Generic Face Alignment using Boosted Appearance Model *

Xiaoming Liu
Visualization and Computer Vision Lab
General Electric Global Research Center, Niskayuna, NY, 12309, USA
liux AT research.ge.com

Abstract

This paper proposes a discriminative framework for efficiently aligning images. Although conventional Active Appearance Models (AAM)-based approaches have achieved some success, they suffer from the generalization problem, i.e., how to align any image with a generic model. We treat the iterative image alignment problem as a process of maximizing the score of a trained two-class classifier that is able to distinguish correct alignment (positive class) from incorrect alignment (negative class). During the modeling stage, given a set of images with ground truth landmarks, we train a conventional Point Distribution Model (PDM) and a boosting-based classifier, which we call Boosted Appearance Model (BAM). When tested on an image with the initial landmark locations, the proposed algorithm iteratively updates the shape parameters of the PDM via the gradient ascent method such that the classification score of the warped image is maximized. The proposed framework is applied to the face alignment problem. Using extensive experimentation, we show that, compared to the AAM-based approach, this framework greatly improves the robustness, accuracy and efficiency of face alignment by a large margin, especially for unseen data.

1. Introduction

Image alignment is the process of moving and deforming a *template* to minimize the *distance* between the template and an image. Since Lucas and Kanade's seminar work [18], image alignment has found many applications in computer vision such as face fitting [19], image coding [3], tracking [5, 13], image mosaicing [23], etc. With the introduction of Active Shape Model (ASM) and Active Appearance Models (AAM) [7, 8, 19], face alignment/fitting has become more popular in the vision community.

*This project was supported by award #2005-IJ-CX-K060 awarded by the National Institute of Justice, Office of Justice Programs, US Department of Justice. The opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the Department of Justice.

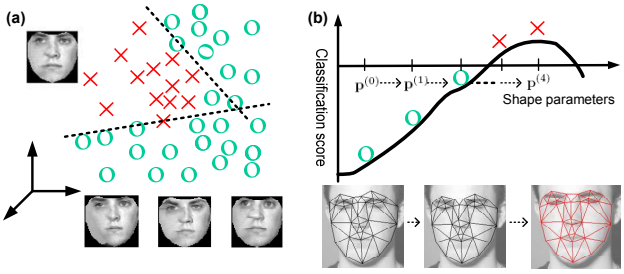

Figure 1. (a) Model training: learn a two-class classifier that distinguishes correct alignment (positive class) from incorrect alignment (negative class) based on warped images; (b) Face alignment: given initial shape parameters, iteratively update the parameter via gradient ascent such that the warped image achieves the maximal score from the trained classifier.

Essentially, there are three elements to image alignment, namely *template representation*, *distance metric*, and *optimization method*. The template can be a simple image patch, or the more sophisticated ASM and AAM. The Mean Squared Error (MSE) between the warped image and the template is one of the most widely used distance metrics. For optimization, gradient descent methods are commonly used to iteratively update the shape parameters, including Gauss-Newton, Newton, Levenberg-Marquardt, etc. The Inverse Compositional (IC) and Simultaneously Inverse Compositional (SIC) methods proposed by Baker and Matthews [2] are excellent examples of recent advances in image alignment. Their novel formulation of warp update in the optimization results in an efficient algorithm for fitting AAM to facial images. However, as indicated by [12], the alignment performance degrades quickly when the AAM are trained on a large dataset and fit to images that were not seen during the AAM training. We assert that this generalization issue is caused by the eigenspace-based appearance modeling and the use of MSE as the distance metric.

To remedy the generalization problem, this paper proposes a novel discriminative framework for image alignment. As shown in Figure 1(a), for the *template representation*, we train a boosting-based classifier that learns the de-

cision boundary between two classes, given a face dataset with ground truth landmarks. The positive class includes images warped with ground truth landmarks; the negative one includes images warped with perturbed landmarks. The set of trained weak classifiers, based on Haar-like rectangular features [21, 25], acts as an appearance model, which is called *Boosted Appearance Model (BAM)*. We then use the score from the trained strong classifier as the *distance metric*, which is a continuous value proportional to the accuracy of alignment, to align the image by maximizing the classification score. As shown in Figure 1(b), the image warped from the initial shape parameters $\mathbf{p}^{(0)}$ will likely have a negative score. The shape parameters are iteratively updated such that the classification score keeps increasing.

The proposed image alignment framework has three main contributions.

◇ We propose a novel discriminative method of appearance modeling via boosting. Unlike the conventional generative model-based AAM that only model the Gaussian distribution of correct alignment, the BAM learns the discriminative properties between correct and incorrect alignment. Also, the BAM is a much more compact representation compared to AAM, since only the weak classifier parameters are stored as the *model*. Furthermore, the local rectangular features used in the BAM makes it robust to partial occlusion.

◇ We propose a novel alignment algorithm through maximizing the classification score. Compared to minimizing the MSE in AAM-based approaches, our method benefits from the fact that the boosting method is known to be capable of learning from a large dataset and generalizing well to unseen data. The final classification score after convergence also provides a natural way to describe the quality of the image alignment.

