S_e, measures the correspondence of the spatial distances between neighboring nodes. Distances are measured in pixels. The measure in (3) is scale dependent. This means that the measure can not distinguish between scaling and a strong distortion. We propose a normalisation by a scaling matrix, U, that can be computed from a global scale factor estimate. This normalisation makes the spatial similarity measure scale invariant.

$$U = \begin{pmatrix} \frac{w^I}{w^M} & 0 \\ 0 & \frac{h^I}{h^M} \end{pmatrix} \tag{4}$$

$$S_{e,U}(\boldsymbol{\Delta_{ij}}^I, \boldsymbol{\Delta_{ij}}^M) = (\boldsymbol{\Delta_{ij}}^I - U\boldsymbol{\Delta_{ij}}^M)^2 \tag{5}$$

with w^I, w^M width and h^I, h^M height of the query and the model region respectively.

Appearance similarity of labels is computed as the Mahalanobis distance of the feature vectors $\boldsymbol{m^I}, \boldsymbol{m^M} \in \mathcal{R}^d$:

$$S_v(\boldsymbol{m^I}, \boldsymbol{m^M}) = (\boldsymbol{m^I} - \boldsymbol{m^M})^T C^{-1} (\boldsymbol{m^I} - \boldsymbol{m^M}) \tag{6}$$

where C is the covariance of the local feature vectors of the training data. The Mahalanobis distance has the advantage to compensate for the covariance between the dimensions of the feature space. This measure is known to be stabler than the Euclidean distance in a features space composed of Gaussian derivatives of different order [2].

The cost function C_{total} is a weighted sum of the spatial similarity and the appearance similarity.

$$\begin{aligned} C_{total}(\{x_i^I\}, \{x_i^M\}) &= \lambda C_e + C_v \\ &= \lambda \sum_{(i,j) \in E} S_{e,U}(\boldsymbol{\Delta_{ij}}^I, \boldsymbol{\Delta_{ij}}^M) - \sum_{i \in V} S_v(\boldsymbol{m_i^I}, \boldsymbol{m_i^M}) \end{aligned} \tag{7}$$

The weighting factor λ controls the acceptable distortions of the query graph by penalising more or less the spatial similarity. The graph rigidity can be varied dynamically during optimization, which allows to employ a two stage algorithm that first places a rigid graph at the locally optimal position. The global cost function is then improved by allowing local distortions.

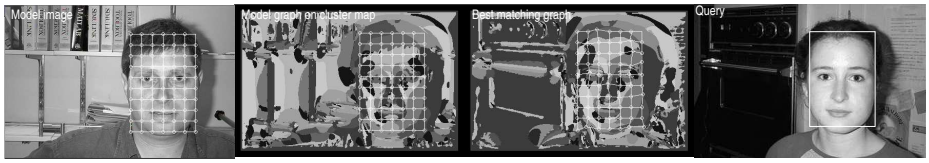

Fig. 4. Example of a detection by elastic graph matching on cluster map representation.

The original algorithm proposed by Lades is non-deterministic. Peters has proposed a deterministic version, that proceeds as follows. First a rigid graph is placed at the best position by raster scanning the image with a coarse step size. Then the graph is distorted locally by updating each node within a small window such that the labels are the most similar. An example is shown in Figure 4.

When scale changes must be considered, we search the optimal position and the optimal scale in the first step of the algorithm. These parameters are kept constant for the local optimisation phase. The optimal scale is selected among a predefined range of discrete values. We envision a preprocessing module that provides a rough estimate of the global object size, and the object position. Such a module would reduce significantly the computation time of the matching algorithm.

Without the preprocessing module, the matching algorithm has a complexity of $O(k_{nodes} k_{win} N) = O(N)$ with k_{nodes} the number of nodes (typically in the

range of 70 to 200), N the number of tested positions in the image (related to the image size), and k_{win} the size of the search window for the local position refinement. We use this graph matching algorithm to compare a query graph to a reference model graph.

5 Experiments

First we explain how the experiments are evaluated. Then the first experiment detects and classifies objects of approximately constant object size. This demonstrates the advantage of the class specific features over raw Gaussian derivative features. In the second experiment, artificially scaled objects are located by elastic graph matching using the model of the first experiment. This demonstrates the robustness to scale changes of our approach. The third experiment shows the performance of the system to detection of target objects in unconstrained images.

