POSE-ROBUST REPRESENTATION FOR FACE VERIFICATION IN UNCONSTRAINED VIDEOS

Hoang Anh B. Nguyen, Wen Li

School of Computer Engineering, Nanyang Technological University, Singapore

ABSTRACT

Face recognition in unconstrained videos has been actively studied with the increasing popularity of surveillance and personal cameras. Compared with the traditional task of face recognition in images, recognizing faces in unconstrained videos is much more challenging due to the large variations in poses, expressions and lighting conditions. To handle varying poses in the videos, in this paper, we propose a two-level representation approach for face verification in unconstrained videos. Specifically, we first recover the full-pose face representation for each video which contains all the pose categories ranging from the most left to the most right profile faces. The missing poses are synthesized using keyframes of known ones. Then, we further propose a cross-pose video pair representation for face verification task, which consists of the similarity scores of all frame-level pairs across two videos. Extensive experiments on the benchmark YouTube Faces (YTF) dataset clearly demonstrate the effectiveness of our proposed method.

Index Terms— Face recognition, video representation.

1. INTRODUCTION

With the rapid adoption of surveillance systems, digital cameras and handphones, the face recognition problem has been widely studied over the last decades. Generally, face recognition applications fall into two categories: face identification which aims to identify the subject of each face, and face verification which aims to decide whether a pair of faces are from the same subject. While many face recognition systems have shown promising results under restricted environments with cooperative users, face recognition under unconstrained environments remain challenging due to large variations in poses, lighting conditions, expressions, and so on. Many works have studied the image-based face identification [1, 2, 3, 4, 5, 6] and face verification [7, 8, 9, 10] problems under unconstrained environments. However, for the video-based face recognition, previous works mainly focus on the face identification problem [11, 12, 13, 14, 15, 16], only few works [17] have been proposed for the face verification under unconstrained environments.

Under unconstrained settings, face recognition in videos is much more challenging than that in images because videos usually contain more intra-class variations in terms of poses, expressions and lighting conditions. Among all these difficulties, pose variation is considered to be most challenging as pointed out in previous works [18, 19]. Different videos may contain different poses and the number of keyframes for each may be different as well. As a result, the two videos of different persons with an overlapped set of poses may appear more similar, when compared with the videos of captured under completely different viewpoints. Hence, using traditional methods to directly compare the videos may not achieve promising results.

In this paper, we present a new method for face verification in unconstrained videos. To explicitly handle the large pose variance in different videos, we first recover full-pose representation for each video which covers all the poses from the most left to the most right profile faces. Specifically, the keyframes are first divided into \( k \) predefined pose categories according to the detected head poses. For pose categories with no keyframes, we synthesize face frames using either a flip operation or a generative model with keyframes of known poses. After that, the keyframes in each pose category are averaged to obtain a representative frame, which both simplifies the problem and reduces the possible variances in expressions and lighting conditions.

The full-pose representation helps to capture the pose structure of each video. However, we are more interested in the specific face verification task in which face matching is performed between each pair of videos. In order to better utilize the information of different poses during the matching stage, we further propose a pair-level cross-pose representation for each pair of videos. In particular, we use \( k \times k \) classifiers targeting at \( k \times k \) different pose combinations between any two videos. For each pair of input videos, we populate the \( k \times k \) pairs of keyframes between them and use the corresponding classifier to obtain the output similarity. Then, the \( k \times k \) outputs from these classifiers are concatenated into a single feature representation for pose-robust face verification. Finally, an SVM classifier is trained on this feature representation to perform the face verification task. Figure 1 shows the pipeline of our proposed method for a pair of input videos.

Our method significantly outperforms the state-of-the-art methods on the YouTube Faces (YTF) dataset. Our proposed
scheme achieves 79.12%, 79.58% and 80.92% in terms of recognition rate by using the intensity, LBP and Gabor features, respectively. It can further achieve 82.24% by fusing different types of features, which outperforms the published state-of-the-art result of 76.4% [17].

