

Tom-vs-Pete Classifiers and Identity-Preserving Alignment for Face Verification

Thomas Berg
tberg@cs.columbia.edu

Columbia University
New York, NY

Peter N. Belhumeur
belhumeur@cs.columbia.edu

Abstract

We propose a method of face verification that takes advantage of a reference set of faces, disjoint by identity from the test faces, labeled with identity and face part locations. The reference set is used in two ways. First, we use it to perform an “identity-preserving” alignment, warping the faces in a way that reduces differences due to pose and expression but preserves differences that indicate identity. Second, using the aligned faces, we learn a large set of identity classifiers, each trained on images of just two people. We call these “Tom-vs-Pete” classifiers to stress their binary nature. We assemble a collection of these classifiers able to discriminate among a wide variety of subjects and use their outputs as features in a same-or-different classifier on face pairs. We evaluate our method on the Labeled Faces in the Wild benchmark, achieving an accuracy of 93.10%, significantly improving on the published state of the art.

1 Introduction

In face verification, we are given two face images and must determine whether they are the same person or different people. The images may vary in pose, expression, lighting, occlusions, image quality, etc. The difficulty lies in teasing out features in the image that indicate identity and ignoring features that vary with differences in environmental conditions.

One might argue it is easy to find features that correspond with identity. For example, to distinguish the faces of actors Lucille Ball and Orlando Bloom from each other, we can consider hair color. “Red hair” is a simple feature that consistently indicates Lucille Ball. To distinguish images of Stephen Fry and Brad Pitt, the best feature might be “crooked nose.” With a sufficiently large and diverse set of these features, we should have a discriminating feature for almost any pair of subjects. Kumar *et al.* [10] explored this approach, calling these features “describable visual attributes,” implementing them as classifiers, and using them for verification. A limitation of the approach is that the set of reliable features can only be as big as the relevant vocabulary one can come up with and get training data labelers to consistently label.

In this paper, we *automatically* find features that can distinguish between two people, without requiring the features to be describable in words and without requiring workers to label images with the feature. A simple way to find such a feature is to train a linear classifier to distinguish between two people. If the training data includes many images of each person under varied conditions, the projection found by the classifier will be insensitive

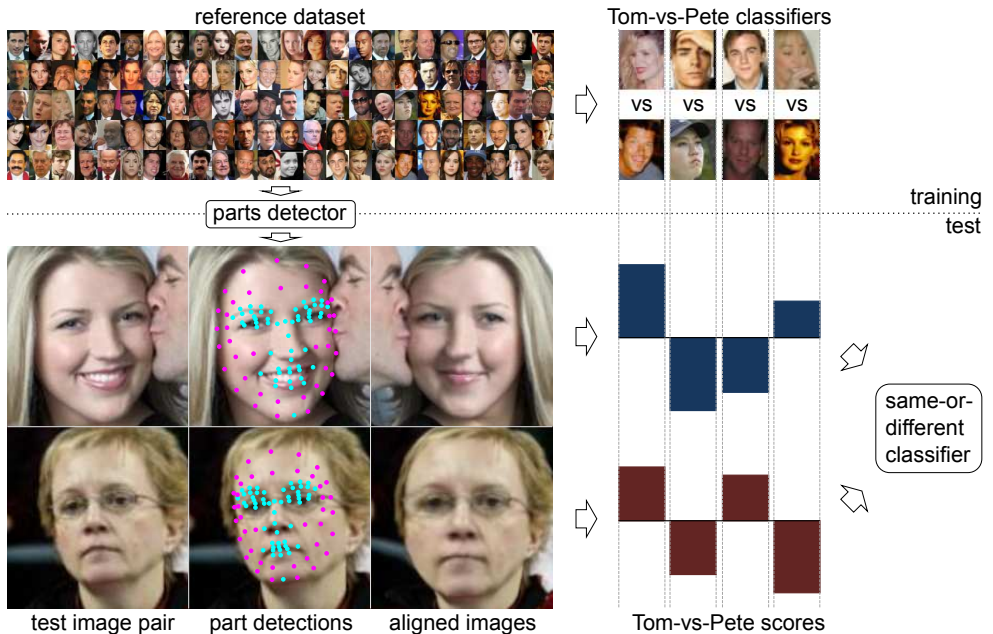


Figure 1: An overview of the verification system. A reference set of images labeled with parts and identities is used to train a parts detector and a large number of “Tom-vs-Pete” classifiers. Then given a pair of test image, parts are detected and the detected parts are used to perform an “identity-preserving” alignment. The Tom-vs-Pete classifiers are run on the aligned images, with the results passed to a same-or-different classifier to produce a decision.

to the conditions and consistently correspond to identity. We call classifiers trained in this way “Tom-vs-Pete” classifiers to emphasize that each is trained on just two individuals. We will show that they can be applied to *any* individual and used for face verification.

