Describable Visual Attributes for Face Verification and Image Search

Neeraj Kumar, *Student Member, IEEE,* Alexander C. Berg, *Member, IEEE,*
Peter N. Belhumeur, and Shree K. Nayar, *Member, IEEE*

Abstract—We introduce the use of *describable visual attributes* for face verification and image search. Describable visual attributes are labels that can be given to an image to describe its visual appearance. This paper focuses on images of faces and the attributes used to describe them, although the concepts also apply to other domains. Examples of face attributes include gender, age, jaw shape, nose size, *etc*. The advantages of an attribute-based representation for vision tasks are manifold: they can be composed to create descriptions at various levels of specificity; they are generalizable, as they can be learned once and then applied to recognize new objects or categories without any further training; and they are efficient, possibily requiring exponentially fewer attributes (and training data) than explicitly naming each category. We show how one can create and label large datasets of real-world images to train classifiers which measure the presence, absence, or degree to which an attribute is expressed in images. These classifiers can then automatically label new images. We demonstrate the current effectiveness (and explore the future potential) of using attributes for face verification and image search via human and computational experiments. Finally, we introduce two new face datasets – named FaceTracer and PubFig – with labeled attributes and identities, respectively.

Index Terms—Face recognition, attribute classification, classifier training, content-based image retrieval, image search.

1 Introduction

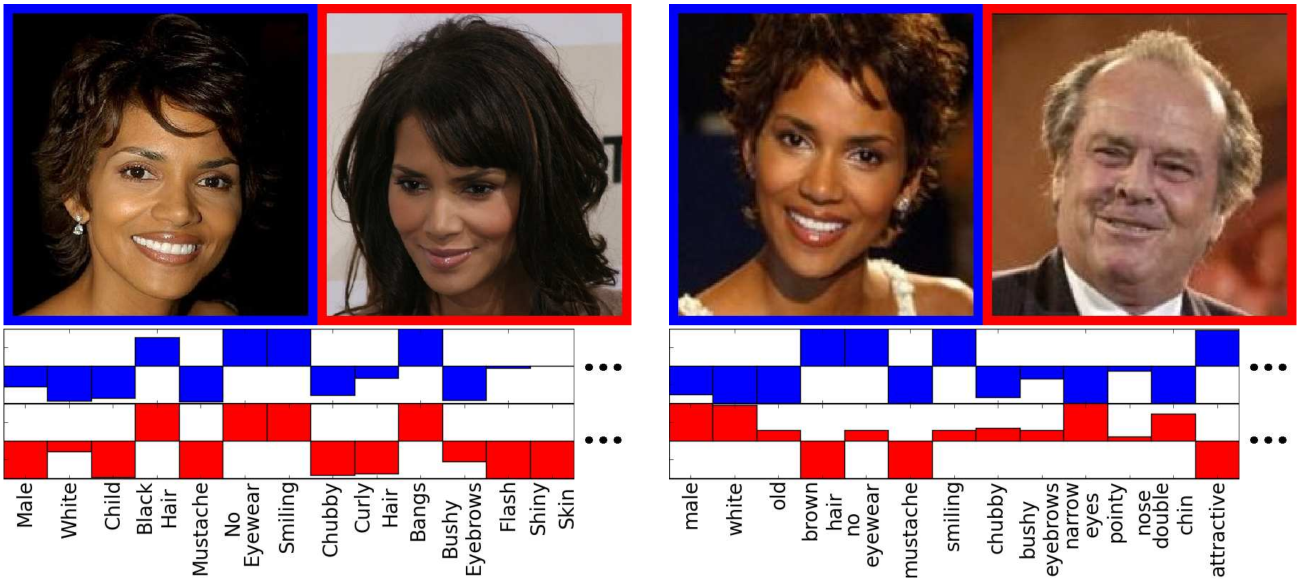
One of history's most successful books was a five-volume pharmacopoeia titled *De Materia Medica*, written in the first century by the Greek botanist and physician Pedanius Dioscorides. It is perhaps the earliest known field guide, giving pictures and written descriptions of nearly 600 plant species, showing how each could be found and identified. This work would be the first in a line of botanical texts including the ninth century medieval agricultural and toxicological texts of Ibn Washiyah and the early eighteenth century *Systema Naturae* of Carl Linneaus, which laid out the rules of modern taxonomy. All of these works have in common an effort to teach the reader how to identify a plant or animal by describable aspects of its visual appearance.

While the use of describable visual attributes for identification has been around since antiquity, it has not been the focus of work by researchers in computer vision and related disciplines. Most existing methods for recognition (*e.g.*, [11], [34], [36], [49]) work by extracting low-level features in images, such as pixel values, gradient directions, histograms of oriented gradients [11], SIFT [32], etc., which are then used to *directly* train classifiers for identification or detection.

In contrast, we use low-level image features to first learn *intermediate representations* [27], [28], in which images are labeled with an extensive list of descriptive visual attributes. Although these attributes could clearly be useful in a variety of domains such as object recognition, species identification, architectural description, action recognition, *etc*., we focus solely on faces in this paper. These face attributes can range from simple demographic information such as gender, age, or ethnicity; to physical characteristics of a face such as nose size, mouth shape, or eyebrow thickness; and even to environmental aspects such lighting conditions, facial expression, or image quality. In our approach, an extensive vocabulary of visual attributes is used to label a large dataset of images, which is then used to train classifiers that *automatically* recognize the presence, absence, or degree to which these attributes are exhibited in new images. The classifier outputs can then be used to identify faces and search through large image collections, and they also seem promising for use in many other tasks such as image exploration or automatic description-generation.

Why might one need these attributes? What do they afford? Why not train classifiers directly for the task at hand? Visual attributes – much like words – are composable, offering tremendous flexibility and efficiency. Attributes can be combined to produce descriptions at multiple levels, including object categories, objects, or even instances of objects. For example, one can describe "white male" at the category level (a set of people), or "white male brown-hair green-eyes scar-on-forehead" at the object level (a specific person), or add "..., smiling lit-from-above seen-from-left" to the previous for an instance of the object (a particular image of a person).

- *The authors are with the Computer Science Department, Columbia University, 450 Mudd Hall, 500 West 120 Street, New York, NY 10027.*
E-mail: {neeraj, aberg, belhumeur, nayar}@cs.columbia.edu



(a) Attribute labels for two images of the same person

(b) Attribute labels for images of two different people

Fig. 1. An attribute classifier can be trained to recognize the presence or absence of a *describable aspect* of visual appearance. The responses for several such attribute classifiers are shown for (a) two images of Halle Berry and (b) two images of different individuals. In (a), notice how most attribute values are in strong agreement, despite the changes in pose, illumination, expression, and image quality. Conversely, in (b), the values differ completely despite the similarity in these same environmental aspects. We train a verification classifier on these outputs to perform face verification, achieving 85.29% accuracy on the Labeled Faces in the Wild (LFW) benchmark [25], comparable to the state-of-the-art.

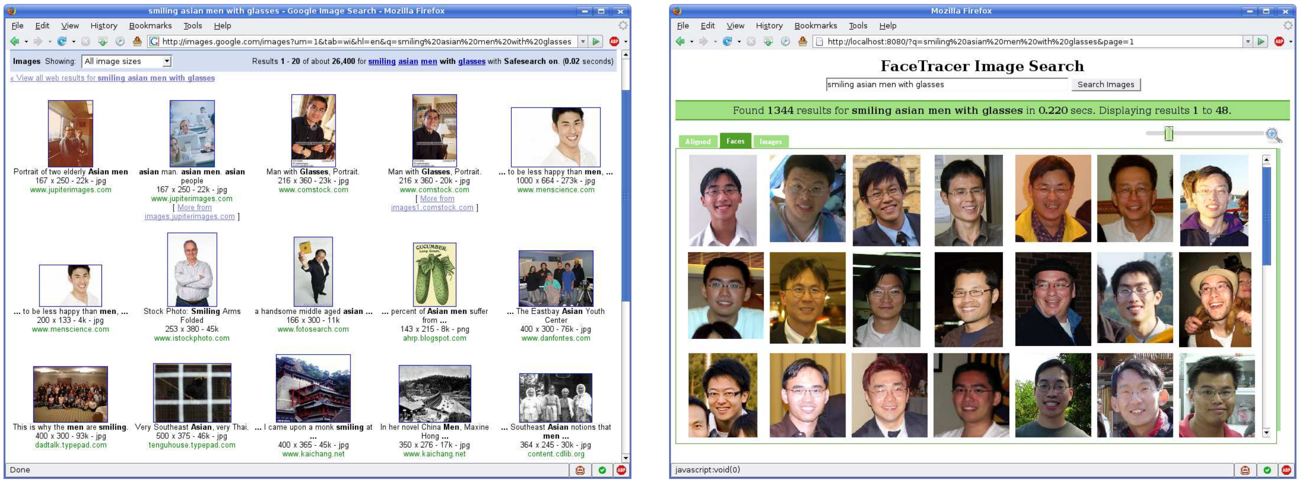
Moreover, attributes are generalizable; one can learn a set of attributes from large image collections and then apply them in almost arbitrary combinations to novel images, objects, or categories. Better still, attributes are efficient: consider that k binary attributes may suffice to identify 2^k categories, clearly more efficient than naming each category individually. (Of course, in practice, the potential benefits are limited by the problem domain, the type of categories being considered, and the accuracy of learned classifiers.) In contrast to existing labeling efforts such as ImageNet [13] and LabelMe [45] that label large collections of images by category or object name, the use of attributes may provide a significantly more compact way of describing objects. This would allow for the use of much smaller labeled datasets to achieve comparable performance on recognition tasks.

Perhaps most importantly, these attributes can be chosen to align with the domain-appropriate vocabulary that people have developed over time for describing faces. This means that they can be used to describe faces from the coarsest level (such as gender and age) to more subtle aspects (such as expressions and shape of face parts) to highly face-specific marks (such as moles and scars).

While describable visual attributes are one of the most natural ways of describing faces, people also frequently use another technique: similes. A person's appearance can be described in terms of the similarity of a part of their face to the same part of another individual's. For example, someone's mouth might be like Angelina Jolie's, or their nose like Brad Pitt's. Dissimilarities also provide useful information – *e.g.*, her eyes are *not* like Jennifer Aniston's.

