Facial Image Analysis Using Local Feature Adaptation Prior to Learning

Rogerio Feris Ying-li Tian Yun Zhai Arun Hampapur

IBM T.J. Watson Research Center
{rsferis,yltian,yunzhai,arunh}@us.ibm.com

Abstract

Many facial image analysis methods rely on learning-based techniques such as Adaboost or SVMs to project classifiers based on the selection of local image filters (e.g., Haar and Gabor filters) from large sets of training data. In general, the learning process consists of selecting discriminative image filters from a large feature pool that contains filters uniformly sampled from the parameter space. In this paper, we argue that we are able to improve these methods by incorporating a local feature adaptation technique prior to learning*, which generates a more compact and meaningful pool of image filters, consequently reducing both learning and detection/recognition computational costs, while at the same time improving accuracies. In the first stage of our approach, local feature adaptation is carried out by a non-linear optimization method that determines image filter parameters (such as position, orientation and scale) in order to match the geometrical structure of each training sample. In the second stage, Adaboost feature selection technique is applied to the adapted feature pool to obtain the final set of discriminative local image filters. We demonstrate the effectiveness and efficiency of the proposed framework in the face detection domain. In the experiments, we have applied our method using a pool of wavelet features, including Haar and Gabor filters. The results showed that with local feature adaptation, significant improvements in terms of detection accuracy and computational cost reduction are achieved over learning based on the same features sampled uniformly from the parameter space.*

1. Introduction

In the past few years, many facial image analysis methods (such as face detection, recognition and expression analysis) have been built upon learning-based techniques which select discriminative local image filters from large sets of training data [13, 15, 1]. The key contribution of our work is to improve these methods with a direct but much useful technique in the feature-level: *adapting local image filters to encode the local geometrical structures of training samples prior to learning*.

As noted by Munder and Gavrila [9], local filters offer advantages over global features such as PCA [16] or FDA [14], which are more sensitive to occlusion and tend to smooth out important object details. However, current learning methods based on local features suffer from a scalability problem in the feature selection process. For a specific feature type (e.g., a Gabor filter), most methods include many feature configurations in a feature pool (e.g., Gabor filters uniformly sampled at different positions, orientations and scales), and then select the most discriminative features using a learning algorithm. Therefore, as new feature types are considered, the feature pool increases dramatically, leading to computational problems. This scalability issue has several implications. First, training time can be excessively long due to the large feature pool and brute-force feature selection. Most methods have hundreds of thousands of local features in the pool and take order of weeks for training on conventional machines. Second, the detection/recognition rate can be significantly affected as important feature configurations may not be included in the feature pool due to the sampling process, whereas many features that are less meaningful for discrimination are present in the pool.

We propose a novel framework to overcome the above limitations by combining and selecting multiple types of visual features in the learning process. Our approach relies on the observation that selected local features tend to match the local structure of the object. Figure 1 shows the first selected features by Adaboost in the context of face detection [13] and recognition [15]. In this example, the selected Haar filters capture the local image contrast. In the middle image of the top row, the dark part of the filter coincides with the dark image region (the eyes), while the bright part of the filter matches the bright image region under the eyes (the cheek and nose). Similarly, in the bottom row, Gabor wavelets capture the local structures of the face. In fact, Liu and Shum [8], in their KullBack-Leibler boosting framework, argued that features should resemble the face seman-

978-1-4244-2154-1/08/$25.00 2008 IE

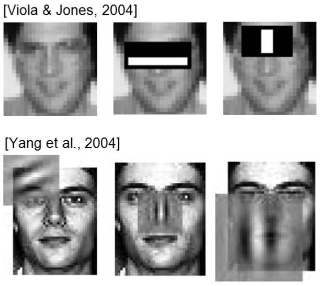

Figure 1. Selected features from Adaboost in the context of face detection (top) and face recognition (bottom). Note that the features tend to adapt to the local face structure.

tics, matching the local or global face structure.

Based on this observation, we present an approach that first pre-selects local image filters based on how well they fit the local image structures of an object, and then use traditional learning techniques such as Adaboost or SVMs to select the final set of features. In this paper, we focus on the face detection problem. Please note that our framework could be potentially extended to other tasks like face recognition, facial feature extraction, and expression analysis.

