Object Detection using a Cascade of Classifiers

Nayyar A.Zaidi, David Suter

Department of Electrical and Computer Systems Engineering
Monash University, Clayton VIC 3800, Australia
Email: nayyar.zaidi@eng.monash.edu.au,d.suter@eng.monash.edu.au

Abstract

Typical object detection systems work by training a classifier on features extracted at different scales of an object. In this paper we investigate the performance of an object detection system in which different classifiers which are trained at various scales of an object are combined and compare the performance with a typical object detection system where a single classifier is trained for all the scales. The notion behind such an approach is that the features extracted over smaller scales give more object's specific information whereas large scale features provide more contextual information. We trained different classifiers for different scales and combined their output to reach a decision about the existence of an object. Confidence rated Ada-boost is used to train the classifiers. It was found that training a single classifier for all the scales results in superior performance as compared to training different classifiers for each scale and than combining their results. We show our results on objects belonging to three categories in TUDarmstadt and one category in Caltech4 [1].

1 Introduction

In this paper we propose a technique for object detection. For example, given an image we would like to detect the instance of a specific object. There exist myriad of approaches for object categorization which classify images based upon the presence of an object inside them irrespective of where exactly that object is present or located inside the image. Object detection is particularly an extension of object classification methods in which an image is not only categorized but also the object of interest is detected and located. Because of scale, rotation, deformation and viewpoint changes of the object, object detection is quite a challenging area of computer vision, therefore many state of the art methods only solve a binary classification problem.

The issues of scale in object detection systems have always been very critical. It is unknown at what scale the features need to be computed. However, it is widely agreed that features computed at higher scales tend to have more contextual information whereas lower scale features tend to be more specific.

Typical object detection system work by training a classifier on features that are extracted at different scales. In this paper we study the effect of different scales on the detection performance. We extract features at different scales and train different classifiers each specialized in making a decision for a single scale. We combine the output of these classifiers and adjust the weights to control the contribution of each classifier, such that classifiers trained at high scales are weighted more or less than the classifiers trained at lower scales and vice-versa. Our series of experiments reveal that training classifiers only from features extracted at a particular scale and combining them using different weighting schemes do not result in a performance gain. Training a single classifier for all scales and letting the learning algorithm choose the best feature and scale can result in much better performance.

We discuss some other object detection systems in section 2, our approach in section 3 and experimental results in section 4. We conclude in section 5.

2 Related Work

There are a lot of methods suggested for object detection in the presence of occlusion, clutter, scale changes, rotation, deformations and viewpoint variations. They vary in their choice of the feature sets and choice of the classifier [2, 3, 4]. Torralba et al in [5] are using shared boosted decision stumps to find shared features among different classes. The features are inspired from [6] and are extracted for 32x32 window of an image (note the window size is fixed). Opelt et al in [3] detects objects using boundary fragment model. The chamfer distance

978-0-7695-3456-5/08 $25.00 © 2008 IEEE
DOI 10.1109/DICTA.2008.55

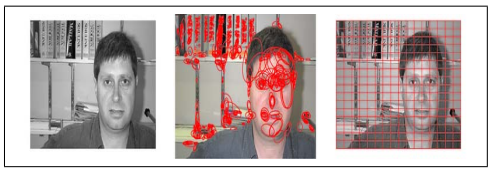

Figure 1: Left: Original Image, Middle: shows the interest points along with the area where features are to be computed, Right: shows the grid and the points at which features are to be computed.

between the boundaries fragments of an object are trained using boosting. Shotton et al [4] approach in is very much similar to Opelt, as they are also using boundaries and a novel formulation of chamfer distance to train classifier to detect object in the image. Leibe et al obtain the best results for object detection in [2] where they propose implicit shape model. Segmentation and recognition are considered as the same process both sharing information. Some other interesting work on object detection has been done in [7, 8].

There has been a significant debate over the use of features extracted around the interest points (sparse features) to features extracted around points on a regular grid (dense feature). The interest points are detected using some key-point detector for example Harris-Affine. The features are computed for the area around these points, so the area around these points are characterized in scale and rotation invariant way. These are typically know as sparse features as contrast to dense features which are computed for some area around grid points (both sparse and dense points shown in figure 1). Using interest points have the advantage that the feature are scale invariant, but they have the disadvantage of being sparse. The grid point's features have the advantage of being dense but they lack to provide the scale information.

In our previous work [9], we found that using dense features is much beneficial than using sparse features for object categorization, thats why in this study we have used dense features. For more information on decision to use dense regular grid (last image of figure 1) instead of interest points, refer to [10] in which authors have provided a comparative evaluation of both techniques. Our feature extraction strategy is most similar to [11, 12] where dense features are computed at grid points with a spacing of 10 pixels. The features are computed at radii of 4,8,12 and 16 pixels. Using these features impressive results have been achieved for image categorization and scene classification, though in this work we use them for object detection.

