Improved Identical Twins Face Recognition with Kin Images

Ivan Chit Hin Ho
Stanford University
Stanford, CA 94305
ivanho1@stanford.edu

Abstract

Recent additions of face recognition technology to mobile devices have prompted discussions on its accuracy when user has identical siblings. There numerous studies on algorithms and techniques to face recognition of identical twins. This paper devises a new algorithm to improve accuracy of identification by correlating features with kin of identical twins .

1. Introduction

Recent additions of face recognition technology to mobile devices has prompted discussions on its accuracy when the user has identical siblings. See `https://9to5mac.com/2017/10/31/face-id-twins/` for example. In academia, there have also been numerous studies (Xu [6], Paone [4], Bowyer [1]) on algorithms and techniques to face recognition of identical twins. Personally, I have always had a hard time recognizing my friend?s twin daughters ever since they were born. As they live in a different city, I rely on looking at their images on social network without name correspondences.

From human experience, one way to improve identification of children is through correlation of features with their parents. In this paper, this technique is applied in computing to see if twin identification could be improved through correlation of images of their kin (mother, father, siblings).

1.1. Baseline Techniques

The following steps and techniques are used to generate a baseline of face recognition pipeline:

1) Identify faces in images (using Histogram of Oriented Gradients:`http://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf` or Convolutional Neural Network from OpenFace: `https://github.com/cmusatyalab/openface/tree/master/models/openface`)

2) Morph various faces usingface landmark estimation (`http://www.csc.kth.se/~vahidk/papers/KazemiCVPR14.pdf`)

3) Generate an 128-point encoding of the face using existing machine learning models from dlib [5]

4) Compute the Euclidean Distance of the unknown image against known images and return the image with the closest Euclidean Distance as match, subject to a tolerance level.

Example of the pipeline on my github: `https://github.com/alrightyi/stanford_cs231a` Much of the baseline algorithm is based on the following open-sourced project: `https://github.com/ageitgey/face_recognition#face-recognition`

1.2. Key Methods

Two Key Methods are used to correlate face encodings with those from the kins.

1) Relative Euclidean Distance:

The basic idea is to calculate the Relative Euclidean Distance (RED) of known image encodings among family members. The result is then used to compare against RED of unknown images against the same family member image encodings.

The family member that has the smallest root-mean-square difference between the known and the unknown RED is selected.

This is implemented in `draw_boxes.py` in `https://github.com/alrightyi/stanford_cs231a`. Algorithm is further elaborated in next section.

2) Key Landmarks

The basic idea is to leverage the key landmarks of face images (*e.g.*chin, left eyebrow, nose tip, etc) among the family members and pick ones that closely resembles one another.

The key landmarks are represented by x and y coordinates on an image plane. If we normalize and center the images we can calculate the Euclidean Distance between two image of the same face landmark.

We can identify the top N key face landmarks that are closest between each pari of family members. For example, a father and son pair may have the closest Euclidean Distance between their left eye brow and chin if setting N=2.

The result is used to compare against key landmarks of unknown face images.

The family member that has the smallest root-mean-square difference between the known and the unknown Euclidean Distance from the top two key landmarks is selected.

Algorithm is further elaborated in next section and it has not yet been implemented.

1.3. Data Sets

Datasets from several sources are used, including: `http://www.kinfacew.com/datasets.html`. Datasets of friends and family from social networks are also used for verification. See github for details.

1.4. Reading Material

2. Proposed Algorithms

1) Relative Euclidean Distance:

a) As part of Step 4) above, compute Relative Euclidean Distances (RED) of the 128-point encodings of known images against the images of their family members.

b) At Step 4), if there are two or more matches to the unknown image (based on tolerance level) and all matches are from the same family, then calculate RED between the unknown image encoding and the known image encodings of the family.

c) Return the match that has the lowest root-mean-square difference between Unknown Image-Family RED vs Known Image-Family RED.

This is implemented in `draw_boxes.py` in `https://github.com/alrightyi/stanford_cs231a`

2) Key Landmarks

a) As part of Step 4) above, identify the key face landmarks from normalized and centered images, using method described by Kazemi and Sullivan [3] i) Chin, ii) Left Eye Brow, iii) Right Eye Brow, iv) Nose Bridge, v) Nose Tip, vi) Left Eye, vii) Right Eye, viii) Top Lip, ix) Bottom Lip

b) As part of Step 4) above, compute Relative Euclidean Distances (RED) of each of the Key Landmarks of known images against the images of their family members.

c) Between any two family members, remember the top N key landmarks that have the closest RED

d) At Step 4), if there are two or more matches to the unknown image (based on tolerance level) and all matches are from the same family, then calculate RED of the top N key landmarks between the unknown image and those from known family member images.

e) Return the match that has the lowest root-mean-square difference between Unknown Image-Family RED vs Known Image-Family RED.

2.1. Work Completed Thus Far

Algorithm of Method 1) has been implemented while Method 2) is still outstanding. See code in github for details.

3. Results

Results of the two methods will be compared against baseline algorithm described in Baseline Techniques

3.1. Method 1: Relative Euclidean Distance

The controlled variables to use in this method are family size and tolerance level, to be plotted against accuracy.

3.2. Method 2: Key Landmarks

The controlled variables to use in this method are family size and number fo key landmarks to use, to be plotted against accuracy.

4. Conclusions

Age group of dataset may play a factor, per Ricanek, Bhardwaj, Sodomsky [2].

5. Acknowledgments

TBD

5.1. Illustrations, graphs, and photographs

TBD

References

[1] K. Bowyer and P. Flynn. Biometric identification of identical twins: A survey. *Biometrics Theory Applications and Systems (BTAS) 2016 IEEE 8th International Conference*, pages 1–8, 2016.

[2] K. R. Jr., S. Bhardwaj, and M. Sodomsky. A review of face recognition against longitudinal child faces. *Lecture Notes in Informatics*, (15), 2016.

[3] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference*, 2014.

[4] J. Paone, P. Flynn, and P. Philips. Double trouble: Differentiating identical twins by face recognition. *Information Forensics and Security IEEE Transactions*, 9:285–295, 2014.

[5] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *Computer Vision and Pattern Recognition (CVPR) 2015 IEEE Conference*, 2015.

[6] F. Xu and M. Savvides. An augmented linear discriminant analysis approach for identifying identical twins with the aid of facial asymmetry features. *Computer Vision and Pattern Recognition Workshops (CVPRW) 2013 IEEE Conference*, pages 56–63, 2013.