Computer Science and Artificial Intelligence Laboratory

Technical Report

MIT-CSAIL-TR-2005-046 July 7, 2005
AIM-2005-023
CBCL-254

Boosting a Biologically Inspired Local Descriptor for Geometry-free Face and Full Multi-view 3D Object Recognition

Jerry Jun Yokono, Tomaso Poggio

massachusetts institute of technology, cambridge, ma 02139 usa — www.csail.mit.edu

massachusetts institute of technology — computer science and artificial intelligence laboratory

Boosting a Biologically Inspired Local Descriptor for Geometry-free Face and Full Multi-view 3D Object Recognition

Jerry Jun Yokono & Tomaso Poggio

AI Memo 2005-023 July 2005
CBCL Memo 254

©2005 massachusetts institute of technology, cambridge, ma 02139 usa — www.csail.mit.edu

Abstract.

Object recognition systems relying on local descriptors are increasingly used because of their perceived robustness with respect to occlusions and to global geometrical deformations. Descriptors of this type -- based on a set of oriented Gaussian derivative filters -- are used in our recognition system. In this paper, we explore a multiview 3D object recognition system that does not use explicit geometrical information. The basic idea is to find discriminant features to describe an object across different views. A boosting procedure is used to select features out of a large feature pool of local features collected from the positive training examples. We describe experiments on face images with excellent recognition rate.

1 Introduction

In the last decade, many object recognition systems have been proposed both at the level of categorization [4][5][23][24][25][26] and of identification of specific objects [8][9]. Recent reports describe good performances under various conditions such as different illuminations and clutter backgrounds. Lowe [8] developed an object identification system that works well in cluttered scenes and achieves rotational and scale invariance by using a unique local descriptor called "SIFT" inspired by Edelman and Poggio [18]. Matching is performed by efficient nearest neighbor search and by generalized Hough transform followed by solving for the affine parameters. The system can learn an object model from a single image. Inspired by his work, using local features for object recognition becomes trend and several recognition systems combined with statistical learning have been proposed [21][22].

Wallraven [21] proposed local kernels for SVM to learn the object model from multiple image samples. Their system practically provides good performance while the kernel turns out to be having theoretical problems (the kernel is actually not positive definite). Their kernel takes an average over the max correlation (score) between local features and it causes the features less-distinctive. We solve this problem by taking every max correlation value as one feature and select good features out of the huge feature pool. We show later how distinctive features are selected during the learning procedure. Csurka [22] introduced "bags of keypoints" to object recognition. Their system computes SIFT local features and clustering technique was used to build "visual vocabulary" and an SVM classifier is trained to classify each category such as faces, cars. The input features to SVMs are the number of occurrences of each key feature. Confusion matrix was used to evaluate the performances with relatively small set of negative test examples. Our report here differs from their report in that we are exploring the level of full-multiview object identification and therefore, we do not perform clustering since we found clustering makes each local feature less-distinctive. Additionally, we use max correlation values for classified feature while they use binned histograms where features are thresholded using Euclidian distance. In that sense, we do not use any heuristic and those thresholds are learned for each feature during the learning. Moreover, we consider object detection in the real world clutter environment and so we prepared great number of negative test images to examine the system by plotting the ROC curves.

Another difficulty of object recognition is that due to the viewpoint change, local patterns change drastically, especially for the rigid objects which causes recognition difficult. In this report, we show how viewpoint-robust local features are selected during the learning procedure which none of the above authors have focused on.

As we show later, our system can also be used for face identification task. Although there are several works in the literature which used local features and AdaBoost classifiers for the face identification [27] [28], their performances are limited because of the geometry constraints on the face. Main advantage of our system is that our system is invariant to change in position and rotation (both in plane and in depth) of the face.

In [1], we have evaluated the performance of local descriptors based on sets of oriented Gaussian derivatives which are the filter responses biologically found in human's primary visual cortex. We have performed the comparison in terms of criterion of selectivity and invariance by seeing how local patterns change through various transformations and shown that our Gaussian descriptors are robust against affine changes. We also implemented a simple recognition system similar to Lowe's that can learn object model from a single image. We achieved good recognition ability even if objects are partially occluded and even with different illuminations. The system uses the same threshold for all the features to find one to one correspondences of the local features followed by homography estimation. It became clear that some local features are better at discriminating a specific object. We describe here a system capable of finding discriminant and specific features while tuning the threshold for those features. Motivated by Morgenstern and Heisele [11], our system uses correlation features computed from a "feature pool". Our system differs from the one in [11] in that we use rotation invariant Gaussian derivative based features at the corner-like points detected in the image. In addition, the recognition does not use any explicit geometrical information. Of course, objects might have different geometry depending on the viewpoint (see [11]). Therefore, in this report, we focus on how oriented filter helps geometry-free recognition. We also report the face recognition performance on the well-used ORL database. Comparisons with two other statistical classifiers, SVM and boosting, are also described. The main contribution of this paper is (1) to propose a framework that can effectively select viewpoint-distinctive features for full-multiview object recognition, (2) to propose a simple

yet powerful framework for integrating full object views into just one model (3) to show extensive and excellent experimental results on face recognition and full-multiview object recognition.

In section 2, we review a simple-cell type local descriptor based on the Gaussian derivatives, implemented using "steerable filters" [6]. Section 3 overviews the system. We describe the main experiments in section 4 and 5. Section 6 concludes the paper.

2 A simple-cell type local descriptor based on sets of oriented Gaussian derivatives

2.1 Steerable filters

Gaussian derivatives are widely used filters with spatial orientation selectivity as well as frequency selectivity. The steerable filter [6] response for the *n*th order Gaussian derivative $G_n(\theta)$ to an arbitrary orientation θ is, given the Gaussian distribution *G:*

$$G = e^{\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

$$G_1(\theta) = \cos(\theta)G_1(0^o) + \sin(\theta)G_1(90^o)$$

$$G_2(\theta) = k_{21}(\theta)G_2(0^o) + k_{22}(\theta)G_2(60^o) + k_{23}(\theta)G_2(120^o)$$
$$k_{2i}(\theta) = \frac{1}{3}\{1 + 2\cos(2(\theta - \theta_i))\}$$

$$G_3(\theta) = k_{31}(\theta)G_3(0^o) + k_{32}(\theta)G_3(45^o) + k_{33}(\theta)G_3(90^o) + k_{34}(\theta)G_3(135^o)$$
$$k_{3i}(\theta) = \frac{1}{4}\{2\cos(\theta - \theta_i) + 2\cos(3(\theta - \theta_i))\}$$

where $k_{in}(\theta)$ is the coefficient of the basis.

We use Gaussian derivatives up to the third order, with four orientations and three widths: 1,2, and 4. The vector length of the jet descriptor associated with a location in the image is 3x3x4=36. The descriptor can be made more powerful by combining the neighboring four jets, which are five pixels away from the center pixel. In this case the length of the local descriptor is 36x5=180. The local descriptor used in the experiments is shown in Figure 1. See [1][2][3] about the detail of the descriptor.

2.2 Rotation invariant local descriptor

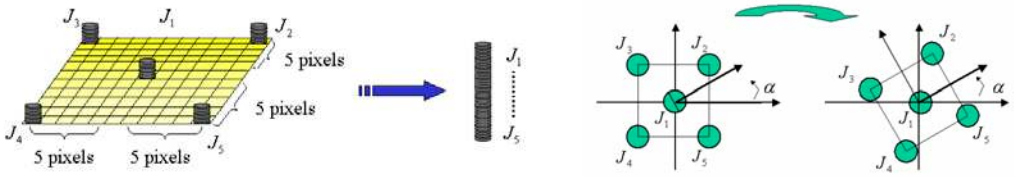

Figure1. Gaussian derivatives up to the third order with four orientations and three scales are computed and expanded using four neighboring pixel locations $J = \{J_1, J_2 \cdots, J_i\}$,

where $J_i = \{G(\lambda, \theta, \sigma)\}$, $\lambda : order, \theta : orientation, \sigma : GaussianWidth$

Rotational invariance of the descriptor is achieved by computing a gradient orientation at a center pixel location, and normalizing the jet by "steering" the filter responses. The gradient orientation can be computed

