Components for Object Detection and Identification

Bernd Heisele[1,2], Ivaylo Riskov[1], and Christian Morgenstern[1]

[1] Center for Biological and Computational Learning, M.I.T.,
Cambridge, MA 02142, USA
riskov@mit.edu, christian.morgenstern@upc.edu
[2] Honda Research Institute US
Boston, MA 02111, USA
bheisele@honda-ri.com

Abstract. We present a component-based system for object detection and identification. From a set of training images of a given object we extract a large number of components which are clustered based on the similarity of their image features and their locations within the object image. The cluster centers build an initial set of component templates from which we select a subset for the final recognizer. The localization of the components is performed by normalized cross-correlation. Two types of components are used, gray value components and components consisting of the magnitudes of the gray value gradient.

In experiments we investigate how the component size, the number of the components, and the feature type affects the recognition performance. The system is compared to several state-of-the-art classifiers on three different data sets for object identification and detection.

1 Introduction

Object detection and identification systems in which classification is based on local object features have become increasingly common in the computer vision community over the last couple of years, see e.g. [24,8,11,26,4]. These systems have the following two processing steps in common: In a first step, the image is scanned for a set of characteristic features of the object. For example, in a car detection system a canonical gray-value template of a wheel might be cross-correlated with the input image to localize the wheels of a car. We will refer to these local object features as the components of an object, other authors use different denotations such as parts, patches or fragments. Accordingly, the feature detectors will be called component detectors or component classifiers. In a second step, the results of the component detector stage are combined to determine whether the input image contains an object of the given class. We will refer to this classifier as the combination classifier.

An alternative approach to object classification is to search for the object as a whole, for example by computing the cross-correlation between a template of the object and the input image. In contrast to the component-based approach, a single classifier takes as input a feature vector containing information about the

J. Ponce et al. (Eds.): Toward Category-Level Object Recognition, LNCS 4170, pp. 225–237, 2006.

whole object. We will refer to this category of techniques as the global approach; examples of global face detection systems are described in [23,13,18,14,7]. There are systems which fall in between the component-based and the global approach. The face detection system in [25], for example, performs classification with an ensemble of simple classifiers, each one operating on locally computed image features, similar to component detectors. However, each of these simple classifiers is only applied to a fixed x-y-position within the object window. In the component-based approach described above, the locations of the components relative to each other are not fixed: each component detector performs a search over some part of the image to find the best matching component.

In the following we briefly motivate the component-based approach:

(a) A major problem in detection is the variation in the appearance of objects belonging to the same class. For example, a car detector should be able to detect SUVs as well as sports cars, even though they significantly differ in their shapes. Building a detector based on components which are visually similar across all objects of the class might solve this problem. In the case of cars, these indicator components could be the wheels, the headlights or the taillights.

(b) Components usually vary less under pose changes than the image pattern of the whole object. Assuming that sufficiently small components correspond to planar patches on the 3D surface of the object, changes in the viewpoint of an object can be modeled as affine transformations on the component level. Under this assumption, view invariance can be achieved by using affine invariant image features in the component detector stage as proposed in [4]. A possibility to achieve view invariance in the global approach is to train a set of view-tuned, global classifiers as suggested in [15].

(c) Another source of variations in an objects appearance is partial occlusion. In general it is difficult to collect a training set of images which covers the spectrum of possible variations caused by occlusion. In the component-based approach, partial occlusion will only affect the outputs of a few component detectors at a time. Therefore, a solution to the occlusion problem might be a combination classifier which is robust against changes in a small number of its input features, e.g. a voting-based classifier. Another possibility is to add artificial examples of partially occluded objects to the training data of the combination classifier, e.g. by decreasing the component detector outputs computed on occlusion-free examples. Experiments on detecting partially occluded pedestrians with a component-based system similar to the one describe in our chapter have been reported in [11].