◇ We greatly improve the performance of generic face alignment. The AAM-based approach performs well for person-specific or small population-based face alignment. Our proposal improves it toward the goal that a face alignment algorithm should be able to fit to faces from any subject and with any expression in real time.

2. Related Work

Due to the needs of many practical applications such as face recognition, expression analysis and pose estimation, extensive research has been conducted in face alignment, especially using model-based approaches. AAM, ASM [2, 7, 8] and their variations [4, 9, 15, 27, 28] are probably the most popular model-based face alignment methods because of their elegant mathematical formulation and efficient computation. For the template representation, their basic idea is to use two eigenspaces to model the facial shape and shape-free appearance respectively. For the distance metric, the MSE between the appearance instance synthesized from the appearance eigenspace and the warped appearance from the image observation is minimized by iteratively updating the shape and/or appearance parameters.

It is well known that AAM-based face alignment has difficulty with generalization [12]. That is, the alignment tends to diverge on images that are not included as the training data for learning the model, especially when the model is trained on a large dataset. In part, this is due to the fact that the appearance model only learns the appearance variation retained in the training data. When more training data is used to model larger appearance variations, the representational power of the eigenspace is very limited even under the cost of a much higher-dimensional appearance subspace, which in turn results in a harder optimization problem. Also, using the MSE as the distance metric essentially employs an analysis-by-synthesis approach, further limiting the generalization capability by the representational power of the appearance model. Researchers have noticed this problem and proposed methods to handle it. Jiao *et al.* [16] suggest using Gabor wavelet features to represent the local appearance information. Hu *et al.* [15] utilize a wavelet network representation to replace the eigenspace-based appearance model, and demonstrate improved alignment with respect to illumination changes and occlusions.

The basic idea of our proposal is optimization via *maximizing a classification score*. Similar ideas have been explored in object tracking research [1, 14, 26]. Avidan [1] estimates the 2D translation parameters by maximizing the Support Vector Machine (SVM) classification score. Limitations of this method include dealing with partial occlusions and the large number of support vectors which might be needed for tracking, burdening both computation and storage. Williams *et al.* [26] build a displacement expert, which takes an image as input and returns the displacement, by using Relevance Vector Machine (RVM). Since RVM is basically a probabilistic SVM, it still suffers from the problem of requiring a large set of support vectors. The recent work by Hidaka *et al.* [14] performs face tracking (2D translation only) via maximizing the score from a Viola and Jones face detector [25], where a face versus non-face classifier is trained. Our proposal differs from these works in that we are dealing with a much larger shape space than object tracking, where often only 2D translation is estimated.

3. Shape and Appearance Modeling

3.1. Point Distribution Model

The Point Distribution Model (PDM) is trained with a representative set of facial images [7]. Given a face database, each facial image is manually labeled with a set of 2D landmarks, $[x_i, y_i]$ $i = 1, 2, ..., v$. The collection of landmarks of one image is treated as one observation from the random process defined by the shape model,

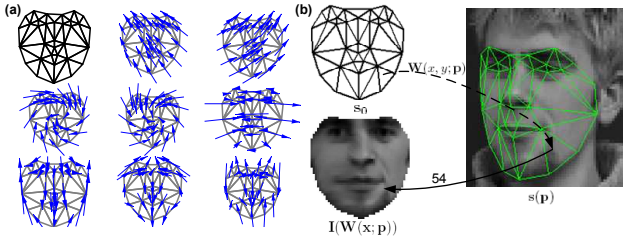

Figure 2. (a) The mean shape and first 8 shape bases of the PDM; (b) Image warping from the image observation to the mean shape. Given a pixel coordinate (x, y) in the mean shape $\mathbf{s}_0$, $\mathbf{W}(x, y; \mathbf{p})$ indicates the corresponding pixel in the image observation, whose intensity value (54) is obtained via bilinear interpolation and treated as one element of the N-dimensional vector $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$.

$\mathbf{s} = [x_1, y_1, x_2, y_2, ..., x_v, y_v]^\mathbf{T}$. Eigen-analysis is applied to the observation set and the resultant model represents a shape as,

$$\mathbf{s}(\mathbf{p}) = \mathbf{s}_0 + \sum_{i=1}^{n} p_i \mathbf{s}_i, \tag{1}$$

where $\mathbf{s}_0$ is the mean shape, $\mathbf{s}_i$ is the i^{th} shape basis, and $\mathbf{p} = [p_1, p_2, ..., p_n]^\mathbf{T}$ are the shape parameters. Figure 2(a) shows an example of PDM. By design, the first four shape bases represent global translation and rotation. Together with other bases, a warping function from the model coordinate system to the coordinates in the image observation is defined as $\mathbf{W}(x, y; \mathbf{p})$, where (x, y) is a pixel coordinate within the face region defined by the mean shape $\mathbf{s}_0$. Figure 2(b) shows one example of the warping process.