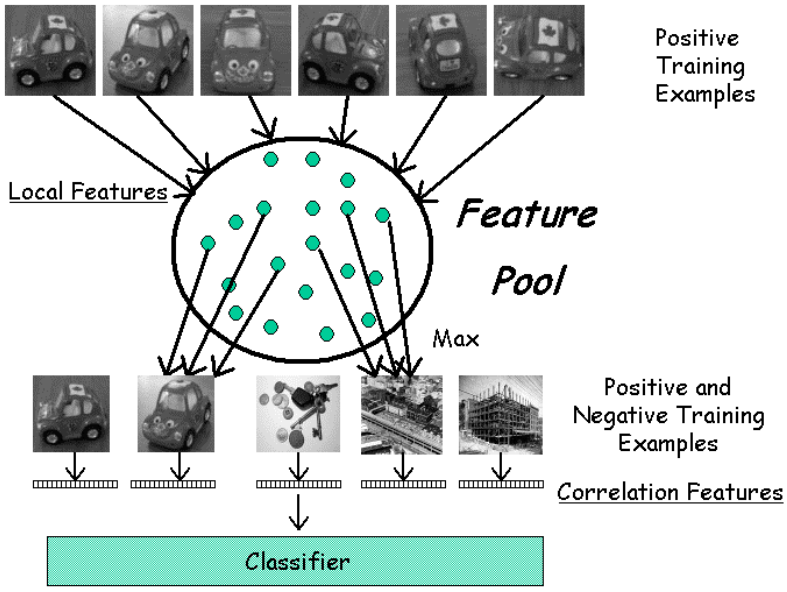

Figure 2. System Overview: Local features are collected from positive training examples and maximum correlations are computed to make correlation features fed into a classifier. Note that when considering multiclass recognition, different features are used to make feature pool.

from the first order derivative responses of our descriptor.

$$\alpha = atan2(Gx(\sigma), Gy(\sigma)) \quad (2)$$

where *Gx* and *Gy* are the first order derivatives of the Gaussian and σ is the largest width of the Gaussian (σ =4), respectively. Once the main orientation of the center pixel is computed, the filter responses at the center pixel are steered. Four neighboring pixel locations and their filter responses are also steered to create a rotational invariant descriptor. The steered *n*th order Gaussian derivative is computed by following:

$$G_n = G_n(\theta_i + \alpha), \quad \theta_1 = 0, \theta_2 = \frac{pi}{2}, \theta_3 = pi, \theta_4 = \frac{3pi}{2} \quad (3)$$

where α is the main orientation at the center pixel.

3 Boosting local features without geometric information

3.1 Correlation Features

The basic idea of our approach -- motivated by Morgenstern and Heisele[11] -- is to collect local features from the positive training images and thus create a feature pool. The pool features represent a "dictionary" of features that describes the object. When considering multi-class recognition, a feature pool is created for each class with a different set of local features. Local features based on the Gaussian derivatives are computed on corner-like points detected by Harris measure [7]. Once the feature pool is created, all the positive and negative training images are used to compute "correlation features": for each feature in the pool, the maximum of the (normalized) correlation over all the local features in a sample is computed for each training sample. Therefore, if N features are in the pool, every training image has a N-dimensional feature vector. We call this vector a "correlation feature" and this is the input to the classifier. An overview of the system is shown in Figure 2.

We can use any kind of classifiers such as Support Vector Machine and boosting. Our expectation is that even without geometric information, the descriptors are sufficiently discrimnant. The correlation features might be

Discrete AdaBoost Algorithm (Freund & Schapire [14])

1.Initialize weights $w_i = 1/N, \quad i = 1,2,...,N$, where N is the number of samples.

2. Repeat for $m = 1,2,...,M$:

(a) Train the weak classifier (ex. stump) $f_m(x) \in \{-1,+1\}$ using weights w_i.

(b) Compute error $err_m = E_w[1_{(y \neq f_m(x))}], c_m = \log((1 - err_m)/err_m)$.

(c) Update weights by $w_i \leftarrow w_i \exp[c_m \cdot 1_{(y_i \neq f_m(x_i))}], i = 1,2,...,N$ and normalize

3. Final strong classifier output is

$$sign[\sum_{m=1}^{M} c_m f_m(x)]$$

Gentle AdaBoost Algorithm (Friedman, Hastie & Tibshirani [12])

1.Initialize weights $w_i = 1/N, \quad i = 1,2,...,N$, where N is the number of samples.

2.Repeat for $m = 1,2,...,M$:

(a) Fit the regression stump $f_m(x)$ by minimizing a weighted squared error

(b) Update the function by $F(x) \leftarrow F(x) + f_m(x)$

(c) Update weights by $w_i \leftarrow w_i e^{-y_i f_m(x_i)}$

3.Final strong classifier output is

$$sign[F(x)] = sign[\sum_{m=1}^{M} f_m(x)]$$

common features across the viewpoint of the object or distinctive features for a specific viewpoint. Our report differs from [11] in that we detect corner-like points for the local features and do not use geometric information. The advantage of using corner-like points is that those points usually have high information [10] (we do not need local features on the white background as Morgenstern [11] reported on "bird" object). At run time local features on the interest points are used to compute a correlation features which is fed into the previously trained classifier.

3.2 SVM, Discrete AdaBoost, Gentle AdaBoost

SVM and boosting are successful classifiers in a host of real-world applications [4]. AdaBoost was originally proposed by Freund and Schapire [13][14]. Later, Friedman et. al. [12] proposed a modified version of AdaBoost that uses additive regression as a weak learner and adaptive Newton steps for the optimization. They called the original AdaBoost as Discrete AdaBoost and claimed that their new Gentle AdaBoost often outperforms Discrete AdaBoost. Both algorithms are listed in the Algorithm Box. As can be seen, Gentle AdaBoost uses real-valued regression rather than the *{-1, +1}* of Discrete AdaBoost.

3.3 Boosting local features

In terms of performance, SVM and boosting (and as a matter of fact also square-loss regularization) are usually quite similar. An advantage of using boosting is that it might effectively perform feature selection during the learning. For instance, if the system holds initial 6000 features in the feature pool, it is necessary when using SVM to compute at run time all the correlation values for all the features. On the other hand, boosting may selects fewer features, say 200, thereby considerably speeding up computation at run time. Since we are effectively using a dicision stump (binary split decision tree) as weak classifiers, the learning procedure tunes the threshold for each feature. Good features can be selected by taking the minimum error of the features. In the experiments, we use both SVM and boosting and compare the results.

4 Face identification

4.1 CBCL datasets

4.1.1 Datasets

In the first experiment, we performed the face identification on our CBCL face database. The CBCL face database contains faces of 10 people with approximately 200 images per person. It has both male and female face images collected from various ethnic subjects. Sample images are shown in Figure 3. As shown in these images, there are variations in illuminations, face positions (not aligned to center), slight scale changes, and pose changes up to about 45 degrees of depth rotation and up to 30 degrees of in-plane rotation. Images are 70x70 gray value.

Figure 3. Sample images from CBCL face database. This database contains 10 people with approximately 200 images per person. As can be seen, it has various changes in illumination, scale, pose, and facial expression.

4.1.2 Experiment setup

For each run, N images are randomly chosen as training images and remaining images are used for testing. Gaussian derivative based local features are computed on the corner-like points detected by the Harris corner detector for every training image. Approximately 50 points per image are detected in our experiments. These local features from positive examples are collected to build a feature pool. For instance, when we use 30 training images, approximately 50x30 = 1500 features are in the pool. Then, all the positive and negative images are used to make the correlation feature vectors where each image represents approximately 1500-dimensional correlation feature vector. All the results are averages of 10 runs.

4.1.3 Classification methods

Correlation feature vectors from the positive training examples are used to train the supervised learning machines. As we mentioned in the previous section, basically any kind of classifier can be used in our system. We use SVM and Boosting classifiers and compare the results. Figure 4 shows the ROC curve when 30 images are used in training and 200 stumps for the Boosting. Linear SVM , RBF SVM, AdaBoost, Gentle AdaBoost are plotted in the figure. As shown in the figure, SVM classifiers are slightly better than the Boosting classifiers but not significant difference in performance. The advantage of using Boosting method is that the system can select 200 features out of huge feature pool (in this case, approximately 1500 features) that reduces computational cost.

