EFFECTIVE DISCRETIZATION OF GABOR FEATURES FOR REAL-TIME FACE DETECTION

Feijun Jiang, Bertram Shi

Hong Kong University of
Science and Technology
Department of ECE
Clear Water Bay, Kowloon, Hong Kong

Mika Fischer, Hazım K. Ekenel

Karlsruhe Institute of Technology
Institute for Anthropomatics
Adenauerring 2, 76131 Karlsruhe, Germany

ABSTRACT

We describe a real-time face detector based on Gabor features. While Gabor features often lead to improved performance, they are often avoided as they are perceived as being computationally expensive. We address this in two ways. First, we propose an efficient discrete encoding method for the Gabor feature vector. This enables us to use a computationally efficient multi-stage classifier based on boosting and winnowing. Second, we accelerate computationally complex computations using the parallelization provided by graphics processing units (GPUs). With these innovations, the resulting detector runs at 16.8 fps for 640×480 images on a PC equipped with an i5 CPU and a GTX 465 graphic card.

Index Terms**—** Real-time, face detection, Gabor filter, boosting

1. INTRODUCTION

There have been extensive research efforts on face detection. Recently, there have been real-time face detectors proposed. Among the most popular detectors are Viola-Jones and MCT face detectors [1, 2], which use Haar wavelets or the modified census transform (MCT) as features, and boosting for classification. These systems have shown very promising results.

Since biological systems excel at face detection, one possible way to achieve better performance would be to use a feature representation that resembles what is thought to be used by biological systems. Since the impulse responses of Gabor filters are known to resemble the receptive fields of simple cells in the mammalian primary visual cortex, features based on the outputs of Gabor filters have attracted much attention for face detection. Xiaohua et al. [3] proposed a simplified Gabor filter that uses the subtraction of rectangle regions as an approximation of the output of Gabor filters. Adding these simplified Gabor features to a detector using Haar-like features and training via boosting improves the detector performance. Huang et al. [4] proposed a face detector that uses a polynomial neural network to classify Gabor features computed at four orientations. Kwolek [5] proposed a convolutional neural network face detector using Gabor filter outputs from two scales and two orientations. Chen et al. [6] introduced a hybrid face detector that cascades detectors based on Gabor energy features and Haar-like features.

While these studies suggest that Gabor features are promising for face detection, they are often avoided as they are perceived as being computationally expensive. Here, we address this problem by developing an effective and computationally simple discretization of the Gabor filter output. Previous work has typically used high dimensional continuous valued feature vectors defined by taking the Gabor filter outputs or their magnitudes directly. By using a discrete encoding, we can use a table lookup based boosting classifier. This enables us to quickly run many boosting loops, each of which improves classification performance, and the results of all boosting loops can be combined into a single lookup table. We find that the performance of our face detector exceeds that of detectors using continuous valued Gabor filter outputs as features. Thus, the loss of information due to discretization is compensated by the increased power of the classifier provided by boosting. Meanwhile it is quite efficient; it works at 16.8 fps on our system with image size of 640×480.

To examine whether discretized Gabor features are effective for face detection, we first study a discretization based upon k-means clustering and vector quantization (VQ). While this performs well, the k-means process of searching for the closest center is quite time consuming. To address this problem, we have examined the classifiers obtained by the k-means clustering, and developed a discrete encoding called radial phase which only requires several pixel-wise comparison operations on the Gabor continuous outputs, while maintaining performance comparable to the VQ encoding.

This study is funded by a grant from the Germany/Hong Kong Joint Research Scheme sponsored by the Research Grants Council of Hong Kong and the German Academic Exchange Service (DAAD) (Reference No. G_HK014/09), by the InterACT program, by the "Concept for the Future" of Karlsruhe Institute of Technology within the framework of the German Excellence Initiative and by OSEO, the French State agency for innovation, as part of the Quaero Program.

This paper is organized as follows. In Section 2, we describe the architecture of the classifier. In Section 3, we present experimental results comparing the VQ and radial phase encodings of the outputs of Gabor filters tuned to different orientations. We also discuss the implementation of our top performing classifier using GPU based processing. Finally, we conclude in Section 4 with a summary.

2. FACE DETECTOR ARCHITECTURE

The face detector is applied to 20×20 pixel sub-images within each image. The location of the sub-image is scanned over the entire image and at multiple scales. The scales are constructed using an image pyramid, where each scale differs by a factor of 1.2. The detector consists of three steps: Gabor feature extraction, discrete encoding of the Gabor features and finally a multi-stage classifier. We describe these steps in more detail below.

