Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments

Gary B. Huang[1], Marwan Mattar[1], Tamara Berg[2], and Erik Learned-Miller[1]

[1] University of Massachusetts Amherst, Amherst, MA
{gbhuang, mmattar, elm}@cs.umass.edu
[2] Stony Brook University, Stony Brook, NY
tlberg@cs.sunysb.edu

Abstract. Most face databases have been created under controlled conditions to facilitate the study of specific parameters on the face recognition problem. These parameters include such variables as position, pose, lighting, background, camera quality, and gender. While there are many applications for face recognition technology in which one can control the parameters of image acquisition, there are also many applications in which the practitioner has little or no control over such parameters. This database, Labeled Faces in the Wild, is provided as an aid in studying the latter, unconstrained, recognition problem. The database contains labeled face photographs spanning the range of conditions typically encountered in everyday life. The database exhibits "natural" variability in factors such as pose, lighting, race, accessories, occlusions, and background. In addition to describing the details of the database, we provide specific experimental paradigms for which the database is suitable. This is done in an effort to make research performed with the database as consistent and comparable as possible. We provide baseline results, including results of a state of the art face recognition system combined with a face alignment system. To facilitate experimentation on the database, we provide several parallel databases, including an aligned version.

1 Introduction

This paper describes a database of human face images designed as an aid in studying the problem of *unconstrained face recognition*.[1] The database can be viewed and downloaded at http://vis-www.cs.umass.edu/lfw/.

[1] We note that for more general classes of objects such as cars or dogs, the term "recognition" often refers to the problem of recognizing a *member of the larger class*, rather than a specific instance. When one "recognizes" a cat (in the context of computer vision research), it is meant that one has identified a particular object as a cat, rather than a particular cat. In the context of general objects, the term *identification* is used to refer to recognizing a specific instance of a class (such as Bob's Toyota), as in [1],[2],[3]. However, in the literature on human faces, the term *recognition* is typically used to refer to the identification of a particular individual, not just a human being, and we adopt this latter terminology here.

Face recognition is the problem of identifying a specific individual, rather than merely detecting the presence of a human face, which is often called *face detection*. The general term "face recognition" can refer to a number of different problems including, but not limited to, the following.

Face Verification: Given a picture of a face, decide which person from among a set of people the picture represents, if any.

Pair Matching: Given two pictures, each of which contains a face, decide whether the two people pictured represent the same individual.

Our database, which we called Labeled Faces in the Wild (LFW), can be used to study these problems in unconstrained environments, as well as other face processing tasks, such as face alignment and face segmentation.

The primary contribution of LFW is providing a large set of relatively unconstrained face images. By unconstrained, we mean faces that show a large range of the variation seen in everyday life. This includes variation in pose, lighting, expression, background, race, ethnicity, age, gender, clothing, hairstyles, camera quality, color saturation, and other parameters. The reason we are interested in natural variation is that for many tasks, face recognition must operate in real-world situations where we have little to no control over the composition, or the images are pre-existing. For example, there is a wealth of unconstrained face images on the Internet, and developing recognition algorithms capable of handling such data would be extremely beneficial for information retrieval and data mining. Since LFW closely approximates the distribution of such images, algorithms trained on LFW could be directly applied to web IR applications. In contrast to LFW, existing face databases contain more limited and carefully controlled variation, as we describe in Section 2.

Figure 1 shows images from LFW representative of the diversity in the database. Tables 1 gives statistics of LFW such as number of images and people.

LFW is a valuable tool for studying face verification and pair matching in unconstrained environments, as we discuss in Section 3. To facilitate fair comparison of algorithms, we give specific protocols for developing and assessing algorithms using LFW (Section 4). By construction, algorithm performance on LFW is generalizable to performance in an end-to-end recognition system, as we describe in Section 5. We allow for easy experimentation with LFW by making publicly available parallel versions of the database containing aligned images and superpixel computation (Section 6.1). We give baseline results for LFW using both standard and state of the art face recognition methods (Section 7).

2 Related Databases

There are a number of face databases available to researchers in face recognition. These databases range in size, scope and purpose. The photographs in many of these databases were acquired by small teams of researchers specifically for the purpose of studying face recognition. Acquisition of a face database over a short time and particular location has advantages for certain areas of research, giving the experimenter direct control over the parameters of variability in the database.

