Unsupervised Joint Alignment of Complex Images

Gary B. Huang Vidit Jain Erik Learned-Miller
University of Massachusetts Amherst
Amherst, MA
{gbhuang,vidit,elm}@cs.umass.edu

Abstract

Many recognition algorithms depend on careful positioning of an object into a canonical pose, so the position of features relative to a fixed coordinate system can be examined. Currently, this positioning is done either manually or by training a class-specialized learning algorithm with samples of the class that have been hand-labeled with parts or poses. In this paper, we describe a novel method to achieve this positioning using poorly aligned examples of a class with no additional labeling. Given a set of unaligned examplars of a class, such as faces, we automatically *build an alignment mechanism, without any additional labeling of parts or poses in the data set. Using this alignment mechanism, new members of the class, such as faces resulting from a face detector, can be precisely aligned for the recognition process. Our alignment method improves performance on a face recognition task, both over unaligned images and over images aligned with a face alignment algorithm specifically developed for and trained on hand-labeled face images. We also demonstrate its use on an entirely different class of objects (cars), again without providing any information about parts or pose to the learning algorithm.*

1. Introduction

The identification of certain objects classes, such as faces or cars, can be dramatically improved by first transforming a detected object into a canonical pose. Such registration reduces the variability that an identification system or classifier must contend with in the modeling process. Subsequent identification can condition on spatial position for a detailed analysis of the structure of the object in question. Thus, many recognition algorithms assume the prior rough alignment of objects to a canonical pose [1, 7, 15, 17]. In general, the better this alignment is, the better identification results will be. In fact, alignment itself has emerged as an important sub-problem in the face recognition literature [18], and a number of systems exist for the detailed alignment of specific categories of objects, such as faces [3, 4, 5, 6, 12, 19, 20].

We point out that it is frequently much easier to obtain images that are roughly aligned than those that are precisely aligned, indicating an important role for automatic alignment procedures. For example, images of people can be taken easily with a motion detector in an indoor environment, but will result in images that are not precisely aligned.

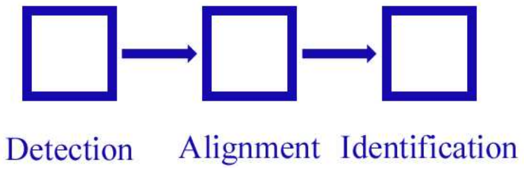


Figure 1. Recognition pipeline

Although there exist many individual components to do both detection and recognition, we believe the absence of a complete end-to-end system capable of performing recognition from an arbitrary scene is in large part due to the difficulty in alignment, the middle stage of the recognition pipeline (Figure 1). Often, the middle stage is ignored, with the assumption that the detector will perform a rough alignment, leading to suboptimal recognition performance.

A system that did attempt to address the middle stage would suffer from two signficant drawbacks of current alignment methods:

- They are typically designed or trained for a single class of objects, such as faces.
- They require the manual labelling either of specific features of an object (like the middle of the eye or the corners of the mouth),[1] or a description of the pose (such as orientation and position information).

As a result, these methods require significant additional effort when applied to a new class of objects. Either they must be redesigned from scratch, or a new data set must be collected, identifying specific parts or poses of the new data set before an alignment system can be built. In contrast,

[1] Some systems identify more than 80 landmarks per face for 200 to 600 faces [6, 19].

systems for the detection and recognition steps of the recognition pipeline only require simple, discrete labels, such as object versus non-object or pair match versus pair non-match, which are straight forward to obtain, making these systems significantly easier to set up than current systems for alignment, where even the form of the supervised input is very often class-dependent.

Some previous work has used detectors capable of returning some information about object rotation, in addition to position and scale, such as, for faces, [8, 16]. Using the detected rotation angle, along with the scale and position of the detected region, one could place each detected object into a canonical pose. However, so far, these efforts have only provided very rough alignment due to the lack of precision in estimating the pose parameters. For example, in [8], the rotation is only estimated to within 30 degrees, so that one of 12 rotation-specific detectors can be used. Moreover, even in the case of frontal faces, position and scale are only roughly estimated, and, in fact, for face images, we use this as a starting point and show that a more precise alignment can be obtained.