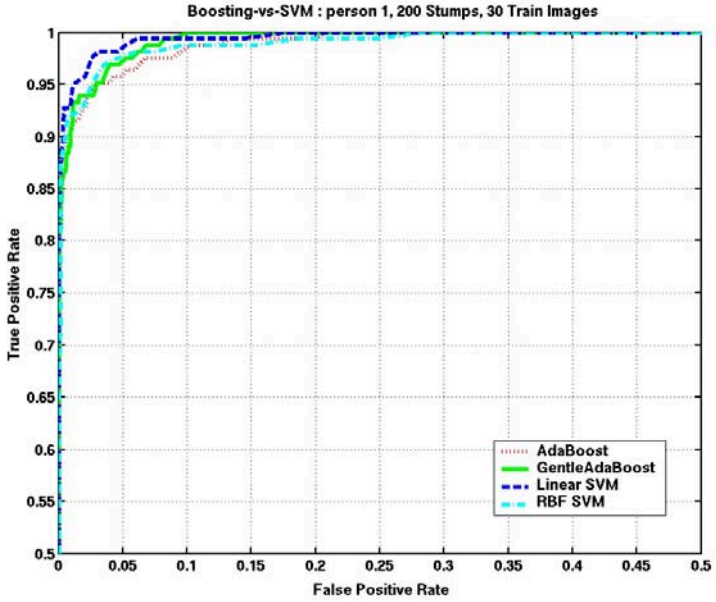

Figure 4. ROC curve comparing performance of SVM and Boosting. 30 training images are used and as can be seen, SVM works slightly better than Boosting. Advantage of using Boosting is that it can effectively select features out of all the features.

4.1.4 Number of training images

We conduct experiments by changing the number of training images.
ROC curve indicating the classifier performance is shown in Figure 5. Due to the difficulty of the database, 10 images are not enough for the high performance. If we use 50 images for the training, recognition is almost perfect.

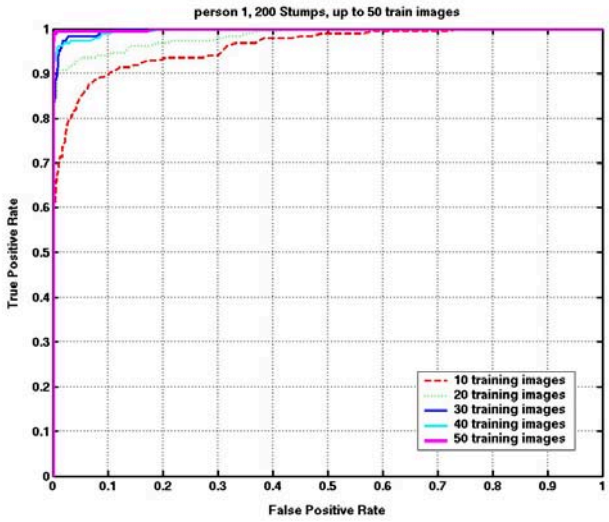

Figure 5. ROC curve of a boosted face classifier using 200 stumps. The performance increases, as more training images are available. When 50 images are used for training, result is perfect.

4.1.5 Number of weak classifiers

We also conduct experiment how number of weak classifiers of the boosting affects the performance. As expected, if we use more weak classifiers, performance increases significantly. When 30 images are used for the training, a classifier using 1000 stumps is almost perfect as shown in ROC curve.

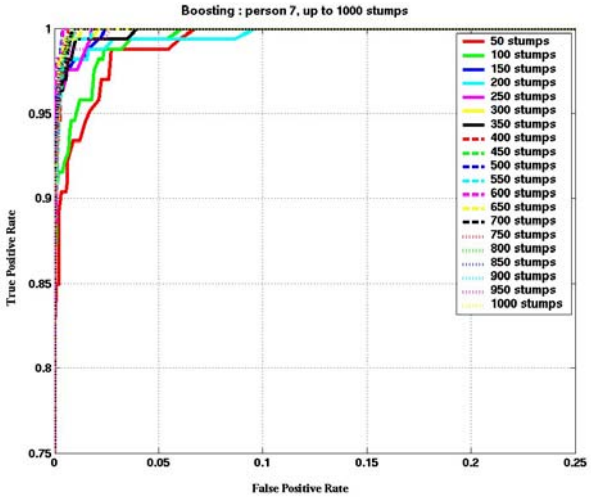

Figure 6. ROC curve as a function of number of weak classifiers. When 1000 stumps are used, performance is even better than SVMs in Figure 4.

4.1.6 Multiclass Recognition

We also perform multiclass face recognition. The system has to identify the specific person out of 10 subjects. We trained 10 one-vs-all binary classifiers and take the maximum output. Results are shown in Table1. As shown in the table, if we have more images available for training, there is no significant difference in performance between SVMs and Boostings. However, the boosting performance is worse than SVMs when less than 10 images are used for training. Figure 7 shows recognition rate of each class. Person 5 and person 6 are difficult due to the much variation in illumination and depth rotation.

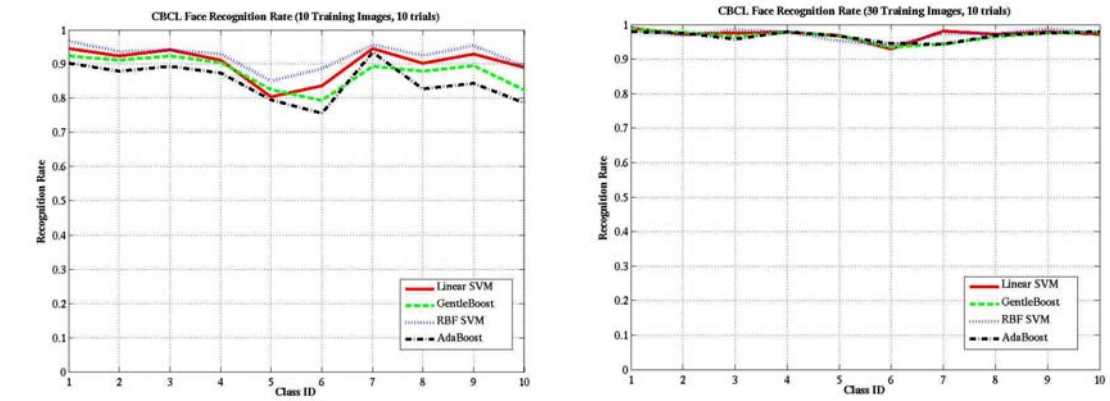

Figure 7. Performance of each person. Figures show average recognition rate on 10 runs when 10 images are used for training (left) and 30 images are used for training. Some people are more difficult than the others to recognize. More the training images are available, performance of SVM and boosting getting closer.

Classification Methods / Number of training images	5 images	10 images	20 images	30 images
Linear SVM	82.1 %	90.3 %	95.0 %	97.2 %
RBF SVM	84.3 %	92.4 %	95.9 %	97.2 %
AdaBoost (200 stumps)	60.8 %	84.9 %	93.8 %	96.7 %
GentleBoost (200 stumps)	68.5 %	87.7 %	93.8 %	96.7 %

Table 1. Recognition performance on CBCL face database.

4.2 ORL face database

We also perform multiclass face recognition on the Cambridge ORL face database. This database is an often used benchmark test set for face identification. The images are 112x92 pixels and have minor variation in facial expression and scale, and pose. Example images are shown in Figure. It contains 40 different subjects with 10 images each. Although this database is relatively easy, compared to the CBCL database (since the faces are aligned in the center of the image and subjects are placed on the uniform background), it is interesting to compare the results with other techniques reported in the literature. When 5 images are used for training and testing, our previous one-shot-learning system [2] achieved 97.5% recognition rate which is state-of-the-art performance. We tested our new approach to this data set. For each subject, N randomly chosen images are used for training and remaining 10-N images are used for the testing. Thus a total of Nx40 images are used for training and (10-N)x40 are used for testing. The number of local features depends on the

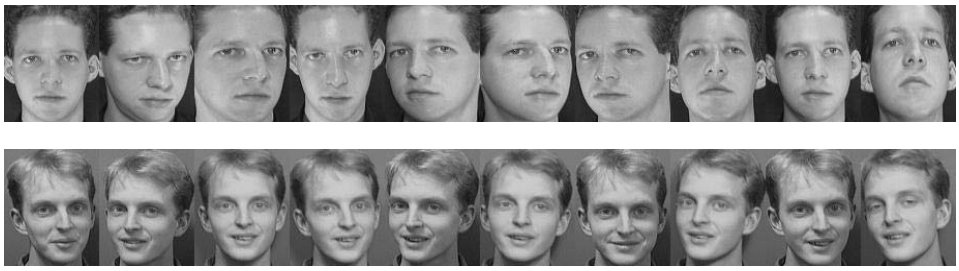

Figure 8. Sample images from ORL face database. The database contains 40 subjects with minor variation of illumination, scale, and pose. Faces are aligned to the center of the image and uniform background is used. Each subject has 10 face images and randomly split set are used for training and testing.

image but approximately 70 features are found in an image. For instance, when 5 images are used for training, there are 70x5=350 features are in the feature pool. Maximum correlations are computed for all the features. We trained the one-vs-all SVM classifiers and boosting classifiers for all the 40 people. In the testing, all the test data are classified into one of the 40 categories. The input feature vector is classified by all the 40 classifiers. The classifier with maximum value provides the final decision. We run the experiment 30 times since the result is slightly different when different images are chosen as training and testing and considered the average result.