We define the warping function with a piecewise affine warp:

$$\mathbf{W}(x, y; \mathbf{p}) = [1\ x\ y]\mathbf{a}(\mathbf{p}), \tag{2}$$

where $\mathbf{a}(\mathbf{p}) = [\mathbf{a}_1(\mathbf{p})\ \mathbf{a}_2(\mathbf{p})]$ is a 3 by 2 affine transformation matrix that is unique to each triangle pair between $\mathbf{s}_0$ and $\mathbf{s}(\mathbf{p})$. Given shape parameters $\mathbf{p}$, the $\mathbf{a}(\mathbf{p})$ matrix needs to be computed for each triangle. However, since the knowledge of which triangle each pixel (x, y) belongs to can be pre-computed, the warp can be efficiently performed via a table lookup, inner product as in Equation 2, and bilinear interpolation of the image observation $\mathbf{I}$. We denote the resultant warped image as a N-dimensional vector $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$, where $\mathbf{x}$ is the set of all pixel coordinates within $\mathbf{s}_0$.

3.2. Boosted Appearance Model

Boosting-based learning iteratively selects weak classifiers to form a strong classifier using summation: $F(x) = \sum_{m=1}^{M} f_m(x)$, where $F(x)$ is the strong classifier and $f_m(x)$'s are the weak classifiers. Different variants of boosting have been proposed in the literature [20]. We use the GentleBoost algorithm [11] based on two considerations. First, unlike the commonly used AdaBoost algorithm [10], the weak classifier in the GentleBoost algorithm is a soft classifier with continuous output. This property allows the output of the strong classifier to be smoother and favorable as an alignment metric. In contrast, the hard weak classifiers in the AdaBoost algorithm lead to a piecewise constant strong classifier, which is difficult to optimize. Second, as shown in [17], for object detection tasks, the GentleBoost algorithm outperforms other boosting methods in that it is more robust to noisy data and more resistant to outliers.

Input: Training data $\{x_i; i = 1, 2, ..., K\}$ and their corresponding class labels $\{y_i; i = 1, 2, ..., K\}$.
Output: A strong classifier $F(x)$.
1. Initialize weights $w_i = 1/K$, and $F(x) = 0$.
2. **for** $m = 1, 2, ..., M$ **do**
   (a) Fit the regression function $f_m(x)$ by weighted least-squares (LS) of y_i to x_i with weights w_i:
   $$f_m(x) = \underset{f \in \mathcal{F}}{\operatorname{argmin}}\, \epsilon(f) = \sum_{i=1}^{K} w_i (y_i - f(x_i))^2. \tag{3}$$
   (b) Update $F(x) = F(x) + f_m(x)$.
   (c) Update the weights by $w_i = w_i e^{-y_i f_m(x_i)}$ and normalize the weights such that $\sum_{i=1}^{K} w_i = 1$.
end
3. Output the classifier $sign[F(x)] = sign[\sum_{m=1}^{M} f_m(x)]$.

Algorithm 1: The GentleBoost algorithm.

We employ the boosting framework (Algorithm 1) to train a classifier that is able to distinguish correct alignment from incorrect alignment. Given a face database with manually labeled landmarks $\mathbf{s}$, the ground truth shape parameters $\mathbf{p}$ for each face image $\mathbf{I}$ are computed based on Equation 1. Then, the set of warped images $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$ are treated as positive samples ($y_i = 1$) for the boosting. For each image, a number of negative samples ($y_i = -1$) are synthesized by randomly perturbing each element of $\mathbf{p}$ up to $\pm\mu$, where μ is the corresponding eigenvalue of the shape basis in the PDM. Note that in our context, a training sample for boosting is a warped image $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$.

We now define the weak classifier. Given that real-time face alignment is desired, we construct the weak classifier based on the Haar-like rectangular features [21, 25], whose fast evaluation is enabled by the integral image [25]. As shown in Figure 3(a), the rectangular feature can be parameterized by (r, c, dr, dc, b), where (r, c) is the top-left corner, (dr, dc) is the height and width, and b is the feature type. Figure 3(b) shows the feature types used in our algorithm. We propose a novel feature type, where two detached rectangles occupy the mirror-position of two sides of the face, based on the fact that the warped face is approximately symmetric in the horizontal direction. The hypothesis space $\mathcal{F}$, where (r, c, dr, dc, b) resides, is obtained via an

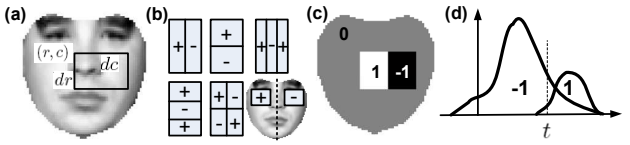

Figure 3. (a) The parametrization of a weak classifier; (b) The six feature types; (c) The notional template $\mathbf{A}$, whose inner product with the warped image is mathematically equivalent to computing a rectangular feature; (d) Let the rectangular features of the positive samples have larger mean than that of the negative samples, by multiplying a sign $g = \{1, -1\}$, and then estimate the threshold t that has the minimal weighted LS error via binary search.

exhaustive construction within the mean shape. For example, there are more than $300,000$ such rectangular features for a mean shape with size of 30×30. The crucial step in the GentleBoost algorithm, Step 2(a) in Algorithm 1, is the feature selection process. It selects a weak classifier with minimal error $\epsilon(f)$ from the hypothesis space using exhaustive search.