More concretely, in this work, we describe a system that, given a collection of images from a particular class, automatically generates an "alignment machine" for that object class. The alignment machine, which we call an *image funnel*, takes as input a poorly aligned example of the class and returns a well-aligned version of the example. The system is fully automatic in that it is not necessary to label parts of the objects or identify their initial poses, or even specify what constitutes an aligned image through an explicitly labeled canonical pose, although it is important that the objects be roughly aligned to begin with. For example, our system can take a set of images as output by the Viola-Jones face detector, and return an image funnel which dramatically improves the subsequent alignment of facial images.

(We note that the term *alignment* has a special meaning in the face recognition community, where it is often used to refer to the localization of specific facial features. Here, because we are using images from a variety of different classes, we use the term alignment to refer to the rectification of a set of objects that places the objects into the same canonical pose. The purpose of our alignments is not to identify parts of objects, but rather to improve positioning for subsequent processing, such as an identification task.)

1.1. Previous Work

The problem of automatic alignment from a set of examplars has been addressed previously by Learned-Miller's *congealing* procedure [10]. Congealing as traditionally described works directly on the pixel values in each image, minimizing the entropy of each column of pixels (a pixel stack) through the data set. This procedure works well when the main source of variability in a pixel value is due to misregistration. Congealing has proven to work well on simple binary handwritten digits [14] and on magnetic resonance image volumes [11, 21]. These data sets are free of many of the most vexing types of noise in images. In particular, the goal of this work was to extend congealing-style methods to handle real-world image complexity, including phenomena such as

- complex and variable lighting effects,
- occlusions,
- highly varied foreground objects (for example, for faces, arising from varying head shape, hair, beards, glasses, hats, and so forth), and
- highly varied backgrounds.

For example, on a realistic set of face images taken from news photographs, straight forward implementations of congealing did not work at all. To make the general approach of congealing work on this type of complex images, we needed to define features for congealing that ignore unimportant variability, such as lighting, have a large capture range, and are not sensitive to the clustering procedure we use to obtain the first two properties. The details of the extension are developed in Section 3.

Another information theoretic method was previously proposed by Kim *et al.* [9]. However, that method solves the separate problem of computing correspondences between two highly similar images taken from a stereo pair using mutual information, whereas our method jointly aligns an entire set of highly variable images using entropy minimization.

We show our system on different classes of images: frontal faces and rear cars. For faces, we show high quality results on the Faces in the Wild data set [2], which contains many different people under different poses and lighting, on top of complex backgrounds, in contrast to the data sets on which many other alignment methods are tested, which contain a limited number of people in front of controlled backgrounds. We then show similar quality alignment results on cars, using the same out of the box code as used for the faces, without the need for any training or labeling.

In addition, we do detailed comparisons of our results in frontal face rectification with previous work by Zhou *et al.* [19]. In particular, we show that face identifiers built using our rectified images outperform an identifier built using images that either have not been pre-processed and even exceeds an identifier built from images aligned using Zhou's supervised alignment method.

2. Background

We first review the basics of congealing. Then in Section 3 we show how to extend this framework to handle complex images.

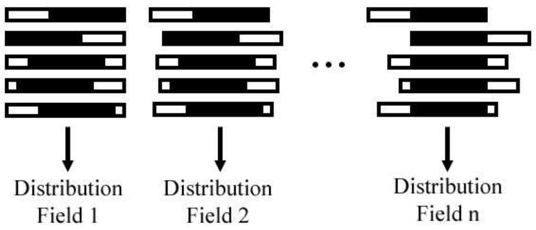


Figure 2. Schematic illustration of congealing

2.1. Distribution Field

A key concept in congealing is the **distribution field**. Let $\mathcal{X} = \{1, 2, \ldots, M\}$ be the set containing all possible feature values at a given pixel. For example, using intensity values as features, for a binary image, $M = 2$, and for a greyscale image, $M = 256$. A distribution field is a distribution over $\mathcal{X}$ at each pixel, so for a binary feature, a distribution field would be a distribution over $\{0, 1\}$ at each pixel in the image.

One can view the distribution field as a generative independent pixel model of images by placing a random variable X_i at each pixel location i. An image then consists of a draw from the alphabet $\mathcal{X}$ for each X_i according to the distribution over $\mathcal{X}$ at the ith pixel of the distribution field.

Another important concept in congealing is the **pixel stack**, which consists of the set of values with domain $\mathcal{X}$ at a specific pixel location across a set of images. Thus, the empirical distribution at a given pixel of a distribution field is determined by the pixel stack at that pixel.