Table 2 shows the results with other techniques such as Eigenface[15], SOM+CN[15], and ARENA[16]. As we can see from the table, the result is excellent: even with only 1 training image, the recognition rate of the SVM classifier is over 80%. When we use 3 images for training, performance is almost perfect. When 5 images are used for the training, 12 out of 30 runs show 100% recognition rate. We should note here that boosting performs poorly compare to SVM. However, when the number of training images are more than 10, the performance of boosting increases significantly. We already showed the result on our own database containing more number of examples.

# of Training images	1 image	3 images	5 images
Eigenface [15]	61.4 %	81.8 %	89.5 %
SOM+CN [15]	70.0 %	88.2 %	96.5 %
ARENA [16]	74.7 %	92.2 %	97.1 %
Linear SVM	84.4 %	96.5 %	99.3 %
RBF SVM	**84.5 %**	**96.8 %**	**99.3 %**
Gentle AdaBoost	74.7 %	74.8 %	78.6 %
One Shot System [2]	50.3 %	82.9 %	97.5 %

Table 2. Recognition performance on ORL face database

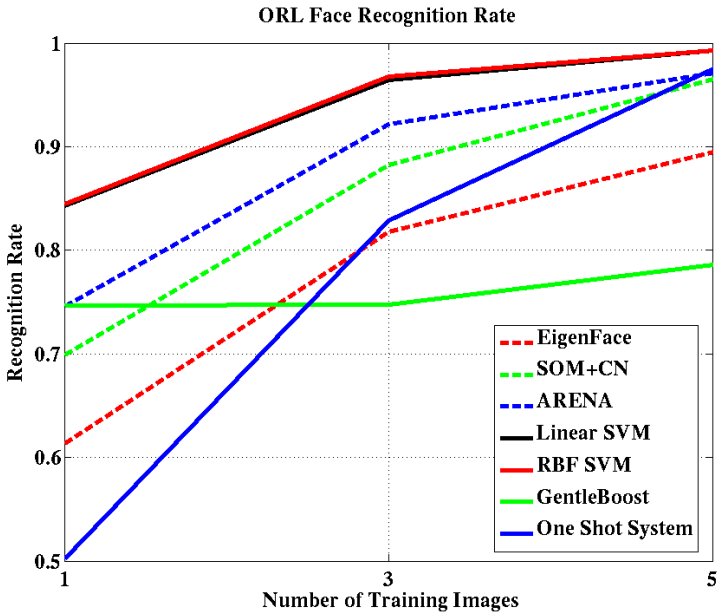

Figure 9. Face recognition performance when different number of images are used for training.

5 Full multiview 3D object recognition

5.1 Integrating multiview to one object model

- finding common features across the viewpoints and distinctive features to a particular viewpoint-

In the previous section, we applied our system to the face recognition task and showed high recognition performance on the ORL and CBCL face databases. The next challenge was to apply the system to multiview 3D object recognition. There are two major approaches for multiview 3D object recognition: 3D model based and 2D image based approaches. We use the latter approach where sequences of multiview images are used for training. One way to accomplish 3D multiview recognition from collection of 2D images is to use reference frames. In that method, a set of images separated by N degrees in depth rotation are chosen as references. For a planar object such as a painting, even if the viewpoint is changed 45 degrees, the images are still similar enough for detection. On the other hand, for a 3D object, 30 degrees interval might not be too much. In our approach, this problem is dealt with same way as framework described in the previous section. Local features are extracted from object images taken from the various viewpoint and good features are selected during the learning procedure. Those selected features are common features across the view or distinctive features of a certain viewpoint. Figure 9 shows one of the objects in our 3D object database. For this toy car object, 74 images (100x100) from the different viewpoint are used for the full multiview object training. As noted in the previous section, local features from the positive training examples are used to create a “feature pool” and “correlation features” are computed for all the positive and negative examples. Randomly cropped 4000 negative images are used as negative training examples. Some of the negative clutter scene images are shown in Figure 11. Other objects in our database are tested in the next sections.

5.2 Results

5.2.1 Toy Car Object

As described in the previous subsection, 74 positive images are used for training. A total of 7089 features are extracted from full multiview images of this object. During the training, a 7089-dimensional correlation feature is computed for each training sample. In the experiments, we also extract 15x15 rounded gray patches at the interest points to compare the performance of our rotation invariant gaussian derivative descriptor(RIGD). SVM classifiers are also trained and compared to boosting with N stumps. (N=50,200,500) For testing, 100 positive images and 9000 negative images are used.

All the positive test images are shown in Figure 12. We can see from the figure that test images are taken under different light conditions, and under significant changes in viewpoint and scales. Another difficulty here is that background where the object is placed also changes. Even though the test images are taken under different light conditions, results are good. The ROC is shown in Figure 13. For all the classifiers we train, RIGD (Rotation Invariant Gaussian Derivatives) descriptor shows better performance than the gray patch descriptor. If we use RIGD descriptor, 90% of the test images are recognized with extremely low false positive rate by the SVM classifier and boosting. With more than 200 stumps, boosting and SVM show almost the same performance. An advantage of the boosting is that it effectively performs feature selection during the learning. In this case, 200 features out of 7089 features are selected and shows almost the same performance. If more training samples taken under various environments were available, results might be even better. This recognition system achieves complete multiview object recognition in one model without considering reference frames nor even without geometric information. Moreover, due to the rotation invariance of the descriptor, basically, object images from any rotation, any viewpoint can be detected.

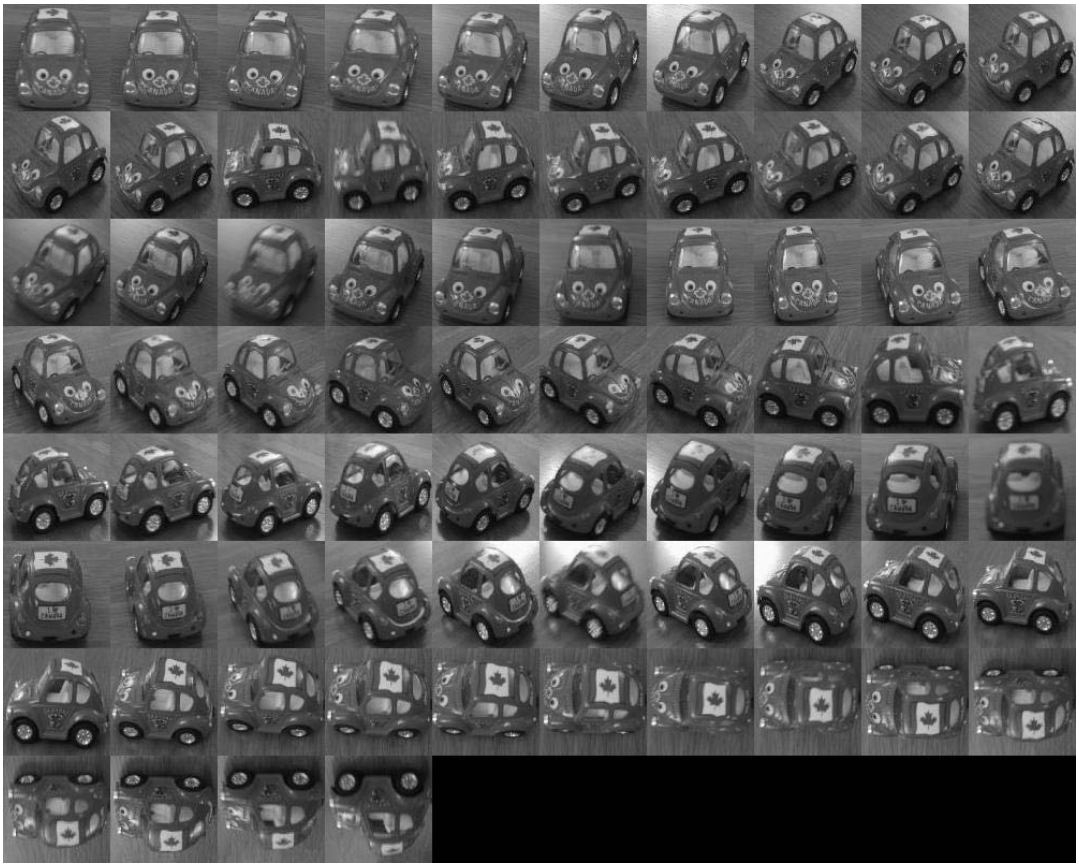

Figure 10. All the 74 training images of toy car object. Images are taken from all the viewpoints and used for the training of multiview recognition system

.

