Pose-invariant face recognition with homography-based normalization

Xiangxiang Fang
2017.05.02
1. Background

2. Method
   2.1 Homography-based pose normalization
   2.2 HPN-based face representation

3. Experiments

4. Conclusion
1. BACKGROUND

- PIER (Pose-invariant face recognition)

  Pose variations
  Image blur and illumination variation

- Existing pose normalization approaches

  2D methods
  3D methods

Fig. 1. Sample images with dramatic pose variations and other factors.
• A pose normalization approach that combines the advantages of both 3D methods and 2D methods.

• First, a grid of dense 3D facial landmarks are projected to the 2D image.

• Next, the transformation of the local patches across pose is efficiently approximated by homography based on landmarks in the patch.
2.METHOD

2.1 Homography-based pose normalization

- **Dense semantic correspondence establishment**
  - Pose estimation
  
  We estimate the pose by the model:
  
  $$\min_{p} \| x - Xp \|^2_2$$

  - $x \in R^{10d}$ is a vector containing the coordinates of five 2D facial landmarks.
  - $X \in R^{10x8}$ is a matrix composed of coordinates of five 3D facial landmarks labeled on the 3D model.
  - $p \in R^{8d}$ is a vector containing the similarity transformation parameters in the orthogonal projection model $T$.

Fig. 2. Principle of dense semantic correspondence establishment.
2. METHOD

2.1 Homography-based pose normalization

- **Dense semantic correspondence establishment**

- 3D-to-2D projection

1) They manually label a grid of 328 facial landmarks which are uniformly distributed on the BFM model.

2) With the estimated projection model parameters $p$, we project the 328 3D landmarks to the 2D face image.
2. METHOD

2.1 Homography-based pose normalization

Patch-wise frontal-view synthesis by homography

\[ x_p^i = \alpha Hx_0^i \]

- \( x_p^i \) is the coordinates of the i-th landmark in the current pose.
- \( x_0^i \) stands for the corresponding landmark in the canonical pose.
- \( \alpha \) is a constant related to the patch.
- \( H \) is a \( 3 \times 3 \) homography matrix.

Fig. 3. Illustration for homography-based patch correction. (a) HPN corrects the patch around each facial landmark one by one; (b) a local patch that covers 9 facial landmarks is cropped. (c) pose normalized patch.
2.2 HPN-based face representation

- Feature extraction: Dual-Cross Patterns (DCP)
- Extract fixed-length face representations across pose based on facial symmetry.
- Using the following method to evaluate the visibility:

\[
\gamma(p_i) = \max \left( -\frac{c^T n_i}{\|c\| \|n_i\|}, 0 \right) = \max \left( -\frac{c^T Rn_i^0}{\|c\| \|Rn_i^0\|}, 0 \right)
\]

- Fuse the feature vectors of the i-th patches from the original face image and its horizontally flipped version by the weighted summation according to their visibility.
2.2 HPN-based face representation

- To reduce redundancy, the representation dimensionality is reduced by Whitened Principal Component Analysis (WPCA) algorithm:

\[ y = \left( U \Lambda^{-\frac{1}{2}} \right)^T x \]

- The similarity score \( \rho \) between two face images \( y_1 \) and \( y_2 \) in the low-dimensional subspace is measured by the cosine metric:

\[ \rho(y_1, y_2) = \frac{y_1^T y_2}{\|y_1\| \|y_2\|} \]
3. EXPERIMENTS

Fig. 4. (a) Sample images of the original Multi-PIE face images. The yaw angles are $-45^\circ$, $-15^\circ$, $+15^\circ$, and $+45^\circ$, respectively. (b) Examples of synthesized frontal face images by HPN.
3. EXPERIMENTS

- Effectiveness of homography-based pose normalization (on Multi-PIE database)

Fig. 6. Performance comparison between HPN and PAF. HPN promotes the performance of PIFR algorithms by a large margin.

Table 1. Performance comparison between HPN and PAF on Multi-PIE.
In this paper, they propose a highly efficient pose normalization approach named HPN which is based on homography.

HPN effectively handles the three major challenges for PIFR.

To improve the ability of HPN to recognize unconstrained large-pose face images.
Face recognition in real-world images

Xiangxiang Fang
2017.05.02
1. Background

2. Method
   2.1 Automatic face alignment
   2.2 Face recognition

3. Experiments

4. Conclusion
1. BACKGROUND

- Face recognition

  Face verification: classify a pair of pictures as belonging to the same individual or not;
  Face identification: put a label on an unknown face with respect to some training set.

