Simile Classifiers for Face Classification

Vadim Konushin, Tatiana Lukina, Artem Kuharenko, Anton Konushin
Lomonosov Moscow State University
{vadim, tlukina, akuharenko, ktosh}@graphics.cs.msu.ru

Abstract

We present a new approach to face classification using simile classifiers. Unlike other methods we explicitly estimate similarity distances to the known reference people and use these similarities as high-level features for the classification of the test face.

We test our algorithm on gender classification problem. Our algorithm shows classification accuracy of 92.96% on LFW dataset.

Keywords: *attribute classification, classifier training, gender classification*

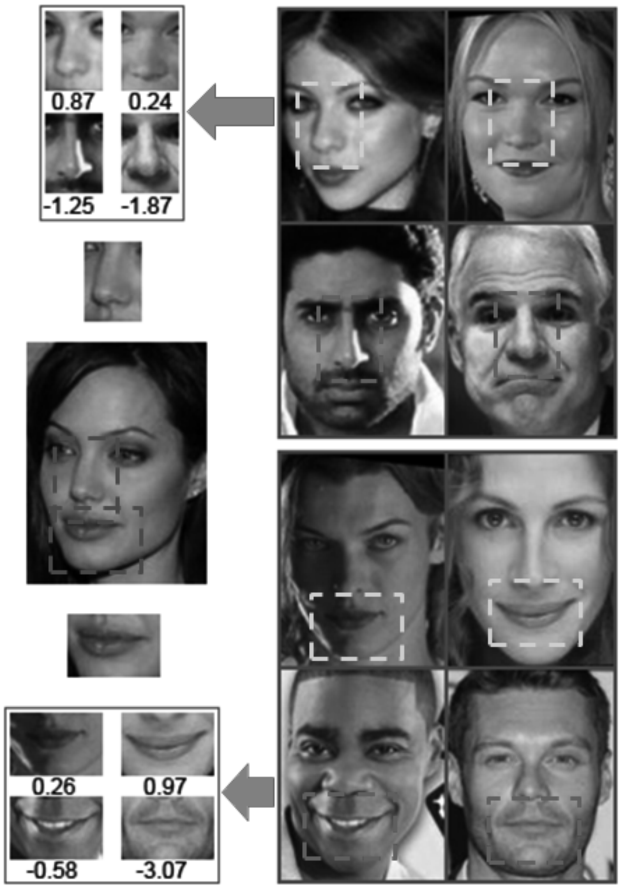

Figure 1. Illustration of our algorithm applied to a gender classification problem. Components of a test face are compared with corresponding components of the reference people. Classification decision is made based on similarities to these people and their gender.

1. INTRODUCTION

Attribute classification from face images is a widely studied problem. Range of attributes includes gender, ethnicity, age, hair style, presence of glasses and so on. Attribute classification is used for content-based image retrieval in web, video surveillance, audience measurement and other applications.

A common approach to face classification is to extract some kind of low-level features from image and then use a machine learning method to find dependencies of the classified attribute on the features. Such methods do not search for similar faces or compare a test face to a set of training faces directly. If SVM classifier is used, a number of training faces are selected as support vectors and used implicitly. In this case the classification results depend on distances to support vectors (in fact, training faces), but the distances are defined by SVM kernels and may correspond poorly to real face similarity.

We propose to explicitly train similarity classifiers to compute distances to the selected set of people and then use these distances as input to a second classifier, which gives the final result. Our work was inspired by the face recognition method [10], where it was proposed to compare people faces based on their attributes (like age, gender, nose size, etc.) and similarities to other "reference" people. For this task they trained so called simile classifiers for individual face components, like eyes or lips.

We use a similar approach, but for the task of face classification. Figure 1 shows an algorithm illustration. We test our algorithm on the gender classification problem and show 92.96% accuracy on LFW dataset [8].