One of the main problems that has to be addressed in the component-based approach is how to choose a suitable set of components. A manually selected set of five components containing the head, the upper body, both arms, and the lower body has been used in [11] for person detection. Although there are intuitively obvious choices of components for many types of objects, such as the eyes, the nose and the mouth for faces, a more systematic approach is to automatically select the components based on their discriminative power. In [24] components of various sizes were cropped at random locations in the training images of an object. The mutual information between the occurrence of a component in a training image and the class label of the image was used as a measure to rank and select components. Another strategy to automatically determine an initial set of components is to apply a generic interest operator to the training images and to select components located in the vicinity of the detected points of interest [5,4,10]. In [4], this initial set was subsequently reduced by selecting components based on mutual information and likelihood ratio. Using interest operators has the advantage of providing a quick and reliable way to locate component candidates in a given input image. However, forcing the locations of the components to coincide with the points detected by the interest operator considerably restricts the choice of possible components—important components might be lost. Furthermore, interest operators have a tendency to fail for objects with little texture and objects at a low pixel resolution.

How to include information about the spatial relationship between components is another important question. In the following we assume that scale and translation invariance are achieved by sliding an object window of fixed size over the input image at different resolutions—the detection task is then reduced to classifying the pattern within the current object window. Intuitively, information about the location of the components is important in cases where the number of components is small and each component carries only little class-specific information.

We adopt the component-based classification architecture similar to the one suggested in [8,12]. It consists of two levels of classifiers; component classifiers at the first level and a single combination classifier at the second level. The component classifiers are trained to locate the components and the combination classifier performs the final detection based on the outputs of the component classifiers. In contrast to [8], where support vector machines (SVM) were used at both levels, we use component templates and normalized cross-correlation for detecting the components and a linear classifier to combine the correlation values.

2 System Description

2.1 Overview

An overview of our two-level component-based classifier is shown in Fig. 1. At the first level, component classifiers independently detect components of the object. Each component classifier consists of a single component template which is matched against the image within a given search region using normalized cross-correlation. We pass the maximum correlation value of each component to the combination classifier at the second level. The combination classifier produces a binary recognition result which classifies the input image as either belonging to the background class or to the object class.

2.2 Features

The applicability of a certain feature type depends on the recognition task at hand–some objects are best described by texture features, others by shape

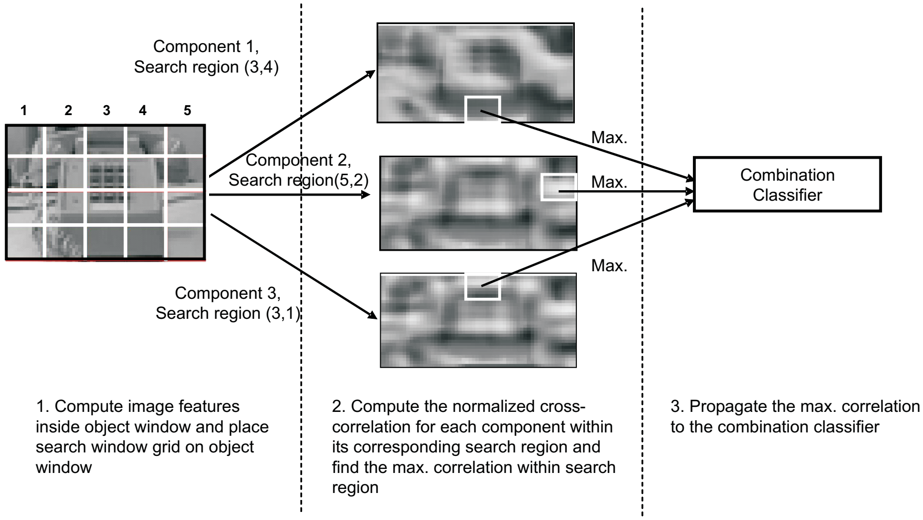
Component 1,
Search region (3,4)
1
2
3
4
5
Component 2,
Search region(5,2)
Component 3,
Search region (3,1)
Max.
Max.
Max.
Combination
Classifier
1. Compute image features inside object window and place search window grid on object window
2. Compute the normalized cross-correlation for each component within its corresponding search region and find the max. correlation within search region
3. Propagate the max. correlation to the combination classifier

Fig. 1. The component-based architecture: At the first level, the component templates are matched against the input image within predefined search regions using normalized cross-correlation. Each component's maximum correlation value is propagated to the combination classifier at the second level.