We use the weak classifier defined as follows:

$$f_m(\mathbf{p}) = \frac{2}{\pi} atan(g_m \mathbf{A}_m^{\mathbf{T}} \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p})) - t_m), \qquad (4)$$

where $\mathbf{A}_m$ is a template, g_m is ± 1 and t_m is a threshold. Given a rectangular feature (r, c, dr, dc, b), we can generate a corresponding template $\mathbf{A}$, as shown in Figure 3(c). The inner product between the template and the warped image is equivalent to computing the rectangular feature using the integral image. Once the rectangular features for a set of training samples are computed, $g_m = -1$ if the mean of the features of positive samples is less than that of the negative samples, otherwise $g_m = 1$. The threshold, t_m, is obtained through binary search along the span of the rectangular features, such that the weighted LS error is minimal. The $atan()$ function makes this weak classifier different from the commonly used stump classifier in the AdaBoost algorithm, since the classifier response $f_m(\mathbf{p})$ is continuous within -1 and 1.

The results of the boosting are a number of weak classifiers, each with 7 parameters $\mathbf{c}_m = (r, c, dr, dc, b, g, t)$. We call the set of weak classifiers $\{\mathbf{c}_m; m = 1, 2, ..., M\}$ a Boosted Appearance Model (BAM). Figure 4 shows the top 3 rectangular features, and the spatial distribution of the top 50 features trained from a face dataset with 400 images.

Compared to the generative model-based AAM, the discriminative model-based BAM has a number of advantages. First, the BAM learns from not only the appearance of correct alignment, which is basically what AAM do, but also the appearance of incorrect alignment. Second, because of the local rectangular features, the BAM is inherently more likely to be robust to partial occlusion. Third, from the storage point of view, the BAM is a much more storage-efficient way of modeling the appearance information. We

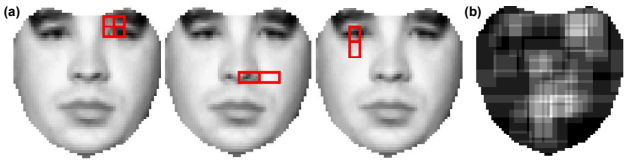

Figure 4. (a) The top 3 features selected by the GentleBoost algorithm. The rectangles are well aligned with the boundary of facial features, such as eyes and nose; (b) The brightness indicates the density of the top 50 rectangular features. Most classifier features are located on facial features.

do not store the training data. The knowledge of the training data is absorbed in the selected rectangular features. Hence, the BAM only requires a 7 by M matrix to be saved as the *model*. In contrast, AAM need a N by Q matrix where Q is the number of appearance bases, $N \gg M$, and $Q > 7$. The storage-efficient property of the BAM enables the potential of performing model-based face alignment from mobile devices such as cell phones.

4. Face Alignment

4.1. Problem Definition

Given the trained PDM and BAM, we formally define the problem we are trying to solve: *Find the shape parameters* $\mathbf{p}$ *to maximize the score of the strong classifier*

$$\max_{\mathbf{p}} \sum_{m=1}^{M} f_m(\mathbf{p}). \qquad (5)$$

In the context of face alignment, solving this problem means that given the initial shape parameters $\mathbf{p}^{(0)}$, we look for the new shape parameters that lead to the warped image with the maximal score from the strong classifier.

Because image warping is involved in the objective function, this is a nonlinear optimization problem. We choose to use the gradient ascent method to solve this problem iteratively.

4.2. Algorithm Derivation

Plugging Equation 4 into Equation 5, the function to be maximized is

$$F(\mathbf{p}) = \sum_{m=1}^{M} \frac{2}{\pi} atan(g_m \mathbf{A}_m^{\mathbf{T}} \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p})) - t_m). \qquad (6)$$

Taking the derivative with respect to $\mathbf{p}$ gives

$$\frac{dF}{d\mathbf{p}} = \frac{2}{\pi} \sum_{m=1}^{M} \frac{g_m [\nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}]^{\mathbf{T}} \mathbf{A}_m}{1 + [g_m \mathbf{A}_m^{\mathbf{T}} \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p})) - t_m]^2}, \qquad (7)$$

where $\nabla \mathbf{I}$ is the *gradient* of the image $\mathbf{I}$ evaluated at $\mathbf{W}(\mathbf{x}; \mathbf{p})$, and $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is the *Jacobian* of the warp evaluated at $\mathbf{p}$.