In this work, we show two major uses of classifiers trained on describable visual attributes and similes: face verification and image search. Face verification is the problem of determining whether two faces are of the same individual. What makes this problem difficult is that there is enormous variability in the manner in which an individual's face presents itself to a camera: not only might the pose differ, but so might the expression and hairstyle. Making matters worse – at least for researchers in biometrics – is that the illumination direction, camera type, focus, resolution, and image compression are all almost certain to vary as well. These manifold differences in images of the same person have confounded methods for automatic face recognition and verification, often limiting the reliability of automatic algorithms to the domain of more controlled settings with cooperative subjects. [4], [20], [22], [41], [43], [46], [48].

We approach the unconstrained face verification problem with non-cooperative subjects by comparing faces using our attribute and simile classifier outputs instead of low-level features directly. Fig. 1 shows the outputs of various attribute classifiers, for (a) two images of the same person and (b) images of two different people. Note that in (a), most attribute values are in strong agreement, despite the changes in pose, illumination, and expression, while in (b), the values are almost perfectly contrasting. By training


smiling asian men with glasses - Google Image Search - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Images Showing: All image sizes
Done
Mozilla Firefox
File Edit View History Bookmarks Tools Help
Search Images
Aligned
Faces
Images
javascript:void(0)

http://images.google.com/images?um=1&tab=wi&hl=en&q=smiling%20asian%20men%20with%20glasses

Results **1** - **20** of about **26,400** for **smiling asian men with glasses** with **Safesearch on**. (**0.02** seconds)

« View all web results for **smiling asian men with glasses**

Portrait of two elderly **Asian men**
167 x 250 - 22k - jpg
www.jupiterimages.com

asian man. **asian men**. **asian** people
167 x 250 - 22k - jpg
www.jupiterimages.com
[More from images.jupiterimages.com]

Man with **Glasses**, Portrait.
216 x 360 - 23k - jpg
www.comstock.com

Man with **Glasses**, Portrait.
216 x 360 - 20k - jpg
www.comstock.com
[More from images1.comstock.com]

... to be less happy than **men**, **...**
1000 x 664 - 273k - jpg
www.menscience.com

... to be less happy than **men**, **...**
200 x 133 - 4k - jpg
www.menscience.com

Stock Photo: **Smiling** Arms Folded
253 x 380 - 45k
www.istockphoto.com

a handsome middle aged **asian ...**
166 x 300 - 11k
www.fotosearch.com

... percent of **Asian men** suffer from **...**
143 x 215 - 8k - png
ahrp.blogspot.com

... The Eastbay **Asian** Youth Center
400 x 300 - 76k - jpg
www.danfontes.com

This is why the **men** are **smiling**.
400 x 300 - 93k - jpg
dadtalk.typepad.com

Very Southeast **Asian**, very Thai.
500 x 375 - 46k - jpg
tenguhouse.typepad.com

... I came upon a monk **smiling** at **...**
400 x 365 - 45k - jpg
www.kaichang.net

In her novel China **Men**, Maxine Hong **...**
350 x 276 - 17k - jpg
www.kaichang.net

... Southeast **Asian** notions that **men ...**
364 x 245 - 30k - jpg
content.cdlib.org

http://localhost:8080/?q=smiling%20asian%20men%20with%20glasses&page=1

FaceTracer Image Search

smiling asian men with glasses

Found **1344** results for **smiling asian men with glasses** in **0.220** secs. Displaying results **1** to **48**.

(a) Google image search results

(b) Attribute-based image search results

Fig. 2. Results for the query "smiling asian men with glasses," using (a) the Google image search engine (as of July 2009) and (b) our face search engine. Conventional image search engines rely on text annotations, such as file metadata, manual labels, or surrounding text, which are often incorrect, ambiguous, or missing. In contrast, we use attribute classifiers to *automatically* label images with faces in them, and store these labels in a database. At search time, only this database needs to be queried, and results are returned instantaneously. The attribute-based search results are much more relevant to the query.

a classifier that uses these labels as inputs for face verification, we achieve close to state-of-the-art performance on the Labeled Faces in the Wild (LFW) data set [25], at 85.29% accuracy.

LFW is remarkable for its variability in all of the aspects of visual appearance mentioned above, which also makes it a challenging benchmark for face verification algorithms. Our excellent performance on this benchmark shows that our particular approach to building and using attribute classifiers is, at least, adequate; however, how much better could one do? The attribute classifiers we train are currently binary, with continuous outputs approximated by the distance of a sample to the classification boundary. One could instead train regressors to directly estimate real-valued attribute outputs with greater accuracy. An upper-bound on the expected accuracy of attribute classification can be found by asking humans to label attributes. Thus, replacing the automatic classifier outputs with human labels, we found that accuracy on LFW goes up to 91.86%. Going even further, we asked humans to do the entire verification process. This experiment revealed the ideal to which automatic algorithms should aspire – 99.20%.

Given the tremendous strides in face recognition performance over the last two decades, in large part due to the introduction of larger and more realistic data sets, we have publicly released two large datasets: FaceTracer, which contains URLs to 15,000 face images and 5,000 attribute labels; and PubFig, which contains URLs to 59,476 images of 200 public figures – politicians and celebrities.

Another application of describable visual attributes is image search. The ability of current search engines to find images based on facial appearance is limited to images with text annotations. Yet, there are many problems with annotation-based image search: the manual labeling of images is time-consuming; the annotations are often incorrect or misleading, as they may refer to other content on a webpage; and finally, the vast majority of images are simply not annotated. Figs. 2a and 2b show the results of the query, "smiling asian men with glasses," using a conventional image search engine (Google Image Search, as of July 2009) and our search engine, respectively. The difference in quality of search results is clearly visible. Google's reliance on text annotations causes it to find images that have no relevance to the query, while our system returns only the images that match the query.

Both of our systems first require the creation of a large dataset of real-world face images. This is done by downloading images from the internet, running face detection and alignment, and then obtaining ground-truth attribute labels, all of which is described in Sec. 3. From this labeled data, one can train accurate attribute and simile classifiers fully automatically, as described in Sec. 4. Performing face verification using these attributes is described in Sec. 5, which also contains various experiments and looks at how well people perform on verification. Finally, the use of attributes for searching in large collections of images with faces is described in Sec. 6.

2 Related Work

Our work lies at the intersection of attribute classification, face verification and content-based image retrieval. We present an overview of the relevant work, organized by these topics.

2.1 Attribute Classification

Prior research on attribute classification has focused mostly on gender and ethnicity classification. Early works [10], [21] used neural networks to perform gender classification on small datasets. The Fisherfaces work [2] showed that linear discriminant analysis could be used for simple attribute classification such as glasses/no glasses. Later, Moghaddam and Yang [33] used Support Vector Machines (SVMs) [9] trained on small "face-prints" to classify the gender of a face, showing good results on the FERET face database [41]. The works of Shakhnarovich et al. [47] and Baluja and Rowley [1] used Adaboost [19] to select a linear combination of weak classifiers, allowing for almost real-time classification of face attributes, with results in the latter case again demonstrated on the FERET database. These methods differ in their choice of weak classifiers: the former uses the Haar-like features of the Viola-Jones face detector [52], while the latter uses simple pixel comparison operators. In a more general setting, Ferrari and Zisserman [18] described a probabalistic approach for learning simple attributes such as colors and stripes.

This journal paper builds on earlier conference works [27], [28]. Since then, others have also begun to explore the use of attributes for animal species identification [29] and object categorization [16]. The former shows how attributes can be learned from one dataset and transferred to a new one, possibly containing novel categories. They demonstrate results on a newly created animals dataset. The latter shows how one could learn attribute classifiers and apply them to object categorization, *e.g.*, on the Pascal VOC challenge [15]. While clearly demonstrating the potential of using attributes in a more general domain, their performance on VOC does not yet match the current state-of-the-art, showing how challenging this problem is. In contrast, by focusing on faces, we are able to train more accurate classifiers and use them to achieve competitive performance on an established real-world benchmark, as well as apply to them other tasks such as image search.

2.2 Face Verification

Early work in appearance-based face verification [26], [51] looked at the L_2 distance between pairs of images in a lower dimensional subspace obtained using Principal Components Analysis. This was extended and improved upon by using linear discriminant analysis [2]. However, these algorithms are mostly limited to images taken in highly controlled environments with extremely cooperative subjects. It is well understood that variation in pose and expression and, to a lesser extent, lighting cause significant difficulties for recognizing the identity of a person [56]. Illumination changes can be mostly handled using a variety of different approaches – the direction of the image gradient [7] and related image features such as SIFT [32], the phase of Gabor jets [53], and gradient pyramids [31] are all highly insensitive to lighting variation. The Pose, Illumination, and Expression (PIE) data set and follow-on results showed that sometimes alignment, especially in 3D, can overcome the other difficulties [4], [5], [8], [22], [48].

Unfortunately, in the setting of real-world images such as those in the "Labeled Faces in the Wild" (LFW) benchmark data set [25] and similar data sets [3], [14], 3D alignment is difficult and has not (yet) been successfully demonstrated. Various 2D alignment strategies have been applied to LFW – aligning all faces [23] to each other, or aligning each pair of images to be considered for verification [17], [35]. Approaches that require alignment between each image pair are computationally expensive. Our work does not require pairwise alignment. Neither do many other recent methods on LFW [44], [50], [54], [55], all of which use a large set of carefully designed local features. The best-performing of these [55] ranks the similarity of each face in an input pair to those in a "background set," which is similar in spirit to our simile classifiers.

2.3 Content-Based Image Retrieval (CBIR)

Our search application can be viewed as a form of CBIR, where our content is limited to images with faces. Interested readers can refer to the work of Datta et al. [12] and Lew et al. [30] for a recent survey of this field. Most relevant to our work is the "Photobook" system [40], which allows for similarity-based searches of faces and objects using parametric eigenspaces. However, their goal is different from ours. Whereas they try to find objects similar to a chosen one, we locate a set of images starting only with simple text queries. Although we use vastly different classifiers and methods for feature selection, their division of the face into functional parts such as the eyes, nose, etc., is echoed in our approach of training classifiers on functional face regions. While in this paper we ignore existing text annotations for images, one could envision using describable attributes in combination with such annotations for improved search performance, somewhat akin to the idea presented in the "Names and Faces" work [3].