To demonstrate the Tom-vs-Pete classifiers, we collect a “reference set” of face images, labeled by identity and with many images of each subject. We build a library of Tom-vs-Pete classifiers by considering all possible pairs of subjects in the reference set. We then assemble a subset of these classifiers such that, for any pair of subjects, it is highly likely that we have at least a few classifiers able to distinguish them from each other. When presented with a pair of faces (of subjects *not* in the reference set) for verification, we apply these classifiers to each face and use the classifier outputs as features for a second-stage classifier that makes the “same-or-different” verification decision. Figure 1 shows an overview of the method.

To allow us to build a large and diverse collection of Tom-vs-Pete classifiers, and to make it more likely that each classifier will generalize beyond the two subjects it is trained on, each classifier looks at just a small portion of the face. These small regions must correspond to each other across images and identities for the classifiers to be effective, so alignment of the faces becomes particularly important. With this in mind, we adopt an alignment procedure, based on the detection of a set of face parts, that enforces a fairly strict correspondence across images. Our alignment procedure also includes a novel use of the reference dataset to distinguish geometry differences due to pose and expression, which should be normalized out by the alignment, from those that pertain to identity (such as thicker lips or a wider nose) and should be preserved. We call this an “identity-preserving alignment.”

We evaluate our method on the Labeled Faces in the Wild (LFW) [16], a face verification

benchmark using uncontrolled images collected from Yahoo News. We achieve an accuracy of 93.10%, reducing the error rate of the previous state of the art by 26.86%.

2 Related Work

The field of face recognition and verification has too long a history to survey here, so we focus on previous approaches that are similar to ours. As shown in Figure 1, our method is essentially a hierarchical classifier, where the outputs of the Tom-vs-Pete *face* classifiers are combined to form features for input to a second stage *pair* classifier. Wolf *et al.* [52, 53] also use a two-level classifier, for each test pair training a small number of “one-shot” and “two-shot” classifiers using one or both test images as positive samples and an additional fixed set of reference images as negative samples. Yin *et al.* [34] take a similar approach but augment the positive training set with a set of images of subjects similar to the test image, chosen from a reference set independent from the test set. In both cases specialized classifiers must be trained for each test image. In contrast, we train a single set of classifiers as an offline preprocessing step and have no need to train during testing.

Kumar *et al.* [18, 19] also adopt this two-level classifier structure, using a set of attribute (gender, race, hair color, etc.) classifiers in the first stage. This requires a great deal of manual effort in choosing and labeling attributes, while our method automatically generates a very large number of classifiers from data labeled only with identity. The “simile classifiers” method from the same work uses a set of one-vs-all identity classifiers trained on an independent reference dataset for the first stage. This is most similar to our work, but the one-vs-one nature of the Tom-vs-Pete classifiers allow us to produce a much larger set of classifiers, quadratic instead of linear in the number of subjects. The simplicity of the one-vs-one concept to be learned also allows us to use fast linear SVMs, while the attribute and simile classifiers use an RBF kernel.

Other well-performing verification methods that do not follow this precise two-level structure still follow the pattern of first learning how to extract features, then learning the same-vs-different classifier. For example, Pinto and Cox [25] use a validation set of face pairs to experiment with a large number of features and choose those most effective for verification, then feed these to an SVM for verification of the test data. Nguyen and Bai [23] split their training data and use one part to learn metrics in the image feature space and the other to learn a verification classifier whose input is the learned distances between the image pairs.

Our Tom-vs-Pete classifiers themselves are based on features extracted at detected parts of the face such as the corners of the mouth and tip of the nose. Similar parts-based features are demonstrated for face recognition by [11, 8, 34]. Generation of the Tom-vs-Pete classifiers can also be viewed as a feature selection process where we seek features suitable for face verification. There is an extremely large body of work on feature selection from the machine learning community (for surveys, see [13, 22]), but we do not adopt a formal feature selection methodology here.