2.1. Gabor features

Gabor filters have an impulse response given by a 2D Gaussian modulated by a 2D complex exponential with spatial frequency parameters Ω_x and Ω_y, which determine the center spatial frequency $\Omega = \sqrt{{\Omega_x}^2 + {\Omega_y}^2}$ and orientation $\theta = \arctan \Omega_x/\Omega_y$. Here, we use a center spatial frequency (Ω) of $2\pi/3$. The Gaussian covariance matrix is diagonal with entries $\sigma^2 = (1.6/\Omega)^2$. In order to remove the DC component in the real part of the Gabor filter, we subtract from the real part the output of a circularly symmetric Gaussian filter with standard deviation σ multiplied by a constant.

We construct a continuous valued Gabor feature vector by taking the real and imaginary parts of the complex valued outputs of N filters tuned to orientations equally spaced from 0 to $(N-1)\pi/N$, where $N = 4$. In order to increase the robustness of our detector against illumination variations, we normalize each element of the Gabor feature vector by the square root of the sum of the energies of the responses from all orientations.

2.2. Discrete feature encoding

We examined two strategies for discretizing the normalized filter outputs: vector quantization of the entire feature vector using a codebook obtained by k-means clustering and a method based on discretizing each complex valued filter output individually based on its phase and energy.

For the vector quantization encoding, we use a set of 512 vectors. We generate these vectors by feeding 11,338,056 normalized Gabor vectors into a k-means clustering function. Half the training vectors are taken from all 18×18 feature locations of the face images in 70% of the training data. Half are taken from non-face images, which are chosen randomly from among all the negative images.

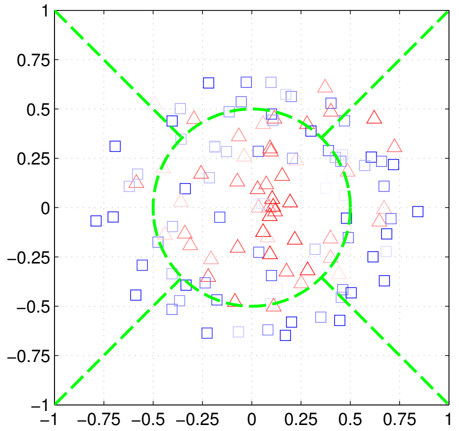

Fig. 1. Graphical representation of the lookup table for one classifier in the first boosting stage. Color intensities of triangle/rectangle represent positive/negative weights. The green lines show the boundaries used for the radial phase encoding.

Unfortunately, the VQ encoding is computationally expensive, since we must compare the Gabor feature with each of the k vectors. Thus, we seek to leverage our knowledge about the structure of the feature space gleaned from designing the VQ based classifiers to develop a discrete encoding that can be computed more efficiently.

We define a discrete encoding that encodes the normalized Gabor filter output at each of the four orientations separately. We map each Gabor filter output to one of five indices depending upon which of the five regions of the complex plane shown in Fig. 1 the filter output falls into. In total, there are $5^4 = 625$ total possibilities for each Gabor feature vector.

Filter outputs falling into the central circular region with radius 0.5 correspond to orientations whose Gabor filter energy is smaller than the mean energy among the four orientations. These are less significant. The more significant filter outputs are quantized into one of four regions, corresponding to different phases. The phase indicates the edge type, which is necessary for face detection. Phases of 0 and π correspond to cosine modulations in the Gabor function, and thus indicate light or dark bars at the given orientation. Phases of $-\pi/2$ and $\pi/2$ correspond to sine modulations in the Gabor function, and thus indicate step edges with different polarity. By quantizing the output into four regions centered around these values we ignore small position shifts of the edges. We refer to this encoding as the radial phase encoding.

2.3. Multi-stage classifier

For the classification, we use the same lookup table based cascaded classifier as C. Küblbeck and A. Ernst used in their MCT based face detector [1], which contains four cascaded stages: three Ada-boosting stages and a final winnow stage.

Stages are trained sequentially. Each stage is trained only on samples which pass through all the previous stages. The threshold for comparison at each stage is set so that the true positive detection rate is 99.5%. Negative samples are chosen randomly from a set of 20 × 20 sub-windows taken from natural images which contain no faces.