Input: Input image $\mathbf{I}$, initial shape parameters $\mathbf{p}$, PDM $\{\mathbf{s}_i; i = 0, 1, ..., n\}$, BAM $\{\mathbf{c}_m; m = 1, 2, ..., M\}$, and pre-computed Jacobian $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$.
Output: Shape parameters $\mathbf{p}$.
0. Compute the 2D gradient of image $\mathbf{I}$.
repeat
1. Warp $\mathbf{I}$ with $\mathbf{W}(\mathbf{x}; \mathbf{p})$ to compute $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$.
2. Compute the feature for each weak classifier: $e_m = g_m \mathbf{A}_m^{\mathbf{T}} \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p})) - t_m; m = 1, 2, ..., M$.
3. Bilinearly interpolate the gradient of image $\mathbf{I}$ at $\mathbf{W}(\mathbf{x}; \mathbf{p})$.
4. Compute the steepest descent image $SD = \nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$.
5. Compute the integral images for each column of SD and obtain the rectangular features for each weak classifier: $\mathbf{b}_m = g_m SD^{\mathbf{T}} \mathbf{A}_m; m = 1, 2, ..., M$.
6. Compute $\triangle \mathbf{p}$ using $\triangle \mathbf{p} = \lambda \frac{2}{\pi} \sum_{m=1}^{M} \frac{\mathbf{b}_m}{1+e_m^2}$.
7. Update $\mathbf{p} = \mathbf{p} + \triangle \mathbf{p}$.
until $|| \sum_{i=1}^{n} \triangle \mathbf{p}_i \mathbf{s}_i || \leq \tau$.

Algorithm 2: The boosting-based alignment algorithm.

The derivative $\frac{dF}{d\mathbf{p}}$ indicates the direction to modify $\mathbf{p}$ such that the classification score increases. Thus, during the alignment iteration, the shape parameters $\mathbf{p}$ are updated via

$$\mathbf{p} = \mathbf{p} + \lambda \frac{dF}{d\mathbf{p}}, \tag{8}$$

where λ is the step size, until the change of the facial landmark locations is less than a certain threshold τ.

We now discuss how to compute $\frac{dF}{d\mathbf{p}}$ efficiently. Based on Equation 2 and the chain rule,

$$\frac{\partial \mathbf{W}}{\partial \mathbf{p}} = [\frac{\partial \mathbf{W}}{\partial \mathbf{a}_1} \frac{\partial \mathbf{a}_1}{\partial \mathbf{p}} \quad \frac{\partial \mathbf{W}}{\partial \mathbf{a}_2} \frac{\partial \mathbf{a}_2}{\partial \mathbf{p}}], \tag{9}$$

where $\frac{\partial \mathbf{W}}{\partial \mathbf{a}_1}$ and $\frac{\partial \mathbf{W}}{\partial \mathbf{a}_2}$ are both N by 3 matrices and N is the number of pixels in the warped images. Since the affine parameter $\mathbf{a}$ is a linear function of $\mathbf{p}$, $\frac{\partial \mathbf{a}_1}{\partial \mathbf{p}}$ and $\frac{\partial \mathbf{a}_2}{\partial \mathbf{p}}$ are independent of $\mathbf{p}$. Thus $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ does not depend on $\mathbf{p}$. In other words, it can be pre-computed and does not need updating in each alignment iteration. Note that we have this computational gain only because we use the piecewise affine warp, which is linear on $\mathbf{p}$. In theory, $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ needs to be re-evaluated if $\mathbf{p}$ are updated, when for example the warp is polynomial on $\mathbf{p}$.

We call $SD = \nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ the steepest descent image, which is a N by n matrix where n is the number of shape bases and N is defined above. Similar to $\mathbf{A}_m^{\mathbf{T}} \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$, we do not need to perform the actual matrix multiplication between SD and $\mathbf{A}_m$. Instead, we compute the integral images of each column in SD and then calculate the rectangular features of $\mathbf{A}_m$ by a fast table lookup.

The alignment algorithm is summarized in Algorithm 2. Basically $\mathbf{b}_m$ in Step 5 can be considered as the gradient direction derived from each weak classifier. However, its contribution to the final gradient $\frac{dF}{d\mathbf{p}}$ is determined by $\frac{1}{1+e_m^2}$. The weak classifiers with low $|e_m|$ are less certain in their own classification decision. These weak classifiers contribute more to the final *travel direction*. Obviously this observation conforms well with intuition.

We summarize the computation cost for each step during one iteration in Table 1. Note that because of using integral images, the most computationally intensive step, Step 5, can be computed in a relatively efficient way.

Table 1. The computation cost of the alignment algorithm at one iteration. n is the number of shape bases, N is the number of pixels within the mean shape, and M is the number of weak classifiers.

Step 1	**Step 2**	**Step 3**	**Step 4**
$O(nN)$	$O(N+M)$	$O(N)$	$O(nN)$
Step 5	**Step 6**	**Step 7**	**Total**
$O(n(N+M))$	$O(nM)$	$O(N)$	$O(n(N+M))$

Figure 5. Examples of the face dataset: ND1 database (left), FERET database (middle), and IMM database (right).