Congealing proceeds by iteratively computing the empirical distribution defined by a set of images, then for each image, choosing a transformation (for example, over the set of affine transformations) that reduces the entropy of the distribution field. An important point is that, under an independent pixel model and uniform distribution over transformations, minimizing the entropy of the distribution field is equivalent to maximizing the likelihood according to the distribution field [10].

Therefore, an equivalent formulation of congealing is the following: compute the empirical distribution field of a set of images, find the transformation for each image that maximizes the likelihood of the image under the transformation according to the distribution field, then recalculate the distribution field according to the transformed images, and iterate until convergence.

2.2. Image Funnel

Once congealing has been done on a set of images, for example a training set for a face recognition algorithm, there is the question of how to align additional images, such as from a new test set. Theoretically, one could align new images by inserting them into the training set and re-running the congealing algorithm on all the images, but a more efficient technique can be used by keeping the distribution fields produced at each iteration of congealing [10].

By maintaining the sequence of distribution fields from each iteration of congealing, one can align a new image by transforming it, at each iteration, according to the saved distribution field from the corresponding iteration of the original congealing. The sequence of distribution fields begins at higher entropy as the images are initially unaligned, and decreases in entropy as the images are iteratively aligned during congealing. When aligning a new image according to this sequence of distribution fields, the image is sharpened from the initial "wide" distribution to the final "narrow" distribution, and for this reason we refer to the learned sequence of distribution fields of the training congealing as an **image funnel**, and we will refer to the alignment of a new image according to the image funnel as **funneling** to distinguish it from the original congealing.

Figure 2 illustrates the process of congealing on one dimensional binary images. At each iteration, the distribution field is a function of the set of transformed images, and the sequence of distribution fields forms an image funnel that can be later used to align new images.

3. Methodology

3.1. Congealing with SIFT descriptors

We now describe how we have adapted the basic congealing algorithm to work on realistic sets of images. We consider a sequence of possible choices for the alphabet $\mathcal{X}$ on which to congeal. In particular, we discuss how each choice improves upon the previous choice, eventually leading to an appropriate feature choice for congealing on complex images.

In applying congealing to complicated images such as faces from news photographs, a natural first attempt is to set the alphabet $\mathcal{X}$ over the possible color values at each pixel. However, the high variation present in color in the foreground object as well as the variation due to lighting will cause the distribution field to have high entropy even under a proper alignment, violating one of the necessary conditions for congealing to work.

Rather than considering color, one could set $\mathcal{X}$ to be binary, corresponding to the absence or presence of an edge at that pixel. However, another necessary condition for congealing to work is that there must be a "basin of attraction" at each point in the parameter space toward a low entropy distribution.

For example, consider two binary images a and b of the number 1, identical except for an x-translation. When searching over possible transformations to align b to a, unless the considered transformation is close enough to the exact displacement to cause b and a to overlap, the transformation will not cause any change in the entropy of the resulting distribution field.

Another way of viewing the problem is that, when $\mathcal{X}$ is over edge values, there will be plateaus in the objective function that congealing is minimizing, corresponding to neighborhoods of transformations that do not cause changes in the amount of edge overlap between images, creating many local minima problems in the optimization.

Therefore, rather than simply taking the edge values, instead, to generate a basin of attraction, one could integrate the edge values over a window for each pixel. To do this, we calculate the SIFT descriptor [13] over an 8x8 window for each pixel. This gives the desired property, since if a section of one pixel's window shares similar structure with a section of another pixel's window (need not be the corresponding section), then the SIFT descriptors will also be similar. In addition, using the SIFT descriptor gives additional robustness to lighting.

Congealing directly with the SIFT descriptors has its own difficulties, as each SIFT descriptor is a 32 dimensional vector in our implementation, which is too large of a space to estimate entropy without an extremely large amount of data. Instead, we compute the SIFT descriptors for each pixel of each image in the set, and then cluster these using kmeans to produce a small set of clusters (in our experiments, we have been using 12 clusters), and let $\mathcal{X}$ be over the possible clusters. In other words, the distribution fields consist of distributions over the possible clusters at each pixel.

After clustering, rather than assigning a cluster for each pixel, we instead do a soft assignment of cluster values for each pixel. Congealing with hard assignments of pixels to clusters would force each pixel to take one of a small number of cluster values, leading to local plateaus in the optimization landscape. For example, in the simpliest case, doing a hard assignment with two clusters would lead to the same local minima problems as discussed before with edge values.