Figure 11. Negative samples: randomly cropped cluttered scene images are used for both training and testing.

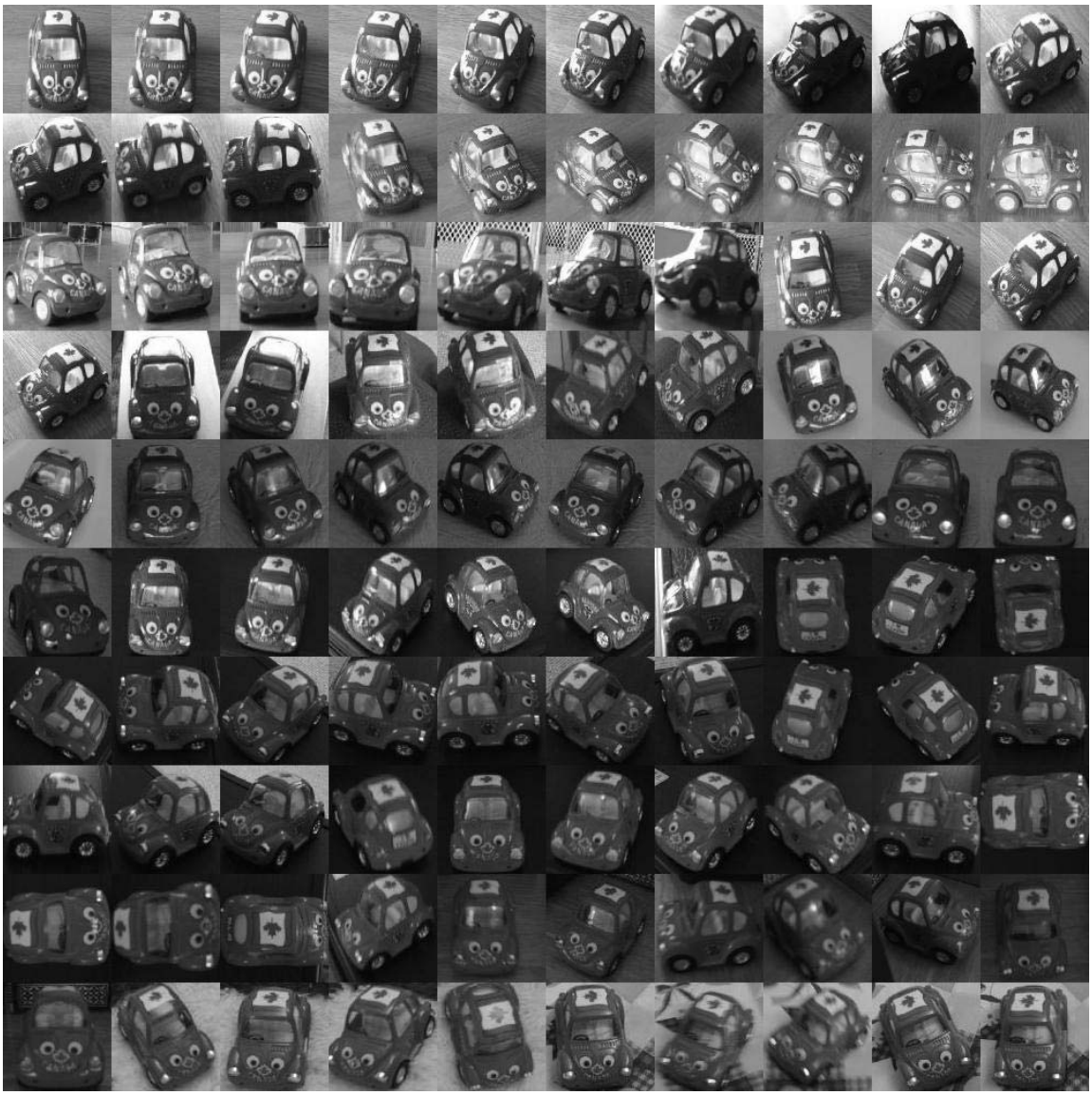

Figure 12. All the 100 test images used in the experiment. Note that images are taken under different viewpoints, different light conditions, different backgrounds from the training set. Some images have motion blurs and also changed in scales.

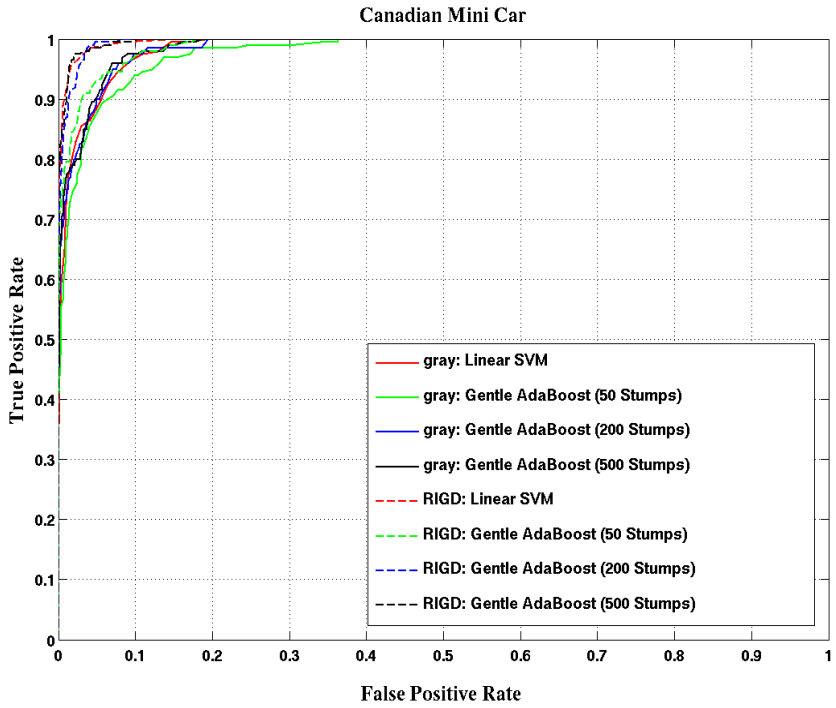

Figure 13. ROC of the toy car object. RIGD (Rotation Invariant Gaussian Derivatives) shows better curve than the gray patches for all the classifiers. With more than 200 stumps, SVM and Boosting shows almost the same performance. An advantage of Boosting is that 200 features out of 7089 features are selected during the learning.

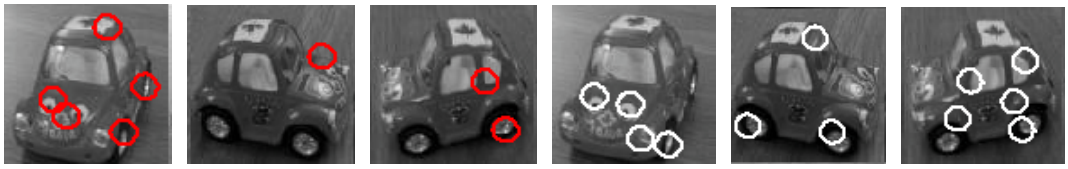

Figure 14. Some of the selected features by the Boosting procedure. Initially, 7089 features are in the pool. Note that different features are selected with gray patches (three left columns with red circles) and RIGD descriptor (three right columns with white circles). Diameter of the circle is 15x15pixels which is a support region of the gray patches. Wheels are selected for both descriptors.

5.2.2 “METRO” newspaper

We also tested different issues of newspaper with the same logo title on the top page.
We use Boston “METRO” newspaper. 22 images from 9 different issues are used for training. All the training images are shown in Figure 15. Our expectation was that the system could automatically find what the “METRO” is : common feature, “METRO” logo . For the testing, we use different issues of the newspaper as shown in Figure 16. As we can see from the figure, images are rotated in various directions and has scale changes and also some test samples are occluded. ROC curve is shown in Figure 17. The result of RIGD descriptor is excellent. Due to the rotation of the test images, gray patch descriptor shows much less performance. Figure 18 shows the selected features during the boosting procedure. Many of the selected features are found on “METRO” logo with overlapping allowing redundant representation.