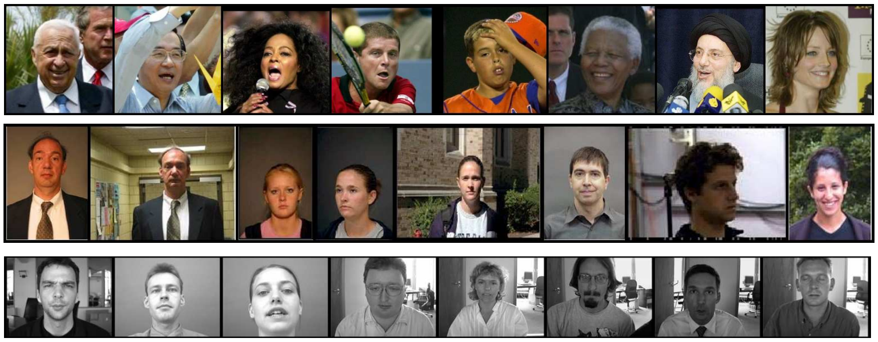

Fig. 1. Sample images from LFW (first row), FRGC (second row), and BioID (third row), representative of variation within each database (**best viewed in color**)

Table 1. Face Database Statistics

(a) Comparison of LFW, FRGC, and BioID

Database	# of people	Total images
LFW	5749	13233
FRGC	>466	>50000
BioID	23	1521
FERET	1199	14126

(b) Distribution of LFW

# of images /person	# of people (% of people)	# of images (% of images)
1	4069 (70.8)	4096 (30.7)
2-5	1369 (23.8)	3739 (28.3)
6-10	168 (2.92)	1251 (9.45)
11-20	86 (1.50)	1251 (9.45)
21-30	25 (0.43)	613 (4.63)
31-80	27 (0.47)	1170 (8.84)
> 81	5 (0.09)	1140 (8.61)
Total	5749	13233

On the other hand, in order to study more general, unconstrained face recognition problems, in which faces are drawn from a very broad distribution, one should train and test face recognition algorithms on highly diverse sets of faces. While it is possible to manipulate a large number of variables in the laboratory in an attempt to make such a database, there are two drawbacks to this approach. The first is that it is extremely labor intensive. The second is that it is difficult to gauge exactly which distributions of various parameters one should use to make the most useful database. What percentage of subjects should wear sunglasses, or have beards, or be smiling? How many backgrounds should contain cars, boats, grass, deserts, or basketball courts?

One possible solution to this problem is simply to measure a “natural” distribution of faces. Of course, no single canonical distribution of faces can capture a natural distribution that is valid across all possible application domains. Our database uses a set of images that was originally gathered from news articles on the web. This set clearly has its own biases. For example, there are not many images which occur under very poor lighting conditions. Also, because we use the Viola-Jones detector as a filter for the database, there are a limited number of side views of faces, and few views from above or below. However, the range and diversity of pictures present is very large. We believe such a database will be an important tool in studying unconstrained face recognition.

Existing face databases generally differ from LFW in one of two key aspects. Labeled databases for recognition, such as the Face Recognition Grand Challenge [4], BioID [5], FERET [6], and CMU PIE [7], are typically taken under very controlled conditions, with less people and diversity than LFW. For instance, images in LFW often contain complex phenomena such as headgear, additional people and faces in the background, and self-occlusion. Moreover, variations in parameters such as pose, lighting, and expression are carefully controlled in other databases, as compared with the uncontrolled variation in LFW that approximates the conditions in every day life. On the other hand, databases such as Caltech 10000 Web Faces [8] present highly diverse image sets similar to LFW, but are designed for face detection and do not contain person labels, making them unsuitable for recognition.

We now discuss the origin for LFW and comparisons with two of the more similar existing face recognition databases.[2]

Faces in the Wild [10],[11]. In this work, it was shown that a large, partially labeled database of face images could be built using imperfect data from the web. The database was built by jointly analyzing pictures and their associated captions to cluster images by identity. The resulting data set, which achieved a labelling accuracy of 77% [11], was informally referred to as "Faces in the Wild".