For the first three stages, the classification at each stage is determined by the weighted sum of the outputs of a number of weak classifiers, where the choice of the weak classifiers and their weighting coefficients are determined by boosting. For each boosting loop, each sub-image location is considered as a potential weak classifier, where we consider only the center 18 × 18 (324) locations within each 20 × 20 pixel sub-image to avoid boundary effects. The weak classifier at each sub-image location is a lookup table, where the indices correspond to the indices of the discrete encoded features and the value of the lookup table is either 0 or 1 depending on whether the classification result is face or non-face.

We limit the first stage to use only 20 pixel locations and the second stage to use only 80 pixel locations. The third stage is not limited. For each boosting stage, we run 1500 boosting loops. In the initial stages, many weak classifiers at each pixel location are used, but they can be combined into a single lookup table.

The fourth winnow stage makes use of all the locations. At each location there are separate lookup tables of weights for face and non-face. Initially all weights are 0, but are set to 1 the first time they are addressed during training. The weights are updated according to the Winnow Update Rule in [1]. After training, we generate a single lookup table for each location by subtracting the lookup tables for face and non-face.

3. EXPERIMENTS AND RESULTS

For training, we use a database of 25000 cropped face images with size of 20 × 20 pixels, which includes faces with slight variations in pose angle and under different illumination conditions [7]. The negative samples are from a dataset of more than 25000 background images. While training the first stage, the negative samples are obtained by randomly choosing 20 × 20 pixels windows from the background images. For the later stages, the negative samples are randomly chosen from the false positives that pass through all of the previous stages.

We used the results of this VQ classifier to develop the radial phase discrete encoding. Fig. 1 shows an example of this analysis with a VQ codebook size of 128. Here the final combined lookup table for the classifier at one sub-image location chosen during the first boosting phase is shown graphically as a set of points on the complex plane. The location of each point corresponds to the normalized Gabor output for the horizontal orientation. The triangle/rectangle encodes the sign of the lookup table weight which indicates positive/negative judgement and the intensity encodes the weight. As we can see, most of the bright rectangle points lie outside the circle with radius 0.5, while most of the bright triangle points lie inside.

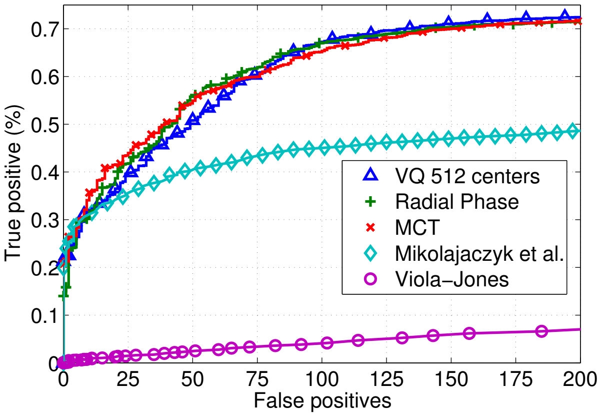

Fig. 2. The ROC curves of different detectors on FDDB set [8].

We benchmark these detectors using evaluation protocol for the FDDB dataset proposed by V. Jain and E. Learned-Miller [8]. Each detector generates a ROC curve representing its performance on the FDDB benchmark dataset, which contains 2845 images. We compare our results with the published data from the FDDB dataset, as well as the result of an MCT based detector that uses the same multi-stage classifier and is trained with the same training data. The MCT encodes the intensities in each 3 × 3 window around a pixel location as one of 511 possible patterns.

Fig. 2 shows the ROC curves of detectors based on the radial phase coding, the VQ based detector with 512 centers and the MCT baseline [1]. The performance of both the radial phase encoder and the VQ based detector exceeds that of the baseline results published from the FDDB dataset by Mikolajazyk et al. [9] and Viola-Jones [2], as well as that obtained using the MCT based detector. For example, while the true positive is 70%, the corresponding number of false detections for VQ, radial phase, MCT, Mikolajazyk et al. and Viola-Jones are 132, 141, 150, 30739 and 44147, respectively. This demonstrates that appropriately chosen discretizations based on Gabor filter outputs can provide more effective representations of the image structure than the ones used in state of the art face detectors.

As an additional comparison with prior work, we ran our detectors on the CMU/MIT dataset which contains 130 images and 511 faces. Table 1 compares the results with previously reported work using the output of Gabor detectors. It shows that our Gabor based detector works significantly better than the Gabor detector proposed by Huang et al. [4], with an increased number of true positives and reduced number of false positives. In addition, our detectors show a decreased number of false positives at a similar rate of true positives compared to the Viola-Jones detector.