This problem of local minima was borne out by preliminary experiments we ran using hard cluster assignments, where we found that the congealing algorithm would terminate early without significantly altering the initial alignment of any of the images.

To get around this problem, we model the pixel's SIFT descriptors as being generated from a mixture of Gaussians model, with one Gaussian centered at each cluster center and σ_i's for each cluster that maximize the likelihood of the labeling. Then, for each pixel, we have a multinomial distribution with size equal to the number of clusters, where the probability of an outcome i is equal to the probability that the pixel belongs to cluster i. So, instead of having an intensity value at each pixel, as in traditional congealing, we have a vector of probabilities at each pixel.

The idea of treating each pixel as a mixture of clusters is motivated by the analogy to gray pixels in the binary image case. In the binary image case, a gray pixel is interpreted as being a mixture of underlying black and white "subpixels" [10]. In the same way, rather than doing a hard assignment of a pixel to one cluster, we treat each pixel as being a mixture of the underlying clusters.

3.2. Implementation

Following the notation in [10], suppose we have N face images, each with P pixels. Let x_i^j be the multinomial distribution of the ith pixel in the jth image, $x_i^j(k)$ be the probability of the kth element of the multinomial distribution in x_i^j, and let $x_i^{j'}$ be the multinomial distribution of the ith pixel of the jth image under some transformation U^j. Denote the pixel stack $\{x_i^{1'}, x_i^{2'}, \ldots x_i^{N'}\}$ as x_i'.

In our congealing algorithm, we first compute the empirical distribution field defined by the images under a particular set of transformations. Define $D_i(k)$ as the probability of the kth element in the distribution at the ith pixel of the distribution field. Then, $D_i(k) = \frac{1}{N}\sum_j x_i^{j'}(k)$. The entropy of a distribution at a particular pixel i is equal to

$$H(D_i) = -\sum_k D_i(k) \log_2 D_i(k) \qquad (1)$$

Thus, at each iteration in congealing, we wish to minimize the total entropy of the distribution field $\sum_{i=1}^{P} H(D_i)$. This is equivalent to finding, for each image, the transformation that maximizes the log-likelihood of the image with respect to the distribution field, *e.g.* the transformation that maximizes

$$\sum_{i=1}^{P} \sum_k x_i^{j'}(k) \log D_i(k) \qquad (2)$$

for a given image j. In our case, this maximization is done over the transformations defined by the four parameters, x-translation, y-translation, rotation, and scaling (uniform in x and y), for each image. In our implementation, we do a hill climbing step at each iteration that increases the likelihood with respect to the distribution field at that iteration.

4. Experimental Results

4.1. Alignment on Faces in the Wild

We ran our alignment algorithm on 300 faces selected randomly from the first 300 clusters of the Faces in the Wild data set [2]. This data set consists of news photographs that cover a wide variety of pose, illumination, and background. We used the Viola-Jones face detector to extract the faces from the images, and ran the images through the congealing alignment algorithm. A representative sample of 50 of the resulting aligned images after congealing are given in Figure 5. The original images, together with the corresponding bounding boxes of the final alignments, are given in Figure 6. We also show animations of images under the transformations at each iteration of congealing on our project webpage.[2]

For comparison, we aligned the same set of images using the Zhou face alignment [19] using their web interface,[3] which returns the alignment as a set of connected landmark points. The results are given in Figure 7, and one can see that the two alignment methods are comparable, despite congealing being unsupervised. Both methods do a good job of finding the correct scale of the face, though in a few instances the Zhou alignment is thrown off, such as by partial occlusion due to a tennis raquet or confusing the bottom of the lip as the chin. Both methods also do a good job with respect to rotation, as is most evident in the first picture of the sixth row.

4.2. Cars

We also show results on a separate data set of 125 rear car images, taken from different parking lots with variable background and lighting. Since our algorithm is fully automatic, we were able to obtain these results using the same code as with faces without any labeling or training. A representative sample of the final aligment bounding boxes are given in Figure 4. Of the 50 images, only one is a clear error (6th row, 2nd column), and one is a case where the algorithm rotated the image in the right direction but not enough (7th row, 4th column). Of the other 75 images, the final bounding box captures the correct scale, rotation, and position of the car, with the exception of one other car where the algorithm again rotated the image in the right direction but not sufficiently. We emphasize again that *no changes of any kind were made to the code* before running the car examples; the algorithm ran directly as it did on the faces. We believe this is a dramatic demonstration of the generality of this method.