It is well known that alignment is critical for good performance in face recognition with uncontrolled images [11, 81, 33]. One method often applied is Huang *et al.*’s [15] “funneling,” which extends the congealing method of Learned-Miller [20] to handle noisy, real-world images. These methods find transformations that minimize differences in images that are initially only roughly aligned. Funneled images provided by the LFW maintainers have been widely used [12, 17, 26, 32, 35], but the Tom-vs-Pete classifiers require a correspondence more precise than funneling provides.

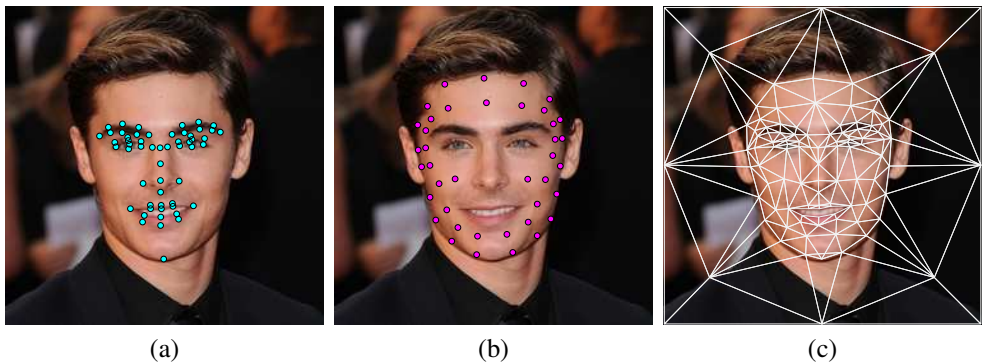


Figure 2: Labeled face parts. (a) There are fifty-five “inner” points at well-defined landmarks and (b) forty “outer” points that are less well-defined but give the general shape of the face. (c) The triangulation of the parts used to perform a piecewise affine warp. (Additional fixed points are placed well outside the face to ensure a rectangular warped image.)

Another common technique is to apply a transformation to the images based on the locations of detected parts such as the corners of the eyes and mouth. This is the approach of Kumar *et al.* [18] and Wolf *et al.* [63], who detect a small number of parts with a commercial system and then apply an affine transformation to bring the parts close to fixed locations. The aligned LFW images from [63] have been released as the LFW-a dataset, which has subsequently been used by many authors [14, 23, 25, 28, 29, 65], but we desire an alignment that gives tighter correspondence between the faces.

Alignments based on active appearance models [2, 6, 4] and 3D models [3, 9] are appealing, but have not yet been demonstrated on images captured in the wild and displaying simultaneous variation in pose, lighting, expression, occlusion, and image quality. AAM-based methods perform alignment by creating a triangulation on a set of detected parts and performing a piecewise affine transformation that takes the triangles to their positions in the desired pose. We follow this warping procedure, but detect the parts in a different way, and introduce an adjustment to the part locations to avoid losing identity information in the transformation.

3 Reference Dataset

The identity-preserving alignment and the Tom-vs-Pete classifiers both rely on a dataset of reference face images labeled with identities and face part locations. We describe this dataset here for clarity of explanation in the following sections.

The dataset consists of 20,639 images of 120 people. So that we can train on our dataset and evaluate our methods on LFW, none of the people in our dataset are in LFW. The images were collected by searching for the names of public figures on web sites such as Flickr and Google Images. We filtered the resulting images by running a commercial face detector [24] to discard images without faces and using Amazon Mechanical Turk to discard images that were not of the target person, following the procedure outlined in [18]. In addition we removed the majority of “near-duplicate” images – images derived from the same original, but with different crops, compression, or other processing – following the method of [27], which is based on a simple image similarity measure. Images for 60 of the 120 people are from the “development” part of PubFig [18] dataset (which was collected as described above), while the remainder are new.

For all 20,639 images, we have obtained the human-labeled locations of 95 face parts, again using Mechanical Turk. Each point was marked by five labelers, with the mean of