Figure 15. All the training images of "METRO" Boston newspaper. 9 different issues of newspaper are used for training.

Figure 16. All the test images of Boston "METRO" newspaper. Those test images are all different issues of the newspaper from the training samples. The system successfully find the common features "METRO" logo in the training images and knows what the "METRO" is. Of course, if we use different label to each issue, the system can learn what issue it is.

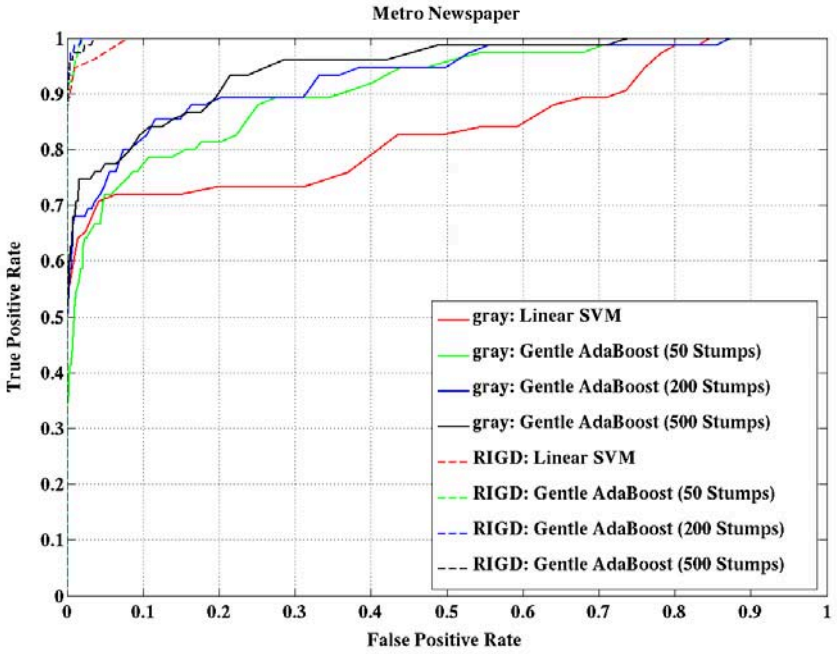

Figure 17. ROC curve of "METRO" newspaper. Performance of rotation invariant descriptor is almost perfect while gray patches fail to find correspondences. Both Linear SVM and Boosting get perfect results.

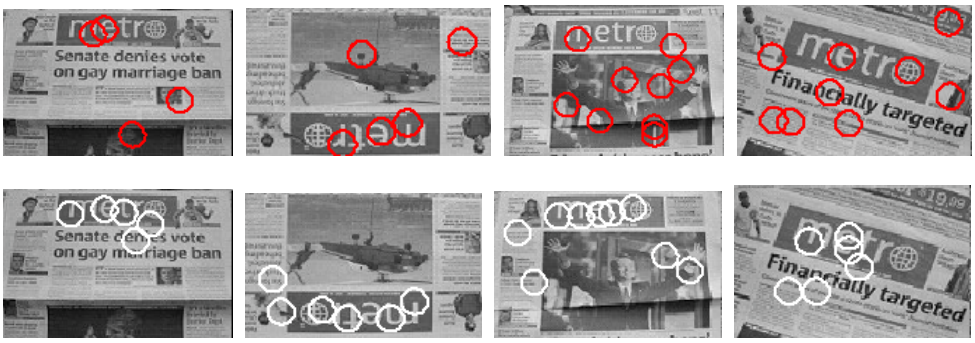

Figure 18. Examples of selected local features during the boosting procedure. Top row (red circle) indicates the gray patch features and low row (white circles) indicates the RIGD local descriptor. The diameter of the circle is the support region of the descriptor: 15x15 pixels. Note that how different features are selected for gray patches and the RIGD descriptor. Rotation invariant features are mostly selected in "METRO" logo.

5.2.3 Office Phone

To compare the results to Heisele and Morgenstern [11], we use the same office phone object. Experimental setup is the same as theirs. 34 positive and 4000 negative images are used in the training and 114 positives and 9000 negatives are used in the testing. All the training and testing images are shown in Figure 19 and Figure 20. As same as the previous objects, we can see the ROC curve in Figure 21, RIGD shows better results than the gray patches. The system shows slightly worse results than the "with-grid" results of Heisele and Morgenstern system. As shown in Figure 22, features selected by the Boosting algorithm are most in the

object parts, not in the backgrounds.

Figure 19. Positive training sequences. 34 images are taken from different viewpoints of the office phone. Local features are extracted from these images and good features are efficiently selected by the boosting framework.

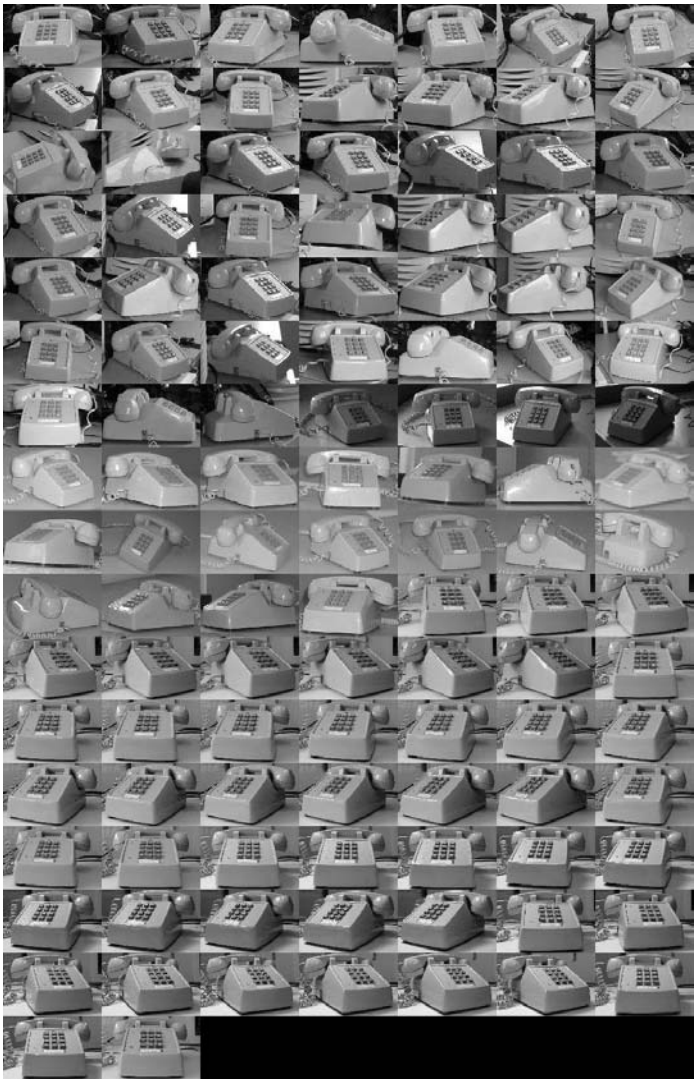

Figure 20. All the test images of the phone. Note that illuminations, scales, viewpoints, backgrounds are different from the training images.

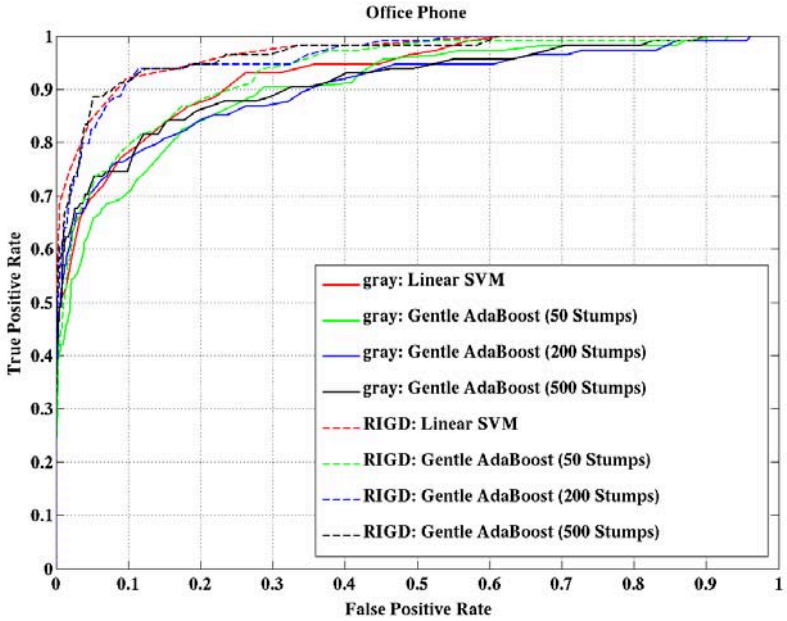

Figure 21. ROC curve of the office phone object. RIGD descriptor shows better performance than the gray patches.