2. RELATED WORK

Most face classification algorithms follow the same pipeline, which consists of face normalization, features extraction and classification. The difference between algorithms is mainly in specific methods, used in these steps.

A common choice for face normalization is consecutive image rotation, scaling and cropping, so that eyes are placed into specific pixels. As features researchers have tried raw pixel intensities [1, 13], Haar-like features [12], Active Appearance Model parameters [2, 6, 12], Local Binary Patterns [11, 17] and Biologically Inspired Features [7, 15], which are based on Gabor functions.

For the classification step boosting [1, 12, 17], SVM [2, 6, 9, 10, 11, 12, 13, 15, 17] and Random Forest [14] have been used, with SVM being the most popular choice.

Several papers [4, 7] investigated an influence of one attribute on another, for example preliminary gender classification on age estimation [7] or ethnicity classification on gender classification [4].

We are not aware of any face classification method, which explicitly searches for similar faces and makes a decision, based on these similarities. Algorithms, which use SVM, do this implicitly by distances to the support vectors, but the used distance metrics are defined by the SVM kernels and may be a bad approximation to the real face similarities.

The most similar method to ours is a face recognition algorithm of [10]. Face recognition algorithms are in general very close to face classification algorithms and often use the same normalization and features. In [10] for face verification it was proposed to use similarities to some other known people as features. For this task special 'simile' classifiers are learned, which, for example, could estimate "how similar are these lips to Angelina Jolie's lips" or "how similar is this nose to the nose of Brad Pitt's".

One another somewhat related method to ours is a recent face recognition algorithm of [19]. For rotated test faces they search for the most similar components of known reference people. During face comparison these components are switched to the components of the found reference people, but in a frontal view. By this trick the algorithm becomes more robust to face out-of-plane rotation, which is possibly the main problem of current face recognition algorithms.

We use a similar approach to face classification. We measure similarity distances of a test person to the set of known people and use their known attribute labels for the classification.

3. PROPOSED ALGORITHM

Our algorithm consists of 2 steps:

- Apply simile classifiers to the test face components and find similarity distances to all reference people
- Use computed similarities as input to the final classifier

3.1 Simile Classifiers

We follow approximately the same approach to simile classifiers training as in the original paper [10]. Some details can be different due to somewhat brief description of implementation in [10].

At first a face is detected by a Viola-Jones face detector [18]. Then we find fiducial points, including eye and lips corners by our own implementation of [3]. Then we select face components, as illustrated on Figure 2. For each component we train separate simile classifiers.

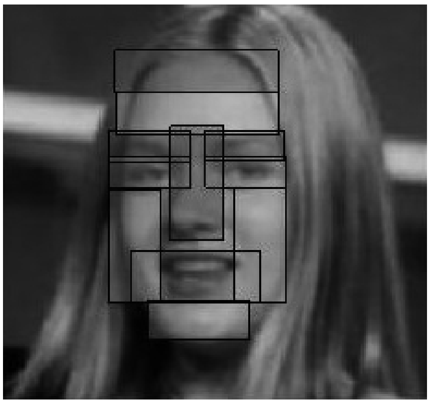

Figure 2. Used components illustration.

For simile classifiers a number of different low-level features are used. Features are constructed by choosing a pixel value type, normalization and the aggregation type. For pixel value types we use colors in RGB and HSV color spaces, intensities, edge magnitudes and orientation. Normalization types include mean normalization $\hat{x} = \frac{x}{\mu}$ or energy normalization $\hat{x} = \frac{x-\mu}{\sigma}$, where x refers to the input value, μ and σ - to the mean and standard deviation of x, $\hat{x}$ is the normalized output value. Aggregation types include a histogram or the mean and variance. All these feature types are summarized in the Table 1.

RGB HSV Image Intensity Edge Magnitude Edge Orientation	None Mean-Normalization Energy-Normalization	None Histogram Statistics
Pixel Value Types	Normalizations	Aggregations

Table 1. Feature types used for simile classifiers.