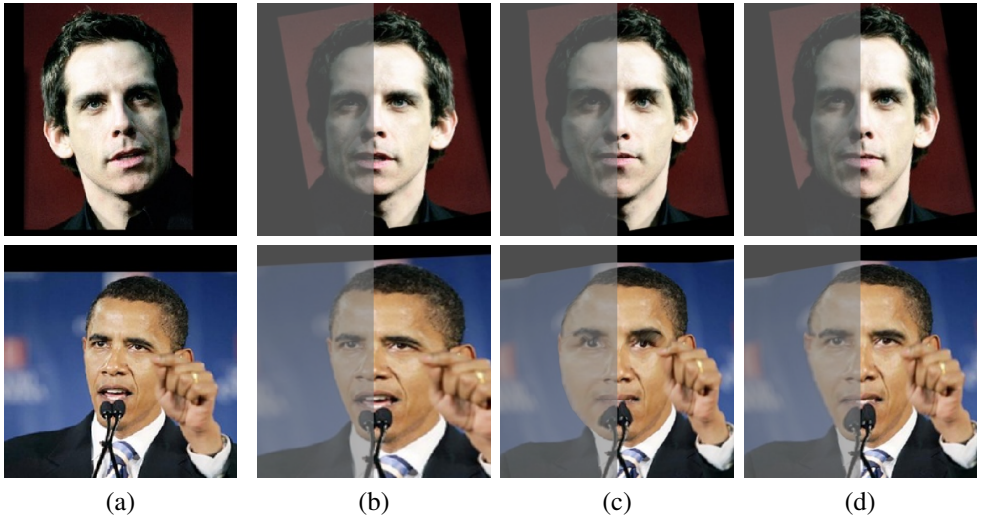


Figure 3: Warping images to frontal. (a) Original images. (b) Aligning by an affine transformation based on the locations of the eyes, tip of the nose, and corners of the mouth does not achieve tight correspondence between the images. (c) Warping to put all 95 parts at their canonical positions gives tight correspondence, but de-identifies the face by altering its shape. (d) Warping based on genericized part locations gives tight correspondence without obscuring identity. In all methods, we ensure that the side of the face presented to the camera is on the right side of the image by performing a left-right reflection when necessary. This restricts the worst distortions to the left side of the image (shown with a gray wash here), which the classifiers can learn to weight less important than the right.

the three-label subset having the smallest variance taken as the final location. We divide the parts into two categories: a set of 55 “inner” parts that occur at edges and corners of relatively well-defined points on the face, such as the corners of the eyes and the tip of the nose, and a set of 40 “outer” parts that show the overall shape of the face but are less well-defined and so harder to precisely localize. The part locations are shown on a sample image in Figure 2.

4 Identity-preserving Alignment

We have constructed our alignment procedure with three criteria in mind. First, although our classification problem concerns pairs of faces, the Tom-vs-Pete classifiers used in the first stage operate on single faces. To accommodate this, all images must be aligned to a standard pose and expression. A “pairwise” alignment in which the images in each pair are brought into correspondence only with each other, which can produce less distortion than a single “all images” alignment, is not sufficient. We design our alignment to bring all images to a frontal pose with neutral expression.

Second, for the Tom-vs-Pete classifiers to be effective, the regions on which they are trained must have very good correspondence. This is because each classifier uses only a small part of the face, so the regions will have little or no overlap if the correspondence is not good, and because the linear nature of the classifiers makes it difficult to learn the more complex concepts that would be required to deal with poor alignment. Global similarity or affine transformations, for example, are not ideal because they can bring only two or three points into perfect alignment, respectively.

Third, we must be careful not to *over-align*. The perfect alignment procedure for face

recognition removes differences due to pose and expression but not those due to identity. Our alignment should turn faces to a frontal pose and close open mouths, but it should not warp a prominent jaw to a receding chin.

The alignment procedure we have designed to satisfy these criteria requires a set of part locations on each face. We use the ninety-five parts defined in the reference dataset. To find them automatically in a test image, we first use the detector of [4], which combines the results of an independent detector for each part with global models of the parts’ relative positions, to detect the fifty-five inner parts. Then we find the image in the reference dataset whose inner parts, under similarity transformation, are closest in an L_2 sense to the detected inner parts, and “inherit” the outer part positions from that image. The parts detector is trained on a subset of the images in the reference set.

Each part also has a canonical location, where it occurs in an average, frontal face with neutral expression. To align the image, we adopt the piecewise affine warp often used with parts detected using active appearance models [6, 7]. We take a Delaunay triangulation of the canonical part positions and the corresponding triangulation on the part positions in the image, then map each triangle in the image to the corresponding canonical triangle by the affine transformation determined by the three vertices. The three correspondences at the vertices of each triangle produce a unique, exact solution for the affine transformation, so all the parts are mapped perfectly to their canonical locations. Provided we have a sufficiently dense set of parts, this ensures the tight correspondence we require.