However, the database was not intended to act as training and test data for new experiments, and contained a high percentage of label errors and duplicated images. As a result, various researchers derived ad hoc subsets of the database for new research projects [2],[12],[13],[14]. The need for a clean version of the data set warranted doing the job thoroughly and publishing a new database.

The Face Recognition Grand Challenge Databases (FGRC) [4]. The FGRC was designed to study the effect of new, richer data types on face recognition, and thus includes high resolution data, three-dimensional scans, and image sequences. In contrast, LFW consists of faces extracted from previously existing images and hence can be used to study recognition from images that were not taken for the special purpose of face recognition by machine.

Another important difference between the data sets associated with the FRGC and our data set is the general variety of images. For example, while there are large numbers of images with uncontrolled lighting in the FRGC data sets, these images contain a great deal less natural variation than the LFW images. For example, the FRGC outdoor uncontrolled lighting images contain two images of each subject, one smiling and one with a neutral expression. The LFW images, in contrast contain arbitrary expressions. Variation in clothing, pose, background, and other variables is much greater in LFW than in the FRGC

[2] See [9] for more detailed comparisons and a more complete list of existing face databases.

databases. As mentioned earlier, the difference is one of *controlled variation* (FRGC) versus *natural* or *random* variation (LFW).

The BioID Face Database [5]. Similar to LFW, BioID strives to capture realistic settings with variability in pose, lighting, and expression. Unlike LFW, however, the distribution of images is more limited, focusing on a small number of home and office environments. Images for a given individual are generally different views of the same scene, whereas images in LFW for a given individual tend to be from a variety of venues. In addition, LFW has much more variability with respect to race, as the large majority of people in BioID are Caucasians. Finally, BioID is targeted at the face detection problem, and no person labels are given, so images would need to be manually labeled to be used for recognition.

While BioID is an interesting database of face images which may be useful for a number of purposes such as face detection in indoor enviroments, LFW will be useful for solving more general and difficult face *recognition* problems with large populations in highly variable environments.

In summary, there are a great number of face databases available, and while each has a role in the problems of face recognition or face detection, LFW fills an important gap for the problem of unconstrained face recognition.

3 Intended Uses

As mentioned in the introduction, this database is aimed at studying face recognition in realistic, unconstrained environments. Specifically, we focus on the two formulations of face verification and pair matching.

3.1 Face Verification

In the face verification paradigm (*e.g.* [15], [16]), there is a pre-specified gallery consisting of face images of a set of people, where the identity of each face image is known. The problem is to take a new query image, and decide which person in the gallery the new image represents. For instance, the gallery may consist of 10 images each of 10 different people, and the task would be to decide which of the 10 people a new input image represents.

Generally, face verification has been tested in situations where both the gallery images and query images are taken under controlled environments. For instance, even in Experiment 4 of the FRGC [4], which was designed to test the case in which the query images are taken in a more uncontrolled environment, the gallery images are still controlled.

This assumption is reasonable for certain tasks, such as recognition for security access, where gallery images can be taken ahead of time in a fixed environment, and query images can be taken in the same environment. On the other hand, for a large range of tasks, this assumption does not hold. For instance, as an information retrieval task, a user may wish to have photos automatically tagged with the names of the people, using a gallery of previously manually annotated photographs, which would not be taken in a controlled environment.

For studying unconstrained face verification, LFW contains 158 people with at least 10 images in the database, and in the next section we describe a specific protocol for designing and testing verification using this subset of the database.

3.2 Pair Matching

An alternative formulation of face recognition is the pair matching paradigm: given a pair of face images, decide whether the images are of the same person. Within the pair matching paradigm, there are a number of subtly, but importantly different recognition problems. Some of these differences concern the specific organization of training and testing subsets of the database. **A critical aspect of our database is that for any given training-testing split, the people in each subset are mutually exclusive.** In other words, for any pair of images in the training set, neither of the people pictured in those images is in any of the test set pairs. Similarly, no test image appears in a corresponding training set. We refer to this case, in which neither of the individuals pictured in the test pair have been seen during training, as the *unseen pair match* problem.