3 Creating Labeled Image Datasets

Two recent trends in internet services have made collecting and labeling image data dramatically easier. The first, large internet photo-sharing sites such as flickr.com and picasa.com are growing exponentially and host billions of public images, many with textual annotations and comments. In addition, search engines such as Google Images allow searching for images of particular people (albeit not perfectly). The second, efficient marketplaces for small amounts of

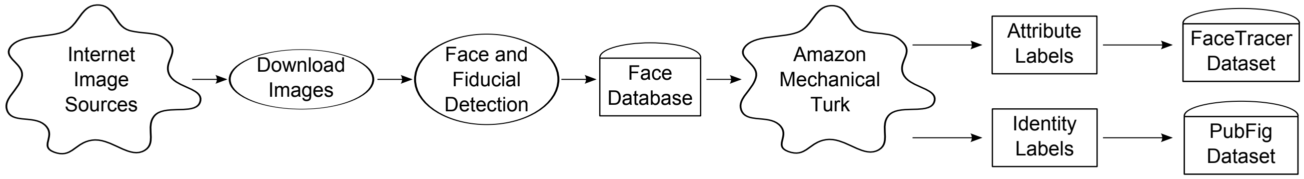


Fig. 3. Creating labeled image datasets. Our system downloads images from the internet. These images span many sources of variability, including pose, illumination, expression, cameras, and environment. Next, faces and fiducial points are detected using a commercial detector [37] and are stored in a face database. Faces are submitted to the Amazon Mechanical Turk service, where they are labeled with attributes or identity, which are used to create the FaceTracer and PubFig datasets, respectively. Both datasets have been publicly released for non-commercial use.

labor, such as Amazon's Mechanical Turk (MTurk)[1], make it possible to purchase small amounts of web-based labeling effort with very low overhead. We exploit both of these trends to create a large dataset of real-world images with attribute and identity labels, as shown in Fig. 3 and described next.

3.1 Collecting Face Images

We use a variety of online sources for collecting a large dataset of real-world images, including search engines such as Google Images and photo-sharing websites such as flickr.com. Depending on the type of data needed, one can either search for particular people's names (to build a dataset labeled by identity) or for default image filenames assigned by digital cameras (to use for labeling with attributes). The latter technique allows one to find images that are otherwise not returned in users' queries, *i.e.*, images which are effectively "invisible." Relevant metadata such as image and page URLs are stored in the EXIF tags of the downloaded images.

Next, we apply the OKAO face detector [37] to the downloaded images to extract faces. This detector also returns the pose angles of each face, as well as the locations of six fiducial points: the corners of both eyes and the corners of the mouth. These fiducial points are used to align faces to a canonical pose by applying an affine transformation. This transform is computed using linear least squares on the detected points and corresponding points defined on a template face. The 3.1 million aligned faces collected using this procedure comprise the Columbia face database.

We would like to draw attention to three observations about this database. First, from the statistics of the randomly-named images, it appears that a significant fraction of them contain faces (25.7%), and on average, each image contains 0.5 faces. Second, our collection of aligned faces is the largest such collection of which we are aware. It is truly a "real-world" dataset, with completely uncontrolled lighting and environments, taken using unknown cameras and in unknown imaging conditions, with a wide range of image resolutions. In contrast, existing face datasets such as Yale Face A&B [20], CMU PIE [48], and FERET [41] are either much smaller in size and/or taken in highly controlled settings. Even the more expansive FRGC version 2.0 dataset [42] has a limited number of subjects, image acquisition locations, and all images were taken with the same camera type. The most comparable dataset is LFW [25], itself derived from earlier work [3]. These images were collected from news sources, and exhibit most of the same types of variation as the Columbia face dataset.

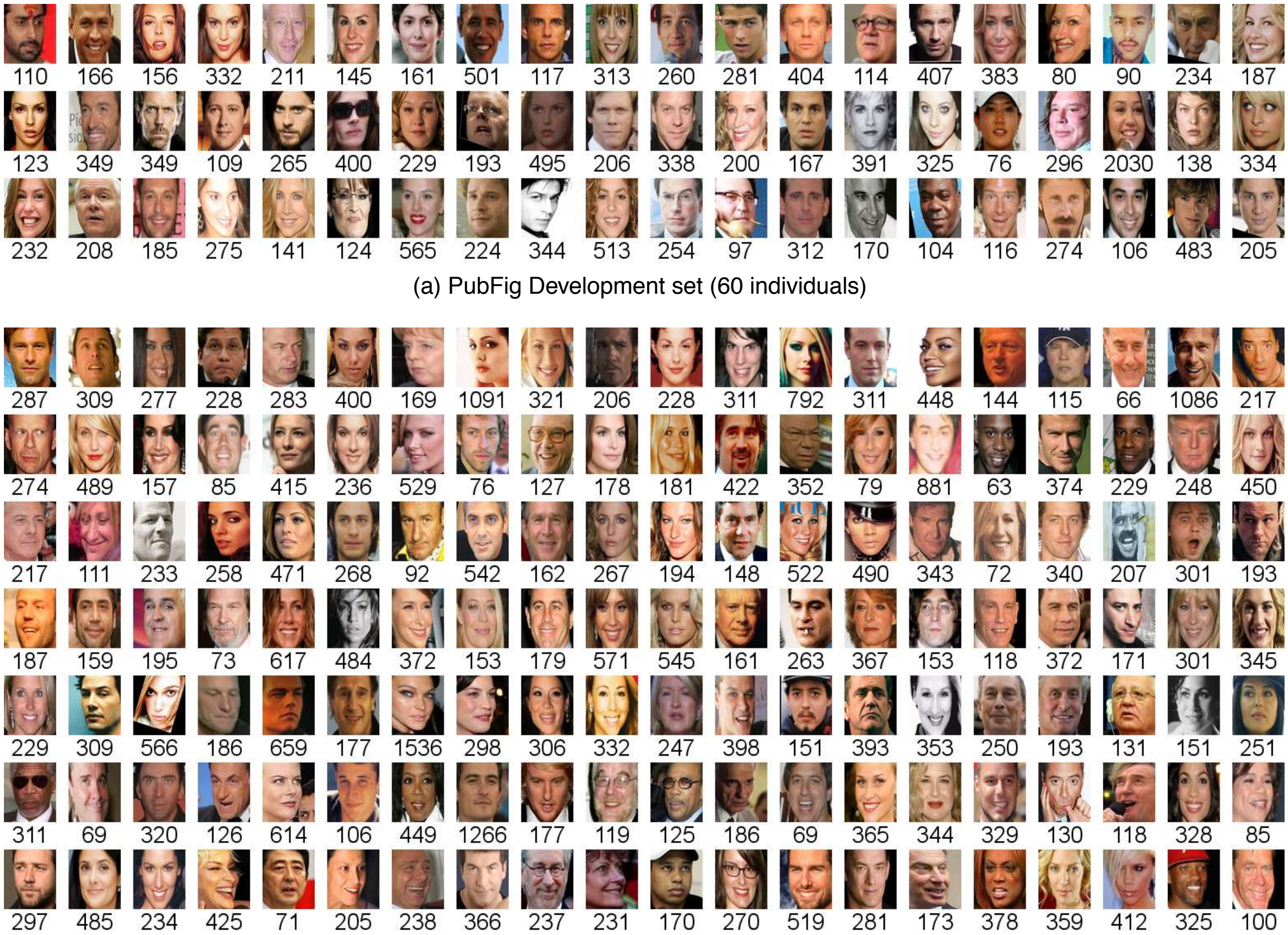
3.2 Collecting Attribute and Identity Labels

For labeling images in our Columbia face database, we use the Amazon Mechanical Turk (MTurk) service. This service matches online workers to online jobs. *Requesters* can submit jobs to be completed by *workers*, optionally setting various quality controls such as confirmation of results by multiple workers, filters on minimum worker experience, *etc*.

We submitted 73,000 attribute labeling jobs showing 30 images to 3 workers per job, presenting a total of 6.5 million images to users. The jobs asked workers to select face images which exhibited a specified attribute. (A few manually-labeled images were shown as examples.) Only labels where all 3 people agreed were used. From this raw data, we were able to collect over 125,000 triply-verified positive attribute labels in a month, for under $5,000.

Although this approach is somewhat similar to other labeling efforts in the computer vision community – such as ImageNet [13] and LabelMe [45], which focus on naming objects, images, and regions of images using nouns – there are several important differences. One is that attributes need not be binary or even discrete; a person's age or the thickness of their eyebrows are both continuous attributes. Another critical difference is that visual attributes can be composed more freely than names, which generally exist in a tree-structured hierarchy. This allows for the use of a set of general attributes, which can be combined in an exponential number of ways to describe many objects at different levels of specificity. Attributes can therefore compactly provide a great deal of information, both about object properties and their identity. Finally, for many objects, it can be

[1] http://mturk.com



(a) PubFig Development set (60 individuals)

(b) PubFig Evaluation set (140 individuals)

Fig. 4. The PubFig dataset consists of $59,476$ images of 200 public figures – celebrities and politicians – partitioned into (a) a development set of 60 individuals and (b) an evaluation set of 140 individuals. Below each thumbnail is shown the number of photos of that person. There is no overlap in either identity or image between the development set and any dataset that we evaluate on, including Labeled Faces in the Wild (LFW) [25].

prohibitively expensive to obtain a large number of labeled training images. In contrast, the same attribute can be exhibited by many otherwise-unrelated objects, making it easier to find more example images of the attribute.

For gathering identity labels, we used the images downloaded from keyword searches on people's names. The MTurk jobs shown to users asked them to select only the face images of a given person (of whom a few examples were shown). We also ran additional jobs pruning images for quality, good alignment, and some conservative duplicate-removal.

From these attribute and identity labels and our face database, we have created two publicly available face datasets, described next.

3.3 FaceTracer Dataset

The FaceTracer dataset is a subset of the Columbia face database, along with attribute labels. Each of the 15,000 faces in the dataset has a variety of metadata and fiducial points marked. The attributes labeled include demographic information such as age and race, facial features like mustaches and hair color, and other attributes such as expression, environment, *etc*. There are 5,000 labels in all. FaceTracer can be used as simply a dataset of real-world images with face detections and fiducials; or by researchers wanting to train their own attribute classifiers; or for any other non-commercial purpose.

The dataset is publicly available as a set of face URLs and accompanying data at http://faceserv.cs.columbia.edu/databases/facetracer/

3.4 PubFig Dataset

The PubFig dataset is a more direct complement to the LFW dataset [25]. It consists of $59,476$ images of 200 public figures. The larger number of images per person (as compared to LFW) allows one to construct subsets of the data across different poses, lighting conditions, and expressions for further study. In addition, this dataset is well-suited for recognition experiments.