This system of alignment produces very tight correspondences and effectively compensates for pose and expression. However the warping is so strict, moving the ninety-five parts to *exactly* the same locations in every image, that features indicating identity are lost; the third criterion for our alignment is not satisfied. This can be seen in Figure 3 (c), whose images are somewhat anonymized compared with (a), (b), and (d). To understand why this happens, note that since there are parts at both sides of the base of the nose, aligned images of all subjects will have noses of the same width. To avoid this over-alignment, we will perform the alignment based not on the part locations in the image itself, but on “generic” parts – where the parts would be for an average person with the pose and expression in the image. For a wide-nosed person, these points will be not on the edge of the nose but slightly inside, and the above-average width of the nose will be preserved by the piecewise affine warp.

To find the generic parts, we modify the procedure for locating the parts in a test image as follows. We run the detector of [4] to get the fifty-five inner part locations as before. Then we find the image with the most similar configuration of parts for each of the 120 subjects in the reference dataset. We include the additional forty outer parts of these images to get a full set of ninety-five parts for each of 120 reference faces. These represent the part locations of 120 different individuals with nearly the same pose and expression as the test image. We take the mean of the 120 sets of part locations to get the generic part locations for the test image. We use these generic part locations in place of the originally detected locations to produce an identity-preserving aligned image with a piecewise affine warp as described above.

For large yaw angles, we cannot produce a warp to frontal that looks good on the side of the face originally turned away from the camera. To reduce the difficulty this presents to the classifiers, we make a very simple guess at the yaw direction of the face (we use the detected parts to find the shorter eyebrow and assume the subject is facing that direction), then reflect the image if necessary so that all faces are facing the left side of the image. In this way, our classifiers can learn to assign more importance to the reliable, right side of the image.

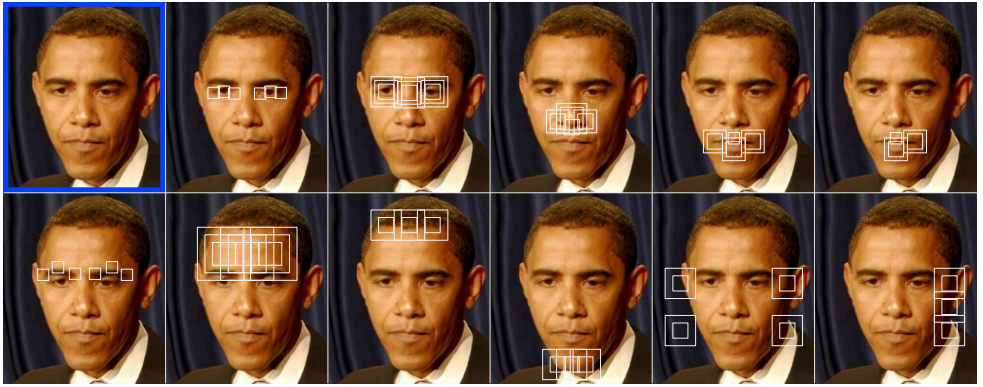


Figure 4: The top left image is produced by the alignment procedure. Each of the remaining images shows the region from which one low-level feature is extracted. SIFT descriptors are extracted from each square and concatenated. Concentric squares indicate SIFT descriptors at the same point but different scales.

5 Tom-vs-Pete Classifiers and Verification

Each Tom-vs-Pete classifier is a binary classifier trained using a low-level feature on images of two people from the reference dataset. If there are N subjects in the reference set and k low-level features, we can train $k \cdot \binom{N}{2}$ Tom-vs-Pete classifiers.

In our experiments, each low-level feature is a concatenation of SIFT descriptors [21] extracted at several points and scales in one region of the face. By limiting each classifier to a small region of the face, we hope to learn a concept that will generalize to individuals other than the two people used for training. The regions, shown in Figure 4, cover the distinctive features inside the face, such as the nose and eyes, as well as the boundary of the face. The classifiers are linear support vector machines trained using the LIBLINEAR package [9].

For each face in a verification pair, we evaluate a set of Tom-vs-Pete classifiers and construct a vector of the signed distances from the decision boundaries. This vector serves as a descriptor of the face. Following the example of [18], we then concatenate the absolute difference and element-wise product of the two face descriptors and pass the result to an RBF SVM to make the same-or-different decision.