Our implementation can be summarized in two stages. In the first stage, we learn adaptive features that adapt to each particular sample in the training set. This is carried out by a non-linear optimization method that determines local image filter parameters (such as position, orientation and scale) that match the geometric structure of each object sample. By combining adaptive features of different types from multiple training samples, a compact and diversified feature pool is generated. In the second stage, Adaboost feature selection is applied to the pool of adaptive features in order to select the final set of discriminative features. Having in mind our target application - face detection - this idea is better illustrated in Figure 2. Throughout this paper, we will use the term *adaptive features* to describe features that match the geometric structure of a particular object training image and *general features* to describe the final set of discriminative features that encode common characteristics of all training face images.

In the face detection domain, we show that current methods based on local image filters can benefit from our approach. In particular, our experiments show that significant improvements in terms of detection accuracy and computational cost reduction are achieved when comparing learning based on adaptive features with learning based on features sampled uniformly from the parameter space. In this comparison, we consider wavelet filters of different types, including Haar and Gabor filters with multiple frequencies, and global features generated by grouping a set of local filters.

The remaining of this paper is organized as follows: in

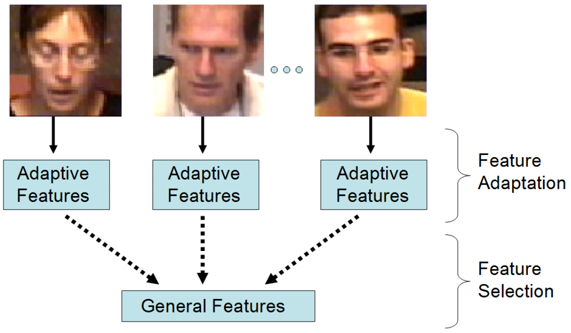

Figure 2. Our approach has two stages: first we compute adaptive features for each training sample and then use a feature selection mechanism to obtain general features.

Section 2, we describe our method in detail to incorporate local feature adaptation in the training process. Section 3 shows the implementation details of our system and the experimental results on the face detection domain. In Section 4, we present a thorough discussion about advantages and potential applications of our framework. Finally, Section 5 concludes our work.

2. Training with Local Adapted Features

In this section, we describe our framework to incorporate local feature adaptation in the training process. We start by showing the feature adaptation algorithm for each individual training image containing the target object. Then, we show how to apply this adaptation method to create a meaningful feature pool containing multiple types of wavelet features. Lastly, the adapted feature pool is used in Adaboost learning to project an object detector classifier. Although we have used wavelet filters in our work, technically other local image filters could also be applied in the same settings.

2.1. Feature Adaptation

Given a particular image of the target object, our goal is to learn the parameters of wavelet features, including position, scale, and orientation, such that the wavelet features match the local structures of the object. This is motivated by the wavelet networks proposed by Zhang [17] and introduced in computer vision by Krueger [5].

We start by including a family of N two-dimensional wavelet functions $\Psi = \{\psi_{n_1}, \ldots, \psi_{n_N}\}$, where $\psi_{n_i}(x, y)$ is a particular mother wavelet (e.g., Haar, Gabor, and etc.) with parameters $n_i = (c_x, c_y, \theta, s_x, s_y)^T$. Here, c_x and c_y denote the translation of the wavelet, s_x and s_y denote the dilation, and θ denotes the orientation. The choice of N is depending on the degree of desired representation precision.

Let I be an input training image. First, we initialize the set of wavelets Ψ along the image in a grid, as shown in Figure 3a, with the wavelets having random orientations,

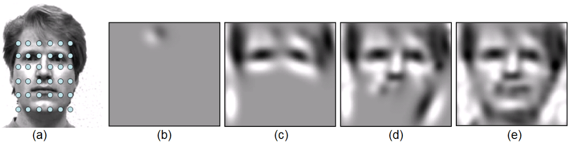

Figure 3. Learning adaptive features for a particular training image. (a) Input training image with wavelets initialized as a grid along the face region, with random orientations and scales. (b)-(e) Wavelet features being optimized one by one, with parameters (position, scale, and orientation) optimized to match the local structure of the input face image.

and scales initialized to a unified value that is related to the density with which the wavelets are distributed. Then, assuming I is dc-free, without loss of generality, we minimize the following energy function:

$$E = \min_{n_i, w_i \text{ for all } i} \|I - \sum_i w_i \psi_{n_i}\|^2, \quad (1)$$

with respect to the wavelet parameter vectors n_i and their corresponding weights w_i. We used the Levenberg-Marquardt method for the optimization process. Figures 3b-e show the wavelet features being optimized one by one to match the local image structure of the object. In this example, a Gabor wavelet was adopted as the mother wavelet.

