Affine Local Descriptors for Viewpoint Invariant Face Recognition

Yongbin Gao*, Hyo Jong Lee*
*Division of Computer Science and Engineering,
Center for Advanced Image and Information Technology,
Chonbuk National University

Abstract

Face recognition under controlled settings, such as limited viewpoint and illumination change, can achieve good performance nowadays. However, real world application for face recognition is still challenging. In this paper, we use Affine SIFT to detect affine invariant local descriptors for face recognition under large viewpoint change. Affine SIFT is an extension of SIFT algorithm. SIFT algorithm is scale and rotation invariant, which is powerful for small viewpoint changes in face recognition, but it fails when large viewpoint change exists. In our scheme, Affine SIFT is used for both gallery face and probe face, which generates a series of different viewpoints using affine transformation. Therefore, Affine SIFT allows viewpoint difference between gallery face and probe face. Experiment results show our framework achieves better recognition accuracy than SIFT algorithm on FERET database.

Keypoints: Face recognition, Lucas-Kanade, Scale Invariant Feature Transform.

1. Introduction

Face recognition is widely investigated for last decades, especially for robust face recognition algorithms that are able to deal with real world face recognition, such as identifying individuals from surveillance camera for public security and annotating people from digital photos automatically. There are some successful commercial face recognition systems available like Google Picasa and Apple iPhoto [1]. However, face recognition research is still far from mature. Earlier face recognition algorithms are only effective under controlled settings, such as the probe and gallery images are frontal. This algorithm fails when it is applied to cases as pose and illumination changes. This paper focuses on the viewpoint invariant face recognition, which identify face when probe faces are from different viewpoints while gallery faces are frontal.

The key issue for face recognition under different viewpoint is the distance between different poses is bigger than distance between different subjects. One solution is to eliminate the distance between different poses. Among which, face normalization is an effective method to remove the pose difference. Face normalization can be used as 2D or 3D model. As for 2D model, Markov Random Fields (MRF) is widely used to find corresponding between frontal face with the profile probe faces [2, 3]. MRF is to find 2D displacement by minimize the energy, which consists of two parts, one is distance of corresponding node, another one represent the smoothness between neighbour nodes. Lucas-Kanade method is also used for face alignment [4, 5]. As for 3D model, Blanz et al. proposes an effective 3D morphable method to fit the 3D model to 2D face [6], the fitting shape and texture coefficients are used for face recognition. Normalization method can be used to construct the frontal face from the probe profile face [2]. It can also be used to directly match between images and the matching score represents the similarity between them [3]. These normalization methods are reported effective at the cost of long computation time. It has been reported that two minutes is needed to normalize one face [2]. Marsico et al. proposes a FACE framework to recognize face for uncontrolled pose and illumination changes [7]. It detects some keypoints using STASM algorithm [8], and construct half face by the middle line keypoints, the rest half face is reflected from the constructed half face. This easy method is fast but not robust for it highly depends on the accuracy of keypoints detection, when the keypoints detection fails, the system performances become bad.

Another solution for face recognition under viewpoint change is to design new classifier or new feature. For the new classifier, one shot similarity (OSS) or two shot similarity (TSS) are proposed by introducing another dataset, which contains no probe and gallery images [9]. Each dataset contains different images of a single subject or different subjects viewed from a single pose. Similarity scores between two faces are calculated by the model built by one of faces and the introduced dataset using LDA or SVM. Cross posed face recognition shares similar idea by introducing a third dataset [10]. Faces from different viewpoints are all linearly represented by the introduced dataset using subspace method, similarity between these faces is then calculated indirectly by the linear coefficients. As for new feature extraction, tied factor analysis is proposed to estimate the linear transformation and noise parameters in "identity" space [11]. Local descriptor is also a effective way to deal with affine transformation between two images, such as Harris-Affine [12], Hessian-Affine [13], and Affine SIFT [14] algorithms.

The rest of this paper is organized as follows. Section II reviews the SIFT algorithm. We describe Affine SIFT algorithm and its application to face recognition in Section III. Section IV applies the above algorithm to FERET database, and presents the experiment results. Finally, we conclude this paper with future work in Section V.

2. Scale Invariant Feature Transform

Local features are effective methods for matching and recognition for it is robust to occlusion, scale, rotation or even affine transformation to some extent. Among these algorithms, Scale Invariant Feature Transform (SIFT) is an scale, rotation invariant local feature. It transforms image data into scale-invariant coordinates and localizes the keypoint. Each keypoint is assigned a descriptor. The major steps for SIFT algorithm are as following [15]:

1. Scale-space extrema detection: Image is transformed into different scales and size. Extrema are searched by finding maxima and minima over all scales using a difference-of-Gaussian scheme, which are invariant to scale and orientation.