We use 5,000 Tom-vs-Pete classifiers to build the face descriptors. Experiments suggest that additional classifiers beyond this number are of little benefit. With 120 subjects in the reference dataset and 11 low-level features, we can train tens of thousands of classifiers, so we have to choose a subset. There are many reasonable ways to do so, but we design our procedure motivated by the desire for a subset of classifiers that (a) can handle a wide variety of subject pairs and (b) complement each other. To achieve the first, we will choose evenly from classifiers that excel at each reference subject pair. To achieve the second, we will use Adaboost [10]. We begin by constructing a ranked list of classifiers for each subject pair (S_i, S_j) , as follows:

1. Let H_{ij} be the set $\{h_1, \dots, h_n\}$ of Tom-vs-Pete classifiers that are *not* trained on S_i or S_j .
2. Consider each h_k in H_{ij} as an S_i -vs- S_j classifier. Do this by fixing the subject with the greater median output of h_k as the positive class, then finding the threshold that produces equal false positive and false negative rates.
3. Treating H_{ij} as a set of weak S_i -vs- S_j classifiers, run the Adaboost algorithm. This

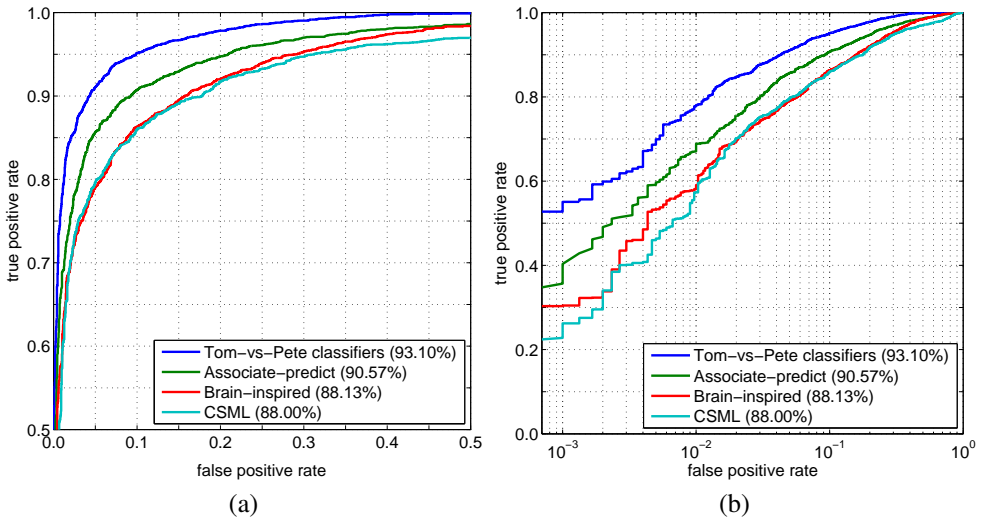


Figure 5: (a) A comparison with the best published results on the LFW image-restricted benchmark, including the Associate-predict method [64], Brain-inspired features [25], and Cosine Similarity Metric Learning (CSML) [23], (b) The log scale highlights the performance of our method at the low-false-positive rates desired by many security applications.

assigns a weight to each h_k .

4. Sort H_{ij} by descending Adaboost weights to get a list of classifiers L_{ij} . An initial subsequence of L_{ij} will be a set of classifiers, not trained on S_i or S_j , that combine effectively to distinguish S_i from S_j .

We construct an overall ordered list of classifiers, L , by taking the first classifier in each of the L_{ij} , then the second in each, then the third, etc. Within each group we randomly order the classifiers, and each classifier is included in L only the first time it occurs. To choose a subset of classifiers of any size, we take an initial subsequence of L .

6 Results

We evaluate our system on Labeled Faces in the Wild (LFW) [16], a face verification benchmark using images collected from Yahoo News. The LFW benchmark consists of 6,000 pairs of faces, half of them “same” pairs and half “different,” divided into ten folds for cross-validation. In our method the parts detection, alignment, and Tom-vs-Pete classifiers are based on the reference dataset, so the LFW training folds are used only to train the final same-vs-different classifier. Note that none of the subjects in our reference set appear in LFW, so neither the Tom-vs-Pete classifiers nor the parts detector have seen these individuals in training. We follow the “image-restricted” protocol, in which the training face pairs are marked only with a same or different label and not with the identities of each face (which would allow the generation of additional training pairs).