At training time, it is essentially impossible to build a model for any person in the test set, making this problem substantially different from the face verification paradigm. In particular, for LFW, since the people in test images have never been seen before, there is no opportunity to build models for such individuals, except to do this at test time from a single image. Instead, this paradigm is meant to focus on the generic problem of differentiating *any two individuals* that have never been seen before. Thus, a different type of learning is suggested–learning to discriminate among any pair of faces, rather than learning to find exemplars of a gallery of people as in face verification. Recently, there have been several important developments in this area of face recognition research [1],[14],[2].

A closely related problem to unseen pair matching is learning from one example [17], although there are subtle differences between the two.[3]

3.3 Pair Matching versus Face Verification

As mentioned earlier, we believe that unseen pair matching is one of the most general and fundamental face recognition problems. At a basic level, human beings are capable of recognizing faces after only seeing one example image, and thus are fundamentally different from algorithms that are only capable of performing matching against a fixed gallery of exemplars. Moreover, as we attempt to scale recognition systems to be able to deal with orders of magnitude more people, algorithms designed to learn general variability will be less computationally and resource intensive than methods that attempt to learn a specific model for each person, and likely perform better as well.

From a practical standpoint, pair matching algorithms require less supervision, only requiring examples of matching and mismatching pairs, rather than exemplars of each person to be identified. For instance, this would significantly

[3] See [9], Section IIIA for more detail.

simplify the previously mentioned image annotation problem. A pair matching algorithm could be trained independently on separate existing data, then used to label photographs in a collection with the names of the people pictured by clustering face images that were likely to be the same person. In comparison, a face verification algorithm would require manually labeled examples and would only be able to recognize from among the people appearing in the labeled examples.

For these reasons, we believe the unseen pair matching problem is an important area of face recognition and that having the LFW database as a benchmark for developing and comparing algorithms will help push new developments in this area. In addition to containing a larger variety of images matching real-life complexity than existing databases, LFW also contains a larger number of people, an important aspect for pair matching, allowing algorithms to discriminate between general faces rather than a specific small number of faces within a gallery.

4 Protocols

Proper use of training, validation, and testing sets is crucial for the accurate comparison of face recognition algorithms. For instance, performance will be improperly biased upward if the parameters of the algorithm are inadvertently tuned to the test set. We provide clear guidelines for the use of this data to minimize "fitting to the test data". Also, the size and difficulty of the data set may mitigate the degree to which unintended overfitting problems may occur.

For each recognition paradigm (verification and pair matching), we organize our data into two "Views", or groups of indices. View 1 is for algorithm development and general experimentation, prior to formal evaluation. This might also be called a model selection or validation view. View 2, for performance reporting, should be used only for the final evaluation of a method. The goal of this methodology is to use the final test sets as seldom as possible before reporting.

View 1: Model selection and algorithm development. The main purpose of this view of the data is so that researchers can freely experiment with algorithms and parameter settings without worrying about overusing test data. For example, if one is using support vector machines and trying to decide upon which kernel to use, it would be appropriate to test various kernels on View 1 of the database. Training and testing algorithms from this view may be repeated as often as desired without significantly biasing final results.

View 2: Performance reporting. The second view of the data should be used sparingly, and only for performance reporting. Ideally, it should only be used once, as choosing the best performer from multiple algorithms, or multiple parameter settings, will bias results toward artificially high accuracy. Once a model or algorithm has been selected (using View 1 if desired), the performance of that algorithm can be measured using View 2. For both recognition paradigms, View 2 consists of 10 splits of training and test sets, and the experimenter should report aggregate performance of a classifier on these 10 separate experiments.

It is critical for performance reporting that the final parameters of the classifier under each experiment be set using either the data in View 1 or **only the training data for that experiment.** An algorithm may not, during performance reporting, set its parameters to maximize the combined accuracy across all 10 training sets. The training and testing sets overlap across experiments, thus optimizing a classifier simultaneously using all training sets is essentially fitting to the test data, since the training set for one experiment is the testing data for another. In other words, each of the 10 experiments (both the training and testing phases) should be run completely independently of the others, resulting in 10 separate classifiers (one for each test set).