5.1 Set up

Fergus evaluates the results by ROC (receiver operator characteristics) equal error rates against the background dataset. His system evaluates a maximum likelihood ratio $\frac{p(Obj|X,S,A)}{p(BG|X,S,A)}$ where the background is modeled from the Caltech background set. In this way, few insertions are observed for this particular background set, but an equivalent performance on a different background set is not guaranteed. As stated by Agarwal [1], the ROC measures a system's accuracy as a classifier not as a detector. To evaluate the accuracy of a detector, we are interested in how many objects are detected correctly and how often the detections are false. These aspects of a system are captured by a recall-precision curve.

$$\text{Recall} = \frac{\text{Number of correct positives}}{\text{Total number of positives in dataset}} \tag{8}$$

$$\text{Precision} = \frac{\text{Number of correct positives}}{\text{Number of correct positives} + \text{Number of false positives}} \tag{9}$$

In order to suppress multiple detections on nearby locations, Agarwal implements the scheme of a classifier activation map, that allows to return only the activation extrema within a rectangular window with parameters w_{win}, h_{win}. A point (i_0, j_0) is considered an object location if

$$\text{cost}(i_0, j_0) \leq \text{cost}(i, j), \forall (i, j) \in N \tag{10}$$

where $N = \{(i, j) : |i - i_0| \leq w_{win}, |j - j_0| \leq h_{win}\}$ and no other point in N has been declared an object location.

We use the image database provided by Caltech[1] and the BioID database [6], with known object position and size. Figure 5 shows some example images. We consider a detection correct when following constraints are fulfilled.

[1] available at http://www.vision.caltech.edu/html-files/archive.html

1. $|i_{true} - i_{det}| < \delta_{width}$ and $|j_{true} - j_{det}| < \delta_{height}$, and
2. the detection and the ground truth region have an overlap of at least θ_{area}.

The parameters are set as a function of the object size, (w^M, h^M). We use $\delta_{width} = \frac{1}{3}w^M, \delta_{height} = \frac{1}{3}h^M$ and $\theta_{area} = 50\%$. This corresponds to the parameter setting used by Agarwal.

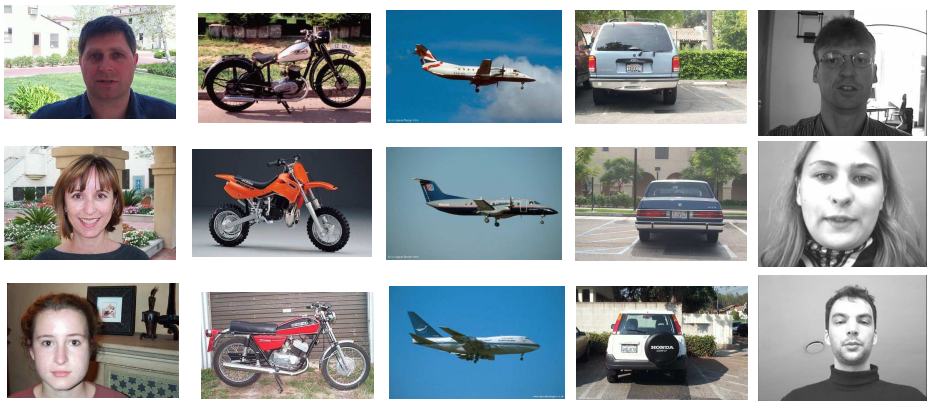

Fig. 5. Example images of the Caltech databases and the BioID face database.

5.2 Detection without scale changes

Table 1 summarises the detection results for different object classes, evaluated with different maximum cost thresholds (rectangles mark the best results with a false detection rate of $< 10\%$). There is a tradeoff between recall and precision according to this threshold. We compare elastic graph matching on 5 dimensional Gaussian derivative features (first and second derivatives, scale invariant) and elastic graph matching on the cluster map representation. For faces, the Gaussian derivatives have a higher precision than the cluster map representation. This is due to the significant data reduction which increase the frequency of insertions. However, we obtain very good detection rates with both techniques.

For the other data bases, the cluster map representation produces superior results. Motorbikes and airplanes can be reliably localised ($\geq 95\%$) with false detection rates $< 10\%$. Elastic graph matching on Gaussian derivative features has a higher false positive rate. This confirms that the cluster map representation detects the class-specific feature robustly to intra-class variations.

The data base of rear views of cars has a much lower precision rate. This is due to the lack of structure of the target objects. Many false positives are found. The targets display a large variance in appearance and also in the spatial arrangement which explains the lower detection rate. The current non-optimised implementation requires an average of $3.3s$ for processing an image of size 252x167 pixels

on a Pentium 1.4GHz (automatic scale selection, 5 Gaussian derivative filter operations on all image pixels, transformation into cluster map representation and optimising elastic graph matching function).