3 Approach

We extract SIFT features from an intensity image. SIFT feature are extracted at grid points as shown in figure 2, where the grid size is 10x10. Features are extracted at a radius of 4, 8, 16, 24 and 32 pixels around the grid points. A typical object and a non-object point along with its multi scale range is shown in figure 2. Once the features are extracted they are clustered into a codebook of size 200. The clustering is performed by k-means clustering of the features. [1]. The Codebook size of 200 has been used widely in previous works [11, 13] and has been proven to give optimal results. Once the features in an image are quantized, we have used a sliding window based approach, where each window is represented by the histogram of features contained with in that window (the bag of words model). The size of the window over which histograms are computed is not fixed (since some images are vary large for example faces and some very small for example cars). We have computed histograms over the windows equal to the size of the cell at pyramid level 3 [13] ,with a spacing of 50 pixels.

Typical object detection approaches requires background images to train the classifier for any certain object. The idea is to extract histograms from the object and from the background images and train a classifier using these object's and non-object's features. We have not used any background images in this work. All the categories we used are annotated (bounding box), refer to figure 4. Once the bag-of-words are computed for each cell (belonging either to object or to non-object), they are fed into learning algorithm. We have used a boosting [14] learning algorithm to train classifiers. There are a lot of variants of boosting. We have used confidence rated Ada-boost as it appears to perform better than other versions of boosting for generic object categorization as illustrated in [15].

As mentioned, we extract features at different scales and train a classifier for each scale. Once classifiers are trained for different scales, their output is combined. Let C_i denotes the output of the classifier at scale $S(i)$, where $S = (4, 8, 16, 24, 32)$ and i range from 1 to N We consider four combination strategies as follows (P denotes the output of the combination strategy and $N = 5$):

- Scheme 1: We give equal weights to all classifiers (equal weighting), such that

$$P = \frac{1}{N} \sum_{i=1}^{N} C_i \qquad (1)$$

[1] Actual features clustered vary by experiment (refer to section 4 for more details)

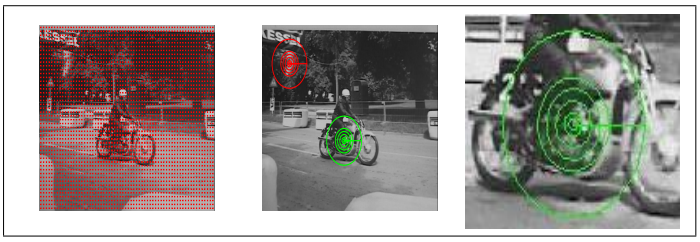

Figure 2: Left: Shows typical points across the image at which features are extracted, Middle: Typical scale at which features are computed (a typical object point is shown in green, whereas non-object point is shown in red), Right: Image point zoomed to illustrate feature extraction scales.

- Scheme 2: We give more weight to the classifier trained at higher scales and less weight to the one trained at lower scales.

$$P = \sum_{i=1}^{N} \frac{1}{2^{N-i+1}} C_i \tag{2}$$

- Scheme 3: We give more weight to the classifier trained at lower scales and more to the higher scales classifiers.

$$P = \sum_{i=1}^{N} \frac{1}{2^{i}} C_i \tag{3}$$

- Scheme 4: In this scheme we give maximum weight to the classifier in the middle, weights decreases linearly on both sides.

In order to detect the instance of an object, each cell in the testing image is classified as object and non-object with a certain confidence. Some of the classification results are shown in figure 3. The output of each classifier for each cell acts as a vote, in a Hough voting space. Votes are accumulated in a circular search Window with a radius of 3 around the center of the cell (represented by a Mean-Shift-Mode estimation [16]). The Mean-Shift modes that are above a certain threshold t_{det} are taken as detections of object instance. (Note. We have used different t_{det} for different categories). An object is deemed correctly detected if the overlap of the bounding boxes (detection vs ground truth) is greater than 70%.

4 Experimental Results

In this section, we present the results on three categories in TUDarmstadt database and one category in the Caltech database, [1]. TUDarmstat consist of approximately 100 images of each category i.e., cows, bikes and cars. Bikes and cars categories are pretty challenging as there are significant scale and viewpoint changes. The cows category is quite easy as compared to other two as there is not much

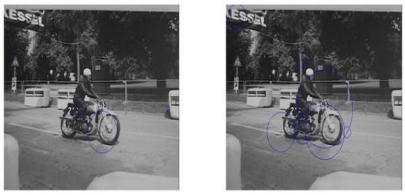

Figure 3: Left and Right Image: Illustration of the classifiers output trained at two different scales, Detected object's points are shown in red with blue circle's radius depicting its confidence, more the radius high is the confidence (note the difference in the output of classifiers trained at different scales).

scale and rotation. Faces category in the Caltech database is perfect for testing detection tasks as each image is rich in background. Statistics of training and testing images are shown in table 1. In Bikes category there were some images containing more than one bike (around 7 images) We have not incorporated multi object detection, so we have not used those images as part of testing or training. Databases are fully annotated (bounding box present around the object). Some sample images along with annotations are shown in figure 4. SIFT descriptors are computed on intensity images as described in the above section. No color or other information has been used.