features. The range of variations in the pose of an object might also influence the choice of features. For example, the recognition of cars and pedestrians from a car-mounted camera does not require the system to be invariant to in-plane rotation; the recognition of office objects on a desk, on the other hand, requires invariance to in-plane rotation. Invariance to in-plane rotation and to arbitrary rotation of planar objects can be dealt with on a feature level by computing rotation-invariant or affine invariant features, e.g. see [10,4]. In the general case, however, pose invariance requires an altogether different classification architecture such as a set of view-tuned classifiers [15]. Looking at biological visual systems for clues about useful types of features is certainly a legitimate strategy. Recently, a biologically motivated system which uses Gabor wavelet features at the lowest level has shown good results on a wide variety of computer vision databases [21]. With broad applicability and computational efficiency in mind, we chose gray values and the magnitudes of the gradient as feature types.[1]

2.3 Geometrical Model

Omitting any spatial information leads to a detection system similar to the biologically plausible object recognition models proposed in [16,24,21]. In [16],

[1] We computed the gradient by convolving the image with the derivatives of a 2D-Gaussian with $\sigma = 1$.

the components were located by searching for the maximum outputs of the detectors across the full image. The only data propagated to the higher level classifiers were the outputs of the component detectors.

A framework in which the geometry is modeled as a prior on the locations of the components by a graphical model has been proposed in [3]. The complexity of the graphical model could be varied between the simple naïve Bayesian model and the full joint Gaussian model. The experimental results were inconclusive since the number of components was small and the advantages of the two implemented models over the naïve Bayesian model were in the range a few percent. In cases where the number of components and the number of detections per component are large, complex models might become computationally too expensive. A standard technique to keep the number of detections small is to apply interest operators to the image. The initial detections are then solely appearance-based which has the disadvantage that configurations with a high geometrical prior might be discarded early in the recognition process. A technique in which both appearance and geometry are used at an early stage has been proposed in [2].

We introduce geometrical constraints by restricting the location of each component to be within a pre-defined search region inside the object window. The search regions can be interpreted as a simple geometrical model in which the prior for finding a component within its search region is uniform and the prior of finding it outside is zero.

2.4 Selecting Components

As shown in Fig. 1 we divided the image into non-overlapping search regions. For each of the object images in the training set and for each search region we extracted 100 squared patches of fixed size whose centers were randomly placed within the corresponding search region. We then performed a data reduction step by applying k-means clustering to all components belonging to the same search region. The k-means clustering algorithm has been applied before the context of computing features for object recognition [17,20]. The resulting cluster centers built our initial set of component templates. For each component template we built a corresponding component classifier which returned a single output value for every training image. This value was computed as the maximum of the correlation between the component template and the input image within the search region. The next step was to select a subset of components from the pool of available component templates. We added a negative training set containing non-object images which had the same size as the object images and either used Adaboost [19] or Gentle-boost [6] to select the component templates. In a previous study [12] we evaluated two other feature selection techniques on an object identification database: a method based on the individual performance (ROC area) of the component templates and forward stepwise regression [27]. The technique based on the ROC area performed about the same as Adaboost, forward stepwise regression did worse.

3 Experiments

3.1 The MIT Office Object Database

In this identification task, the positive training and test data consisted of images of four objects, a telephone, a coffee machine, a fax machine, and a small bird figurine.[2] The object images were manually cropped from high resolution color images recorded with a digital camera. The aspect ratio of the cropping window was kept constant for each object but varied between the four objects. After cropping we scaled the object images to a fixed size. For all objects we used randomly selected 4,000 non-object training images and 9,000 non-object test images. Some examples of training and test images of the four objects are shown in Fig. 2. We kept the illumination and the distance to the object fixed when we took the training images and only changed the azimuthal orientation of the camera. When we took the test pictures, we changed the illumination, the distance to the object, and the background for the small objects. We freely moved the hand-held camera around the objects allowing all three degrees of freedom in the orientation of the camera.