While there are many methods for reporting the final performance of a classifier, including ROC curves and Precision-Recall curves, we ask that each experimenter, at a minimum, report the **estimated mean accuracy** and the **standard error of the mean** for View 2 of the database. The estimated mean accuracy is $\hat{\mu} = \sum_{i=1}^{10} p_i/10$, where p_i is the percentage of correct classifications on subset i of View 2. It is important to note that accuracy should be computed with parameters and thresholds chosen independently of the test data, ruling out, for instance, simply choosing the point on a Precision-Recall curve giving the highest accuracy. The standard error of the mean is $S_E = \hat{\sigma}/\sqrt{10}$, where $\hat{\sigma}$ is the estimate of the standard deviation, $\hat{\sigma} = \sqrt{\sum_{i=1}^{10}(p_i - \hat{\mu})^2/9}$.

The *training sets* in View 2 overlap, therefore the standard error may be biased downward somewhat relative to what would be obtained with fully independent training sets and test sets. However, because the test sets of View 2 are independent, we believe this quantity will be valuable in assessing the significance of the difference among algorithms.[4]

4.1 Face Verification

For face verification, View 1 of LFW consists of 892 images from 38 randomly selected people from the 158 people with at least 10 images in the database. These images, along with images of people with less than 10 images, may be used in any manner for model selection or parameter estimation. View 2 consists of 3432 images from the remaining 120 people. Each of the 10 runs consists of a stratified split, such that for each person, ninety percent of that person's images appears as training and the other ten percent as testing.

In addition to reporting the estimated mean accuracy (the *micro-average* of the individual results), experimenters should also report the *macro-average*, which is the mean of the accuracies for each person $\hat{\mu}_{\mathrm{macro}} = \sum_{i=1}^{10}\sum_{j=1}^{120} p_{ij}/1200$, where p_{ij} is the percentage of correct classifications for person j on run i, and standard error (where $\hat{\sigma}$ is the standard deviation of the p_{ij}'s). This average accounts for the differing number of test instances for each person.

[4] We remind the reader that for two algorithms whose standard errors overlap, one may conclude that they their difference is not statistically significant at the 0.05 level. However, one *may not conclude,* in general, that algorithms whose standard errors do not overlap are statistically different at the 0.05 level.

4.2 Pair Matching

For pair matching, View 1 of LFW consists of two subsets of the database, one for training, containing 2200 pairs, and one for testing, containing 1000 pairs. The people appearing in the training and testing sets are mutually exclusive. View 2 consists of 6000 pairs, divided into ten subsets, and performance is computed using 10-fold cross validation using those subsets.

It should be noted that some images in View 1 may appear in View 2 as well, as the two views were selected randomly and independently from the entire database. This multiple-view approach has been used, rather than a traditional training-validation-testing split of the database, in order to maximize the amount of data available for training and testing. Ideally, one would have enough images in a database so that training, validation, and testing sets could be non-overlapping. However, in order to maximize the size of our training and testing sets, we have allowed reuse of the data between View 1 of the database and View 2 of the database. The bias introduced into the results by this approach is very small and outweighed by the benefit of the resulting larger training and test set sizes.

Forming training pairs. Whenever one works with matched and mismatched data pairs, the issue of creating auxiliary training examples by using the transitivity of equality arises. For example, in a training set, if one matched pair consists of the 10th and 12th images of George_W_Bush, and another pair consists of the 42nd and 50th images of George_W_Bush, then it might seem reasonable to add other image pairs, such as (10, 42), (10, 50), (12, 42) and (12, 50), to the training data using an automatic procedure. One could argue that such pairs are *implicitly present* in the original training data, given that the images have been labeled with the name George_W_Bush. Auxiliary examples could be added to the mismatched pairs using a similar method.

Rather than disallowing such augmentation or penalizing researchers who do not wish to add many thousands of extra pairs of images to their training sets, we give two separate methods for using training data. These methods and details on how the training sets can be generated can be found in Section IV of [9].

5 The Detection-Alignment-Recognition Pipeline

Many real world applications wish to automatically detect, align, *and* recognize faces in a larger still image, or in a video of a larger scene. Thus, face recognition is often naturally described as part of a Detection-Alignment-Recognition (DAR) pipeline, as illustrated in Figure 2. To complete this pipeline, we need automatic algorithms for each stage of the pipeline. In addition, each stage of the pipeline must either accept images from, or prepare images for, the next stage of the pipeline. To facilitate this process, we have purposefully designed our database to represent the output of the detection process.