For each component and for each reference person we train a simile classifier, which recognizes similarity to this component of this specific person. For the positive sample we take the cropped component from different images of this person, and for the negative sample we take the same component images taken from random images of other persons in the training set.

We train a separate SVM classifier with RBF kernel for each feature type, component and reference person.

3.2 Face Classification using Simile Classifiers

At this step we have a $N \cdot M \cdot K$ dimensional descriptor F for a test face, where N stands for the reference people number, M stands for the feature type number and K is a number of components. Each $F_{n,m,k}$ is a similarity between k^{th} components of a test person and n^{th} reference person using m^{th} feature type and in fact is an output of a corresponding SVM classifier.

We consider only the case of binary face classifications. Multi-class classification problems could be reduced to binary problems by one-against-one or one-against-all strategy. In [2] it was proposed to reduce age estimation problem to series of binary classifications, which answer the question "Is a test person older, than given age?" This strategy shows one of the top age estimation accuracies so far.

Let $l_n \in \{-1,1\}$ be a binary label of a reference person n. We need to estimate binary label of a test face $L(F)$

We have tested several classification algorithms, listed below. The resulting accuracies are given in the Experiments section.

- The most simple option is to use a nearest neighbor approach – to take a label of the most similar person considering only 1 component k and 1 feature type m:

  $$n'(m,k) = \arg(\max_n(F_{n,m,k}))$$

  $$L(F) = l_{n'(m,k)}$$

- To use information from all components and features types we can combine all $K \cdot M$ values $F_{n'(m,k),m,k}$ in a single feature vector and feed it to a single SVM classifier with RBF kernel. To account for values of binary labels $l_{n'(m,k)}$, we switch sign of those $F_{n'(m,k),m,k}$, which correspond to reference people with a label $l_{n'(m,k)} = -1$. In our experiments $F_{n'(m,k),m,k}$ was positive for all k, m, so this switch introduces information about label values into feature vector.
- A logical extension is a k-nearest neighbor approach. In this case we find p most similar reference persons for each component and each feature type, concatenate $p \cdot K \cdot M$ corresponding $F_{n,m,k}$ elements (again with sign switching), and input them into a SVM classifier.

- Another approach would be just to sum all $F_{n,m,k}$ elements. We multiply on -1 those $F_{n,m,k}$, for which corresponding $l_n = -1$, and keep other $F_{n,m,k}$ unchanged.

  Note, that in this case some of $F_{n,m,k}$, as SVM outputs, are already negative. So we treat dissimilarity to a reference person as a similarity to a person with the opposite label.

  $$L(F) = \text{sign}(\sum_{n,m,k} F_{n,m,k} + C),$$

  where C is a parameter.

  The more similar are the reference people with label l to the test face, the greater the chance that the test face has the same label l.

  However $F_{n,m,k}$ for different feature types usually have different variations and in $\sum_{n,m,k} F_{n,m,k}$ only some feature types would matter.

  So we summed $F_{n,m,k}$ only with the same feature type, and combined results for different m with AdaBoost or SVM classifier:

  $$L(F) = \text{classifier}(\sum_{n,k} F_{n,1,k}, \sum_{n,k} F_{n,2,k}, ..., \sum_{n,k} F_{n,M,k})$$

- Instead of a simple summing we could use a separate classifier for each feature type. So we have a two-layer classifier: M different SVM classifiers, that take concatenated $N \cdot K$ values $F_{n,m_i,k}$, and one SVM classifier, that takes M outputs of the previous classifiers.
- We also test another variant of two-layer classifier. First, we train K different SVM classifiers for each component, where each classifier uses $N \cdot M$ values F_{n,m,k_i} Then we train one SVM on top of these K outputs.

It is important to note, that in the last two cases, where we use only classifiers, we don't need reference people labels, because an order of reference people in feature vector is fixed. Classifiers automatically learn dependencies, like "similarity to this reference person means, that the test face is more likely to have label 1, and similarity to that reference person probably means, that it has label -1".