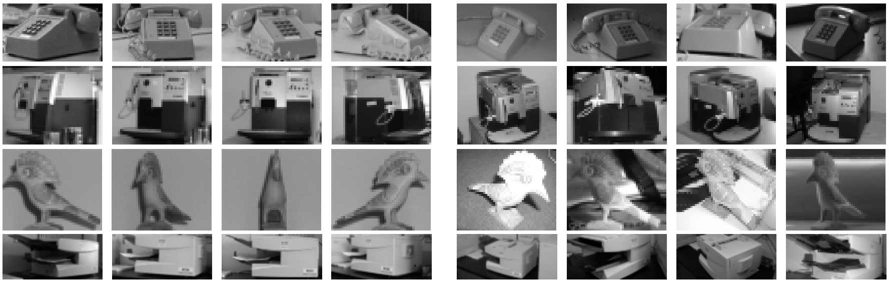

Fig. 2. Examples of training and test images for the four objects. The first four images in each row show training examples, the last four were taken from the test set.

Before we trained the final recognition systems we performed a couple of quick tests on one of the four objects (telephone) to get an idea of how to choose the size the components and the number of components for further experiments. We also verified the usefulness of search regions:[3]

- We compared a system using search regions to a system in which the maximum output of the component classifiers was computed across the whole object window. Using search regions improved the recognition rate by more

[2] Fax machine: 44 training images and 157 test images of size 89 × 40. Phone: 34 training images and 114 test images of size 69 × 40 pixels. Coffee machine: 54 training and 87 test images of size 51 × 40. Bird: 32 training images and 131 test images of size 49 × 40.

[3] The ROC curves for the experiments can be found in [12].

than 20% up to a false positive (FP) rate of 0.1. Search regions were used in all following experiments.
- We trained four classifiers on gray value components of size 3×3, 5×5, 10×10 and 15 × 15. All four classifiers used 30 components selected by Adaboost. The classifiers for sizes 5 × 5, 10 × 10 and 15 × 15 performed about the same while the 3 × 3 classifier was significantly worse. We eliminated components of size 3 × 3 from all further experiments.
- We trained a classifiers on gray value components of size 5 × 5 with the number of components ranging between 10 and 400. For more than 100 components the performance of the classifier did not improve significantly.
- We evaluated the two feature types by selecting 100 components from a set of gray value components, a set of gradient components, and the combination of both sets. The gray values outperformed the combination slightly, the gradient components performed worst. The differences, however, were subtle (in the 2% range) and did not justify the exclusion of gradient features from further experiments.

In the final experiment we trained a separate classifier for each of the four objects. We randomly cropped gray value and gradient components from the positive training images at sizes 5 × 5, 10 × 10, and 15 × 15. Components of the same size and the same feature type, belonging to the same search region were grouped into 30 clusters of which only the components at the cluster centers entered the following selection process. Of the 3,600 components we selected a subsets of 100 and 400 components. As as baseline system we trained four SVMs with on the raw gray values of the objects.[4] Fig. 3 shows the ROC curves for the four different objects for the component-based system and the global SVM classifier. Except for the fax machine, where both systems were on par, the component based system performed better. Both systems had problems recognizing the bird. This can be explained by the strong changes in the silhouette of the figure under rotation. Since we extracted the object images with a fixed aspect ratio, some of the training images of the bird contained a significant amount of background. Considering the fact that the background was the same on all training images but was different on the test images, the relatively poor performance is not surprising.

3.2 The MIT Face Database

In this set of experiments we applied the system with a 4 × 4 search region grid to a face detection database. The positive training set consisted of about 9,000 synthetic face images of size 58 × 58, the negative training set contained about 13,700 background patches of the same size. The test set included 5,000 non-face patterns which were selected by a 19 × 19 low-resolution LDA classifier as the most similar to faces out of 112 background images. The positive test set

[4] We did experiments with Gaussian and linear kernels and also applied histogram-equalization in the preprocessing stage. Fig. 3 shows the best results achieved with global systems.