In particular, every face image in our database is the output of the Viola-Jones face detection algorithm [18]. The motivation for this is as follows. If one

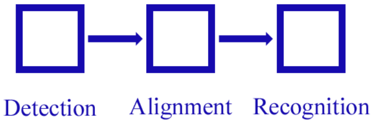

Fig. 2. The Detection-Alignment-Recognition (DAR) pipeline. The images of the LFW database represent the output of the Viola-Jones detector. By working with such a database, the developer of alignment and recognition algorithms know that their methods will fit easily into the DAR pipeline.

can develop a face alignment algorithm (and subsequent recognition algorithm) that works directly on LFW, then it is likely to also work well in an end-to-end system that uses a face detector as a first step. This alleviates the need for each researcher to worry about the process of detection, or about the possibility that a manually aligned database does not adequately represent the true variability seen in the world. In other words, it allows the experimenter to focus on the problems of alignment and recognition rather detection.

6 Construction and Composition Details

The process of building the database can be broken into the following steps[5]:

1. gathering raw images,
2. applying a face detector and manually eliminating false positives,
3. eliminating duplicate images,
4. hand labeling (naming) the detected people,
5. cropping and rescaling the detected faces, and
6. forming pairs of training and testing pairs for View 1 and View 2 of the database.

As a starting point, we used the raw images from the Faces in the Wild database [10]. The OpenCV[6] trained version of the Viola-Jones face detector [18] was run on each image. False positives, as well as duplicate images, which we defined as images judged to have a common original source photograph, where manually removed. The face region returned by the Viola-Jones detector generally contains only a subset of the whole head, so the region was automatically expanded by 2.2 in each dimension to capture the entire head. This expanded region was then cropped and rescaled to output a 250x250 JPEG 2.0 image.

6.1 Parallel Databases

To facilitate experimentation on LFW, we also present several parallel versions of our database. We created an aligned version of the database, and for both the original and the aligned versions, we computed superpixels for each image.

[5] See [9] Section VI for specific details of each step.

[6] `http://opencvlibrary.sourceforge.net/`

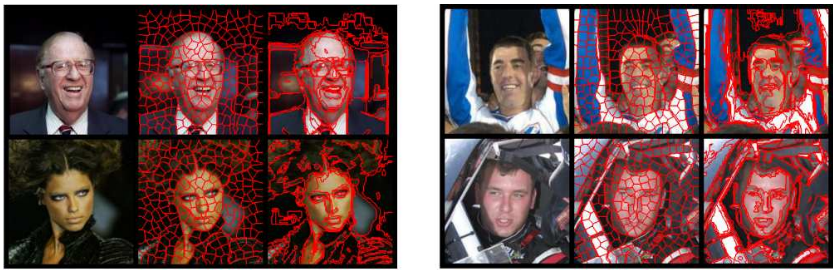

Fig. 3. Examples of superpixels. The left column is the original image, the middle column is the Mori segmentation (N_sp=100, N_sp2=200, N_ev=40), and the right column is the Felzenszwalb-Huttenlocher segmentation (sigma=0.5, K=100, min=20).

Alignment. To create an aligned version of our database, we used an implementation of the congealing and funneling method of Huang *et al.* [12].[7] We took one image each of 800 people selected at random to learn a sequence of distribution fields, which we then used to funnel every image in the database.

Superpixels. A superpixel representation of an image is a division of the image into a number of small contiguous regions where the pixel values in each region are homogeneous. It is thus a type of oversegmentation of an image. Superpixels have recently started replacing pixels as the basic building block for an image in several object recognition and segmentation models [19], [20], [21], [22].[8] This transition is partly due to the larger spatial support that superpixels provide, allowing more global features to be computed than on pixels alone.

Superpixel representations have already been successfully applied to face segmentation [22] and we believe they can also be useful for detection and recognition. Therefore, we provide superpixel representations for all the images in the database based on Mori's online implementation [19].[9] We also experimented with the Felzenszwalb and Huttenlocher [24][10] algorithm but found that Mori's method, while more computationally expensive, did a much better job at preserving the face-background boundary, a crucial property for superpixel-based segmentation. Figure 3 contains sample superpixel results of both methods on four diverse images from the database.