2. RELATED WORKS

Video-based face recognition can be seen as a problem of measuring the similarities between two video sequences. Previous methods can be generally divided into two main directions considering the use of temporal information of the video. The first one extensively makes use of the spatio-temporal information and dynamic structure of the video when modeling the video sequence. The temporal and face motion have been encoded in the learning of Hidden Markov Models [20] or joint probabilistic model [21]. In the second direction, without the temporal information each video is transformed into an unordered set of images. Researches in this direction modelled the image sets as parametric distributions [15], non-parametric subspaces [14] or manifolds [13, 16].

There are two major components in a verification framework: face representation and face matching. Correspondingly, most previous works can be divided into two main approaches, descriptor-based methods [22, 7, 23] and similarity-based methods [8, 9, 24, 25]. The former one mainly focuses on finding effective features for face representation while the latter one aiming at a novel similarity metric for comparing facial descriptors.

Currently, descriptor-based methods have showed promising result for the face verification task. In [22], Wolf et al. proposed two simple extensions of the Local Binary Patterns feature so as to capture the relationship of neighbourhood patches in addition to the pixel surrounding. In [7], Kumar et al. proposed to use the face attributes and the similarity of faces to specific reference people as features. Taigman et al. [23] also utilized the similarity scores of one face with another training on different negative sets as a high-level representation.

3. OUR APPROACH

In this section, we present the details of our proposed approach for video-based face verification task. The first step is to recover the missing poses for each video such that each video contains all face poses, which leads to our full-pose video representation. Thereafter, we extract the cross-pose video pair representation by concatenating the similarities of all frame-level pairs between the two input videos.

3.1. Full-pose video representation

In unconstrained videos, different videos may contain persons in different poses and the number of keyframes may be different for each pose as well. If two videos of different persons have similar or overlapped set of poses, they may appear more similar than videos of the same person captured under completely different viewpoints. As a result, directly comparing the videos using traditional methods may not achieve satisfying results. To reduce such pose variation in different videos, we therefore propose to represent each video with the full-pose face representation, which covers all the poses from the most left to the most right profile.

Formally, let us denote a video as $V = \{f_i\}_{i=1}^m$ where $f_i$ is a keyframe and $m$ is the total number of keyframes. We first define $k$ pose categories $\{P_j\}_{j=1}^k$ which cover all head poses from the most left to the most right profile. Then, we estimate the head pose of each frame $f_i$ in the original video and allocate this frame to the corresponding pose category accordingly. However, a real-world video usually does not cover all of the $k$ poses but rather only several of them; for example,
in the news or interview videos, the person (narrator) is often captured in frontal view with only slight head movements. As a result, some categories may contain no keyframes, which means the pose variance between different videos still exists.

To fill these categories, we therefore propose two operations to synthesize face frames for them.

**Flip Operation**: Since the face is approximately symmetric, our first solution is to horizontally flip keyframes in the category on one side (e.g., the most left profile) to fill its corresponding category on the other side (e.g., the most right profile) if that is empty. In other words, we flip and allocate the keyframes in category \( P_1 \) to the category \( P_{k-j+1} \) if \( P_{k-j+1} \) does not have any keyframes. With this simple operation, we can reduce the number of missing poses by more than 30%.

**Transfer Operation**: To fill the remaining empty categories, we propose to use keyframes in the non-empty categories to synthesize virtual frames for the empty ones. Particularly, we employ the deformable model in [26]. Although their initial intention is for face alignment, the model can be applied to transfer the face appearance from one viewpoint to another. Similar to [26], we firstly obtain the response maps for 65 fiducial landmarks using the feature point detector in [27] and jointly optimize the locations using the subspace constrained mean-shifts. Then we define the operation to rotate face frames in the non-empty pose categories to each empty pose category. Thus, we obtain a set of synthesized keyframes to fill each empty pose category.

By using the above two operations, we have represented the video with a set of pose categories, each of which contains some keyframes with similar poses. To further reduce expression and lighting variations within each category, we propose to average all the keyframes in one pose category into a single face frame. Thus, we finally represent each video as a set of keyframes, each of which represents a distinct pose and all of which cover the poses from the most left to the most right profile, i.e., \( V = \{r_j\}^{k}_{j=1} \) frames where \( r_j \) is the average frame for the \( j \)-th pose category.

