Face Verification Using Boosted Cross-Image Features

Dong Zhang
University of Central Florida
Orlando, FL
dzhang@cs.ucf.edu

Omar Oreifej
University of California, Berkeley
Berkeley, CA
oreifej@eecs.berkeley.edu

Mubarak Shah
University of Central Florida
Orlando, FL
shah@crcv.ucf.edu

Abstract

This paper proposes a new approach for face verification, where a pair of images needs to be classified as belonging to the same person or not. This problem is relatively new and not well-explored in the literature. Current methods mostly adopt techniques borrowed from face recognition, and process each of the images in the pair independently, which is counter intuitive. In contrast, we propose to extract cross-image features, i.e. features across the pair of images, which, as we demonstrate, is more discriminative to the similarity and the dissimilarity of faces. Our features are derived from the popular Haar-like features, however, extended to handle the face verification problem instead of face detection. We collect a large bank of cross-image features using filters of different sizes, locations, and orientations. Consequently, we use AdaBoost to select and weight the most discriminative features. We carried out extensive experiments on the proposed ideas using three standard face verification datasets, and obtained promising results outperforming state-of-the-art.

1. Introduction

Facial image analysis is a widely investigated area in computer vision and multimedia, with several face-relevant problems explored in the literature. In face detection [16, 9, 18], an image is classified as a face or not by capturing the facial features and landmarks which distinguishes a face from other types of objects. On the other hand, in face recognition [19, 21, 23, 2], given an image of a face, possibly detected using a face detector, the person's identity is recognized by capturing facial features which distinguish individuals.

Lately, a new face-relevant problem has been receiving

increasing attention, namely, face verification. Given a pair of face images, the task is to verify whether they belong to the same person or not. Face verification is still poorly addressed with only few reported results. In [7], Guillaumin et al. employed a metric learning method to obtain the optimal Mahalanobis metrics over a given representation space of faces. On the other hand, in [17], Schroff et al. proposed a new similarity measure which increased the verification robustness against pose, illumination, and expression differences. Also in [22], a SVM-based method was proposed for video face verification, where a SVM is trained on a video of a person, then used to classify another video in order to determine if it corresponds to the same identity. Moreover, Kumar et al. [11] successfully employed attribute classifiers for face verification which significantly improved the results of face verification. Also Nguyen et al. [13] used an adaptive learning method to obtain the optimal transformation for the cosine similarity metric in the face space. Additionally, the Wavelet LBP features were proposed in [5] and employed for face verification. It can be clearly observed that current face verification methods mostly employ techniques borrowed from face recognition and apply them separately on each image of the pair to be verified.

In this paper, we propose a novel method for face verification, which we believe will set a new direction in this problem. Our method is inspired by the following observation: Since face verification is a two-image problem, using features extracted individually from each image is counter intuitive; rather, the most discriminative features should be extracted across the pair of images jointly. Methods such as [4] and [3] employ LBP and SIFT features which are designed to describe a certain region in a single image. In contrast, we propose to use cross-image features which describe the relation between two faces, thus better fits the face verification problem. Note that our observation can be

generalized to verification of any type of objects. For instance, given two images of animals, we can verify whether both images are for cats by running each of them separately through a cat detector trained on images of cats vs non-cats, then combine the detector's confidences. This is inherently different from using a verification classifier which receives a pair of images and classifies them jointly as being both cats or not. The latter classifier would be trained on cross-image features extracted from cat-cat pairs vs cat-noncat pairs, which is more discriminative for object similarity and dissimilarity.

Our cross-image features are embarrassingly simple, we use a large bank of normalized correlation filters between patches across the pair of images at different sizes, locations, and orientations. Additionally, we use Haar-like features similar to the ones used in [20], however, obtained across the pair of images. Consequently, we use AdaBoost to weight and select the most discriminative filters. Our method is derived from Viola and Jones face detector [20]; however, instead of using filters within an individual image, we use across-image filters such that we capture the relation between the pair of images, rather than capturing individual image features. It is important to note that cross-image features are extracted from two images in order to encode their similarity. This is inherently different from relating pairs or triplets of features extracted from the same image, which is typically used to encode the spatial relations between the features as in for example [6] and [14]. The rest of the paper is organized as follows: In the next section, we present our proposed cross-image features, followed by AdaBoost classifier training. The experiments and the results are described in Section 3. Finally, Section 4 concludes the paper.