Difference-of-Gaussian is an approximation to the Laplacian of Gaussian, which can be calculated by the difference of two neighbouring scales as:

$$D(x,y,\sigma)=(G(x,y,k\sigma)-G(x,y,\sigma))*I(x,y) \qquad (1)$$

where $D(x,y,\sigma)$ is the difference-of-Gaussian function, G is the Gaussian function and k is the factor of nearby scales. I is the input image. Extrema are detected by comparing a pixel to its 26 neighbours at the current and adjacent two scales (8, 9, 9 pixels for each scale, respectively).

2. Keypoint localization: Extrema are refined by excluding poor localized or low contrast points by checking the refined location, scale and ratio of principal curvatures. This increases stability of keypoint localization.

3. Orientation assignment: Each keypoint is assigned to one or more orientations based on local image gradient histogram. To provide scale and rotation invariance, local image data is transformed to the corresponding orientation and scale for further keypoint descriptor calculation.

4. Keypoint descriptor: Local keypoint descriptor is calculated around each keypoint by histogram of gradients. The descriptor is transformed into a representation that allows for significant levels of local shape distortion and change in illumination.

A keypoint descriptor is created based on the gradient and orientation in a region around the centre keypoint. The region is weighted by a Gaussian window. The region is divided into 4*4 subregions, and histogram of orientation with number of 8 bins is accumulated for each subregion. For each orientation in the histogram corresponds to the sum of the gradient magnitudes near that direction.

There are several methods reported for image matching and recognition of SIFT algorithm, such as BBF [16], Hough transform [17]. Nearest neighbour is the original and effective matching method for SIFT features. SIFT features are first pre-extracted from gallery images and stored in a database. When matching with a probe image, each SIFT feature from the probe image is compared with all gallery features in database. Nearest neighbour and second nearest neighbour are searched based on the Euclidean distance. The ratio of these two distances is compared with a threshold. Ratio that is smaller than the threshold is considered as a matching face.

3. Affine SIFT

The SIFT is scale and rotation invariant feature, but it is not affine invariant. Affine SIFT is the extension of SIFT algorithm. There are several parameters for affine transformation as:

$$A = H_\lambda R_1(\psi) T_t R_2(\phi) = \lambda \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} t & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \qquad (1)$$

where λ, $\mathbf{R}_t$ and $\mathbf{T}_t$ are a scale parameter, rotated angle, and tilted angle, respectively. Fig. 1 shows the geometric interpretation of these parameters. SIFT algorithm is just scale (λ) and rotation (ψ) invariant. The left t and ϕ are not invariant, Therefore, SIFT algorithm is not fully affine invariant. Affine SIFT is trying to fulfil the t and ϕ invariant.

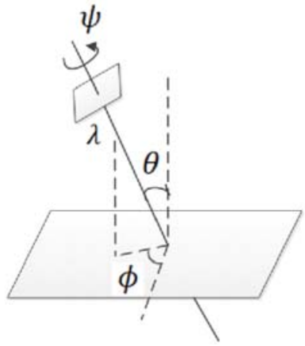

Fig. 1 Geometric interpretation of affine decomposition. λ and ψ are scale and rotation from camera. θ and ϕ is tilt and rotation of subject, which named latitude and longitude respectively. Where $t=1/\cos\theta$.

Affine SIFT transforms a frontal image into a series of simulated images by the change of longitude ϕ and latitude θ. These simulated images are sampled to achieve a balance between accuracy and sparsity. The ASIFT algorithm in detail is as following [14]:

1. The latitude θ is changed as a geometric series 1, a, a^2,…, a^n, where $a>1$. In our experiment, $a=\sqrt{2}$, which is a good compromise between accuracy and sparsity. For digital image, tilt is conducted by a directional t-subsampling with an antialiasing filtered in advance, where $t=1/\cos\theta$.
2. The longitudes ϕ follows an arithmetic series for each tilt as 0, b/t, …, kb/t, where $b \approx 72^\circ$ achieves a balance, and k is the last integer satisfying $kb/t < 180^\circ$.
3. SIFT algorithm is used to detect keypoints from these simulated images.

In our framework, Affine SIFT is adopted to gallery face (frontal face), and probe faces use SIFT algorithm. Affine SIFT is used to detect keypoints and local features for gallery face, and stored in the keypoints database. For the recognition part, SIFT algorithm is used to compute keypoints and local features for each probe face. This SIFT keypoints are compared with Affine SIFT keypoints that stored in the database. The subject that has the maximum number of matching keypoints with the probe face is considered as recognized subject.

4. Experiment Results

In our experiments, we used FERET [18] grey database to evaluate our algorithm. This database contains 200 people, each subject contains 7 images (resolution: 80*80) captured from different pose and illumination. Since this paper focus on face recognition under different viewpoint, we use part of them to test our algorithm. For each subject, we use frontal image as gallery, and other 4 pose images as probe, the pose angle of which are -25, -15, 15, and 25 degrees, respectively.