7 Results

To establish baseline results as well as validate the difficulty of LFW, we used the standard face recognition method of Eigenfaces [16].

[7] http://vis-www.cs.umass.edu/code/congealingcomplex/

[8] While the term superpixels has only recently been defined, the idea of using oversegmentations has existed in the vision community dating back to at least 1989 [23]

[9] http://www.cs.sfu.ca/~mori/research/superpixels/

[10] http://people.cs.uchicago.edu/~pff/segment/

Table 2. Accuracy on View 2

(a) Verification

method	database	$\hat{\mu} \pm S_E$ $\hat{\mu}_{\text{macro}} \pm S_E$
Eigenfaces	unaligned	0.1270 ± 0.0045 0.0702 ± 0.0055

(b) Pair Matching

method	database	$\hat{\mu} \pm S_E$
Eigenfaces	unaligned	0.6002 ± 0.0079
Nowak	unaligned	0.7245 ± 0.0040
Nowak	funneled	0.7333 ± 0.0060

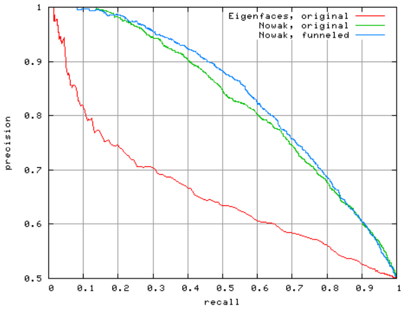

Fig. 4. Precision-Recall curves for pair matching

Eigenfaces for verification. We divided View 1 into a training and test set and ran Eigenfaces with different numbers of eigenvectors. Using enough eigenvectors to capture 80% of the variance in the training set gave the optimal performance on the test set. For each run of the 10 runs of View 2, we computed eigenvectors from the training set, using enough to capture 80% of the variance, and classified test instances using nearest-neighbor.

Eigenfaces for pair matching. We computed eigenvectors from the training set of View 1 and determined the threshold value for classifying pairs as matched or mismatched that gave the best performance on the test set of View 1. For each run of View 2, the training set was used to compute the eigenvectors, and pairs were classified using the threshold on Euclidian distance from View 1.

State of the art pair matching. To determine the current best performance on pair matching, we ran an implementation of the current state of the art recognition system of Nowak and Jurie [14].[11] The Nowak algorithm gives a similarity score to each pair, and View 1 was used to determine the threshold value for classifying pairs as matched or mismatched. For each of the 10 folds of View 2 of the database, we trained on 9 of the sets and computed similarity measures for the held out test set, and classified pairs using the threshold.

Alignment for pair matching. We also ran the Nowak algorithm on the parallel aligned database of LFW, again using View 1 to pick the threshold that optimized performance on the test set.

[11] `http://lear.inrialpes.fr/people/nowak/similarity/index.html`

The mean classification accuracy $\hat{\mu}$ and the standard error of the mean S_E are given in Table 2. In addition, the mean precision-recall curves for pair matching are given in Figure 4. Each point on the curve represents the average over the 10 folds of (recall, precision) for a fixed threshold.

Chance performance is 0.008 on face verification and 0.5 on pair matching. The low accuracy of Eigenfaces reflects the difficulty of the images in LFW and of unconstrained face recognition in general. For verification, half of the 120 classes had 0 accuracy over all 10 runs. While the Nowak method significantly outperforms Eigenfaces, it is still far below human-level performance and there is a large amount of room for improvement.

Comparing the accuracy between the Nowak recognizer on the unaligned and funneled images, the standard errors of the mean overlap. Therefore, the difference between the two is not statistically significant. Nonetheless, combining the Nowak recognition system out of the box with the funneling alignment provides a higher baseline to compare against. In addition, judging from the precision-recall curves, the advantage of using the aligned images may be more pronounced for a cost function emphasizing higher precision at the expense of a lower recall of approximately 0.5. As a general comment, while simply running an algorithm on the aligned database is likely to improve performance over the same algorithm on the original database, modifying the algorithm to take advantage of the tighter correspondence of faces in the aligned version can potentially do even better.