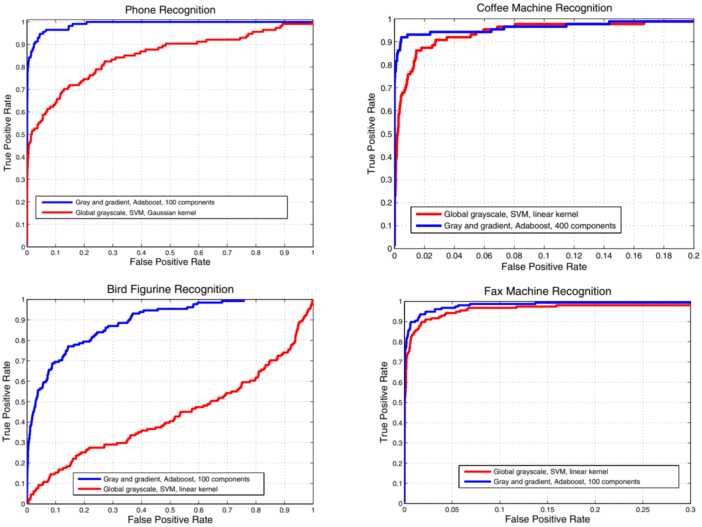

Fig. 3. Final identification results for the four different objects using a combination of gray value and gradient components in comparison to the performance of a global classifier

consisted of a subset of the CMU PIE database [22] that we randomly sampled across the individuals, illumination and expressions. The faces were extracted based on the coordinates of facial feature points given in the CMU PIE database. We resized these images to 70 × 70 such that the faces in test and training set were at about the same scale. Some examples from the training and test sets are shown in Fig. 4. When testing on the 70 × 70 images we applied the shifting object window technique.

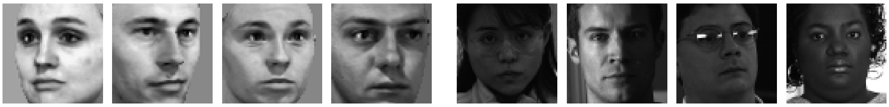

Fig. 4. Examples from the face database. The images on the left half show training examples, the images on the right test examples taken from the CMU PIE database. Note that the test images show a slightly larger part of the face than the training images.

In the following we summarize the experiments on the face database:

- We compared Adaboost, previously used on the office database, with Gentle-boost. Gentle-boost produced consistently better results. The improvements were subtle, the largest increase in ROC area achieved in any of the comparisons was 0.01. In all of the following experiments we used Gentle-boost to select the components.

– We compared systems using gray value components of size 5×5, 10×10, and 15×15. The systems performed about the same.
– When increasing the number of gray value components of size 5×5 from 10 up to 80 the ROC area increased by 0.016. Adding more components did not improve the results.
– Gradient components performed poorly on this database. In a direct comparison using 100 5×5 components the ROC area of the gradient system was about 0.2 smaller than that of the gray value system.

In conclusion, systems with 80 gray value components of size 5×5 or 10×10 selected by Gentle-boost gave best results for face detection. Gradient components were not useful for this database, adding them to the pool of gray value components lead to a decrease in the system's performance. A comparison to the 14 component system using SVMs [8][5] and the biologically inspired model in [21] is given in Table I.

Table 1. Comparison between our system with 80 gray value components and two baseline systems. Given are the ROC area and the recognition rate at the point of equal error rates (EER).

	Our system	14 components SVM [8]	Biological model [21]
ROC area	0.995	0.960	0.993
1− EER	0.962	0.904	0.956

3.3 The MIT Car Database

This database was collected at MIT as part of a larger project on the analysis of street scenes [1]. It includes around 800 positive images of cars of size 128×128 and around 9,000 negative background patterns of the same size. Since no explicit separation of training and testing images was given, we followed the procedure in [1] and randomly selected two thirds of the images for training and the rest for testing. As for faces, we used a 4×4 search region grid. As the samples in Fig. 5 show, the set included different types of cars, strong variations the viewpoint (side, front and rear views), partial occlusions, and large background parts.

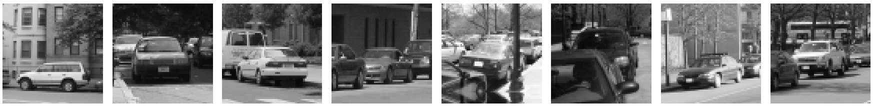

Fig. 5. Examples from the car database. Note the large variations in pose and illumination.