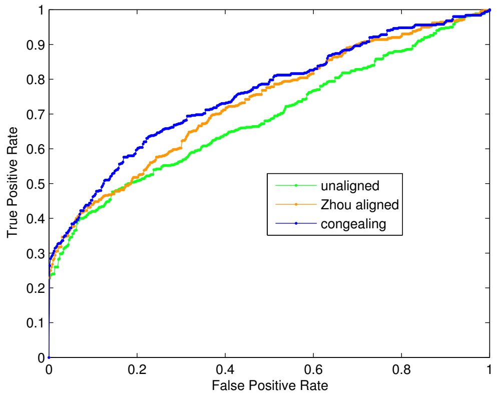
4.3. Improvement in Recognition

In addition, we also tested the performance of a face recognizer on three different alignment processes. We used a hyper-feature based recognizier of Jain *et al.* [7] with 500 randomly selected training pairs and 500 randomly selected test pairs from the Faces in the Wild data set.

For the baseline of our comparison, we trained and tested the recognizer with the unaligned face images found by the Viola-Jones face detector. Next, we examined how aligning the face images with the Zhou method and with congealing would affect the results. We used the unaligned images from the Viola-Jones face detector as input into the two systems, which, for each image, produce a similarity transformation used to align that particular image. For the congealing alignment, we aligned the images by funneling the output of the Viola-Jones face detector using the image funnel learned from congealing on the 300 faces above.

We chose to compare against the Zhou alignment algorithm rather than the Berg method presented in [3]. The Berg algorithm uses support vector machines to detect specific facial features, such as corners of eyes and tip of nose, that are then used to align the images to a canonical pose. Although this method works well for a subset of the images in their data, they throw out images with low alignment score, eliminating a large number of faces. While discarding bad alignments is appropriate for their application, for the purpose of recognition, one cannot discard difficult to align images.

On the other hand, the Zhou system is designed for detection and face point localization in addition to pose estimation, and not specifically to improve classification accuracy. However, it is reasonable to adopt the system for the purposes of alignment to a fixed coordinate system and seemed to align faces as well as anything else we found. We took care to make the comparison fair (by using the default unaligned image when no face was detected by the Zhou system and by manually picking the best face when the Zhou system detected multiple faces for a given image).



	unaligned	Zhou aligned	congealing
AUC	0.6870	0.7312	0.7549

Figure 3. ROC curves and area under curves for recognition. Using face images aligned with congealing during both training and testing of a face identifier uniformly improves accuracy, not only over images directly from the Viola-Jones detector ("unaligned") but also on images that have been aligned using the method of Zhou *et al.*

The ROC curves for the recognition, as well as the area

[2] http://www.cs.umass.edu/~gbhuang/alignment

[3] http://facealignment.ius.cs.cmu.edu/alignment/webdemo.html

under the curves, are given in Figure 3. From this figure, it is clear that our method, which is completely automated and requires no labeling of pose or parts, dramatically improves the results of recognition over the outputs of the Viola-Jones face detector, and even exceeds the supervised alignment method of Zhou in performance benefit to recognition.

5. Discussion and Conclusion

In summary, we have presented an unsupervised technique for jointly aligning images under complex backgrounds, lighting, and foreground appearance. Our method obviates hand-labeling hundreds of images while maintaining comparable performance with supervised techniques. In addition, our method increases the performance of a face recognizer by precisely aligning the images. Of course, our method is not completely unsupervised in the sense that it must be provided with images of objects of a particular class. However, in many scenarios, such images can be automatically acquired, especially since detailed alignment is not a requirement.

One extension of our work we are pursuing is to align images in a two part process. First, all the images are aligned using congealing, then the quality of the alignment is estimated for each image so that poorly aligned images can be re-aligned in a separate second stage. The quality of the alignment is estimated from the likelihood of each image under its alignment according to the final distribution field.

Another possible extension is to use the multi-view face detector in [8] to first separate face images into three separate categories: frontal, left profile, and right profile, and then attempt to align each category of faces individually.

Acknowledgements

This work was supported by Learned-Miller's CAREER Award, NSF Grant #IIS:0546666.