Detection by elastic graph matching using					
Cluster map			Gaussian derivatives, 5 dims		
Faces, 435 images, graph 7x10 nodes, 5 classtons					
Max cost	Recall	Precision	Max cost	Recall	Precision
25	92.2%	95.7%	160	96.4%	91.6%
35	94.3%	87.7%	180	99.5%	43.8%
45	96.4%	77.7%			
Motorbikes, 200 images, graph 9x15 nodes, 5 classtons					
50	91.5%	96%	600	69.8%	75.5%
70	97%	91.1%	1000	82.4%	66.4%
Airplanes, 200 images, graph 7x19 nodes, 5 classtons					
55	95.3%	90.3%	800	74.8%	98.8%
65	96.3%	84.4%	1000	94.4%	87.1%
75	96.3%	74.6%			
Cars (rear view), 200 images, graph 11x15 nodes, 10 classtons					
100	72%	62.1%	1000	65.5%	62.6%
120	91.2%	54.3%	1200	84%	58.3%

Table 1. Detection results without scale changes (rectangles mark best results with precision $> 90\%$).

5.3 Detection under scale changes

To evaluate the performance of our system to objects of different sizes, we have created artificially scaled images from the Caltech database and a database with natural scale changes (the first 99 images of the BioID database [6]). The cluster map representation is scale invariant due to the scale invariant feature extraction (see Figure 3). The spatial relation model in form of a labelled graph is scale dependent. When searching a best fitting graph, we search the best position and the best scale among a set of discrete positions and scales (we test positions that are evenly spaced by 3 pixels and the tested scales are $0.56, 0.75, 1.0, 1.25$). Table 2 shows the detection results. We observe high detection rates. However, many more false positives are observed, elastic graph matching optimises the matching function in space, distortion and scale. As a consequence we observe more false positives which decreases the precision rate. Figure 6 shows an example of typical insertions. The object is detected correctly and in addition several subparts of the motorbike are detected as well. An algorithm which removes multiple detections would help to reduce the high number of insertions and improve the detector precision.

	Scale range	Recall	Precision	Max cost
Faces	0.56 - 1.0	96.2%	42.1%	35
		93.6%	54.5%	30
Motorbikes	0.56 - 1.0	82.9%	57.3%	85
		80.9%	65%	70
Airplanes	0.56 - 1.25	89.6%	58.9%	65
		83.6%	68%	55
BioID faces	natural	89.9%	75.4%	30

Table 2. Detection of objects on images with scale changes (method elastic graph on cluster map).

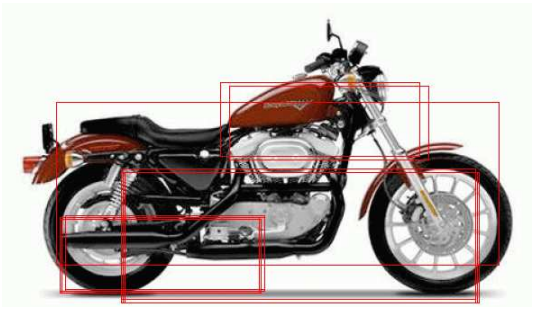

Fig. 6. Example of typical insertions under scale changes.

5.4 Detection in unconstrained images

We tested our method on images with natural scale changes. For detection of faces, we use the cluster map representation that is learned from the faces of the Caltech face database. For detection we perform elastic graph matching on cluster map over different scales. Figure 7 shows an detection example. The model image is significantly different from the query faces. All faces are detected and we observe no false positives.