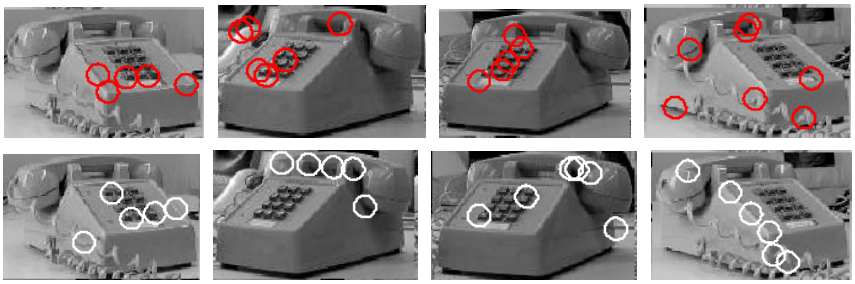

Figure 22. Some of the features selected by the boosting algorithm. Notice that different features for gray patches and RIGD descriptor are selected by the algorithm. Also note that most of the selected features are from the object and not from the background.

5.2.4 AIBO Latte

One of the advantages of geometry free recognition system is that even though the object parts changes their positions, still a chance to detect the object. We use Sony AIBO Latte miniature toy: the quadruped type toy that we can change its positions by moving its legs and a head. For the training images, AIBO is posed standing and take pictures from multiple viewpoints. Test images are taken by changing its position different such as sitting, lying as in the Figure 24. The result is interesting where gray patch descriptor shows better curve than the RIGD descriptor. This is due to the rotation invariance of the descriptor and simplicity of the object. In the expense of the rotation invariance, RIGD descriptor shows similar response on many points. For instance, local features along the edges around its face shows similar patterns when they are rotated. However, overall the system shows good curve. We will extend this system by using orientations of the descriptors in the future to solve this problem.

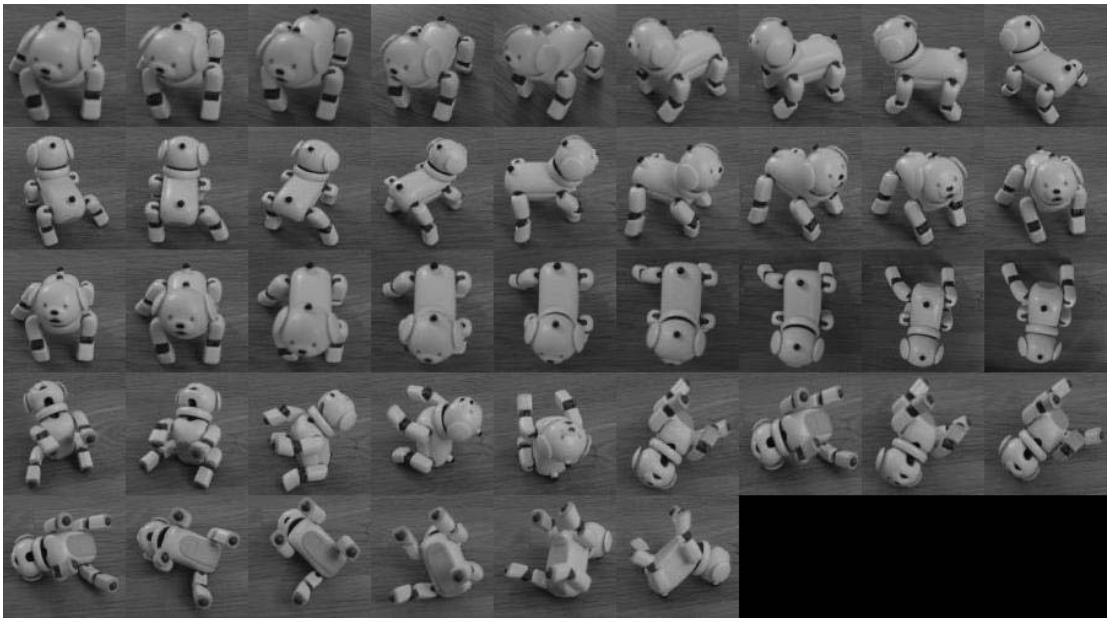

Figure 23. All 42 training images of AIBO Latte ERS-310. AIBO is placed with standing posture and complete full views.

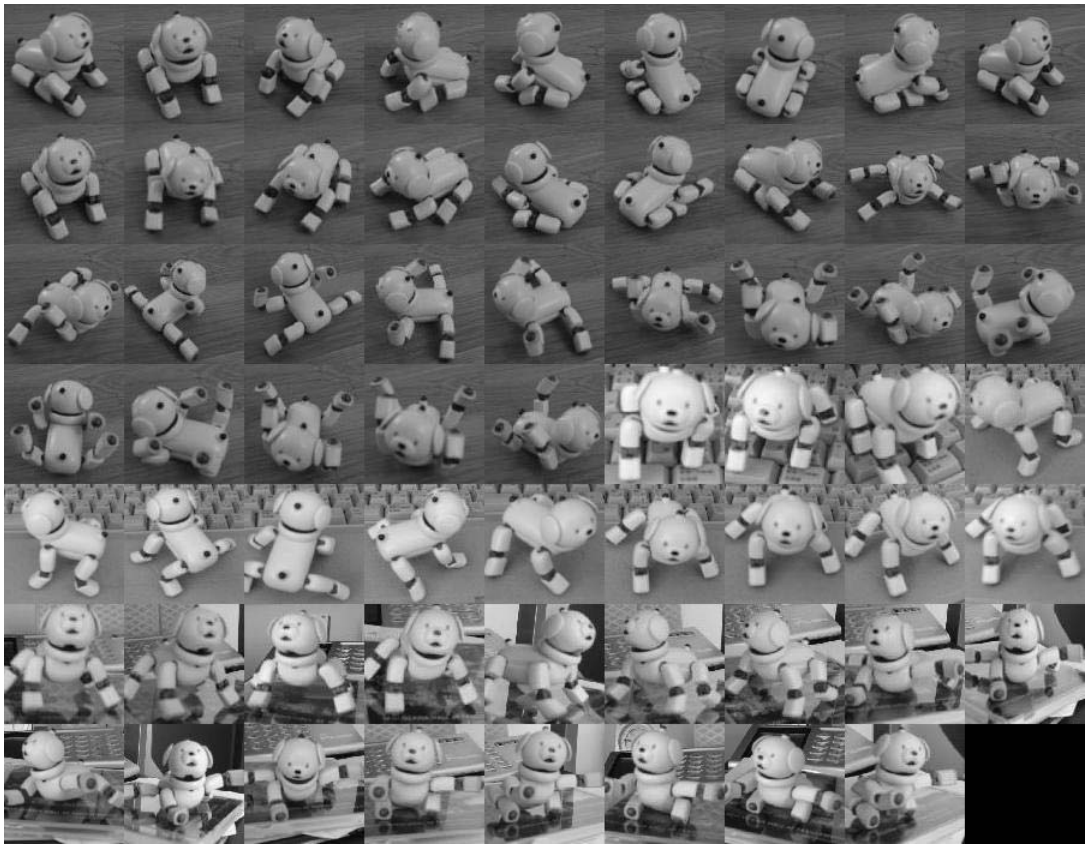

Figure 24. All the test images of AIBO Latte ERS-310. AIBO is placed with different postures from the training set and also in the different backgrounds.