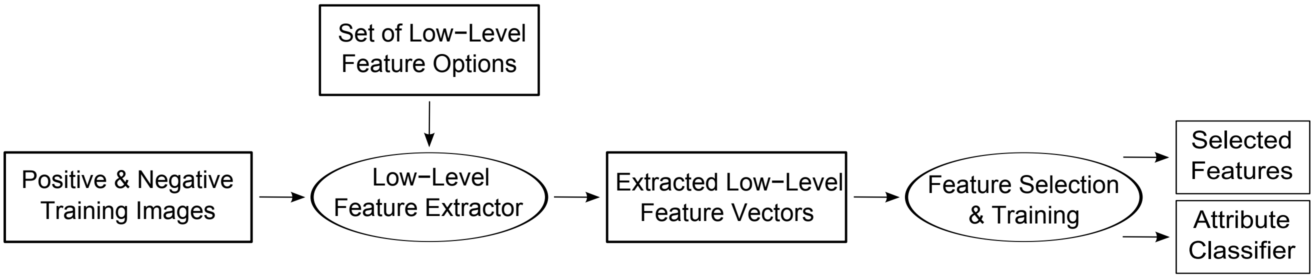

Set of Low−Level Feature Options
Positive & Negative Training Images
Low−Level Feature Extractor
Extracted Low−Level Feature Vectors
Feature Selection & Training
Selected Features
Attribute Classifier

Fig. 5. Overview of attribute training architecture. Given a set of labeled positive and negative training images, low-level feature vectors are extracted using a large pool of low-level feature options. (Each feature option consists of a region chosen from Fig. 6 and a feature type chosen from Table 1.) An automatic iterative selection process then picks the best set of features for correctly classifying the input data. The outputs are the selected features and the trained attribute classifier.

PubFig is divided into a development set of 60 people, on which we trained our simile classifiers (described in Sec. 4.4), and an evaluation set of 140 people. The evaluation set was used to create a face verification benchmark similar to that from LFW.

All the data (again, as URLs to images) and evaluation benchmarks from PubFig are publicly available for non-commercial use at http://faceserv.cs.columbia.edu/databases/pubfig/

4 Learning Visual Attributes

Given a particular describable visual attribute – say "gender" – how can one train a classifier for the attribute? Let us answer this question by first formalizing our notion of attributes. Attributes can be thought of as functions $\mathbf{a_i}$ that map images I to real values a_i. Evaluating the function can be thought of as measuring the attribute. Large positive values of a_i indicate the presence or strength of the ith attribute, while negative values indicate its absence.

Consider the attribute "gender." If images I_1 and I_2 are of males and image J is of a female, we would like our gender function $\mathbf{a_g}$ to map these input images onto the real line, with males assigned positive values and females negative values. Notice that images I_1 and I_2 could differ in many respects – lighting, pose, age, expression, and other attributes – and yet the gender classifier should still mark them as male. We would like these classifiers to measure the degree of the attribute as well. For instance, if I_1 were an image of Clint Eastwood and I_2 were an image of Orlando Bloom, we would want $\mathbf{a_g}(I_1) > \mathbf{a_g}(I_2)$. Similarly, the classifier for a different attribute – age, for example – should give reliable results despite changes in gender.

Similes are another class of describable visual traits. Simile functions describe the similarity of a face region between two different individuals. For example, we could say a person has "eyes like Penelope Cruz's" or a "mouth like Angelina Jolie's." We can formalize these two simile functions as $\mathbf{s_{cruz_{eyes}}}$ and $\mathbf{s_{jolie_{mouth}}}$, and someone who shared Cruz's eyes but not Jolie's mouth would thus have a positive value for the former and a negative value for the latter.

Learning an attribute or simile classifier consists of fitting a function to a set of labeled training data. If the training labels are ±1, this can be seen as fitting a classification function; real-valued labels imply regression; and if only ordering constraints are given, it becomes a problem of learning ranking functions. In all cases, regularization is important because the inputs, low-level image features, are very high-dimensional with complex variation, and there is always limited training data. This regularization could be biased by the distribution of features actually observed, which can be acquired from both labeled and unlabeled data. In this work, we consider mainly binary classifiers; regressors would likely behave very similarly, though possibly with greater accuracy.

4.1 Training Architecture

An overview of the attribute training architecture is shown in Fig. 5. The key idea is to leverage the many efficient and effective low-level features that have been developed by the computer vision community, choosing amongst a large set of them to find the ones suited for learning a particular attribute. This process should ideally be done in a generic, application- and domain-independent way, but with the ability to take advantage of domain-specific knowledge where available.

For the domain of faces, this knowledge consists of an affine alignment procedure and the use of low-level features which have proven to be very useful in a number of leading vision techniques, especially for faces. The alignment takes advantage of the fact that all faces have common structure – *i.e.*, two eyes, a nose, a mouth, *etc.*– and that we have fiducial point detections available from a face detector [37]. The low-level features are described in the next section.

4.2 Low-Level Features

As described in Sec. 3.1, face images are first aligned using an affine transformation. A set of k low-level feature extractors $\mathbf{f_j}$ are applied to an aligned input image I to form a feature set $\mathcal{F}(I)$:

$$\mathcal{F}(I) = \{\mathbf{f_1}(I), \cdots, \mathbf{f_k}(I)\}\,. \tag{1}$$

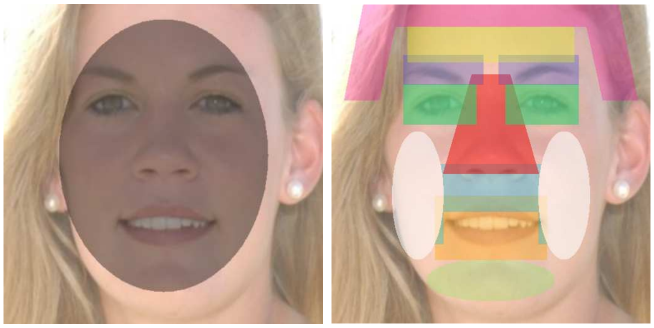


Fig. 6. The face regions used for automatic feature selection are shown here on an affine-aligned face image. There is (a) one region for the whole face, and (b) nine regions corresponding to functional parts of the face, such as the mouth, eyes, nose, *etc*. Regions are large enough to contain the face part across changes in pose, small errors in alignment, and differences between individuals. The regions are manually defined, once, in the affine-aligned coordinate system, and can then be used automatically for all aligned input faces.

We describe each extractor $\mathbf{f_i}$ in terms of four choices: the region of the face to extract features from, the type of pixel data to use, the kind of normalization to apply to the data, and finally, the level of aggregation to use.

The complete set of our 10 regions are shown in Fig. 6. The regions correspond to functional parts of a face, such as the nose, mouth, *etc*., similar to those defined in the work on modular eigenspaces [39]. Regions are defined manually in the affine-aligned coordinate system. This only has to be done once, after which all aligned faces can use the same region definitions. Our coarse division of the face allows us to take advantage of the common geometry shared by faces, while allowing for differences between individual faces as well as robustness to small errors in alignment. Prior to feature extraction, we mask out the background to avoid contaminating the classifiers. We also use the detected yaw angles of the face to first flip images so that they always face left. This small tweak makes the classifier's job slightly easier, as the "good" side of the face is always on the same half of the image.

From each region, one can extract different types of information, as categorized in Table 1. The types of pixel data to extract include various color spaces (RGB, HSV) as well as edge magnitudes and orientations. To remove lighting effects and better generalize across a limited number of training images, one can optionally normalize the extracted values. One method for normalization is mean normalization, $\hat{x} = \frac{x}{\mu}$, which removes illumination gains. Another option is energy normalization, $\hat{x} = \frac{x-\mu}{\sigma}$, which removes gains as well as offsets. (In these equations, x refers to the input value, μ and σ are the mean and standard deviation of all the x values within the region, and $\hat{x}$ refers to the normalized output value.)

TABLE 1
Feature type options

Pixel Value Types	Normalizations	Aggregation
RGB HSV Image Intensity Edge Magnitude Edge Orientation	None Mean Normalization Energy Normalization	None Histogram Mean/Variance

Finally, one can aggregate information over the region rather than simply concatenating all values. This can be as simple as using only the mean and variance, or include more information by computing a histogram of values over the region. A complete feature type is created by choosing a region from Fig. 6 and one entry from each column of Table 1.

4.3 Attribute Classifiers

In creating a classifier for a particular attribute, we could simply extract all types of low-level features from the whole face, and let a classifier figure out which are important for the task and which are not. This, however, puts too great a burden on the classifier, confusing it with non-discriminative features. Instead, we design a selection procedure which automatically chooses the best features from a rich set of feature options. The chosen features are used to train the final attribute or simile classifier.

Attribute classifiers C_i are built using a supervised learning approach. Training requires a set of labeled positive and negative images for each attribute, examples of which are shown in Fig. 7. The goal is to build a classifier that best classifies this training data by choosing an appropriate subset of the feature set $\mathcal{F}(I)$ described in the previous section. We do this iteratively using forward feature selection. In each iteration, we first train several individual classifiers on the current set of features in the output set concatenated with a single region-feature combination. Each classifier's performance is evaluated using cross-validation. The features used in the classifier with the highest cross-validation accuracy are added to the output set. We continue adding features until the accuracy stops improving, up to a maximum of 6 low-level features. For computational reasons, we drop the lowest-scoring 70% of features at each round, but never dropping below 10 features.

Our classifiers are Support Vector Machines (SVMs) [9] with RBF kernels, trained using libsvm [6]. For each classifier, we use between 500 to 2000 positive and negative examples each, and perform a grid search over the C and γ parameters. The entire process is fully automatic, and takes a few hours of computation time per attribute trained, using a small grid of roughly 20 Intel Xeon processors, running at 3.0 Ghz each. We note that our procedure is by no means optimal; picking optimal features for non-

TABLE 2
Comparison of attribute classification performance

Classification Method	Gender Error Rate	Smiling Error Rate
Attribute Classifiers	**8.62%**	**4.67%**
Pixel comp. feats. [1]	13.13%	7.41%
Haar-like feats. [47]	12.88%	6.40%
Full-face SVM [33]	9.52%	13.54%

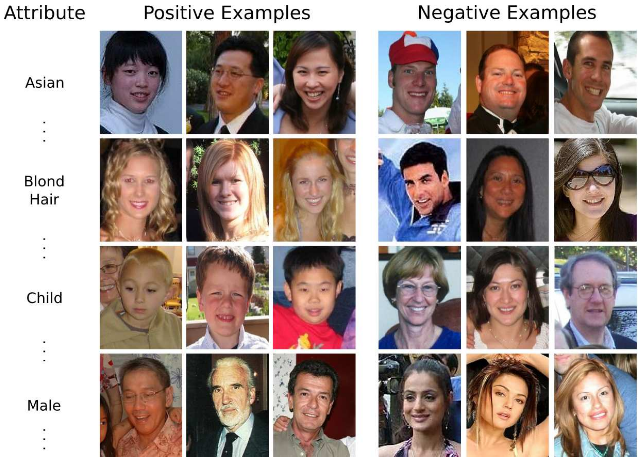


Fig. 7. Training data for the attribute classifiers consists of face images that match the given attribute label (positive examples) and those that don't (negative examples). Shown here are a few of the training images used for four different attributes. Final classifier accuracies for all 65 attributes are shown in Table 3.