It is important to note that the parameters n_i are optimized in the *continuous* domain and the wavelets are positioned with sub-pixel accuracy, contrasting with most existing discrete approaches. This assures that a maximum of the image information can be encoded with a small number of wavelets.

Using the optimal wavelets ψ_{n_i} and weights w_i, the image I can be closely reconstructed by a linear combination of the weighted wavelets:$\hat{I} = \sum_{i=1}^{N} w_i \psi_{n_i}$.

Rather than relying on the optimization process described above, the wavelet weights w_i can be computed directly once the wavelet parameters n_i are optimized. If the wavelet functions are orthogonal, this can be done by just computing the inner products of the image I with each wavelet filter, i.e., $w_i = \langle I, \psi_{n_i} \rangle$.

In the more general cases where the wavelet functions may not be orthogonal, a family of dual wavelets $\tilde{\Psi} = \{\tilde{\psi}_{n_1} \dots \tilde{\psi}_{n_N}\}$ has to be considered. The wavelet $\tilde{\psi}_{n_j}$ is the dual wavelet of ψ_{n_i} if it satisfies the bi-orthogonality condition: $\langle \psi_{n_i}, \tilde{\psi}_{n_j} \rangle = \delta_{i,j}$, where $\delta_{i,j}$ is the Kronecker delta function.

Given an image I and a set of wavelets $\Psi = \{\psi_{n_1}, \dots, \psi_{n_N}\}$, the optimal weights are given by: $w_i = \langle I, \tilde{\psi}_{n_i} \rangle$. It can be shown that $\tilde{\psi}_{n_i} = \sum_j \left(A_{i,j}^{-1}\right) \psi_{n_j}$, where $A_{i,j} = \langle \psi_{n_i}, \psi_{n_j} \rangle$. This is a faster and more accurate solution than using Marquardt optimization to compute the wavelet weights.

2.2. Integrating Visual Features

In the previous section, we have described how to obtain adaptive features for a single object training image. Now we proceed to generate a pool of adaptive features obtained from multiple training images.

Let $\chi = \{I_1, \dots, I_M\}$ be a set of object training images. For each image I_i, we generate a set of adaptive features Ψ_i, using the optimization method described in Section 2.1.

Integration of multiple feature types is possible by using different wavelet settings for each object sample. More specifically, each set Ψ_i is learned with different parameters, including: (1) number of wavelets; (2) wavelet type; (3) wavelet frequency; (4) group of features treated as a single feature.

The number of wavelets indicates how many wavelet functions will be optimized for a particular object image. From this set, we can further select a subset of functions that have the largest weights. Wavelet filters with larger associated weights in general tend to coincide with more significant local image variations. In our system, we used only Haar and Gabor wavelets for the wavelet type parameter, but other feature types could also be considered. The frequency parameter controls the number of oscillations for the wavelet filters. Finally, we also allow a group of wavelet functions to be treated as a single feature, which is important to encode global object information.

All the generated adaptive features for all the object images are then put together in a single pool of features Ω, defined as: $\Omega = \bigcup_{i=1}^{M} \Psi_i$.

Figure 4 shows different adaptive features (such as Haar and Gabor wavelets with different frequencies, orientations and aspect ratios) learned from a dataset of frontal face images. In the resulting feature pool, different types of local wavelet filters as well as global features which are obtained by grouping individual wavelet functions are present.

The wavelet settings (type, frequency, how many features to group, etc.) for each training face image can be initialized randomly to allow a variety of different features in the pool. This initialization process is fully automatic and allows the creation of a compact and diversified feature pool.