8 Conclusion

We have created a set of resources for researchers interested in unconstrained face recognition. Specifically, we have

1. Introduced a new labeled database, Labeled Faces in the Wild, that contains 13,233 images of 5749 unique individuals with highly variable image conditions. The natural variability and difficulty of this database allows models learned to be applied to new unseen images (taken from the web, for example). This database also fits neatly into the DAR pipeline.
2. Devised model selection and performance reporting splits for the verification and pair matching problems. The splits and suggested evaluation metrics were designed to facilitate fair comparisons of algorithms and avoid inadvertantly overfitting to the test data.
3. Provided baseline results using Eigenfaces, both as an example of how to set algorithm parameters and to validate the difficulty of this database for both recognition problems.
4. Provided results using the state of the art method [14] for pair matching.
5. Provided parallel versions of the database. The aligned version can be used to improve the performance and run time (by reducing the search space) and computed superpixels preserve the face-background boundary well and can be reliably used for detection, recognition, and segmentation.

We believe this database and accompanying resources will provide another stimulus to the vibrant research area of face recognition.

References

1. Ferencz, A., Learned-Miller, E., Malik, J.: Building a classification cascade for visual identification from one example. In: ICCV. (2005)
2. Jain, V., Ferencz, A., Learned-Miller, E.: Discriminative training of hyper-feature models for object identification. In: BMVC. (2006)
3. Ferencz, A., Learned-Miller, E., Malik, J.: Learning hyper-features for visual identification. In: NIPS. Volume 18. (2005)
4. Phillips, P.J., Flynn, P.J., Scruggs, T., Bowyer, K., Chang, J., Hoffman, K., Marques, J., Min, J., Worek, W.: Overview of the Face Recognition Grand Challenge. In: CVPR. (2005)
5. Jesorsky, O., Kirchberg, K., Frischolz, R.: Robust face detection using the Hausdorff distance. In Bigun, J., Smeraldi, F., eds.: Audio and Video Based Person Authentication. Springer (2001) 90–95
6. National Institute of Standards and Technology: The Color FERET Database. http://www.itl.nist.gov/iad/humanid/colorferet/home.html (2003)
7. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination, and expression database. PAMI **25** (2003) 1615–1618
8. Angelova, A., Abu-Mostafa, Y., Perona, P.: Pruning training sets for learning of object categories. In: CVPR. Volume 1. (2005) 495–501
9. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst (2007)
10. Berg, T.L., Berg, A.C., Maire, M., White, R., Teh, Y.W., Learned-Miller, E., Forsyth, D.A.: Names and faces in the news. CVPR (2004)
11. Berg, T.L., Berg, A.C., Edwards, J., Forsyth, D.A.: Who's in the picture. NIPS (2004)
12. Huang, G.B., Jain, V., Learned-Miller, E.: Unsupervised joint alignment of complex images. In: ICCV. (2007)
13. Ozkan, D., Duygulu, P.: A graph based approach for naming faces in news photos. In: CVPR. (2006)
14. Nowak, E., Jurie, F.: Learning visual similarity measures for comparing never seen objects. In: CVPR. (2007)
15. Phillips, P., Moon, H., Rizvi, S., Rauss, P.: The FERET evaluation methodology for face-recognition algorithms. PAMI **22** (2000) 1090–1104
16. Turk, M., Pentland, P.: Face recognition using eigenfaces. CVPR (1991)
17. Beymer, D., Poggio, T.: Face recognition from one example view. Technical Report AIM-1536, MIT Artificial Intelligence Laboratory (1995)
18. Viola, P., Jones, M.: Robust real-time face detection. IJCV (2004)
19. Mori, G.: Guiding model search using segmentation. In: ICCV. (2005)
20. Hoiem, D., Efros, A., Hebert, M.: Recovering surface layout from an image. IJCV **75** (2007)
21. Ren, X., Malik, J.: Learning a classification model for segmentation. In: ICCV. (2003)
22. Arora, H., Loeff, N., Forsyth, D., Ahuja, N.: Unsupervised segmentation of objects using efficient learning. In: CVPR. (2007)
23. Beveridge, J.R., Griffith, J., Kohler, R., Hanson, A., Riseman, E.: Segmenting images using localized histograms and region merging. IJCV **2** (1989)
24. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. IJCV **59** (2004)