2. Proposed Method

Our cross-image features for face verification are based on the simple rectangle filters presented by Viola and Jones [20]. However, we extend the features to operate across pairs of images rather than within individual images. While these features seem simple, the experiments demonstrate their superior discriminative capabilities in face verification. Figure 1 illustrates the difference between the features of [20] and ours. We capture the differences between the two images, instead of the variation within the image. In particular, given a pair of face images I_1 and I_2, which we aim to classify as belonging to the same identity or not, let r_i denote a box i positioned at location (x_i, y_i), with width and height (w_i, h_i), and orientation θ_i. We define two types of filters:

- Haar-like cross-image filters: This type of filter compares the box rectangular sum between the image pair
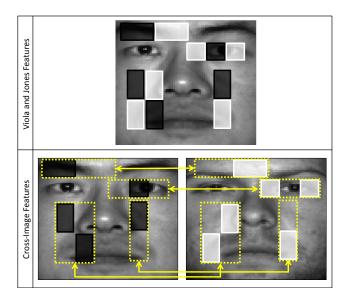

Figure 1. The top row shows the four rectangle features from [20]. The bottom row shows the corresponding cross-image features, which are similar to the original Haar-like features except that the white part of the filters are obtained from the second image in order to capture the difference between the two images in the pair.

$$f_i = \sum_{(x,y)\in r_i} I_1(x,y) - \sum_{(x,y)\in r_i} I_2(x,y). \quad (1)$$

We use four versions of this type similar to [20]; however, the features are obtained across the image pair as shown in figure 1. The black part of the rectangle boxes is obtained from the first image, while the white part is obtained from the second.

- NCC cross-image filters: This type of filter computes the normalized cross correlation (NCC) of the rectangular box between the image pair

$$f_j = \sum_{(x,y)\in r_i} \frac{(I_1(x,y) - \bar{I}_1)(I_2(x,y) - \bar{I}_2)}{\sigma_1 \sigma_2} \quad (2)$$

We use a single version of this feature where the NCC is obtained between corresponding patches in the pair at the same location (i.e. both black and white boxes are placed at the same spatial coordinates in the pair). Note that since the correlation is normalized, these features are robust against illumination changes.

We quantize all possible positions, sizes, and orientations for each of the filters and obtain about $25,000$ features for each pair of images. Calculating the cross-image features for thousands of image boxes is time consuming. Therefore, in order to rapidly compute the features, we adopt the integral image method [20] and apply it on the

cross-image features. The integral image δ for an image I is defined as

$$\delta(x,y) = \sum_{x' \leq x, y' \leq y} I(x', y'). \qquad (3)$$

Each box summation in I can be obtained using four anchors from $\delta(x,y)$ similar to [20]. For the Haar-like cross-image filters, we first obtain the integral image for each of the images in the pair. Consequently, every box summation is obtained from its corresponding integral image. On the other hand, in order to use the integral image for efficient computation of the NCC cross-image features, we first expand equation 2 and apply a few simple operations to reformulate it as

$$f_j = \frac{n \sum I_1 I_2 - \sum I_1 \sum I_2}{\sqrt{(n \sum I_1^2 - \sum^2 I_1)(n \sum I_2^2 - \sum^2 I_2)}}, \qquad (4)$$

where the summation is over all the pixels in the box filter, and n is the number of pixels. Consequently, we obtain five integral images corresponding to each of the terms, in particular, $(I_1, I_2, I_1 I_2, I_1^2, I_2^2)$. Using these internal images, each of the summation terms in equation 4 is efficiently computed using four anchors from the corresponding integral image.