4. EXPERIMENTS

As an example of a face classification problem we chose a gender classification because it is a well-studied problem and, unlike age estimation, it is relatively easy to prepare ground truth data.

Unfortunately, unlike face recognition or to some extent age classification, there is no standard test dataset for gender classification problem. In different papers a subset of FERET [16], CAS-PEAL face database [5], FaceTracer [9] and LFW databases were used.

A natural choice for our experiments was to use a face image dataset PubFig [10] and LFW [8] database, as they were used in the original paper [10], where simile classifiers were proposed.

Figure 3. Example images from LFW dataset.

Figure 4. Example images from PubFig dataset.

LFW database consists of 13233 images of 5749 different persons, all images are collected from the web, and the only condition on image is a detected face by Viola-Jones face detector. Nowadays LFW is a standard database for face recognition algorithms comparison.

PubFig is a complement to LFW database, proposed in [10]. It consists of 60000 images of 200 people. The important property of PubFig is that there are at least 50 different images for each person and therefor it is well suited to be a reference people dataset (as there are enough images for simile classifiers).

Example images of LFW and PubFig are shown in Figure 3 and 4. We have manually annotated gender in both PubFig and LFW datasets.

In our experiments we used PubFig Development Set as reference people set (59 people, originally there were 60 people in this set, but the database is uploaded as a list of URLs and some links are currently missing) and 5-fold cross-validation on LFW.

For comparison with existing gender classification methods we selected two algorithms, with testing procedure closest to our:

- In [10] a gender classification along with other attributes classification are used for further face recognition. Exact training and test subsets are not described, but it seems that some parts of PubFig and LFW were used. An accuracy of 81.22% was reported.
- In [17] a 5-fold cross-validation on a subset of LFW was used for the experiments. The faces, that are not near frontal and those, for which it is difficult to establish the ground truth, were not considered. This resulted in 7443 out of 13233 images. The resulting classification accuracy was impressive 94.81%. Authors promised to make this dataset publicly available, but to our knowledge haven't done it yet.

Our results are summarized in Table 2.

Nearest neighbor approach performs poorly: 67% with 1 feature type and 1 component and 77% when combined by one SVM. When we switch it to k-nearest neighbors approach we gradually get better results with increasing k.

But even simple summing of all the similarities $F_{n,m,k}$ gives equal result. Note that summing $F_{n,m,k}$ for only one feature type can result in 84%. Applying classifier to these sum values improves the accuracy up to 89%. SVM with RBF kernel performs better than AdaBoost.

The best approach is to use a two-layer classifier. If on the first level we use separate SVM for each feature type, we get 91.6% (90.8% by a single SVM classifier for the best feature). If we use separate SVM classifier for each component, and then combine them by another SVM, we get our best 92.96%.

It is significantly better than 81.2% of [10], where the same kind of low-level features were used. Though, as we don't know the exact training/testing samples of [10], it could be a result of a smaller training sample size.

We lose to 94.81% of [17], but we have used a more complex test sample (though maybe there is more sense in discarding strongly rotated photos) and possibly less discriminating low-level features.

Algorithm	Classification accuracy
Nearest neighbor, only 1 best combination of feature and component types	67%
Nearest neighbor and SVM	77%
K-nearest neighbors, k = 2, SVM	80.5%
K-nearest neighbors, k = 3, SVM	81.6%
K-nearest neighbors, k = 4, SVM	82%
Summing all $F_{n,m,k}$	82%
Summing all $F_{n,m,k}$ only for one (best) feature type	84%
Summing all $F_{n,m,k}$ with the same feature type, boosting, 10 iterations	86%
Summing all $F_{n,m,k}$ with the same feature type, boosting, 50 iterations	87.5%
Summing all $F_{n,m,k}$ with the same feature type, SVM	89%
SVM over $N \cdot K$ elements $F_{n,m_i,k}$ with the same feature type (best one)	90.8%
Combination of M SVMs over $N \cdot K$ elements $F_{n,m_i,k}$ with the same feature type by one SVM	91.6%
Combination of K SVMs over $N \cdot M$ elements F_{n,m,k_i} with the same component by one SVM	**92.96%**

Table 2. Classification accuracy results by different proposed algorithms.