The parameters used in our experiment for SIFT algorithm are: image is resized to 240*240 resolution, and the ratio for nearest neighbour is set to 0.85. Table I shows the comparison results of recognition with SIFT on FERET database. Where num_tilt is the parameters for Affine SIFT algorithm, which means the transformation time of tilt t. When it sets to 2, ASIFT generates 5 viewpoints for a image (1 viewpoint for $t=1$, and 4 viewpoints for $t=2$). In the table, (2, 2) of column "num_tilt" means the two num_tilt for gallery image and probe image. When $t=1$, Affine SIFT degenerates to SIFT algorithm. From the table, we know that SIFT can get comparable results with Affine SIFT when a pose degree is between -15 to 15 degree, but Affine SIFT achieves better result than SIFT under large pose different.

Table I Experiment results of face recognition on FERET database

Alg	num_tilt	-25	-15	15	25	average
ASIFT	2,2	84.00%	96.00%	96.00%	81.50%	89.38%
ASIFT	2,1	79.50%	95.00%	95.50%	81.00%	87.75%
SIFT	-	75.50%	95.50%	96.50%	82.00%	87.38%

5. Conclusion

In this paper, Affine SIFT is used to detect affine invariant local descriptors for face recognition under viewpoint change. Affine SIFT is an extension of SIFT algorithm. SIFT algorithm is scale and rotation invariant, which is powerful for small viewpoint changes in face recognition, but it fails when large viewpoint change exists. In our scheme, Affine SIFT is used for both gallery face and probe face, which generates a series of different viewpoints using affine transformation. FERET database is used to test Affine SIFT, and experiment results show SIFT can get comparable results with Affine SIFT when a pose degree is between -15 to 15 degree, but Affine SIFT achieves better result than SIFT under large pose different.

Acknowledgement: This work (Grants No. C0112553) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2013. This work was also supported by the Brain Korea 21 PLUS project, National Research Foundation of Korea.

References

[1] G. Hua, M. H. Yang, E. L. Miller, Y. Ma, M. Turk, D.J. Kriegman and T. S. Huang, "Introduction to the Special Section on Real- World Face Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 10, pp. 1921–1924, Oct. 2011.

[2] H. T. Ho, R.Chellappa, "Pose-Invariant Face Recognition Using Markov Random Fields," *IEEE Trans. Image Processing*, vol.22, no.4, pp.1573-1584, Apr. 2013.

[3] S. R. Arashloo and J. Kittler, "Energy Normalization for Pose- Invariant Face Recognition Based on MRF Model Image Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no.6, pp. 1274-1280, June. 2011.

[4] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *International Journal of Computer Vision,* vol. 56, no.3, pp.221 – 255, Mar. 2004.

[5] A. B. Ashraf, S. Lucey and T. Chen, "Learning Patch Correspondences for Improved Viewpoints Invariant Face Recognition," *in Proc. IEEE conf. Computer Vision and Pattern Recognition*, 2008, pp. 1-8.

[6] V. Blanz and T. Vetter, "Face Recognition Based on Fitting a 3D Morphable Model," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1-12, sep. 2003.

[7] M. D. Marsico, M. N. D. Riccio and H. Wechsler, "Robust Face Recognition for Uncontrolled Pose and Illumination Changes," *IEEE Trans. Systems, Man, and Cybernetics: Systems*, vol. 43, no. 1, pp. 149-162, Jan. 2013.

[8] S. Milborrow and F. Nicolls, "Locating facial features with an extended active shape model," *in Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 504–513.

[9] L. Wolf, T. hassner, and Y. Taigman, " Effective Unconstrained Face recognition by Combining Multiple Descriptors and Learned Background Statistics," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 10, pp. 1978-1990, Oct. 2011.

[10] A. Li, S. Shan, and W. Gao, "Coupled Bias–Variance Tradeoff for Cross-Pose Face Recognition," *IEEE Trans. Image Processing*, vol. 21, no. 1, pp. 305-315, Jan. 2012.

[11] S. J. D. Prince, J. H. Elder, j. Warrell, and Fatima, "Tied Factor Analysis for Face Recognition across Large Pose Differences," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 970-982, June. 2008.

[12] K. Mikolajczyk and C. Schmid, "Scale and Affine Invariant Interest Point Detectors," *Int'l J. Computer Vision*, vol. 1, no. 60, pp. 3-86, 2004.

[13] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L.V. Gool, "A Comparison of Affine Region Detectors," *Int'l J. Computer Vision*, vol. 65, no. 1-2, Nov. 2005.

[14] J. M. Morel and G. Yu, "ASIFT, A new framework for fully affine invariant image comparison," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp.438-469, 2009.

[15] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *Int’l J. Computer Vision*, vol. 60, no. 2, pp. 91-110, Nov. 2004.
[16] J. Beis, and D. G. Lowe, “Shape indexing using approximate nearest-neighbour search in highdimensional spaces,” *In Proc. Computer Vision and Pattern Recognition*, Puerto Rico, 1997, pp. 1000-1006.
[17] Hough, P.V.C. 1962. *Method and means for recognizing complex patterns.* U.S. Patent 3069654.
[18] P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss, “The FERET database and evaluation procedure for face-recognition algorithms,” *Image Vis. Comput.*, vol. 16, no. 5, pp. 295–306, Apr. 1998.