In the training process, we use AdaBoost to select a subset of features and construct the classifier. In each round, the learning algorithm chooses from a heterogenous set of filters, including the Haar-like filters and the NCC filters. The AdaBoost algorithm also picks the optimal threshold for each feature. The output of AdaBoost is a classifier which consists of a linear combination of the selected features. For details on AdaBoost, the reader is referred to [20].

3. Experiments

We extensively experimented on the proposed ideas using three standard datasets: Extended Yale B [4], CMU PIE [1], and Labeled Face in the Wild (LFW) [10]. Figure 2 shows the the highest weighted rectangle features obtained after boosting of the cross-image features for theses three datasets. Figure 3 shows example verification results from the three datasets. In all experiments, we report our performance using the accuracy at EER (100%-ERR) similar to [17], where EER is the equal error rate, which is the average value of the false accept rate (FAR) and the false reject rate (FRR).

- **Extended Yale B**: This is a standard face database consisting of 2414 frontal-face images of 38 individuals. The face images are normalized to the size of
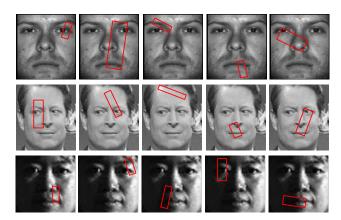

Figure 2. The highest weighted rectangle features obtained after Boosting for Extended Yale B dataset (top), LFW dataset (middle), and CMU PIE dataset(bottom).

192×168. These face images are captured under different laboratory-controlled lighting conditions [12]. There are about 64 images for each individual. We follow a standard experimental setup similar to [23] and randomly select half of the images of each subject for training and the other half for testing. Figure 4 top left shows the obtained ROC curve for Extended Yale B dataset. We compared our method with state-of-the-art methods in table 1 and figure 4. It is clear that our cross-image features outperforms the other approaches.

Table 1. Face verification results for Extended Yale B and CMU PIE datasets.

Method	Accuracy at EER (%)		
	YaleB	PIE Illum	PIE Light
Heusch et al. [8]	73.64	84.85	89.63
Zhang et al. [24]	85.09	79.40	82.77
WLBP-HS [5]	88.46	86.80	90.07
WLFuse [5]	91.25	86.89	90.83
Our method	**95.70**	**92.49**	**98.61**

- **CMU PIE**: This dataset contains 68 subjects with 13 different poses and 4 different expressions under 43 different lighting conditions. We follow the experiment setup in [5] and use two subsets from this database, namely, "illumination" (without ambient light) and "lighting" (with ambient light). Similar to [5], in the illumination subset we use 3 images per person for training, and another 8 images with different illumination conditions are randomly selected for testing. In the lighting subset, we use 5 images for training and 10 for testing. Table 1 shows the performance of our method compared to several other approaches.
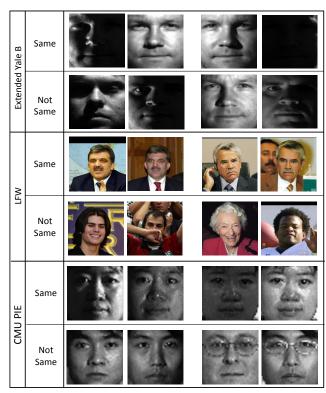
Figure 3. Example face verification results from Extended Yale B, LFW, and CMU PIE datasets.
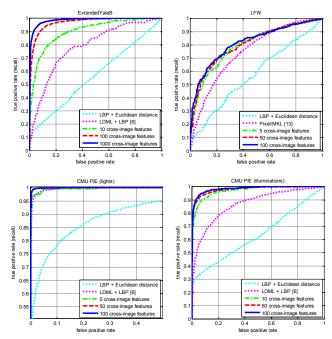

Figure 4. The performance of our method using the highest weighted cross-image features. In the ROC curves, we compare our method with LBP + Euclidean distance, where the face similarity is measured by the distance between LBP features extracted from the faces. Additionally, we compare our performance with the methods from [6] and [15].