TABLE 3
Accuracies of our 65 attribute classifiers

Attribute	Accuracy	Attribute	Accuracy
Asian	92.3%	Mouth Closed	89.3%
Attractive	81.1%	Mouth Open	85.1%
Baby	90.5%	Mustache	91.9%
Bags Under Eyes	86.2%	No Beard	89.5%
Bald	83.2%	No Eyewear	93.5%
Bangs	88.7%	Nose Shape	86.9%
Black	88.7%	Nose Size	87.5%
Black Hair	80.3%	Nose-Mouth Lines	93.1%
Blocked Forehead	79.1%	Obscured Forehead	77.0%
Blond Hair	78.0%	Oval Face	70.3%
Blurry	92.1%	Pale Skin	89.4%
Brown Hair	72.4%	Posed Photo	69.7%
Child	83.6%	Receding Hairline	84.2%
Chubby	77.2%	Rosy Cheeks	85.8%
Color Photo	95.5%	Round Face	74.3%
Curly Hair	68.9%	Round Jaw	67.0%
Double Chin	77.7%	Senior	88.7%
Environment	84.8%	Shiny Skin	84.7%
Eye Width	90.0%	Sideburns	71.1%
Eyebrow Shape	80.9%	Smiling	95.3%
Eyebrow Size	93.4%	Soft Lighting	67.8%
Eyeglasses	91.6%	Square Face	81.2%
Eyes Open	92.5%	Straight Hair	76.8%
Flash Lighting	72.3%	Sunglasses	94.9%
Frowning	95.5%	Teeth Not Visible	91.6%
Gaping	89.6%	Teeth Visible	91.6%
Goatee	80.3%	Visible Forehead	89.4%
Gray Hair	87.2%	Wavy Hair	64.5%
Harsh Lighting	78.7%	Wearing Hat	86.0%
High Cheekbones	84.7%	Wearing Lipstick	86.8%
Indian	86.5%	White	91.5%
Male	85.2%	Youth	85.8%
Middle-Aged	78.4%		

linear classifiers is still an open problem in machine learning. Nevertheless, we obtain excellent results in practice.

While we have designed our classifier architecture to be flexible enough to handle a large variety of attributes, it is important to ensure that we have not sacrificed accuracy in the process. We therefore compare our approach to three previous state-of-the-art methods for attribute classification: full-face SVMs using brightness normalized pixel values [33], Adaboost using Haar-like features [47], and Adaboost using pixel comparison features [1]. Since these works have mostly focused on gender classification, we use that attribute as the first testing criteria. In addition, we also test performance on the "smiling" attribute – which we expect to be localizable to a small region of the face: the mouth.

Results are shown in Table 2. Our method performs the best in all cases (in some cases significantly so). This highlights the power of doing feature selection; in particular, we see that the full-face SVM method, while performing reasonably well on gender, did much worse on a localized attribute like smiling. Note that for the purposes of this test, we limited images to mostly frontal faces.

Using the Columbia face database and the training procedure just described, we trained a total of 65 attribute classifiers. Their accuracies on held out data are shown in Table 3. Typical accuracies range from 80% to 90%.

4.4 Simile Classifiers

Simile classifiers measure the similarity of part of a person's face to the same part on a set of reference people. We use the 60 individuals from the develop-

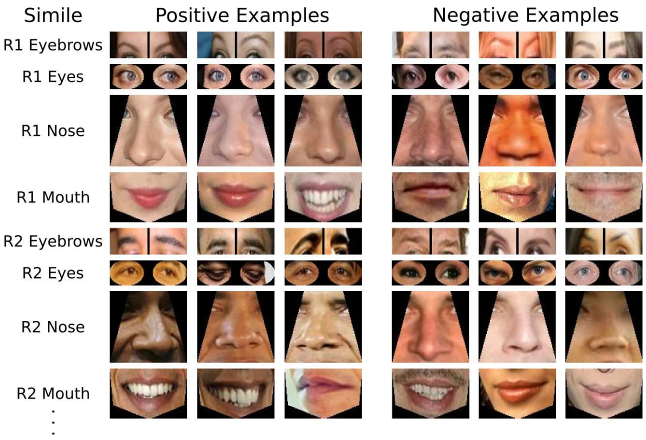


Fig. 8. Each simile classifier is trained using several images of a specific reference person, limited to a small face region such as the eyes, nose, or mouth. We show here three positive and three negative examples, for four regions on two of the reference people used to train these classifiers.

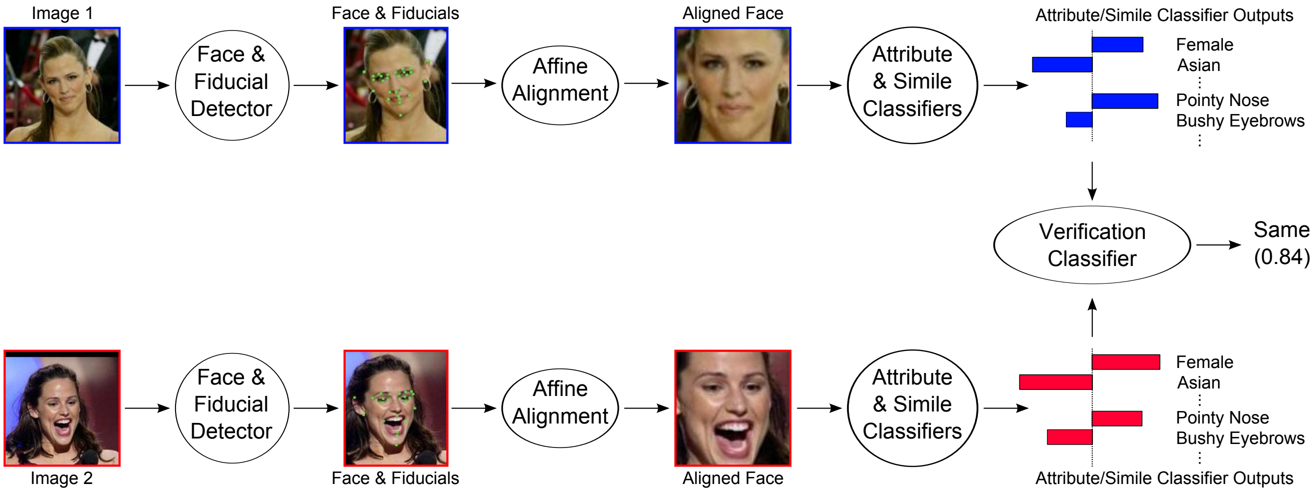


Fig. 9. The face verification pipeline. A pair of input images are run through a face and fiducial detector [37], and the fiducials are then used to align both faces to a common coordinate system. The aligned face images are fed to each of our attribute and simile classifiers individually to obtain a set of attribute values. Finally, these values are compared using a verification classifier to make the output determination, which is returned along with the distance to the decision boundary.

ment set of PubFig as the reference people. The left part of Fig. 8 shows examples of regions selected from reference persons *R1* (top) and *R2* (bottom) as positive examples. On the right are negative examples, which are simply the same region extracted from other individuals' images.

We emphasize two points. First, the individuals chosen as reference people *do not appear* in LFW or other benchmarks on which we produce results. Second, we train simile classifiers to recognize similarity to *part* of a reference person's face in *many* images, not similarity to a single image. The use of face parts increases the number of classifiers, but makes each one easier to learn, while the use of several input images allows for much better generalizability.

For each reference person, we train support vector machines to distinguish a region (*e.g.*, eyebrows, eyes, nose, mouth) on their face from the same region on other faces. We choose eight regions and six feature types from the set of possible features described in Sec. 4.2 and train classifiers for each simile using the training procedure described in the previous section. Each simile classifier is trained using at most 600 positive samples (face images) of the reference person, and at most 10 times as many negative samples, randomly chosen from images of other people in the training set.

5 Face Verification

Existing methods for face verification – "are these two faces of the same person" – often make mistakes that would seem to be avoidable: men being confused for women, young people for old, asians for caucasians, *etc*. On the other hand, small changes in pose, expression, or lighting can cause two otherwise similar images of the same person to be misclassified by an algorithm as different. Based on this observation, we hypothesized that the attribute and simile classifiers could avoid such mistakes.

5.1 Training a Verification Classifier

Fig. 9 illustrates how attribute-based face verification is performed on a new pair of input images. In order to make a decision about whether two face images I_1 and I_2 show the same person, one can train a verification classifier V to compare attribute vectors $\mathbf{C}(I_1)$ and $\mathbf{C}(I_2)$. These vectors are constructed by concatenating the result of n different attribute and/or simile classifiers.

To build a final verification classifier, let us make some observations about the particular form of our classifiers:

1) Values $C_i(I_1)$ and $C_i(I_2)$ from the ith classifier should be similar if the images are of the same individual, and different otherwise.
2) Classifier values are raw outputs of binary classifiers (in the range $[-1, 1]$), where the objective function is trying to separate examples around 0. Thus, the signs of values should be important.
3) The particular choice of classifier used, SVMs, optimize only for separating data at the separation boundary, and so differences in values close to 0 are more important than differences between those with greater absolute values, *e.g.*, the difference between -0.1 and 0.1 matters more than that between 0.8 and 1.0, even though they both have a difference of 0.2.

Let $a_i = C_i(I_1)$ and $b_i = C_i(I_2)$ be the outputs of the ith trait classifier for each face ($1 \leq i \leq n$). One would like to combine these values in such a way that our second-stage verification classifier V can make sense of the data. This means creating values that are large (and positive) when the two inputs are of the same individual, and negative otherwise.

Based on the above 3 observations, let us construct this combination. From observation (1), one can see that using the absolute difference $|a_i - b_i|$ will yield the desired outputs. However, based on the 3rd observation, this will unduly punish differences at high magnitudes. This effect can be ameliorated by multiplying with a Gaussian of variance 1 centered at 0: $g\left(\frac{1}{2}(a_i + b_i)\right) \cdot |a_i - b_i|$, where g is the given Gaussian. Finally, using the 2nd observation, one can add another term to the combination – the product: $a_i \cdot b_i$.

Putting both terms together yields the tuple p_i:

$$p_i = \left\langle g\left(\frac{1}{2}(a_i + b_i)\right) \cdot |a_i - b_i|, a_i \cdot b_i \right\rangle \qquad (2)$$

(We found that changing the exact nature of this tuple – *e.g.*, by using either the difference or the product, or not applying the gaussian weighting – did not affect accuracy by more than 1%.)