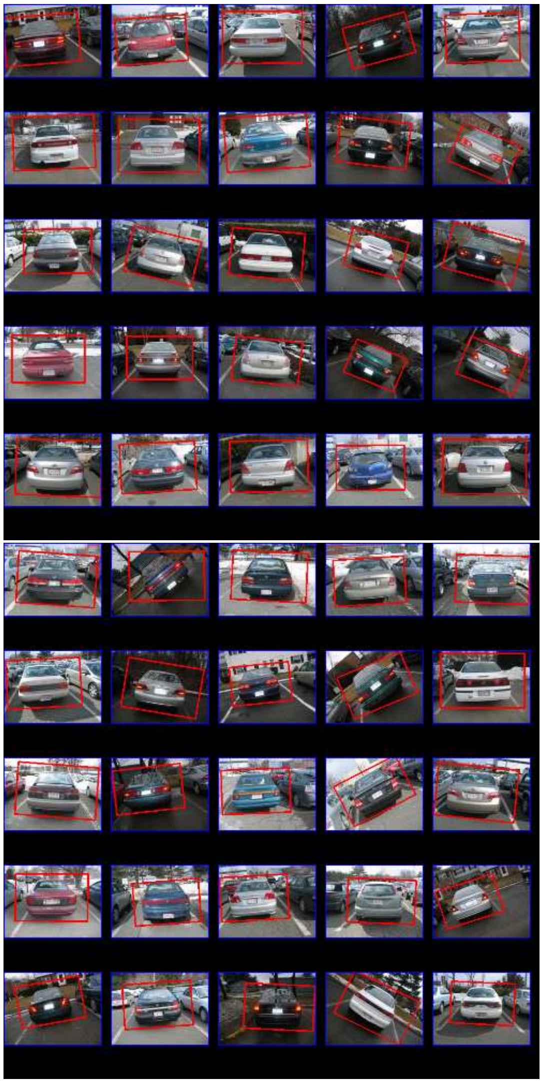


Figure 4. Input to congealing with bounding boxes of final alignment

References

[1] P. N. Belhumeur, J. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.

[2] T. L. Berg, A. C. Berg, J. Edwards, and D. A. Forsyth. Who's in the picture. *NIPS*, 2004.

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

[4] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *In Proceedings of 5th European Conference on Computer Vision*, 1998.

[5] C. Hu, R. Feris, and M. Turk. Real-time view-based face alignment using active wavelet networks. *International Workshop on Analysis and Modeling of Faces and Gestures*, 2003.

[6] Y. Huang, S. Lin, S. Z. Li, H. Lu, and H.-Y. Shum. Face alignment under variable illumination. *In Proceedings of 6th IEEE International Conference on Automatic Face and Gesture Recognition*, 2004.

[7] V. Jain, A. Ferencz, and E. Learned-Miller. Discriminative training of hyper-feature models for object identification. *In Proceedings of British Machine Vision Conference*, 2006.

[8] M. Jones and P. Viola. Fast multi-view face detection. *Mitsubishi Electric Research Laboratories Technical Report*, 2003.

[9] J. Kim, V. Kolmogorov, and R. Zabih. Visual correspondence using energy minimization and mutual information. *ICCV*, 2003.

[10] E. Learned-Miller. Data driven image models through continuous joint alignment. *PAMI*, 2005.

[11] E. Learned-Miller and V. Jain. Many heads are better than

one: Jointly removing bias from multiple mrs using nonparametric maximum likelihood. In *Proceedings of Information Processing in Medical Imaging*, pages 615–626, 2005.

[12] S. Z. Li, Y. ShuiCheng, H. Zhang, and Q. Cheng. Multiview face alignment using direct appearance models. *In Proceedings of 5th IEEE International Conference on Automatic Face and Gesture Recognition*, 2002.

[13] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2003.

[14] E. Miller, N. Matsakis, and P. Viola. Learning from one example through shared densities on transforms. *In Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[15] B. Moghaddam, T. Jebara, and A. Pentland. Bayesian face recognition. *Pattern Recognition*, 2002.

[16] H. Rowley, S. Baluja, and T. Kanade. Rotation invariant neural network-based face detection. In *Proceedings of CVPR*, 1998.

[17] M. Turk and A. Pentland. Face recognition using eigenfaces. *In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1991.

[18] P. Wang, L. C. Tran, and Q. Ji. Improving face recognition by online image alignment. *Pattern Recognition*, 2006.