2.3. Learning General Features

In the previous section we have described a method to generate a pool of adaptive features from a set of training images. Now, in order to project a face detector, we select general features, i.e., the features from the pool that encode common characteristics to all face samples.

We used Adaboost learning [13] to both selecting general features and projecting a classifier. A large set of non-face (background) images is used in addition to the training images. The general features are those that best separate

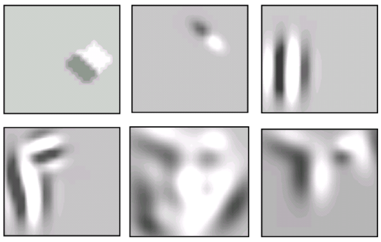

Figure 4. Some examples of features present in the pool of learned adaptive features for a frontal face dataset. A large variety of diferent wavelet filters are considered. The top row shows local wavelet functions, whereas the bottom row shows global features generated by combining a set of local filters.

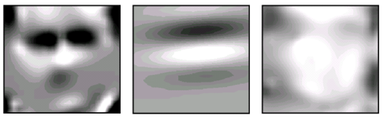

Figure 5. The first three selected general features for a frontal face training database.

the whole set of face samples from non-face samples during classification. We refer the reader to [13] for more details about the Adaboost classifier and the feature selection mechanism.

Figure 5 shows the first three general features selected by Adaboost, using a pool of adaptive features obtained from a database containing frontal faces. The first selected feature gives more importance to the eyes region. The second selected feature is a local coarse-scale Gabor wavelet with three oscillations, which align with the eyes, nose, and mouth regions. The third feature is a global feature that encodes the rounded face shape.

The face detector is applied in all possible image locations and scales. We used a cascade technique [13] to improve the efficiency of this operation. However, even using a cascade classifier, real-time performance (25/30Hz) can not be achieved due to the time required to compute our features. We solved this problem by using traditional Haar-like/rectangle features in the first levels of the cascade. This allows for efficient rejection of background patches during classification. The image patches that are not rejected by the Haar cascade are then fed into a cascade of classifiers using our features. The choice of which cascade level should be used to switch from Haar features to our features is application dependent. Switching in lower levels allows for more accuracy, but switching in higher levels allows for more efficiency.

3. Experiments

In this section we describe the implementation of our system and report experimental results. We demonstrate the usefulness of local feature adaptation prior to learning in the face detection domain.

In our experiments, we used a frontal face dataset containing 4000 faces for training purposes. Each training image was re-scaled and cropped to a 24x24 patch size. A pool of adaptive features was generated by running the optimization process described in Section 2.1, with different wavelet settings (wavelet type, frequency, etc.) for each sample. As a result, a pool of 80000 adaptive features was generated, containing a large variety of wavelet filters. It takes less than a second to create hundreds of adaptive features for a particular 24x24 sample in a conventional 3GHz desktop computer.

For learning general features, we used an additional database of about 1000 background (non-faces) images from which 24x24 patches are sampled. A cascade classifier was trained by considering 4000 faces and 4000 non-faces at each level, where the non-face samples were obtained through bootstrap [11]. Each level in the cascade was trained to reject about half of the negative patterns, while correctly accepting 99.9% of the face patterns. A fully trained cascade consisted of 24 levels. A Haar filter corresponding to the first 18 levels of the cascade was used in our experiments.

During detection, a sliding window was moved pixel by pixel at different image scales. Starting with the original scale, the image was re-scaled by a factor of 1.2 in each iteration. Multiple overlapping detection results were merged to produce a single result for each location and scale.

The CMU+MIT frontal face test set, containing 130 gray-scale images with 511 faces was used for evaluation. A face is considered to be correctly detected if the Euclidian distance between the center of the detected box and the ground-truth is less than 50% of the width of the ground-truth box, and also if the width (i.e., size) of the detected face box is within ±70% of the width of the ground-truth box.

In order to show the effectiveness of feature adaptation prior to learning, we compared our approach to a classifier learned from a similar feature pool, containing the same number and type of features (Haar, Gabor, etc.), but were sampled uniformly from the parameter space (at discrete positions, orientations, and scales), rather than adapted to the local structure of the training samples. Figure 6a shows a plot of the Receiver Operating Characteristic (ROC) curves for this comparison, demonstrating the superior performance of our method. In addition to achieving improved detection accuracy, the number of weak classifiers needed for each strong classifier is significantly smaller in our method (see Figure 6b). This clearly has a direct impact in both training and testing computational costs (about 50% reduction).