We obtain a mean accuracy of $93.10\% \pm 1.35\%$. LFW is widely reported on, with accuracies of twenty-five published methods listed on the maintainers’ web site [60] at time of writing. Figure 5 compares our performance with the top three previously published results. We achieve a 26.86% reduction in the error rate of the previous best results reported by Yin *et al.* [64]. Figure 5 (b) demonstrates our performance at the low false positive rates required by many security applications. At 10^{-3} false positive rate we achieve a true positive rate of 55.07%, where the previous best is 40.33% [64].

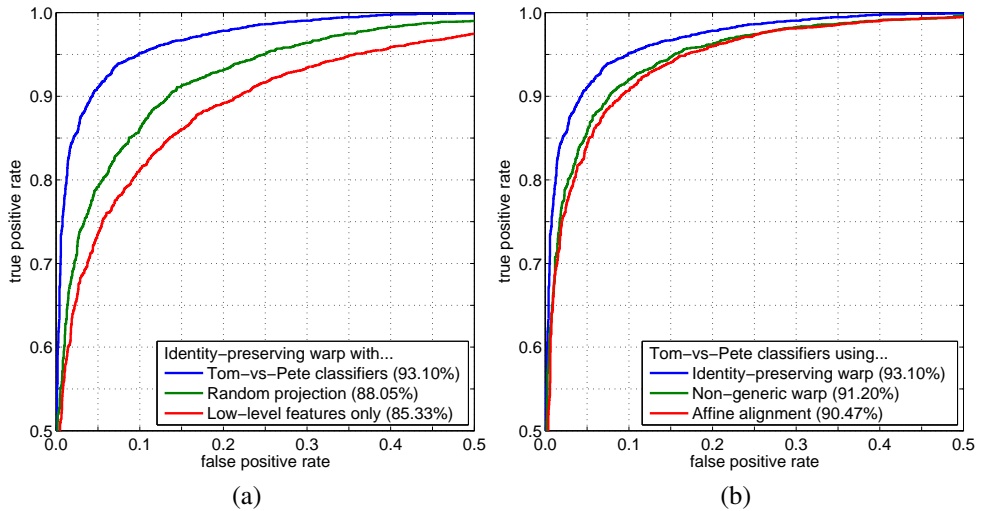


Figure 6: LFW benchmark results. (a) The contribution of the Tom-vs-Pete classifiers, compared with random projection or direct use of the low-level features. (b) The contribution of the alignment method, compared with a piecewise affine warp using non-generic part locations or a global affine transformation.

Kumar *et al.* [18] have made the output of their attribute classifiers on the LFW images available on the LFW web site [60]. These classifiers are similar to our Tom-vs-Pete classifiers, but are trained on images hand labeled with attributes such as gender and age. Appending the attributes classifier outputs to our vector of Tom-vs-Pete outputs boosts our accuracy to $93.30\% \pm 1.28\%$.

Our method is efficient. Training and selection of the Tom-vs-Pete classifiers is done offline. The eleven low-level features are constructed from SIFT descriptors at a total of just 34 points on the face, at one to three scales each. These features are shared by all of the Tom-vs-Pete classifiers. Evaluation of each Tom-vs-Pete classifier requires evaluation of a single dot product. Finally, the RBF SVM verification classifier must be evaluated on a single feature vector.

To demonstrate the relative importance of each part of our system, we run the benchmark with several stripped-down variants of the algorithm:

- **Random projection:** Replace each Tom-vs-Pete classifier with a random projection of the low-level feature it uses. Shown in Figure 6 (a).
- **Low-level features only:** Discarding the Tom-vs-Pete classifiers, concatenate the low-level features to produce a descriptor of each face, and use the absolute difference of these descriptors as the feature vector for the same-or-different classifier. Shown in Figure 6 (a).
- **Non-generic warp:** Train and use Tom-vs-Pete classifiers, but use the detected part locations directly in a piecewise affine warp, rather than the genericized locations that produce the identity-preserving warp. Shown in Figure 6 (b).
- **Affine alignment:** Train and use Tom-vs-Pete classifiers, but align all images with global affine transformations based on the detected locations of the eyes and mouth instead of our identity-preserving warp. Shown in Figure 6 (b).

Figure 6 includes ROC curves from these experiments and from the full system, showing that each part of the method contributes to the high accuracy.

7 Conclusions

We have presented a method of face verification that takes advantage of a reference set of labeled images in two novel ways. First, our identity-preserving alignment warps face images in a way that corrects for pose and expression but preserves geometry distinctive to the subject, based on face parts in similar configurations in the reference set. Second, the verification decision itself is based on the output from our fast, linear Tom-vs-Pete classifiers, which are trained to distinguish anyone from anyone by considering every possible pair of subjects in the reference set.