The key advantage of our proposed discretization is its computational simplicity, while maintaining performance comparable to that of VQ. In the implementation of our real time face detector, we used a computer equipped with i5 2.67 GHz CPU with an NVidia GTX-465 graphics card. For a 640×480 image, there are 18 scales (pyramid levels), each computed by resizing the input image with texture mapping on the graphics card. Computations to filter each level of the pyramid with the four Gabor filters using the algorithm in [10] and to compute the VQ and radial phase encoding are performed on the graphics card. The detector has two main stages: building the pyramid and scanning the pyramid, as well as a clustering stage to merge multiple detections. The computational time required for each stage is shown in Table 2. The whole process of the radial-phase detector takes 59.3 ms, corresponding to 16.8 fps. The efficiency of the VQ depends on the number of centers. On our system, the 512 vector VQ detector takes 450.5ms and the 64 vector VQ detector takes 107.2ms. Both are significantly slower than the radial phase encoding. The main difference is in the time required to build the pyramid, since the VQ process requires comparison with all of the vectors in the codebook. Here, the speedup over VQ ranges from a factor of 3.5 for 64 vectors to over 20 for 512 vectors.

4. CONCLUSION AND FUTURE WORK

In this work, we proposed an efficient discrete encoding of locally normalized Gabor filter outputs, which outperforms state of the art detectors on the FDDB dataset, and other Gabor detector based on continuous Gabor outputs. This encoding only requires several pixel-wise comparison operations on the Gabor output. By implementing this method on a GPU, we achieved a real-time face detector with 16.8 fps.

Gabor filters are widely used as image processing steps for extracting visual cues other than orientation. For example, disparity and motion cues can be calculated based on the outputs of Gabor filter using the disparity and motion energy models. Our work demonstrates that Gabor filter outputs can be effectively incorporated into face detectors. Our future work will seek to use these ideas to incorporate disparity and motion cues efficiently into face detectors. For example, these cues can be used to direct the scanning window instead of scanning every pixel in the pyramid. This could lead to significant gains, since 65% of the computation is now spent for scanning.

5. REFERENCES

[1] C. Küblbeck and A. Ernst, "Face detection and tracking in video sequences using the modifiedcensus transformation," *Image and Vision Computing*, vol. 24, no. 6, pp. 564–572, 2006.

[2] P. Viola and M.J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[3] L. Xiaohua, K.M. Lam, S. Lansun, and Z. Jiliu, "Face detection using simplified Gabor features and hierarchical regions in a cascade of classifiers," *Pattern Recognition Letters*, vol. 30, no. 8, pp. 717–728, 2009.

[4] L.L. Huang, A. Shimizu, and H. Kobatake, "Robust face detection using Gabor filter features," *Pattern Recognition Letters*, vol. 26, no. 11, pp. 1641–1649, 2005.

[5] B. Kwolek, "Face Detection Using Convolutional Neural Networks and Gabor Filters," *Artificial Neural Networks: Biological Inspirations–ICANN 2005*, pp. 551–556, 2005.

[6] J. Chen, S. Shan, P. Yang, S. Yan, X. Chen, and W. Gao, "Novel face detection method based on gabor features," *Advances in Biometric Person Authentication*, pp. 49–94, 2005.

[7] J. Chen, R. Wang, S. Yan, S. Shan, X. Chen, and W. Gao, "Enhancing human face detection by resampling examples through manifolds," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 37, no. 6, pp. 1017–1028, 2007.

[8] V. Jain and E. Learned-Miller, "FDDB: A benchmark for face detection in unconstrained settings," Tech. Rep., Technical report, University of Massachusetts, Amherst, 2010. 111.

[9] K. Mikolajczyk, C. Schmid, and A. Zisserman, "Human detection based on a probabilistic assembly of robust part detectors," *Computer Vision-ECCV 2004*, pp. 69–82, 2004.

[10] X.X. Wang and B.E. Shi, "GPU implemention of fast Gabor filters," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*. IEEE, 2010, pp. 373–376.

Table 1. Comparison of the detector performance on CMU-MIT set

	True positives	False positives
VQ, 512 centers	444	6
radial phase	444	17
MCT baseline	430	7
Gabor detector in [4]	421	57
Viola-Jones in [2]	448	31

Table 2. Comparison on the computational efficiency in (ms)

	Build Pyramid	Scan Pyramid	Clustering
VQ, 512 centers	414.5	36.0	0.003
VQ, 64 centers	71.7	35.5	0.003
radial phase	21.0	38.3	0.027