The concatenation of these tuples for all n attribute/simile classifier outputs forms the input to the verification classifier V:

$$v(I_1, I_2) = V(\langle p_1, \ldots, p_n \rangle) \qquad (3)$$

Training V requires pairs of positive examples (two images of the same person) and negative examples (images of two different people). In our experiments, we use an SVM with an RBF kernel for V, trained using libsvm [6] with default parameters for C and γ.

5.2 Experimental Setup

We explore face verification performance on the Labeled Faces in the Wild (LFW) benchmark [25] and on our PubFig benchmark. For each computational experiment, a set of pairs of face images is presented for training, and a second set of pairs is presented for testing. In all experiments, not only are the images in the training and test sets disjoint, but there is also no overlap in the individuals used in the two sets. In addition, the individuals and images used to train the attribute and simile classifiers are disjoint from the testing sets.

5.3 Attribute Classifier Results on LFW

The LFW dataset consists of 13,233 images of 5,749 people, organized into 2 "views": a development set of 2,200 pairs for training and 1,000 pairs for testing, on which to build models and choose features; and a 10-fold cross-validation set of 6,000 pairs, on which to evaluate final performance. We used View 1 for high-level model selection (*e.g.*, representation for the final classifier V) and evaluated our performance on each of the folds in View 2 using the "image restricted configuration," as described in the LFW paper [25].

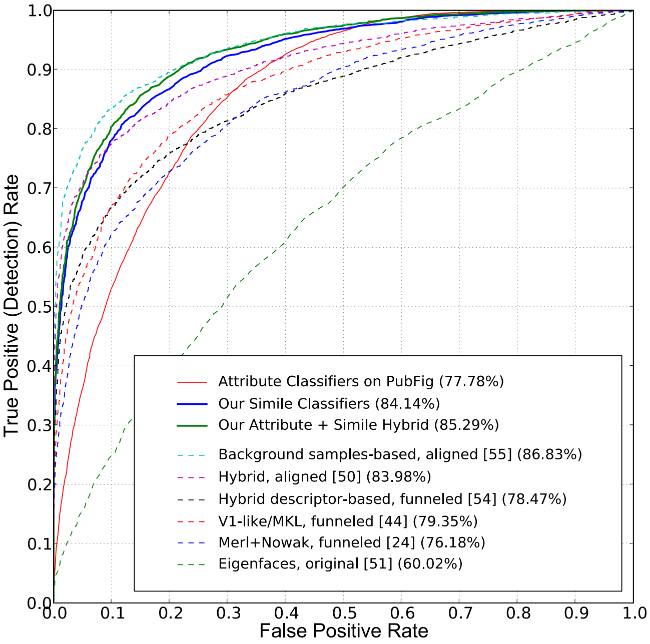


Fig. 10. Face verification performance on LFW of our attribute classifiers, simile classifiers, and a hybrid of the two are shown in solid red, blue, and green, respectively. Dashed lines are existing methods. Our highest accuracy is 85.29%, which is comparable to the current state-of-the-art accuracy of 86.83% [55]. The fact that the hybrid performance is better than either attribute or similes alone suggests that the two capture different kinds of information.

A verification classifier V is trained using nine folds from View 2 of LFW and then evaluated on the remaining fold, cycling through all ten folds. Receiver Operating Characteristic (ROC) curves are obtained by saving the classifier outputs for each test pair in all ten folds and then sliding a threshold over all output values to obtain different false positive/detection rates. An overall accuracy is obtained by using only the signs of the outputs (*e.g.*, thresholding at 0) and counting the number of errors in classification. The standard deviation for the accuracy is obtained by looking at the accuracies for each fold individually.

Fig. 10 shows results on LFW for our attribute classifiers (red line), simile classifiers (blue line), and a hybrid of the two (green line), along with several previous methods (dotted lines) [24], [44], [50], [51], [54], [55]. The accuracies for each of our methods are $83.62\% \pm 1.58\%$, $84.14\% \pm 1.31\%$, and $85.29\% \pm 1.23\%$, respectively.[2] Our highest accuracy of 85.29% is comparable to the 86.83% accuracy of the current state-of-the-art method [55] on LFW.

5.4 Human Attribute Labels on LFW

Although our methods already achieve close to the current best performance on LFW so far, it is inter-

[2] Our face detector [37] was unable to detect one or more faces in 53 of the 6,000 total pairs. For these, we assumed average performance.

esting to consider how well attribute classifiers could potentially do. There are several reasons to believe that the current results are only first steps towards this ultimate goal:

- We have trained only 65 attribute classifiers so far. Adding more attributes, such as the presence and location of highly discriminative facial features including moles, scars, and tattoos, should greatly improve performance.
- Of the 65 attributes, many are not discriminative for verification. For example, facial expression, scene illumination, and image quality are all unlikely to aid in verification. There is also a severe imbalance in LFW of many basic attributes such as gender and age, which reduces the expected benefit of using these attributes for verification.
- The attribute functions were trained as binary classifiers rather than as continuous regressors. While we use the distance to the separation-boundary as a measure of degree of the attribute, using regression may improve results.

With the hope of exploring what might be possible using attribute, we performed an experiment in which our automatic attribute labeling process was replaced by human labels, keeping the verification process identical. MTurk workers were asked to label attributes for all faces in the LFW View 2 benchmark set. We averaged seven user-responses per image to obtain smoothed estimates of the attribute values.

Fig. 11 shows a comparison of face verification performance on LFW using either these human attribute labels (blue line) or our automatically-computed classifier outputs (red line). In both cases, the labels are fed to the verification classifier V and training proceeds identically, as described earlier. The same set of attributes were used for each corresponding point on the graphs. Verification results using the human attribute labels reach 91.86% accuracy with 18 attributes, significantly outperforming our computed labels at 81.57% for the same 18 attributes. Moreover, the drop in error rates from computational to human labels is actually *increasing* with more attributes, suggesting that adding more attributes could greatly improve accuracies even further.

5.5 Human Verification on LFW

The high accuracies obtained in the previous section lead to a natural question: How well do people perform on the verification task itself? While many algorithms for automatic face verification have been designed and evaluated on LFW, there are no published results about how well people perform on this benchmark. Furthermore, it is unknown what characteristics of the dataset might make it easier or harder to perform the verification task. To this end, we conducted several experiments on human verification.

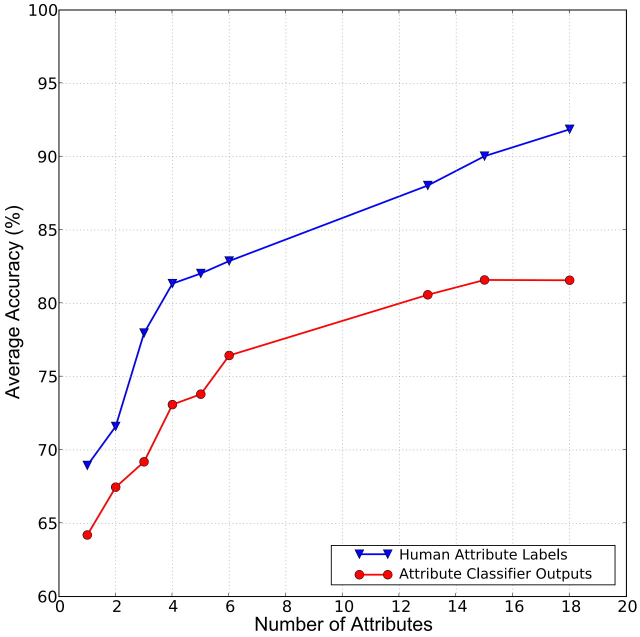


Fig. 11. Comparison of face verification performance on LFW using human attribute labels (blue line) vs. automatically-computed classifier outputs (red line). Verification using human labels consistently outperforms that using classifier outputs. With 18 attributes, human attribute labels reach 91.86% accuracy compared to only 81.57% using classifier outputs. Training better attribute classifiers (or regressors) could thus greatly improve our verification performance.

We followed the procedure of [38] to obtain this data, using Amazon Mechanical Turk. MTurk users were shown pairs of faces from the LFW View 2 benchmark set and asked to mark whether the images showed the same person or not. This was done on a scale of -1 to +1, where the sign of the score was their decision (positive if the pair was of the same individual, negative otherwise), and the magnitude was their confidence in their response. The responses of 10 different users were averaged per face pair to get a score for that pair. (Thus, for the 6,000 image pairs in LFW, we gathered 60,000 data points from users for each of the three tests described below, for a total of 240,000 user inputs.) An ROC curve was created by sliding the confidence threshold from -1 to 1, counting scores less than the threshold as “different” and those above as “same.”

We first performed a test using the original LFW images. The results are shown in red in Fig. 12. At 99.20% accuracy, people are essentially perfect on this task. We now look at tougher variants of this test.

The first variant is to crop the images tightly around the face. We do this by blacking out most of the image, leaving only the face visible (including at least the eyes, nose and mouth, and possibly parts of the hair, ears, and neck). This test measures how much people are helped by the context (sports shot, interview, press conference, *etc.*), background (some images of

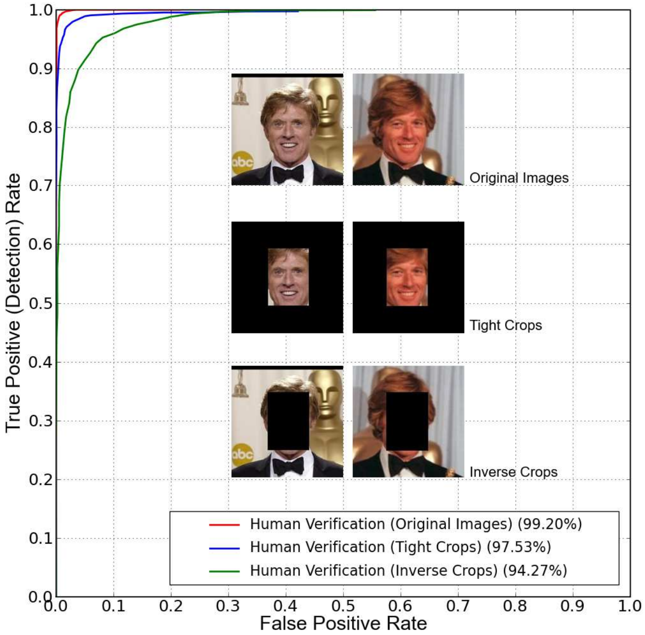


Fig. 12. Face verification performance on LFW by humans is almost perfect (99.20%) when people are shown the original images (red line). Showing a tighter cropped version of the images (blue line) drops their accuracy to 97.53%, due to the lack of context available. The green line shows that even with an inverse crop, *i.e.*, when *only* the context is shown, humans still perform amazingly well, at 94.27%. This highlights the strong context cues available on the LFW dataset. All of our methods mask out the background to avoid using this information.