### 3.2. Cross-pose video pair representation

In the face verification setting, one is given a set of video pairs divided into two categories, same-person and different-person pairs. Specifically, a same-person video pair \( (V_1, V_2) \in S \) means that the two videos are from the same subject, while \( (V_1, V_2) \in D \) means that the videos are from two different subjects. The task of face verification is to predict whether two unseen test videos are from the same person or not.

The face verification task can be treated as a binary classification problem, in which each training sample is a pair of videos. As most traditional classification methods were designed for training samples represented as feature vectors, we propose to represent each video pair into one single high-level feature vector by using a cross-pose video pair representation. In this high-level feature vector, each entry is the similarity of a frame-level pair, i.e., a pair of keyframes between the two input videos. Since each video contains exactly \( k \) average keyframes corresponding to \( k \) pose categories, there are \( k \times k \) frame-level pairs between the two videos. The similarity of a frame-level pair is the output of the classifier learnt from available training data.

Recall that in one video, each keyframe represents a distinct pose, so the \( k \times k \) frame-level pairs actually represent \( k \times k \) cross-pose pairs of the faces from the input videos. As one universal classifier may not well model the similarity measurement for all cross-pose pairs, we propose to learn \( k \times k \) cross-pose classifiers, each corresponds to one cross-pose combination. Formally, let us denote the classifier trained on a cross-pose pair \( (P_1, P_2) \) as \( C_{ij} \), where \( P_i \) is the \( i \)-th pose in the first video and \( P_j \) is the \( j \)-th pose in the second video with \( i, j = 1, \ldots, k \). To train each individual classifier \( C_{ij} \), we employ the Side-Information Linear Discriminant analysis (SILD) [24] approach, which extends the traditional Linear Discriminant Analysis by defining the between-class and within-class scatter matrices using only the side information.

In the training process, we divide frame-level pairs from the \( n \) training video pairs into \( k \times k \) non-overlapped subsets \( \{G_{ij}\}_{i,j=1,\ldots,k} \). Specifically, for the video pair \( V_1 = \{r_{1i}\}_{i=1}^{k} \) and \( V_2 = \{r_{2j}\}_{j=1}^{k} \), we populate all cross-pose frame-level pairs between the two videos as \( \{(r_{1i}, r_{2j})\}_{j=1}^{k} \). The pair of keyframes \( (r_{1i}, r_{2j}) \) is allocated to the set \( G_{ij} \) for training the classifier \( C_{ij} \). Therefore, each subset \( G_{ij} \) contains \( n \) frame-level pairs, in which the first frame in each pair belongs to pose \( P_i \) while the second frame belongs to pose \( P_j \). Following [24], the within-class scatter matrix \( S_{ij}^{w} \) and between-class scatter matrix \( S_{ij}^{b} \) for the pose combination \( (i, j) \) are respectively defined as:

\[
S_{ij}^{w} = \sum_{(r_{1i}, r_{2j}) \in S} (r_{1i} - r_{2j})(r_{1i} - r_{2j})^T, \tag{1}
\]

\[
S_{ij}^{b} = \sum_{(r_{1i}, r_{2j}) \in D} (r_{1i} - r_{2j})(r_{1i} - r_{2j})^T. \tag{2}
\]

Similar to LDA, the projection matrix for pose combination \( (i, j) \) is calculated by solving this optimization problem:

\[
w_{ij} = \arg \max_u \frac{u^T S_{ij}^{b} u}{u^T S_{ij}^{w} u}. \tag{3}
\]

After projecting the original frame-level pair \( (r_{1i}, r_{2j}) \) to a lower dimension using \( w_{ij} \), the classification score \( s_{ij} \) can be obtained by calculating their cosine similarity. Finally, we concatenate the scores from all \( k \times k \) frame-level pairs to form a feature vector of length \( k^2 \) for each pair of videos.

Finally, a binary SVM classifier with RBF kernel [28] is trained on the cross-pose features obtained from the training videos. For any given pair of test videos \( V_1 \) and \( V_2 \), we first represent them as a single feature vector and then use the SVM classifier to predict whether they are from the same person or not.
4. EXPERIMENTAL RESULTS

We conduct the experiments on the benchmark YouTube Faces (YTF) dataset [17] to evaluate our proposed method for face verification in unconstrained environment. The YTF dataset contains 3425 videos of over 1595 subjects. We follow the standard benchmark setting on the YTF dataset, which consists of 5000 video pairs equally divided into 10 independent splits, with 250 same-class pairs and 250 different-class pairs in each split. In all experiments, each keyframe is cropped to the size of $150 \times 80$ pixels and allocated to 1 of 7 pose categories according to the detected head pose provided in YTF dataset.