5. Experiments

5.1. Face Dataset and Experimental Procedure

To evaluate our algorithm, we collect a set of 968 images from multiple public available databases, including the ND1 database [6], FERET database [22] and IMM database [24]. Figure 5 shows sample images from these three databases. We partition all images into three distinct datasets. Table 2 lists the properties of each database and partition. Set 1 includes 400 images (one image per subject) from two databases and is used as the training set for the PDM and BAM. Set 2 includes 334 images from the *same* subjects but different images as the ND1 database in Set 1. Set 3 includes 234 images from 40 subjects in the IMM database that were never used in the training. This partition ensures that we have two levels of generalization to be tested, *i.e.*, Set 2 is tested as the unseen data of seen subjects; Set 3 is tested as the unseen data of unseen subjects. There are 33 manually labeled landmarks for each image. To speed up the training process, we down-sample the image set such that the facial width is roughly 40 pixels across the set.

Given a dataset with ground truth landmarks, the experimental procedure consists of running the alignment algorithm on each image with a number of initial landmarks

Table 2. Summary of the dataset.

	ND1	FERET	IMM
Images	534	200	234
Subjects	200	200	40
Variations	Frontal view	Pose	Pose, expression
Set 1	200	200	
Set 2	334		
Set 3			234

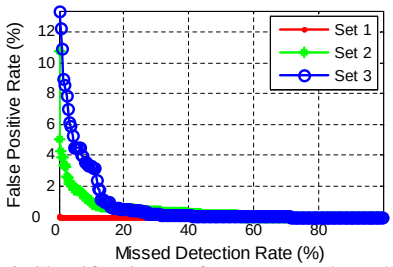

Figure 6. Classification performance on three datasets.

and statistically evaluating the alignment results. The initial landmarks are generated by randomly perturbing the ground truth landmarks by an independent Gaussian distribution whose variances equal to a multiple (sigma) of the eigenvalue of shape basis during PDM training. We declare that the alignment converges if the resultant Root Mean Square Error (RMSE) between the aligned landmarks and the ground truth is less than 1.0 pixel after the algorithm terminates. Two metrics are used to evaluate the alignment results for the converged trials. One is the *Average Frequency of Convergence* (AFC), which is the number of trials where the alignment converges divided by the total number of trials. The other is the Histogram of the resultant RMSE (HRMSE) of the converged trials, which measures how close the aligned landmarks are with respect to the ground truth. These two metrics measure the robustness and accuracy of alignment respectively.

We compare our algorithm with the Simultaneous Inverse Compositional (SIC) algorithm [19], which has been shown to perform best among the family of AAM-based methods. We ensure both algorithms are tested under the same conditions. For example, both algorithms are initialized with the *same* set of randomly perturbed landmarks. Both algorithms have the same termination condition. That is, if the number of iterations is larger than 55 or the RMSE is less than 0.025 pixels. Also, HRMSE is only computed on the trials where both algorithms converge.

5.2. Experimental Results

We train the PDM and BAM on Set 1. There are 400 positive and 4000 negative samples, where each image synthesizes 10 negative samples, used in the boosting-based learning. The resultant PDM has 33 shape bases and the BAM has 50 weak classifiers. We determine the number of weak classifiers by whether the current set of weak classifiers can generate less than 0.1% false alarm rate at 0% missed detection rate on the training set. In contrast, the SIC uses the same PDM model as ours and an appearance model with 24 appearance bases. The number of the appearance bases is chosen such that 99% of the energy is retained in the appearance model for the training set.

To test the generalization capability of the trained BAM, we perform the classification on three datasets. For Set 2,

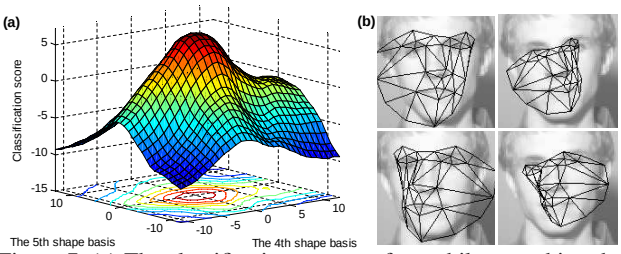

Figure 7. (a) The classification score surface while perturbing the shape parameters in the neighborhood of the ground truth along the 4^{th} and 5^{th} shape basis. The convex surface favors the gradient ascent method; (b) The four perturbed facial landmarks when the perturbation is at the four corners of the surface on the left.

we obtain 334 positive samples by warping images using the ground truth landmarks and 3340 negative samples by perturbing each ground truth landmarks 10 times, using the same methodology as for Set 1. Similarly, 234 positive and 2340 negative samples are generated from Set 3. By setting different thresholds for the classification score $F(\mathbf{p})$, performance curves are shown in Figure 6. Although it is expected that the performances on Set 2 and 3 are worse than that of Set 1, the BAM still achieves reasonable classification capability on the unseen data, which enables the potential of using the BAM in the alignment optimization.