- Robust Sparse Coding (RSC)
2.1 Automatic face alignment

- Face detection—the Viola-Jones approach (OpenCV)
  Haar-like feature:

![Haar-like feature diagram](image)

Fig. 1. Haar-like feature
2. METHOD

2.1 Automatic face alignment

- Landmarks detection—the regression tree method of Kazemi and Sullivan (Dlib)

Fig. 2. The 68 detected landmarks
2.METHOD

2.1 Automatic face alignment

- Face warping
  - Compute the Delaunay triangulation mesh that covers the entire face image.
  - Warping—affine transformation:

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} = \begin{bmatrix}
  a & b \\
  c & d
\end{bmatrix} \begin{bmatrix}
  x \\
  y
\end{bmatrix} + \begin{bmatrix}
  t_x \\
  t_y
\end{bmatrix}
\]

Fig. 3. The steps of alignment using our method on LFW sample image of G.W. Bush. Output (f) is cropped and resized after alignment.
2.METHOD

2.2 Face recognition

- The Robust Sparse Coding (RSC)
  The RSC algorithm solves the weighted-LASSO problem:
  \[
  \min_x \left\| W^{\frac{1}{2}} (y - Dx) \right\|_2^2 \quad s.t. \quad \|x\|_1 \leq \varepsilon
  \]
  - \( \varepsilon \) is a constant representing the noise level.
  - The approach in the paper:
    \[
    \min_x \left\| W^{\frac{1}{2}} (y - Dx) \right\|_2^2 + \lambda \|x\|_2^2
    \]
    - With \( \lambda > 0 \), which has the analytical solution:
      \[
      x = \left( D^T WD + \lambda I \right)^{-1} D^T W y
      \]
3. EXPERIMENTS

The more challenging LFW database is used in this paper.

- Three recognition experiments:
  1. Applying RSC algorithm on the original LFW dataset.
  2. Applying RSC algorithm on the faces detected on the LFW images.
  3. Applying our modified RSC algorithm, after performing our alignment step on the LFW images.

(a) Original image    (b) Triangulation    (c) Warped

Fig. 4. Mesh warping examples on pictures
3. EXPERIMENTS

<table>
<thead>
<tr>
<th>Method</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>10.6%</td>
</tr>
<tr>
<td>SRC [7]</td>
<td>22.3%</td>
</tr>
<tr>
<td>PCRC [13]</td>
<td>25.0 ± 1.8%</td>
</tr>
<tr>
<td>SVDL [10]</td>
<td>30.2%</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td><strong>33.3 ± 3.4%</strong></td>
</tr>
</tbody>
</table>

Our time 0.02 s

Table 3: Recognition rates on the LFWa dataset for the extreme case of using a single training sample per person.

<table>
<thead>
<tr>
<th>Method</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exp. 1</td>
<td>19.6</td>
</tr>
<tr>
<td>Exp. 2</td>
<td>28.8</td>
</tr>
<tr>
<td>Exp. 3</td>
<td>76.4</td>
</tr>
<tr>
<td>Time for one image (s)</td>
<td>3.2</td>
</tr>
</tbody>
</table>

Table 1: Results on the LFWa database with 7 training images and 3 test images.

<table>
<thead>
<tr>
<th>Method</th>
<th>2</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>9.3 ± 1.7%</td>
<td>14.3 ± 1.9%</td>
</tr>
<tr>
<td>SRC [7]</td>
<td>24.4 ± 2.4%</td>
<td>44.1 ± 2.6%</td>
</tr>
<tr>
<td>CRC [12]</td>
<td>27.4 ± 2.1%</td>
<td>42.0 ± 3.2%</td>
</tr>
<tr>
<td>MSPCRC [13]</td>
<td>35.0 ± 1.6%</td>
<td>41.1 ± 2.8%</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td><strong>51.1 ± 2.9%</strong></td>
<td><strong>74.2 ± 2.5%</strong></td>
</tr>
</tbody>
</table>

Our time 0.15 s 0.85 s

Table 2: Recognition rates on the LFWa dataset for different methods and with 2 and 5 training samples per person.
4. CONCLUSION

- The paper presented a computationally efficient face recognition technique for real-world images that requires less than 10 training examples and ordinary hardware to deliver near real-time recognition.
- Compared to existing state-of-the-art this approach nearly doubles the recognition rate while halving the computational runtime.
- We presented results on the LFW dataset that shows that our method significantly outperforms existing non-deep learning algorithms.