The method is efficient and accurate. We evaluate on the standard Labeled Faces in the Wild image-restricted benchmark, on which it achieves state-of-the-art performance.

References

- [1] S. Arca, P. Campadelli, and R. Lanzarotti. A face recognition system based on automatically determined facial fiducial points. *Pattern Recognition*, March 2006.
- [2] A. Asthana, M. J. Jones, T. K. Marks, K. H. Tieu, and R. Goecke. Pose normalization via learned 2d warping for fully automatic face recognition. In *BMVC*, 2011.
- [3] A. Asthana, T. K. Marks, M. J. Jones, K. H. Tieu, and M. V. Rohith. Fully automatic pose-invariant face recognition via 3d pose normalization. In *ICCV*, 2011.
- [4] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *CVPR*, 2011.
- [5] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *PAMI*, September 2003.
- [6] T. F. Cootes, K. Walker, and C. J. Taylor. View-based active appearance models. In *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition*, 2000.
- [7] G. Edwards, C. J. Taylor, and T. F. Cootes. Interpreting face images using active appearance models. In *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition*, 1998.
- [8] M. Everingham, J. Sivic, and A. Zisserman. “Hello! My name is... Buffy” – automatic naming of characters in TV video. In *BMVC*, 2006.
- [9] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9, 2008.
- [10] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, August 1997.
- [11] L. Gu and T. Kanade. A generative shape regularization model for robust face alignment. In *ECCV*, 2008.
- [12] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *ICCV*, 2009.
- [13] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, March 2003.
- [14] C. Huang, S. Zhu, and K. Yu. Large scale strongly supervised ensemble metric learning, with applications to face verification and retrieval. Technical report, NEC, 2011.

- [15] G. B. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007.
- [16] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, University of Massachusetts, Amherst, 2007.
- [17] G. B. Huang, M. J. Jones, and E. G. Learned-Miller. LFW Results Using a Combined Nowak Plus MERL Recognizer. In *Proceedings of the Faces in Real Life Images workshop at ECCV*, 2008.
- [18] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009.
- [19] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Describable visual attributes for face verification and image search. *PAMI*, October 2011.
- [20] E. G. Learned-Miller. Data driven image models through continuous joint alignment. *PAMI*, February 2006.
- [21] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, November 2004.
- [22] H. Motoda and H. Liu, editors. *Computational Methods of Feature Selection*. Chapman and Hall / CRC, 2007.
- [23] H. V. Nguyen and L. Bai. Cosine similarity metric learning for face verification. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2011.
- [24] Omron. OKAO vision. http://www.omron.com/r_d/coretech/vision/okao.html.
- [25] N. Pinto and D. D. Cox. Beyond Simple Features: A Large-Scale Feature Search Approach to Unconstrained Face Recognition. In *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition*, 2011.
- [26] N. Pinto, J. J. DiCarlo, and D. D. Cox. How far can you get with a modern face recognition test set using only simple features? In *CVPR*, 2009.
- [27] N. Pinto, Z. Stone, T. Zickler, and D. D. Cox. Scaling-up biologically-inspired computer vision: A case-study on facebook. In *Proceedings of the Workshop on Biologically Consistent Vision at the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [28] H. J. Seo and P. Milanfar. Face verification using the lark representation. *IEEE Transactions on Information Forensics and Security*, December 2011.
- [29] Y. Taigman, L. Wolf, and T. Hassner. Multiple one-shots for utilizing class label information. In *BMVC*, 2009.
- [30] University of Massachusetts. LFW web site. <http://vis-www.cs.umass.edu/lfw/>.
- [31] P. Wang, L. C. Tran, and Q. Ji. Improving face recognition by online image alignment. In *Proceedings of the International Conference on Pattern Recognition*, 2006.
- [32] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *Proceedings of the Faces in Real Life Images workshop at ECCV*, 2008.
- [33] L. Wolf, T. Hassner, and Y. Taigman. Similarity scores based on background samples. In *Proceedings of the Asian Conference on Computer Vision*, 2009.
- [34] Q. Yin, X. Tang, and J. Sun. An associate-predict model for face recognition. In *CVPR*, 2011.
- [35] Y. Ying and P. Li. Distance metric learning with eigenvalue optimization. *Journal of Machine Learning Research*, March 2012.