Figure 7(a) shows that for a given image, a convex surface of classification scores can be observed while perturbing the shape parameters along two shape bases and setting the shape parameters at other bases to be zero. It is obvious that the gradient ascent algorithm can perform well on this type of surface. The range of the perturbation equals 1.6 times the eigenvalue of these two bases. When the perturbation is at the maximal amount for two bases, the corresponding four perturbed landmarks are plotted at Figure 7(b). In the following experiments, when the sigma equals 1.6, the actual initial landmarks could be further away from the ground truth compared to these four examples because all bases are allowed to be perturbed.

Figure 8 illustrates an example of iterative boosting-based face alignment. Given the initial landmarks, as shown in the first image of Figure 8(a), the alignment iteratively updates the facial landmarks, which has decreasing RMSE with respect to the ground truth and increasing classifica-

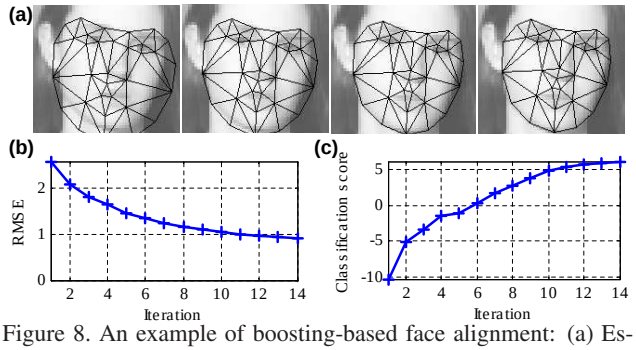

Figure 8. An example of boosting-based face alignment: (a) Estimated landmarks at iteration 1, 5, 10, and 14; (b) Decreasing RMSE during the iterative alignment process; (c) Increasing classification scores during the iterative alignment process.

tion score for the warped image. Note that computing the score is just for illustration purposes and is not a necessary step during the alignment iteration. However, the score after the alignment convergence, which is quickly computed via $\frac{2}{\pi}\sum_{m=1}^{M} atan(e_m)$, can serve as an indication of the quality of the image alignment.

The first experiment is to test the face alignment algorithms on Set 1. The results are shown in the first row of Figure 9. The horizontal axis determines the amount of the perturbation of the initial landmarks. Given one sigma value, we randomly generate 2000 trials, where each one of 400 images has 5 random initializations. Each sample in Figure 9(a) is averaged based on these 2000 trials. For the trials where both algorithms converge, we plot the histogram of their respective converged RMSE in Figure 9(b). The same experiments are performed for Set 2 and Set 3 with 2004 and 2106 trials respectively, using the same PDM and BAM as that of Set 1. The results are shown in the second and third row of Figure 9. The step size λ is manually set to be the same constant for all experiments.

We make a number of observations from this experiment. First, boosting-based alignment performs substantially better than the SIC algorithm, both in terms of alignment robustness (AFC) and accuracy (HRMSE). Second, although both algorithms have worse performance when fitting to unseen images, the BAM has a lower relative performance drop compared to the SIC. For example, for the BAM tests on unseen data of seen subjects (Set 2), the AFC is almost the same as the test on Set 1.

One strength of rectangular features is that they are localized features. Thus inherently they are likely to be robust to partial occlusion. We perform the second experiment to illustrate this. We generate a white square whose size is a certain percentage of the facial width and randomly place it on the tested face area. We perturb the initial landmarks in the usual way by fixing the sigma of the shape bases to be 1.0. As shown in Figure 10, five levels of occlusion are tested on Set 2. This shows that the boosting-based alignment can tolerate a certain level of occlusion because of the nature of features used in the appearance modeling.

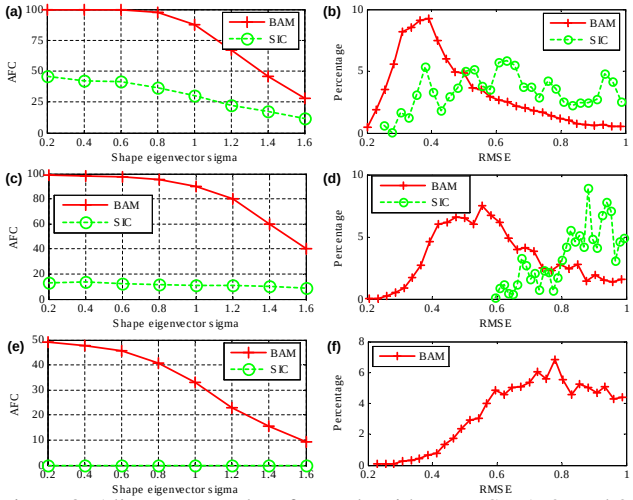

Figure 9. Alignment results of two algorithms on Set 1, 2, and 3. From top to bottom, each row is the result for one set. Left column is the AFC; right column is the histogram of the resultant RMSE for the trials where both algorithms converge. Only the HRMSE of the BAM is plotted at (f) since the SIC has no convergence.