It turned out thats small components performed the best on this database. The ROC area for gray value components of a fixed size decreased by 0.12 when

[5] A different training set of faces was used in this paper.

increasing the size of the components from 5×5 to 15×15. The appearance of the cars in this database varied strongly making it unlikely to find large components which are shared amongst the car images. Since the shadow below the car was a salient feature across most of the car images, it did not surprise that the gradient components outperformed the gray components on this task.

In Fig. 6 we compare the ROC curves published in [1] with our system using 100 gradient components of size 5×5 selected by Gentle-boost. We did not train the systems on the same data since the database did not specify exactly how to split into training and test sets. However, we implemented a global classifier similar to the one used in [1] and applied it to our training and test sets. The right diagram shows two wavelet-based systems labeled "C1" and "C2", the latter is similar to [21], a global gray value classifier using an SVM, labeled "global grayscale", a part-based system according to [9], and a patch-based approach in which 150 out of a pool of 1024 12×12 patches were selected for classification. In a direct comparison, our system performs similar to the "C2" system and slightly worse than the "C1" system. This comparison should be taken with a grain of salt since the global gray value classifier performed very differently on the two tests (compare the two curves labeled "global grayscale").

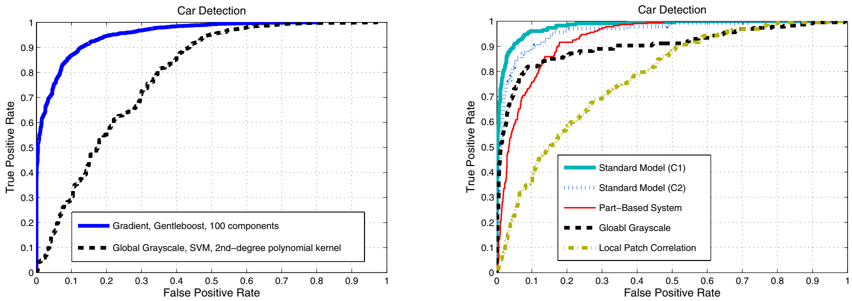

Fig. 6. The ROC curves on the left compare the component system to a global system, the curves on the right are taken from [1]. The curves in both diagrams have been computed on the MIT car detection database, however, the splits into training and test sets were different.

4 Conclusion

We presented a component-based system for detecting and identifying objects.

From a set of training images of a given object we extracted a large number of gray value and gradient components which were split into clusters using the k-means algorithm. The cluster centers built an initial set of component templates. We localized the components in the image by finding the maxima of the normalized cross-correlation inside search regions. The final classifier was built by selecting components with Adaboost or Gentle-boost.

In most of our experiments, selecting around 100 components from a pool of several thousands seemed to be sufficient. The proper choice of the size of the components proved to be task-dependent. Intermediate component sizes between 5×5 and 15×15 pixels led to good results on the objects in our databases, which varied in resolution between 50×50 and 130×130 pixels. We also noticed that the optimal choice of the feature type depends on the task. While the gray value components outperformed the gradient components in the office object and face experiments, the gradient components proved to be better for detecting cars.

We showed that our system can compete with state-of-the-art detection and identification systems. Only on one of the databases our system was outperformed by a detection system using wavelet-type features. We see the main advantages of our approach in its conceptual simplicity and its broad applicability. Since both the computation of the features and the matching algorithm are computationally simple, the system has the potential of being implemented in real-time.

Acknowledgements

The authors would like to thank S. Bileschi for providing the car database and experimental results on this database.

This chapter describes research done at the Center for Biological and Computational Learning, which is in the McGovern Institute for Brain Research at MIT, as well as in the Department of Brain and Cognitive Sciences, and which is affiliated with the Computer Sciences and Artificial Intelligence Laboratory (CSAIL). This research was sponsored by grants from: Office of Naval Research (DARPA) Contract No. MDA972-04-1-0037, Office of Naval Research (DARPA) Contract No. N00014-02-1-0915, National Science Foundation-NIH (CRCNS) Contract No. EIA-0218506, and National Institutes of Health (Conte) Contract No. 1 P20 MH66239-01A1. Additional support was provided by: Central Research Institute of Electric Power Industry (CRIEPI), Daimler-Chrysler AG, Eastman Kodak Company, Honda Research Institute USA, Komatsu Ltd., Merrill-Lynch, NEC Fund, Oxygen, Siemens Corporate Research, Inc., Sony, Sumitomo Metal Industries, and the Eugene McDermott Foundation.