4.1 Experiment 1

In the first experiments, we train a separate classifiers for the features belonging to each scale. Features are extracted from the training images and a separate codebook is formed for all the features belonging to each scale to train a classifier. Results

Table 1: Training/Testing Statistics

	Training	Testing
Bikes	30	67
Cars	30	70
Cows	30	81
Faces	100	350

Figure 4: Some examples of the training images along with their annotations.

are shown in table 2. We got the best performance for bikes at scale of 8, scale of 24 gave best performance for cars. Similarly scale of 16 and 24 gave best performance for cows and faces respectively. From the results it is difficult to infer as to which classifier performs best. The average results of each classifier trained for each scale is shown in figure 5. (obtained by averaging the rows in table 2) It is interesting to note that detection performance increases for higher scale features. We got the best performance for the classifier trained on features with scale of 24. Performance deteriorates as scale is increased from 24 to 32.

Table 2: Results for Experiment 1 (Percentage of correct detection)

Scale	Bikes	Cars	Cows	Faces
4	31	24	69	67
8	73	57	89	72
16	59	76	90	72
24	69	87	84	73
32	56	80	88	70

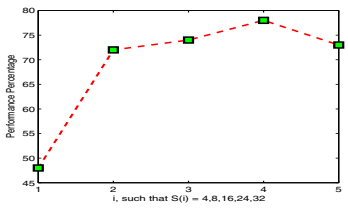

Figure 5: Percentage performance of the classifiers trained for the features extracted at different scale, y-axis is the Percentage performance, x-axis is the scale index i where $S(i) = \{4, 8, 16, 24, 32\}$

Table 3: Results for Experiment 3, Percentage of correct detection (for details regarding schemes refer to section 3)

	Bikes	Cars	Cows	Faces	Average
Scheme 1	60	93	93	77	81
Scheme 2	62	81	93	76	78
Scheme 3	64	94	88	75	80
Scheme 4	64	87	91	76	79

Table 4: Results for Experiment 2 (Percentage of correct detections)

Scale	Bikes	Cars	Cows	Faces	Average
All	89	76	95	77	84

4.2 Experiment 2

In the second experiment, we weighted the output of each individual classifier using different schemes as discussed in section 3. Various mixing strategies definitely boosted the results from the case when single classifiers were used (Experiment 1), but all of them gave almost similar results. We got the best performance when we simply mixed the output of the classifiers, 81%.

4.3 Experiment 3

In the third experiment, we trained a single classifier for all the scales. Features were extracted from training images and a single codebook was formed for all the features. We got the best average performance of 84% in this case. Results are shown in table 4. Some of the good as well as bad detection results are shown in figure 6.