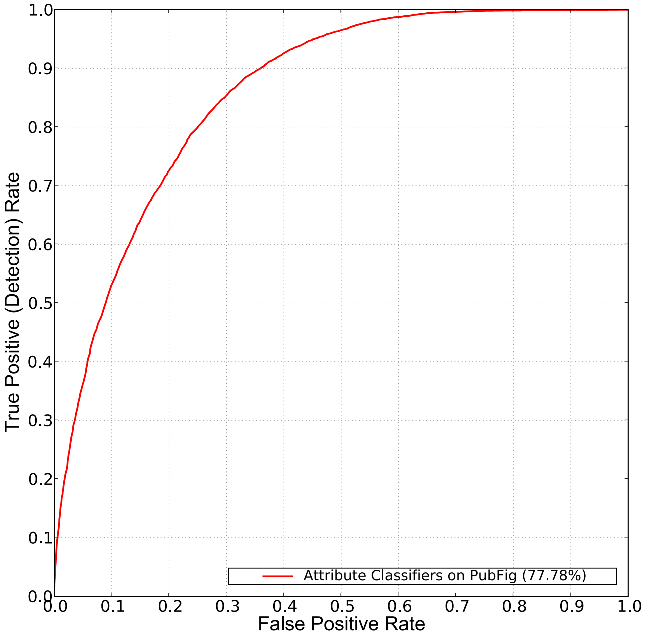


Fig. 13. Face verification results on the PubFig evaluation benchmark using our attribute classifiers. Our accuracy is 77.78% on this benchmark, which consists of 20,000 face pairs partitioned into 10 folds for cross-validation. Our lower performance on this test as compared to LFW suggests that this is a more challenging benchmark.

individuals were taken with the same background), and hair (although sometimes it is partially visible). The results are shown in blue in Fig. 12. Performance drops quite a bit, to 97.53% – a tripling of the error rate.

To confirm that the region outside of the face is indeed helping people with identification, we ran a third test where the mask was inverted – *i.e.*, we blacked out the face but showed the remaining part of the image. Astonishingly, people still achieve 94.27% accuracy, as shown by the green line in Fig. 12. These results suggest that automatic face verification algorithms should avoid regions outside of the face, as they could artificially boost accuracy in a manner not applicable on real data. (In all experiments involving our attribute and simile classifiers, we masked out the background.)

5.6 Attribute Classifier Results on PubFig

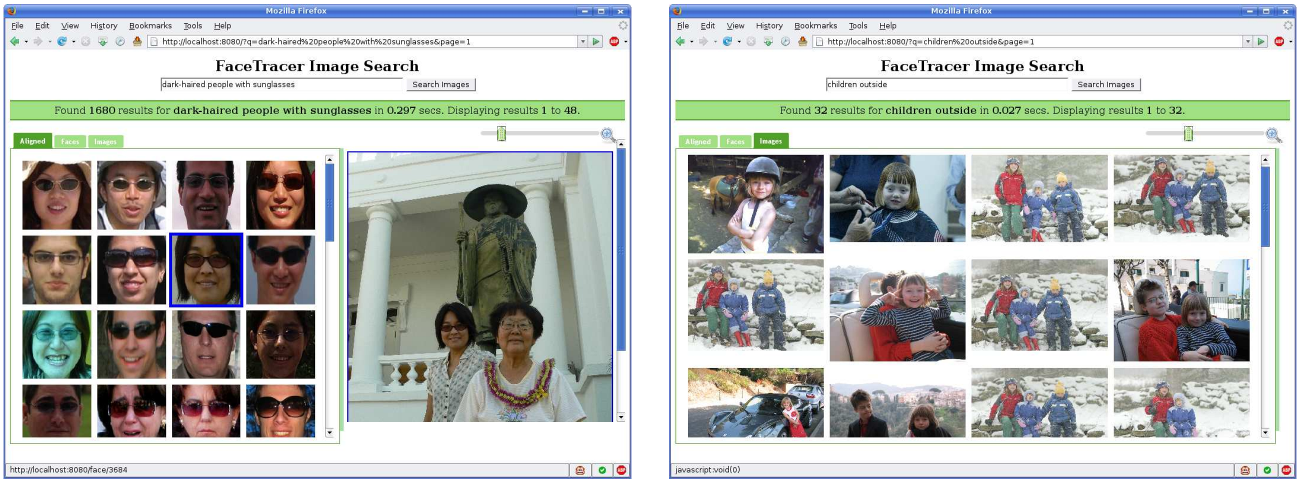
The evaluation benchmark for PubFig is similar to the LFW benchmark, albeit larger. Face verification is performed on 20,000 pairs of images of 140 people, divided into 10 cross-validation folds with mutually disjoint sets of 14 people each. These people are separate from the 60 people in the development set of PubFig, which were used for training the simile classifiers. The larger size and more varied image sources used to gather the PubFig dataset make it a potentially more challenging benchmark than LFW, a hypothesis supported by our lower performance on this test, displayed in Fig. 13 – our attribute classifiers had an accuracy of 77.78%.

All the data and evaluation benchmarks from PubFig are publicly available for non-commercial use at http://faceserv.cs.columbia.edu/databases/pubfig/

6 Face Search

Image search engines currently are completely dependent on textual metadata. This data can be in the form of filenames, manual annotations, or surrounding text. However, for the vast majority of images on the internet (and in peoples' private collections), this data is often ambiguous, incorrect, or simply not present. This presents a great opportunity to use attribute classifiers on images with faces, thereby making them searchable. To facilitate fast searches on a large collection of images, all images are labeled in an offline process using attribute classifiers. The resulting attribute labels are stored for fast online searches using the FaceTracer engine [27].

For a search engine, the design of the user interface is one of the most important aspects for enabling users to easily find what they are looking for. We use simple text-based queries, since these are both familiar and accessible to most internet users, and correspond well to describable visual attributes. Search queries are mapped onto attribute labels using a dictionary of



http://localhost:8080/?q=dark-haired%20people%20with%20sunglasses&page=1

FaceTracer Image Search

dark-haired people with sunglasses

Found **1680** results for **dark-haired people with sunglasses** in **0.297** secs. Displaying results **1** to **48**.

http://localhost:8080/face/3684

http://localhost:8080/?q=children%20outside&page=1

FaceTracer Image Search

children outside

Found **32** results for **children outside** in **0.027** secs. Displaying results **1** to **32**.

javascript:void(0)

(a) Results for “dark-haired people with sunglasses”

(b) Personalized results for “children outside”

Fig. 14. Results of queries (a) “dark-haired people with sunglasses” and (b) “children outside,” using our attribute-based face search engine. In (a), search results are shown in the left panel, while the right panel shows a preview of the original image for the selected face. Clicking the image takes the user to the image’s original webpage. (b) shows search results on a personalized dataset, displaying the results as thumbnails of the original images. Note that these results were correctly classified as being “outside” using only the cropped face images, showing that face images often contain enough information to describe properties of the image not directly related to faces.

terms. Users can see the list of attributes supported by the system on the search page, allowing them to construct searches without having to guess what kinds of queries are allowed. This approach is simple, flexible, and yields excellent results in practice. Furthermore, it is easy to add new phrases and attributes to the dictionary, or maintain separate dictionaries for searches in different languages.

Search results are ranked by confidence, so that the most relevant images are shown first. We use the computed distance to the classifier decision boundary as a measure of the confidence. For searches with multiple query terms, we combine the confidences of different attribute labels such that the final ranking shows images in decreasing order of relevance to all search terms. To prevent high confidences for one attribute from dominating the search results, we first convert the confidences into probabilities by fitting a held-out set of positive and negative examples to gaussian distributions, and then use the product of the probabilities as the sort criteria. This ensures that the images with high confidences for *a*ll attributes are shown first.

Example queries on our search engine are shown in Figs. 14a and 14b. The returned results are all highly relevant. Fig. 14b additionally demonstrates two other interesting things. First, it was run on a personalized dataset of images, showing that this method can be applied to specialized image collections as well as general ones. Second, it shows that we can learn useful things about an image using just the appearance of the faces within it – in this case determining whether the image was taken indoors or outdoors.

This attribute-based search engine can be used in many other applications, replacing or augmenting existing tools. In law enforcement, eyewitnesses to crimes could use this system to quickly narrow a list of possible suspects and then identify the actual criminal from the reduced list, saving time and increasing the chances of finding the right person. On the internet, our face search engine is a perfect match for social networking websites such as Facebook and Myspace, which contain large numbers of images with people. Additionally, the community aspect of these websites would allow for collaborative creation of new attributes. Finally, users can utilize our system to more easily organize and manage their own personal photo collections. For example, searches for blurry or other poor-quality images can be used to find and remove all such images from the collection.

7 Conclusions and Future Directions

In this work, we have shown how to automatically train classifiers for describable aspects of visual appearance – attributes and similes. These classifiers are learned using large collections of labeled images obtained from the internet. We demonstrated the use of these describable attributes for performing face verification and image search. We showed performance comparable to or better than the state-of-the-art in all aspects of the work: attribute classification, face verification, and search (qualitatively). We have also made available two large and complementary datasets for use by the community to make further progress along these lines.

These seem to be promising first steps in a new direction, and there are many avenues to explore. The experiments with human attribute labeling in Sec. 5.4

suggest that adding more attributes and improving the attribute training process could yield great benefits for face verification. Another open question is how attributes can be applied to domains other than faces. It seems that for reliable and accurate attribute training, analogues to the detection and alignment process must be found.

7.1 Dynamic Selection of Attributes to Label

The set of attributes used in this work were chosen in an ad-hoc way; how to select them dynamically in a more principled manner is an interesting topic to consider. In particular, a system with a user-in-the-loop could be used to suggest new attributes. Thanks to Amazon Mechanical Turk, such a system would be easy to setup and could operate almost fully automatically.

The idea is to first evaluate a current set of attribute classifiers on a verification dataset and then look at the mistakes made by the algorithm. Presumably, these mistakes would occur on face pairs which could not be sufficiently distinguished using the current set of attributes. Borrowing terminology from color theory, we term these face pairs “metamers.” The metamers could be shown to users on MTurk, asking them to suggest new attributes which could disambiguate such pairs. By doing this over a large enough number of images and users, one could grow an existing set of attributes in a maximally-efficient way. Measures based on mutual information and information gain could be used in association with this metamer disambiguation strategy to ensure that the best attributes were picked.