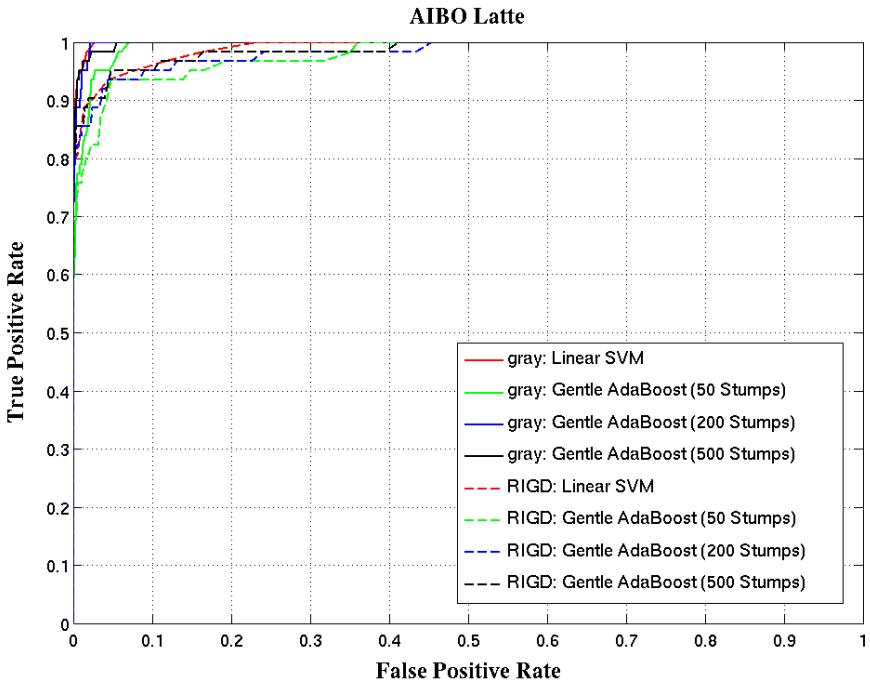

Figure 25. ROC curve of AIBO object. Interestingly, gray patch descriptor shows better results than our RIGD descriptor. This is explained by the fact that rotation invariance

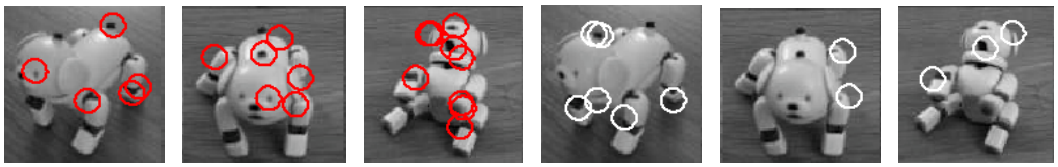

Figure 26. Some of the selected features out of 2900 features in the pool by the boosting algorithm. First three columns show selected gray patches (red circle) and last three columns show selected RIGD descriptors (white circle).

6 Conclusion

In this paper, we do not exploit any geometric constraints for the object recognition. Our system uses local features developed in [2] and statistical learning classifiers. Local features based on sets of oriented Gaussian derivatives are efficiently implemented by "steerable filters". Positive training images are used to extract local features and build a "feature pool". For each feature in the pool, a maximum correlation is computed to make a "correlation feature" to be trained by SVM and boosting. We applied our system to face recognition and multiview 3D object recognition. ORL and CBCL face database showed excellent results. Even without geometric information, the system achieves state-of-the-art performance. A motivation for applying the system to multiview object recognition is that the system effectively integrates automatically multiview object models into one model by collecting sufficient number of distinctive and object specific local features from the training images. Unlike [2] where each local feature has the same priority during the matching, boosting algorithm selects specific features out of a large feature pool while tuning the threshold of matching each feature. Moreover, as we showed by testing the different issues of a newspaper, the system can find common

features in the training images. As for the 3D objects such as a toy car and an AIBO, the system finds distinctive features in a certain viewpoint as well as the common features across viewpoints.

If we consider geometric constraints, we expect that the system is able to learn from a smaller number of examples. We are currently working on an object recognition system that should be capable of learning from few examples using geometric constraints.

Acknowledgements

This research was done during the visit to CBCL, MIT with the support of Sony Corporation. Author would like to thank to Bernd Heisele and Lior Wolf for very useful suggestions and Pascal Paysan and Kevin Chang for preliminary works. J.Y. would like to thank to Christian Morgenstern for providing 3D object database. J.Y. also thanks his son Ryo for lending his favorite toys for the experiments.

References

[1] Jerry Jun Yokono and Tomaso Poggio,"Evaluation of Sets of Oriented and Non-Oriented Receptive Fields as Local Descriptors", AI Memo No.2004-007, CBCL Memo No.237, MIT Center of Biological and Computational Learning, 2004
[2] Jerry Jun Yokono and Tomaso Poggio. "Oriented filters for Object Recognition: an empirical study". Proceedings of the IEEE Conference on Face and Gesture Recognition (FG2004). Seoul, Korea. 2004.
[3] Jerry Jun Yokono and Tomaso Poggio,"Rotation Invariant Object Recognition from One Training Example", AI Memo No.2004-010, CBCL Memo No.238, MIT Center of Biological and Computational Learning, 2004
[4] Paul Viola and Michael J. Jones. "Robust real-time object detection". Technical Report CRL 2001.
[5] Papageorgiou, C., M. Oren, and T. Poggio. "A General Framework for Object Detection". Int. Conf. On Computer Vision (ICCV'98), pp.555-562, 1998.
[6] W. Freeman and E. Adelson. "The design and use of steerable filters". PAMI, 13(9):891–906, 1991.
[7] C. Harris and M. Stephens. "A combined corner and edge detector". Alvey Vision Conference, pp. 147–151, 1988.
[8] D.G. Lowe. "Object recognition from local scale-invariant features". Int. Conf. On Computer Vision (ICCV'99), pp. 1150–1157, 1999.
[9] D.G. Lowe. "Distinctive image features from scale-invariant keypoints". IJCV, 2004
[10] C. Schmid, R. Mohr, and C. Bauckhage. "Evaluation of interest point detectors". IJCV, 37(2):151–172, 2000.
[11] C. Morgenstern and B.Heisele, "Component Based Recognition of Objects in an Office Environment", AI Memo, MIT CBCL, 2003
[12] J. Friedman, T. Hastie, R. Tibshirani, "Additive Logistic Regression: a Statistical View of Boosting". Tech Report, Dept. of Statistics, Stanford Univ, 1998
[13] Y. Freund, R. Schapire, "Experiments with a new boosting algorithm". IEEE Int. Conf. On Machine Learning, pp.148-156, 1996
[14] Y. Freund, "Boosting a weak learning algorithm by majority". Information and Computation 121(2), pp.256-285,1995
[15] S. Lawrence, C. Giles, A. Tsoi, A, Back, "Face Recognition: A Convolutional Neural Network Approach". IEEE Trans. Neural Networks, Vol.8, Number 1, pp.98-113, 1997
[16] T. Sim, R. Sukhankar, M. Mullin, S. Baluja, "High-Performance Memory-based Face Recognition for Visitor Identification". Just Research Technical Report, 1999
[17] T. Sim, R. Sukthankar, M. Mullin, S. Baluja, "Memory-based face recognition for visitor identification". IEEE Int. Conf. On Face and Gesture 2000 (FG2000)
[18] Edelman, S., Intrator, N., and T. Poggio. "Complex cells and object recognition", AI Memo, 1997
[20] R. Schapire, Y. Freund, P. Bartlett, W.S. Lee. "Boosting the margin: A new explanation for the effectiveness of voting methods". Int. Conf. On Machine Learning, 1997.
[21] C. Wallraven, B. Caputo, A. Graf. "Recognition with Local features: the kernel recipe". Int. Conf. On Computer Vision (ICCV2003)
[22] Gabriela Csurka, Cedric Bray, Chris Dance, Lixin Fan. "Visual categorization with bags of keypoints", ECCV 2004
[23] Ullman, S., Vidal-Naquet, M. , and Sali, E. (2002) Visual features of ntermediate complexity and their use in classification. Nature Neuroscience, 5(7), 1-6
[24] Antonio Torralba, Kevin Murphy and William Freeman. "Sharing features: efficient boosting procedures for multiclass object detection"., Int. Conf. On Computer Vision and Pattern Recognition (CVPR2004)

[25] Fergus, R. , Perona, P. and Zisserman, A. “Object Class Recognition by Unsupervised Scale-Invariant Learning”, Int. Conf. On Computer Vision and Pattern Recognition (CVPR 2003)
[26] Heisele, B., P. Ho, J. Wu and T. Poggio. “Face Recognition: Component-based versus Global Approaches”. Computer Vision and Image Understanding, Vol. 91, No. 1/2, 6-21, 2003.
[27] Lei Zhang, Stan Z. Li, ZhiYi Qu, Xiangsheng Huang. "Boosting Local Feature Based Classifiers for Face Recognition". IEEE Workshop on Face Processing in Video, 2004.
[28] Peng Yang, Shiguang Shan, Wen Gao, Stan Li, Dong Zhang, “Face Recognition Using Ada-Boosted Gabor Features”, IEEE Int. Conf. On Automatic Face and Gesture Recognition (FG2004), pp356-361, 2004