Figure 7 shows the comparison between our approach

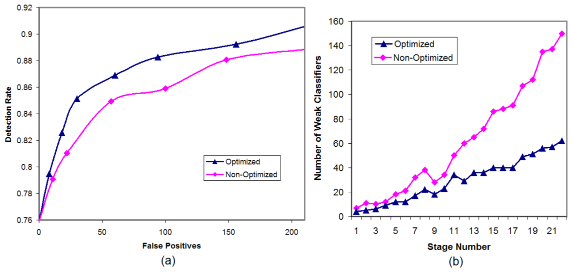

Figure 6. (a) ROC Curve comparing classifiers learned from adaptive (optimized) and non-adaptive (non-optimized) features in the CMU+MIT dataset. (b) Number of classifiers for each level of the cascade in both methods. Our approach offers advantages in terms of detection accuracy and reduced computational costs over traditional methods that use local features uniformly sampled from the parameter space.

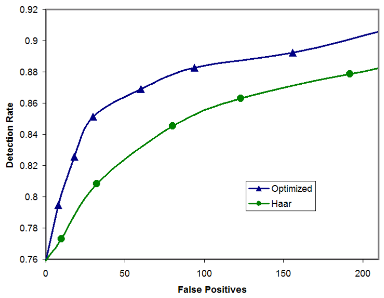

Figure 7. ROC Curves for our approach and traditional Haar features in the CMU+MIT dataset. We used only half of the number of features in the feature pool compared to Haar features and still get superior performance.

and traditional Haar-like/rectangle features. The Haar detector was also trained with a cascade consisting of 24 levels in the same training set. The feature pool, however, was twice as large than our approach, containing about 160000 features. With half of the features in the pool, we achieve superior performance and a faster learning time (see table 4).

4. Discussion

Different methods have been proposed to enhance Adaboost learning with more powerful features. Wang and Ji [14] used FDA and RNDA features for multi-view face detection. Lienhart [7] proposed an extended set of Haar-like features to improve detection performance. Levi and Weiss [6] used edge orientation histograms to learn from a small number of examples. PCA features [16] have also been used as part of the feature pool. More recently, Huang et al. [4] proposed a sparse feature set for object detection.

The uniqueness of our approach has several aspects. First, it can be applied to improve existing methods based on local image filters. As an example, Gabor wavelets in connection with Adaboost Learning have proven to be very successful in facial expression and recognition algorithms [1, 15]. Our method could be directly applied to locally adapt these filters prior to Adaboost learning. Second, it provides a principled mechanism to integrate multiple types of local features, offering a solution to the scalability issue inherent on the feature level of learning algorithms. Finally, our proposed framework allows the pre-selection of relevant local features which are strongly correlated with the face structures, leading to a more discriminative classifier.

In our framework, multiple types of local features can be integrated to form a compact and diversified feature pool. In addition to wavelet filters, other types of features, such as local edge orientation histograms [6], could be adapted/optimized in a similar way. An interesting observation is that the wavelet filters with large coefficients in fact tend to align with the orientation of local intensity edges in the image. Hence, edge-based classifiers could be projected based only on the spatial support of the wavelet features.

Many learning methods based on local features sample the feature space uniformly at discrete positions, scales, and orientations [15, 1]. Thus, the feature pool contains many configurations that are not useful for classification, whereas important feature configurations may not be included due to the sampling process. This becomes more problematic when multiple features of different types are considered. In our method, only meaningful features that match the local structure of the object samples, with subpixel precision, are included in the pool.

The ability to integrate multiple types of features, as well as to pre-select features correlated to the object structures, makes our method more scalable in the domain level. In other words, it is suitable to detect different objects other than faces. This contrasts with traditional Haar-like/rectangle features, which are more appropriated for symmetric objects like frontal faces.

Although adaptive features are projected to reconstruct an object image, they play an important role in discrimination, as motivated in Section 1. An alternative approach would be to bypass the learning of adaptive features and determine the optimal local feature parameters (position, scale, and orientation) to maximize a discrimination criterium between object and non-object samples directly. However, this would not be feasible computationally, especially if we are targeting subpixel precision.