To obtain the full-pose video representation, we employ the pre-trained model from FaceTracker library\(^2\) to synthesize missing pose categories. For the cross-pose video pair representation, we adopt several well-known features in face recognition including intensity, Local Binary Patterns [29] and Gabor wavelet feature [30]. Due to memory limitation, we apply down-sampling and PCA with 98% energy preserved, which leads to about 500 dimensions for these feature vectors. For the cross-pose classifiers projection, 30% of the columns are kept as suggested in [24]. Following the standard setting on YTF dataset, results are reported in terms of recognition rate and standard error by using 10-fold cross validation. There are two main components in our framework as (a) the full-pose video-level representation and (b) the cross-pose pair-level representation for video verification. In order to evaluate the individual contribution of each part towards our final result, we adopt three experiments for comparison:

- **Original**: This is the baseline without using both representations. Instead, one SILD classifier is trained on all frame-level pairs between two input videos and the video similarity is calculated by averaging classification scores of all pairs.
- **Video level representation**: We employ the full-pose video representation in which each video is represented by 7 average keyframes across all pose categories. Similar to the previous baseline, one SILD classifier is used to make the prediction.
- **Pair level representation**: Our unified representation consists of full-pose video descriptor and then learning the similarities between every cross-pose frame-level pairs to form the high-level feature.

<table>
<thead>
<tr>
<th></th>
<th>Intensity</th>
<th>LBP</th>
<th>Gabor</th>
</tr>
</thead>
<tbody>
<tr>
<td>original</td>
<td>77.12±1.51</td>
<td>76.68±2.06</td>
<td>78.24±1.48</td>
</tr>
<tr>
<td>video level</td>
<td>77.60±1.43</td>
<td>77.98±2.16</td>
<td>79.26±1.55</td>
</tr>
<tr>
<td>pair level</td>
<td>79.12±1.81</td>
<td>79.58±2.01</td>
<td>80.92±1.10</td>
</tr>
</tbody>
</table>

Table 1. Recognition rate of our representations and the baseline with different feature types.

The results of the above three experiments are summarized in Table 1. The full-pose video representation helps to improve the results by around 1% from the baseline for all the feature types, which clearly demonstrates that our full-pose video representation better reduces the intra-person variance and helps to normalize all the pose differences between videos.

Moreover, we also observe that using the pair level representation helps to boost the result by more than 1.5%. Since each element in the new feature vector is the similarity score extract from the corresponding cross-pose classifier, the resultant similarity is thus more robust to pose changes. The result reveals that the cross-pose feature is highly suitable for the verification task as it captures the similarities of two faces in every aspects. Finally, our proposed pose-robust representation improves the recognition rate over the original feature by 2.00%, 2.90% and 2.68% for the intensity, LBP and Gabor features, respectively.

Finally, we combine all three feature types with an additional layer of SVM. Figure 2 shows the performance comparison of our proposed method to the state-of-the-art results, in which MBGS [17] is currently the best published result on YTF dataset obtained by training a discriminative classifier for each video sequence. We also include the ROC curves from two recent papers, APEM-FUSION [25] and STFRD+PMML [10], which are accepted for publication when this paper is under review. Our fusion method achieves the best recognition rate of $82.24 \pm 1.14$, which significantly outperforms the published state-of-the-art of $76.4 \pm 1.8$ reported in [17] by 5.8% as well as all latest results from this year [31, 25, 10] by almost 3%.

![Fig. 2. Performance comparison on the YTF dataset.](image)

5. CONCLUSION

We have introduced a novel two-level representation for video-based face recognition by leveraging the pose information. Our two main contributions include a video-level descriptor that helps to reduce intra-person pose variance and a video pair-level representation for face verification task. We achieve better performance than the original feature representation on the YouTube Faces dataset using different features. We also have shown that our proposed method significantly outperforms the current state-of-the-art results.

\(^2\)https://github.com/kylemcdonald/FaceTracker
6. REFERENCES