Acknowledgments

The authors would like to thank Pietro Perona for suggesting the human attribute labels experiment described in Sec. 5.4. This work was supported in part by NSF award IIS-03-25867 and ONR award N00014-08-1-0638.

References

[1] S. Baluja and H. Rowley, “Boosting sex identification performance,” *IJCV*, 2007.

[2] P. Belhumeur, J. Hespanha, and D. Kriegman, “Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection,” *ECCV*, pp. 45–58, 1996.

[3] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y.-W. Teh, E. Learned-Miller, and D. Forsyth., “Names and faces in the news,” *CVPR*, 2004.

[4] V. Blanz, S. Romdhani, and T. Vetter, “Face identification across different poses and illuminations with a 3d morphable model,” *FGR*, 2002.

[5] C. D. Castillo and D. W. Jacobs, “Using stereo matching for 2-d face recognition across pose,” *CVPR*, 2007.

[6] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[7] H. Chen, P. Belhumeur, and D. Jacobs, “In search of illumination invariants,” *CVPR*, 2000.

[8] T. Cootes, K. Walker, and C. Taylor, “View-based active appearance models,” *FGR*, 2000.

[9] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, 1995.

[10] G. W. Cottrell and J. Metcalfe, “Empath: face, emotion, and gender recognition using holons,” in *NIPS*, 1990, pp. 564–571.

[11] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.

[12] R. Datta, J. Li, and J. Z. Wang, “Content-based image retrieval: Approaches and trends of the new age,” *Multimedia Information Retrieval*, pp. 253–262, 2005.

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A Large-Scale Hierarchical Image Database,” in *CVPR*, 2009.

[14] M. Everingham, J. Sivic, and A. Zisserman, “Hello! my name is... Buffy – automatic naming of characters in TV video,” *BMVC*, 2006.

[15] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results,” http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html.

[16] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, “Describing objects by their attributes,” in *CVPR*, 2009.

[17] A. Ferencz, E. Learned-Miller, and J. Malik, “Learning to locate informative features for visual identification,” *IJCV Special Issue on Learning and Vision*, 2007.

[18] V. Ferrari and A. Zisserman, “Learning visual attributes,” in *Advances in Neural Information Processing Systems*, Dec. 2007.

[19] Y. Freund and R. Shapire, “Experiments with a new boosting algorithm,” *ICML*, 1996.

[20] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, “From few to many: Illumination cone models for face recognition under variable lighting and pose,” *PAMI*, vol. 23, no. 6, pp. 643–660, 2001.

[21] B. A. Golomb, D. T. Lawrence, and T. J. Sejnowski, “SexNet: A neural network identifies sex from human faces,” in *NIPS*, 1990, pp. 572–577.

[22] R. Gross, J. Shi, and J. Cohn, “Quo vadis face recognition?” in *Workshop on Empirical Evaluation Methods in Computer Vision*, December 2001.

[23] G. Huang, V. Jain, and E. Learned-Miller, “Unsupervised joint alignment of complex images,” *ICCV*, 2007.

[24] G. Huang, M. Jones, and E. Learned-Miller, “LFW results using a combined Nowak plus MERL recognizer,” in *Real-Life Images workshop at ECCV*, 2008.

[25] 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.

[26] M. Kirby and L. Sirovich, “Application of the karhunen-loeve procedure for the characterization of human faces,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, no. 1, pp. 103 –108, jan 1990.

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

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

[29] C. Lampert, H. Nickisch, and S. Harmeling, “Learning to detect unseen object classes by between-class attribute transfer,” in *CVPR*, 2009.

[30] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, “Content-based multimedia information retrieval: State of the art and challenges,” *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 2, no. 1, pp. 1–19, 2006.

[31] H. Ling, S. Soatto, N. Ramanathan, and D. Jacobs, “A study of face recognition as people age,” *ICCV*, 2007.

[32] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *IJCV*, 2003.

[33] B. Moghaddam and M.-H. Yang, “Learning gender with support faces,” *TPAMI*, vol. 24, no. 5, pp. 707–711, 2002.

[34] D. Nister and H. Stewenius, “Scalable recognition with a vocabulary tree,” in *CVPR*, 2006, pp. 2161–2168.

[35] E. Nowak and F. Jurie, “Learning visual similarity measures for comparing never seen objects,” *CVPR*, 2007.

[36] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," in *In Proc. ECCV*. Springer, 2006, pp. 490–503.

[37] Omron, "OKAO vision," http://www.omron.com/r_d/coretech/vision/okao.html, 2009.

[38] A. O'Toole, P. Phillips, F. Jiang, J. Ayyad, N. Penard, and H. Abdi, "Face recognition algorithms surpass humans matching faces over changes in illumination," *PAMI*, vol. 29, no. 9, pp. 1642–1646, Sept. 2007.

[39] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," *CVPR*, pp. 84–91, 1994.

[40] A. Pentland, R. Picard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases," *IJCV*, pp. 233–254, 1996.

[41] P. Phillips, H. Moon, S. Rizvi, and P. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *PAMI*, vol. 22, no. 10, pp. 1090–1104, 2000.

[42] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge," *CVPR*, pp. 947–954, 2005.

[43] P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, and W. Worek, "Preliminary face recognition grand challenge results," *FGR*, pp. 15–24, 2006.

[44] 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 *Computer Vision and Pattern Recognition*, 2009.

[45] B. Russell, A. Torralba, and K. Murphy, "LabelMe: a database and web-based tool for image annotation," *IJCV*, vol. 77, no. 1-3, pp. 157–173, 2008.

[46] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," *Workshop on Applications of Computer Vision*, 1994.

[47] G. Shakhnarovich, P. Viola, and B. Moghaddam, "A unified learning framework for real time face detection and classification," *FGR*, 2002.

[48] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," *ICAFGR*, pp. 46–51, 2002.

[49] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proceedings of the International Conference on Computer Vision*, vol. 2, Oct. 2003, pp. 1470–1477.

[50] Y. Taigman, L. Wolf, and T. Hassner, "Multiple one-shots for utilizing class label information," in *The British Machine Vision Conference (BMVC)*, 2009.

[51] M. Turk and A. Pentland, "Face recognition using eigenfaces," *CVPR*, 1991.

[52] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *CVPR*, 2001.

[53] L. Wiskott, J.-M. Fellous, N. Krger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, pp. 775–779, 1997.

[54] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Real-Life Images workshop at ECCV*, 2008.

[55] ——, "Similarity scores based on background samples," in *Asian Conference on Computer Vision*, 2009.

[56] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Comput. Surv.*, vol. 35, no. 4, pp. 399–458, 2003.

Neeraj Kumar received BSc degrees in Computer Science and Aeronautical Engineering from the Georgia Institute of Technology in 2005 (both with highest honors). He was awarded a three-year National Defense Science and Engineering Graduate Fellowship (NDSEG) by the American Society for Engineering Education in 2005. He is currently a Ph.D. candidate in Computer Science at Columbia University, where he is co-supervised by Professors Belhumeur and Nayar. His main research interests are at the intersection of computer vision and machine learning – developing techniques for efficient search and recognition in large image databases.

Alexander C. Berg received the PhD degree in Computer Science from U.C. Berkeley in 2005. He is currently a research scientist at Columbia University, and will join the Computer Science faculty at Stony Brook University in Fall 2010. Dr. Berg worked at Yahoo! Research on computer vision for search and advertising from 2007-2008. His research addresses challenges in visual recognition at all levels, including activity recognition, object recognition, feature descriptors, video-based motion synthesis, and face recognition, and a theme of work on machine learning for computer vision.

Peter N. Belhumeur received the ScB degree in Information Sciences from Brown University in 1985. He received the PhD degree in Engineering Sciences from Harvard University under the direction of David Mumford in 1993. He was a postdoctoral fellow at the University of Cambridge's Isaac Newton Institute for Mathematical Sciences in 1994. He was made Assistant, Associate, and Professor of Electrical Engineering at Yale University in 1994, 1998, and 2001, respectively. He joined Columbia University in 2002, where he is currently a Professor in the Department of Computer Science and the director of the Laboratory for the Study of Visual Appearance. His main research focus is on illumination, reflectance, and shape, and their relation to visual appearance. Within these areas, he concentrates on two subproblems: the representation and recognition of objects under variable illumination and the estimation of the geometry of objects from low-level cues like image brightness, binocular stereopsis, and motion. Applications include face and object recognition, image-based rendering, computer graphics, content-based image and video compression, and human computer interfaces. He is a recipient of the Presidential Early Career Award for Scientists and Engineers (PECASE), the US National Science Foundation Career Award, and the Yale University Junior Faculty Fellowship. His papers have received the Siemens Best Paper Award at CVPR 1996, the Olympus Prize at ECCV 1998 and a Best Paper Honorable Mention Award at CVPR 2000.

Shree K. Nayar received the PhD degree in Electrical and Computer Engineering from the Robotics Institute at Carnegie Mellon University in 1990. He is the T.C. Chang Professor of Computer Science at Columbia University and, since 2009, the chairman of the department. He heads the Columbia Automated Vision Environment (CAVE), which is dedicated to the development of advanced computer vision systems. His research is focused on three areas: the creation of cameras that produce new forms of visual information, the modeling of the interaction of light with materials, and the design of algorithms that recognize objects from images. His work is motivated by applications in the fields of computer graphics, human-machine interfaces, and robotics. Dr. Nayar has authored and coauthored papers that have received the Best Paper Award at the 2004 CVPR Conference held in Washington, DC, the Best Paper Honorable Mention Award at the 2000 IEEE CVPR Conference held in Hilton Head, the David Marr Prize at the 1995 ICCV held in Boston, the Siemens Outstanding Paper Award at the 1994 IEEE CVPR Conference held in Seattle, the 1994 Annual Pattern Recognition Award from the Pattern Recognition Society, the Best Industry Related Paper Award at the 1994 ICPR held in Jerusalem, and the David Marr Prize at the 1990 ICCV held in Osaka. He was the recipient of the Columbia Great Teacher Award in 2006, the Excellence in Engineering Teaching Award from the Keck Foundation in 1995, the NTT Distinguished Scientific Achievement Award from NTT Corporation, Japan, in 1994, the National Young Investigator Award from the US National Science Foundation in 1993, and the David and Lucile Packard Fellowship for Science and Engineering in 1992. In February 2008, he was elected to the National Academy of Engineering.