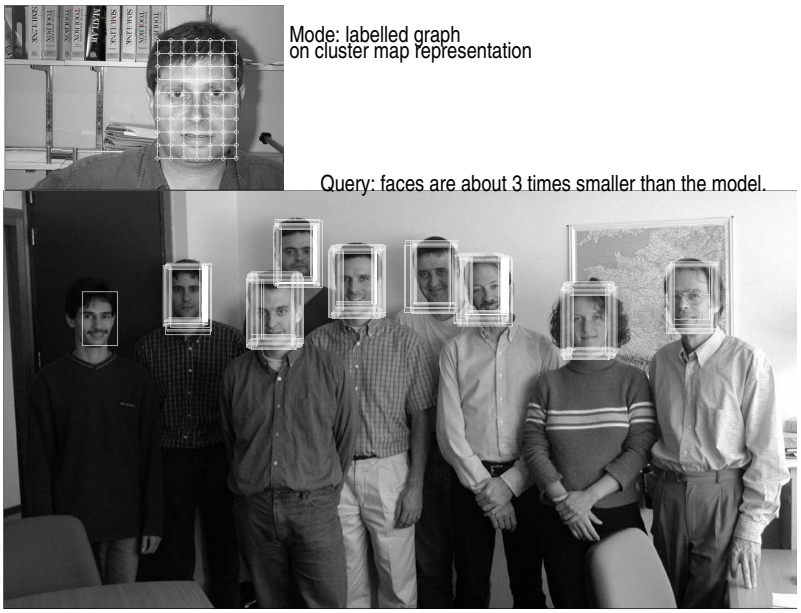

Fig. 7. Successful detection of faces in unconstrained images. The white rectangles mark the position and size of the graphs with lowest cost.

6 Conclusions and future work

In this article we have proposed a method to generate class-specific feature detectors that learn the intra-class variability and allows to represent an image as a cluster map, which preserves the position and the type of the class-specific feature. This is the minimum information required for detection and classification. Reliable detection is obtained by verifying spatial constraints of the features by graph matching. We proposed a method for geometry verification that has a much lower computational complexity than other algorithms. Furthermore, the proposed verification method is invariant to scale and enables successful detection of different kinds of object classes. The method allows to locate objects

observed at various scales and produces good results for a selection of unconstrained images.

The strong data reduction of the cluster maps increases the probability of false positives. This is natural and caused by the information reduction of the cluster map representation. The current implementation is non-optimised and requires to search scale space for optimising the matching cost function. We are working on an additional preprocessing module that extracts candidate locations by image signal properties at very large scale. The a-priori knowledge of the location and approximate size of candidate regions is the key for a fast detection and classification system.

References

1. S. Agarwal and D. Roth. Learning a sparse reprsentation for object detection. In *ECCV*, pages 113–130, 2002.
2. V. Colin de Verdière. *Représentation et Reconnaissance d'Objets par Champs Réceptifs*. PhD thesis, Institut National Polytechnique de Grenoble, France, 1999.
3. G. Dorko and C. Schmid. Selection of scale-invariant parts for object class recognition. In *International Conference on Computer Vision*, Nice, France, October 2003.
4. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, Madison, USA, 2003.
5. D. Hall, V. Colin de Verdière, and J.L. Crowley. Object recognition using coloured receptive fields. In *ECCV*, pages I 164–177, Dublin, Ireland, June 2000.
6. O. Jesorsky, K. kirchberg, and R. Frischholz. Robust face detection using the hausdorff distance. In *Audio and Video based Person Authentication AVBPA 2001*, pages 90–95, 2001.
7. J.J. Koenderink and A.J. van Doorn. Generic neighborhood operators. *PAMI*, 14(6):597–605, June 1992.
8. M. Lades, J.C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Mahlsburg, R.P. Würz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *Transactions on Computers*, 42(3):300–311, March 1993.
9. T.K. Leung, M.C. Burl, and P. Perona. Finding faces in cluttered scenes using random labelled graph matching. In *ICCV*, 1995.
10. T. Lindeberg. Edge detection and ridge detection with automatic scale selection. *IJCV*, 30(2), 1998.
11. D.G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, pages 1150–1157, 1999.
12. J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *IJCV*, 43(1):7–27, June 2001.
13. K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *International Conference on Computer Vision*, pages 525–531, Vancouver, Canada, July 2001.
14. G. Peters. *A view-based approach to three-dimensional object perception*. PhD thesis, Universität Bielefeld, December 2001.
15. R.P.N. Rao and D.H. Ballard. An active vision architecture based on iconic representations. *Artificial Intelligence*, 78(1–2):461–505, 1995.

16. C. Schmid. Constructing models for content-based image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, Kauai, USA, December 2001.
17. C. Schmid and R. Mohr. Local greyvalue invariants for image retrieval. *PAMI*, 1997.
18. L.J. van Vliet, I.T. Young, and P.W. Verbeek. Recursive gaussian derivative filters. In *International Conference on Pattern Recogntion*, pages 509–514, August 1998.
19. L. Wiskott, J.M. Fellous, N. Krüger, and C. von der Mahlsburg. *Face Recognition by Elastic Bunch Graph Matching*, chapter 11, pages 355–396. Intelligent Biometric Techniques in Fingerprint and Face Recognition. CRC Press, 1999.