5. CONCLUSION

We have presented a new approach to face classification. We use a reference people dataset and train simile classifiers, which estimate the similarity between a test and the reference person.

We have tested several algorithms that use such similarity distances to the reference people and their labels for the final classification. Best result was demonstrated by a two-layer classifier, which uses only the similarities, with no information about reference people labels.

We demonstrate the effectiveness of the proposed algorithm on the task of gender classification, where we achieve 92.96% accuracy on LFW dataset.

6. AKNOLEDGMENTS

This work was supported by Russian Foundation for Basic Research, grant N. 11-01-00957-a.

7. REFERENCES

[1] S. Baluja and H. A. Rowley. Boosting sex identification performance. *Int. J. of Comput. Vision*, 71(1): 111–119, 2007.

[2] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung. Ordinal Hyperplanes Ranker with Cost Sensitivities for Age Estimation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2011.

[3] D. Cristinacce and T. Cootes. Boosted Regression Active Shpae Models. *British Machine Vision Conference,* 2007.

[4] P. Du and D. Xiaoqing. The Application of Decision Tree in Gender Classification. *Congress on Image and Signal Processing*, 2008

[5] W. Gao, B. Cao, S. G. Shan, et al. The CAS-PEAL Large-Scale Chinese Face Database and Baseline Evaluations, technical report of JDL, available on http://www.jdl.ac.cn/~peal/peal tr.pdf , 2004

[6] G. Guo, Y. Fu, T. S. Huang, and C. Dyer. A probabilistic fusion approach to human age prediction. In *IEEE CVPRSLAM workshop*, 2008.

[7] G. Guo, G. Mu, Y. Fu, C. Dyer, and T. S. Huang. A Study on Automatic Age Estimation using a Large Database. *ICCV*, 2009

[8] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A database for studying face recognition in unconstrained environments. *UMass Amherst Technical Report 07-49*, October 2007.

[9] N. Kumar, P. N. Belhumeur, and S. K. Nayar. FaceTracer: A search engine for large collections of images with faces. *ECCV*, 2008.

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

[11] H. C. Lian and B. L. Lu. Multi-view gender classification using local binary patterns and support vector machines. *Proceedings of the 5th International Conference on Artificial Neural Networks (ISNN '06)*, 2: 202–209, Chengdu, China, 2006.

[12] E. Makinen and R. Raisamo. Evaluation of gender classification methods with automatically detected and aligned faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30: 541–547, 2008

[13] B. Moghaddam and M. Yang. Learning gender with support faces. *IEEE Trans. Pattern Anal. Machine Intell.* 24(5): 707–711, 2002

[14] A. Montillo and H. Ling. Age Regression from Faces Using Random Forests. *IEEE International Conference on Image Processing*, 2009

[15] G. W. Mu, G. D. Guo, Y. Fu, and T. S. Huang. Human age estimation using bio-inspired features. *Proc. of Computer Vision and Pattern Recognition*, 2009

[16] P. J. Phillips, H. Moon, S. A. Rizvi, and P. Rauss. The FERET Evaluation Methodology for Face Recognition Algorithms. *IEEE PAMI*, 22(10): 1090-1104, 2000.

[17] C. Shan. Learning local binary patterns for gender classification on real-world face images. *Pattern Recognition Letters*, 33(4): 431-437, 2012.

[18] P. Viola and M. J. Jones. Robust Real-Time Face Detection. *Int'l J. Computer Vision*, 57(2): 137-154, 2004

[19] Q. Yin, X. Tang, and J. Sun. An associate-predict model for face recognition. In *CVPR*, 2011