We note that our contribution is made on the feature level of learning-based methods. Our goal in this paper is not to provide an integrated state-of-the-art face detector system, but rather provide a feature selection tool that can be combined with more advanced boosting methods, like GentleBoost [2], Real Adaboost [12], and Vector Boosting [3], in

Feature Pool	Number of Features	Learning Time
Haar Features	160000	around 5 days
Our Approach	80000	around 3 days

Table 1. By learning adaptive and general features, we can use a smaller feature pool, which allows reduced training time, while still keeping superior performance in detection rate, when compared to a traditional pool of Haar features.

order to achieve state-of-the-art results. In a more interesting way, our method could be integrated into the learning method recently proposed by Pham and Cham [10], which achieves extremely fast learning time compared to previous methods. We believe this method would have an even reduced computational learning time by using locally adapted features.

Our approach is related to the work of Krueger [5], who uses adaptive Gabor wavelets as features for object representation. His work is applicable for object recognition, but not for object detection, since only a single training image is considered and no general features are learned for an object class.

5. Conclusions

The main contribution of this paper is to use local feature adaptation prior to learning, which enables training based on a compact and diversified dictionary of visual features. Our experiments show that our approach offers advantages in terms of improved detection accuracy and significant reduced computational costs over traditional methods which use non-adapted local features uniformly sampled from the parameter space.

As future work, we plan to test our approach with different objects and carry out a more extensive evaluation, analyzing parameters such as the size of the feature pool and the training set. We expect to have improved performance with larger feature pools and better ability to learn from a small number of examples. We also plan to extend our method to the problems of face recognition and expression analysis.

References

[1] M. Bartlett, G. Littlewort, I. Fasel, and J. Movellan. Real time face detection and facial expression recognition: Development and application to human-computer interaction. In *CVPR Workshop on Computer Vision and Pattern Recognition for Human-Computer Interaction*, Vancouver, Canada, 2003.

[2] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 38(2):337–374, 2000.

[3] C. Huang, H. Ai, Y. Li, and S. Lao. Vector boosting for rotation invariant multi-view face detection. In *IEEE International Conference on Computer Vision (ICCV'05)*, Beijing, China, October 2005.

[4] C. Huang, H. Ai, Y. Li, and S. Lao. Learning sparse features in granular space for multi-view face detection. In *International Conference on Automatic Face and Gesture Recognition*, Southampton, UK, April 2006.

[5] V. Krueger. *Gabor wavelet networks for object representation*. PhD thesis, Christian-Albrecht University, Kiel, Germany, 2001.

[6] K. Levi and Y. Weiss. Learning object detection from a small number of examples: the importance of good features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'04)*, Washington, DC, July 2004.

[7] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *DAGM 25th Pattern Recognition Symposium*, 2003.

[8] C. Liu and H. Shum. Kullback-leibler boosting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'03)*, Madison, Wisconsin, June 2003.

[9] S. Munder and D. Gavrila. An experimental study on pedestrian classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1863–1868, 2006.

[10] M. Pham and T. Cham. Fast training and selection of haar features using statistics in boosting-based face detection. In *IEEE International Conference on Computer Vision (ICCV'07)*, Rio de Janeiro, Brazil, October 2007.

[11] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.

[12] R. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37:297–336, 1999.

[13] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'01)*, Kauai, Hawaii, December 2001.

[14] P. Wang and Q. Ji. Learning discriminant features for multi-view face and eye detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, June 2005.

[15] P. Yang, S. Shan, W. Gao, S. Li, and D. Zhang. Face recognition using ada-boosted gabor features. In *International Conference on Automatic Face and Gesture Recognition*, Seoul, Korea, May 2004.

[16] D. Zhang, S. Li, and D. Gatica-Perez. Real-time face detection using boosting in hierarchical feature spaces. In *International Conference on Pattern Recognition (ICPR'04)*, Cambridge, UK, August 2004.

[17] Q. Zhang. Using wavelet network in nonparametric estimation. *IEEE Transactions on Neural Networks*, 8(2):227–236, 1997.