Additionally, figure 4 bottom left and right shows the obtained ROC curve for CMU PIE lighting (left) and illumination (right).

- **Labelled Faces in the Wild**: LFW dataset contains 13233 face images collected from the web. The database includes 5749 individuals, 1680 of them have two or more distinct photos. This dataset is very challenging due to the variation in illumination and pose; therefore, it is useful for comparing the effectiveness of different low-level feature descriptors [17]. In our experiments, we use the aligned version of LFW, and follow a standard 10-fold cross validation suggested by the authors of [10]. Figure 4 top right shows the obtained ROC curve for LFW dataset. Moreover, table 2 compares our cross-image features with the state-of-the-art feature descriptors. It is clear that our method is robust to the challenging factors in LFW, and therefore outperforms other features.

Table 2. Comparison of the accuracy at EER for different feature descriptors on LFW dataset

Feature	TPLBP [5]	SIFT	look-alike [17]	ours
Accuracy(%)	69.2	69.1	70.8	**75.4**

4. Conclusion

We proposed a new robust features for face verification based on cross-image similarity. Our approach extracts simple rectangle features from the pair of images jointly, thus capturing discriminative properties of the pair. Through experiments, we demonstrated the power of our proposed approach on challenging datasets. In the future, we will explore the extension of our cross-image features for general object verification problem.

References

[1] S. Baker and M. Bsat. The cmu pose, illumination, and expression (pie) database of human faces. the robotics institute, cmu.

[2] B. Chen, Y. Kuo, Y. Chen, K. Chu, and W. Hsu. Semi-supervised face image retrieval using sparse coding with identity constraint. *ACM Multimedia*, 2011.

[3] M. Everingham, J. Sivic, and A. Zisserman. "Hello! My name is... Buffy" – automatic naming of characters in TV video. In *BMVC*, 2006.

[4] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face

recognition under variable lighting and pose. *TPAMI*, 2001.

[5] Y. Goh, A. Teoh, and M. Goh. Wavelet local binary patterns fusion as illuminated facial image preprocessing for face verification. *Expert Systems with Applications*, 2011.

[6] S. Gu, Y. Zheng, and C. Tomasi. Critical nets and beta-stable features for image matching. *ECCV*, 2010.

[7] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *ICCV*, 2010.

[8] G. Heusch, Y. Rodriguez, and S. Marcel. Local binary patterns as an image preprocessing for face authentication. In *International Conference on Automatic Face and Gesture Recognition (FGR)*, 2006.

[9] R. Hsu, M. Abdel-Mottaleb, and A. Jain. Face detection in color images. *TPAMI*, 2002.

[10] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, University of Massachusetts, Amherst, 2007.

[11] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009.

[12] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *TPAMI*, 2005.

[13] H. Nguyen and L. Bai. Cosine similarity metric learning for face verification. *ACCV*, 2011.

[14] B. Ni, S. Yan, and A. Kassim. Contextualizing histogram. *CVPR*, 2009.

[15] N. Pinto, J. DiCarlo, and D. D. Cox. How far can you get with a modern face recognition test set using only simple features? *CVPR*, 2009.

[16] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *TPAMI*, 1998.

[17] F. Schroff, T. Treibitz, D. Kriegman, and S. Belongie. Pose, illumination and expression invariant pairwise face-similarity measure via doppelgänger list comparison. In *ICCV*, 2011.

[18] W. Tsao, A. Lee, Y. Liu, T. Chang, and H. Lin. A data mining approach to face detection. *Pattern Recognition*, 2010.

[19] M. Turk and A. Pentland. Face recognition using eigenfaces. In *CVPR*, 1991.

[20] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001.

[21] L. Wiskott, J. Fellou, N. Kuiger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *TPAMI*, 1997.

[22] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, 2011.

[23] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *TPAMI*, 2009.

[24] T. Zhang, B. Fang, Y. Yuan, Y. Y. Tang, Z. Shang, D. Li, and F. Lang. Multiscale facial structure representation for face recognition under varying illumination. *Pattern Recognition*, 2009.