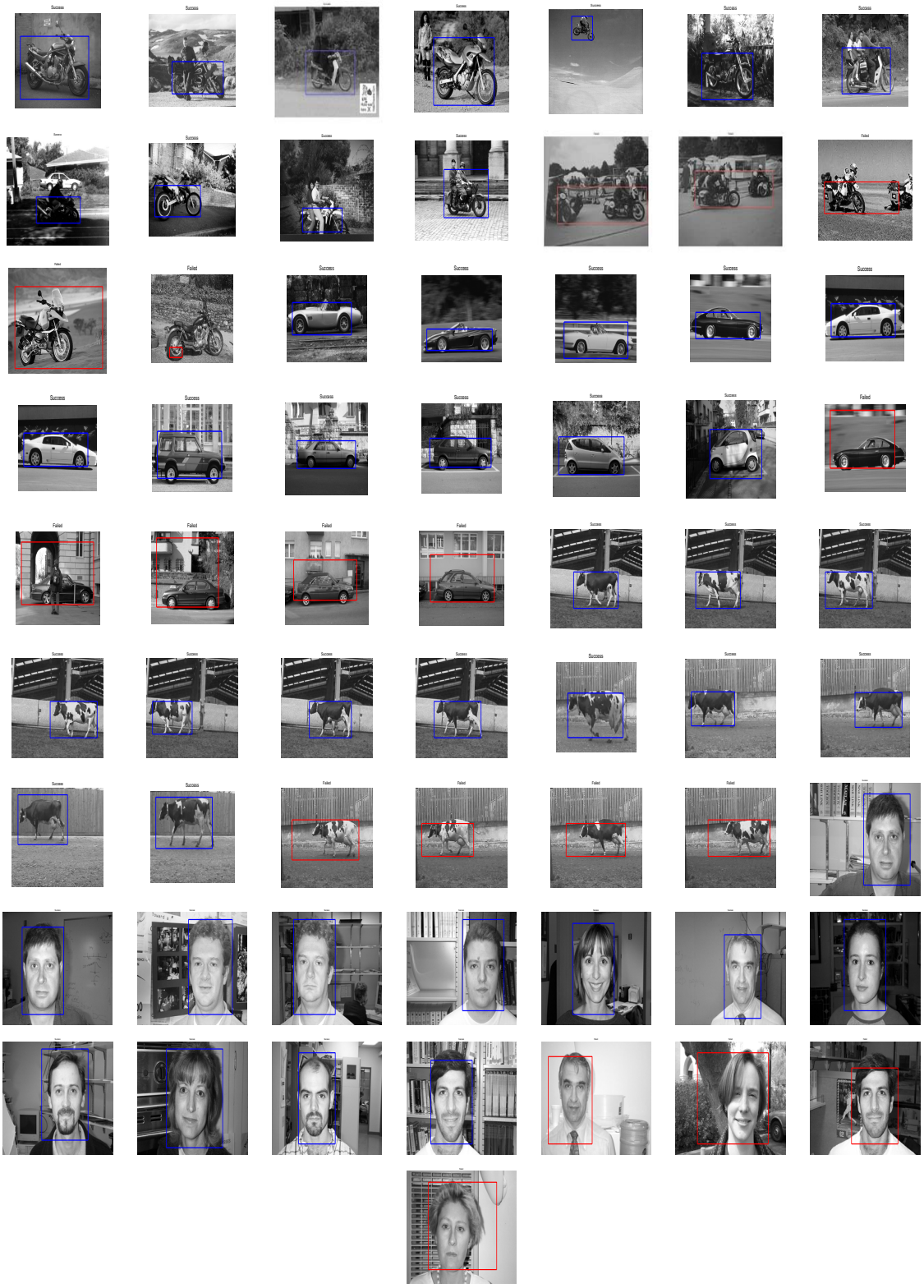

Figure 6: Some of the detection results on three categories. Objects in blue boxes are an example of right detection whereas red boxes are example of bad one.

5 Conclusions

In this paper we trained a cascade of classifiers on dense features computed at different scales around grid points. The output of these classifiers is combined using different weighting schemes to detect an object from the image. The classifiers trained on one particular scale did not reveal promising results. The combination of the classifiers boosted the results but no single scheme result out-performed others. On the other hand classifiers trained on all the features gave us the best performance, which shows that the different features extracted at different scales are required for efficiently detecting an object and the learning algorithm (boosting here) can do the best job of selecting which feature is appropriate and its scale accordingly. A considerably better approach would be to search for all possible combinations of weights in which classifiers can be combined (different schemes we discussed are just four possibilities out of a huge number), we are currently working on it.

References

[1] "The pascal object recognition database collection," 2005. [Online]. Available: http://pascallin.ecs.soton.ac.uk

[2] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *IJCV*, 2005.

[3] A. Opelt, A. Pinz, and A. Zisserman, "A boundary fragment model for object detection," in *ECCV*, 2006.

[4] J. Shotton, A. Blake, and R. Cipolla, "Contour based learning for object detection," in *ICCV*, 2005.

[5] A. Torralba, K. Murphy, and W. Freeman, "Sharing visual features for multiclass and multiview object detecion," *IEEE PAMI*, 2007.

[6] M. Vidal-Naquet and S. Ullman, "Object recognition with informative features and linear classification," in *CVPR*, 2003.

[7] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman, "Discovering objects and their location in images," in *ICCV*, 2005.

[8] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid, "Groups of adjacent contour segments for object detection," *IEEE PAMI*, 2007.

[9] N. Zaidi and D. Suter, "Cascade of classifiers trained on sparse and dense interest points for object categorization," in *submitted to IVCNZ*, 2008.

[10] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *CVPR*, 2005.

[11] A. Bocch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *ACM CIVR*, 2007.

[12] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification using a hybrid generative/discriminative approach," *IEEE PAMI*, 2007.

[13] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features, spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.

[14] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *The Annals of Statistics*, 2000.

[15] N. Zaidi and D. Suter, "Confidence rated boosting algorithm for generic object detection," in *ICPR*, 2008.

[16] D. Comaniciu and P. Meer, "Mean shift: A robust approach towards feature space analysis," *IEEE PAMI*, 2002.