References

1. S. Bileschi and L. Wolf. A unified system for object detection, texture recognition, and context analysis based on the standard model feature set. In *British Machine Vision Conference (BMVC)*, 2005.
2. S. M. Bileschi and B. Heisele. Advances in component-based face detection. In *Proceedings of Pattern Recognition with Support Vector Machines, First International Workshop, SVM 2002*, pages 135–143, Niagara Falls, 2002.
3. D. Crandall, P. Felzenszwalb, and D. Huttenlocher. Spatial priors for part-based recognition using statistical model. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10–17, 2005.

4. G. Dorko and C. Schmid. Selection of scale invariant neighborhoods for object class recognition. In *International Conference on Computer Vision (ICCV)*, pages 634–640, 2003.
5. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 264–271, 2003.
6. J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. Tecnical report, Dept. of Statistics, Stanford University, 1998.
7. B. Heisele, T. Serre, S. Mukherjee, and T. Poggio. Hierarchical classification and feature reduction for fast face detection with support vector machines. *Pattern Recognition*, 36(9):2007–2017, 2003.
8. B. Heisele, T. Serre, M. Pontil, T. Vetter, and T. Poggio. Categorization by learning and combining object parts. In *Neural Information Processing Systems (NIPS)*, Vancouver, 2001.
9. B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit model. In *ECCV'04 Workshop on Statistical Learning in Computer Vision*, 2004.
10. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
11. A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 23, pages 349–361, April 2001.
12. C. Morgenstern and B. Heisele. Component-based recognition of objects in an office environment. A.I. Memo 232, Center for Biological and Computational Learning, M.I.T., Cambridge, MA, 2003.
13. M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 193–199, San Juan, 1997.
14. E. Osuna. *Support Vector Machines: Training and Applications.* PhD thesis, MIT, Department of Electrical Engineering and Computer Science, Cambridge, MA, 1998.
15. T. Poggio and S. Edelman. A network that learns to recognize 3-D objects. *Nature*, 343:163–266, 1990.
16. M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2(11):1019–1025, 1999.
17. M. Riesenhuber and T. Poggio. The individual is nothing, the class everything: Psychophysics and modeling of recognition in object classes. A.I. Memo 1682, Center for Biological and Computational Learning, M.I.T., Cambridge, MA, 2000.
18. H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.
19. R. Schapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the margin: A new explanation of effectiveness of voting methods. *The Annals of Statistics*, 26(5):1651–1686, 1998.
20. T. Serre, J. Louie, M. Riesenhuber, and T. Poggio. On the role of object-specific features for real world recognition in biological vision. In *Biologically Motivated Computer Vision, Second International Workshop (BMCV 2002)*, pages 387–397, Tuebingen, Germany., 2002.
21. T. Serre, L. Wolf, and T. Poggio. A new biologically motivated framework for robust object recognition. A.I. Memo 2004-26, Center for Biological and Computational Learning, M.I.T., Cambridge, USA, 2004.

22. T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination, and expression database. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 25(12):1615–1618, 2003.
23. K.-K. Sung. *Learning and Example Selection for Object and Pattern Recognition.* PhD thesis, MIT, Artificial Intelligence Laboratory and Center for Biological and Computational Learning, Cambridge, MA, 1996.
24. S. Ullman, M. Vidal-Naquet, and E. Sali. Visual features of intermdediate complexity and their use in classification. *Nature Neuroscience*, 5(7):682–687, 2002.
25. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 511–518, 2001.
26. M. Weber, W. Welling, and P. Perona. Towards automatic dscovery of object categories. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 2000.
27. S. Weisberg. *Applied Linear Regression.* Wiley, New York, 1980.