[19] Y. Zhou, L. Gu, and H.-J. Zhang. Bayesian tangent shape model: Estimating shape and pose parameters via Bayesian inference. *CVPR*, 2003.

[20] Y. Zhou, W. Zhang, X. Tang, and H. Shum. A Bayesian mixture model for multi-view face alignment. *CVPR*, 2005.

[21] L. Zollei, E. Learned-Miller, E. Grimson, and W. Wells. Efficient population registration of 3d data. *Workshop on Computer Vision for Biomedical Image Applications: Current Techniques and Future Trends, at ICCV*, 2005.

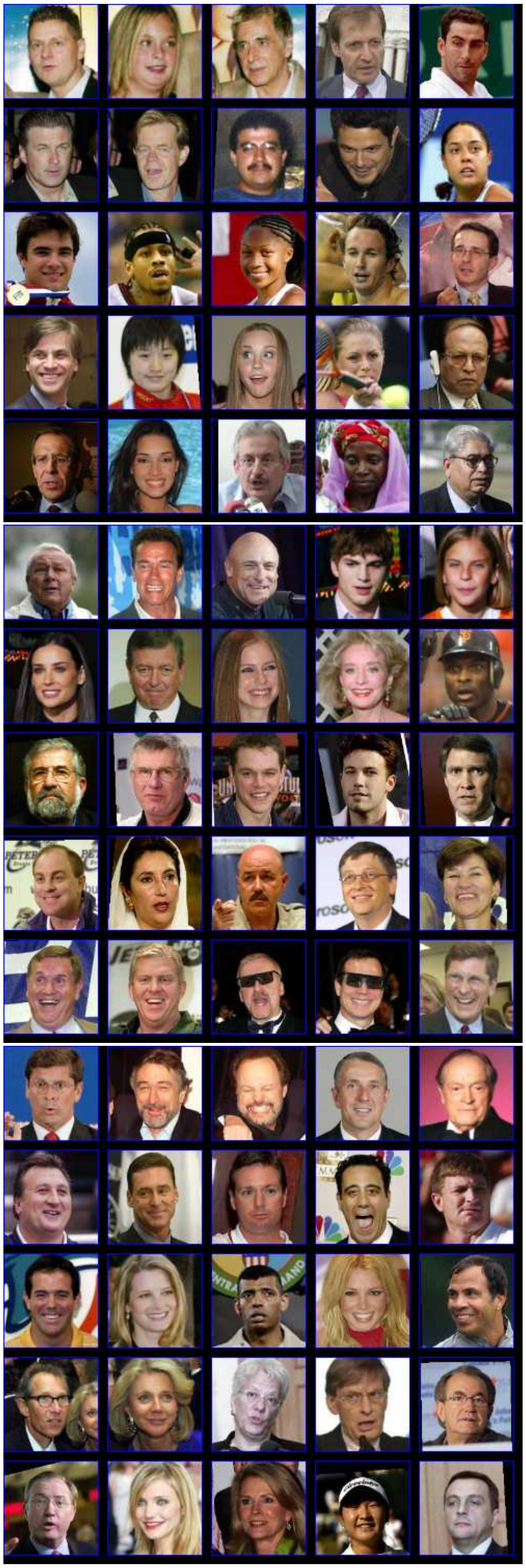


Figure 5. Aligned images output by congealing

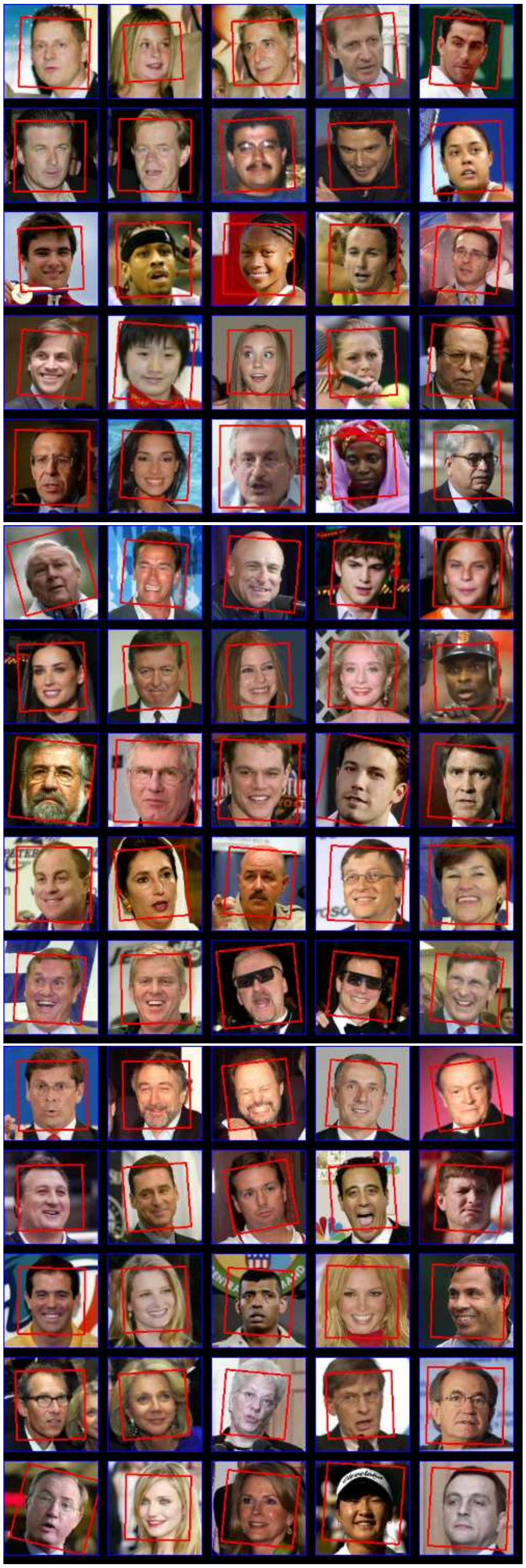


Figure 6. Input to congealing with bounding boxes of final alignment

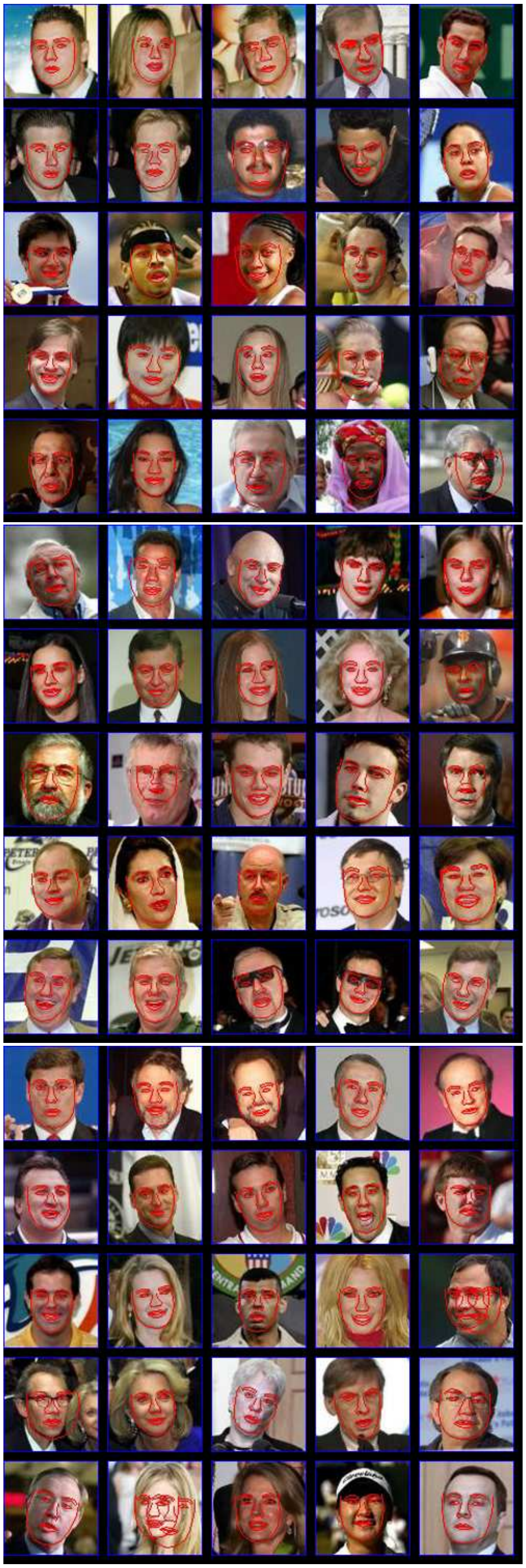


Figure 7. Results of Zhou alignment