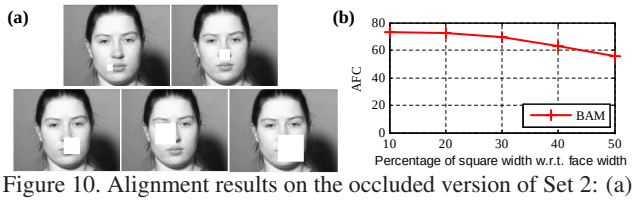

Figure 10. Alignment results on the occluded version of Set 2: (a) Five different levels of occlusions; (b) The average frequency of convergence under five levels of occlusions.

Table 3. The computation cost of the alignment test on Set 2. Our algorithm can run 8 frames per second even with a Matlab™ implementation.

Sigma	0.2	0.6	1.0	1.4	1.6
BAM-iterations	7.1	7.3	7.6	9.1	9.0
SIC-iterations	54.4	54.8	54.6	54.4	55.4
BAM-time (sec.)	0.10	0.11	0.11	0.14	0.14
SIC-time (sec.)	0.62	0.59	0.61	0.62	0.60

Table 3 lists the computation cost of the alignment test on Set 2, without occlusion. The number of iterations and times in fitting one image are averaged for the trials where both algorithms converge. It is clear that with different amount of perturbation, the BAM performs consistently faster than the SIC algorithm and converges in fewer iterations. The cost is based on a Matlab™ implementation of both algorithms running on a conventional 2.13 GHz Pentium™4 laptop. It is anticipated that our algorithm will run faster than real-time (30 frames per second) with a C++ implementation.

6. Conclusions

This paper proposes a novel discriminative framework for the image alignment problem. For the *template representation*, given a face dataset with ground truth landmarks, we train a boosting-based classifier that is able to learn the decision boundary between two classes: the warped images from ground truth landmarks and those from perturbed landmarks. The set of trained weak classifiers based on Haar-like rectangular features is considered as an appearance model, which we call *Boosted Appearance Model (BAM)*. For the *distance metric*, we use the score from the strong classifier and treat the image alignment as the process of maximizing the classification score. On the generic face alignment problem, the proposed framework greatly improves the robustness, accuracy, and efficiency of alignment.

There are several future directions to extend this framework. First, since this paper opens the door of applying discriminative learning in image alignment, many prior art in pattern recognition, such as other boosting variations or pattern classifiers, can be utilized to replace the GentleBoost algorithm for learning a better appearance model. For example, incremental boosting can be used for adding warped images that are hard to classify into the training data, so as to improve the classification capability of the BAM. Second, more sophisticated optimization methods can be used to maximize the classification score. Finally, as a generic image alignment framework, our proposal does not make use of the domain knowledge of the human faces, except the symmetric rectangular feature type. Hence, this framework can be applied to other image alignment problems, such as medical applications.

References

[1] S. Avidan. Support vector tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(8):1064–1072, 2004.

[2] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework. *Int. J. Computer Vision*, 56(3):221–255, 2004.

[3] S. Baker, I. Matthews, and J. Schneider. Automatic construction of active appearance models as an image coding problem. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(10):1380–1384, 2004.

[4] A. Batur and M. Hayes, III. Adaptive active appearance models. *IEEE Trans. Image Processing*, 14(11):1707–1721, 2005.

[5] M. Black and A. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *Int. J. Computer Vision*, 26(1):63–84, 1998.

[6] K. Chang, K. Bowyer, and P. Flynn. Face recognition using 2D and 3D facial data. In *Proc. ACM Workshop on Multimodal User Authentication*, pages 25–32, 2003.

[7] T. Cootes, D. Cooper, C. Tylor, and J. Graham. A trainable method of parametric shape description. In *Proc. 2nd British Machine Vision Conference, Glasgow, UK*, pages 54–61, 1991.

[8] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001.

[9] D. Cristinacce and T. Cootes. Facial feature detection and tracking with automatic template selection. In *Proc. 7th Int. Conf. on Automatic Face and Gesture Recognition, Southampton, UK*, pages 429–434, 2006.

[10] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[11] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 38(2):337–374, 2000.

[12] R. Gross, I. Matthews, and S. Baker. Generic vs. person specific active appearance models. *Image and Vision Computing*, 23(11):1080–1093, 2005.

[13] G. Hager and P. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(10):1025–1039, 1998.

[14] A. Hidaka, K. Nishida, and T. Kurita. Face tracking by maximizing classification score of face detector based on rectangle features. In *Proc. IEEE International Conference on Computer Vision Systems, New York, NY*, page 48, 2006.

[15] C. Hu, R. Feris, and M. Turk. Active wavelet networks for face alignment. In *Proc. 14th British Machine Vision Conference, Norwich, UK*, 2003.

[16] F. Jiao, S. Li, H.-Y. Shum, and D. Schuurmans. Face alignment using statistical models and wavelet features. In *Proc. IEEE Computer Vision and Pattern Recognition, Madison, Wisconsin*, volume 1, pages 321–327, 2003.

[17] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *Proc. 25th Pattern Recognition Symposium, Madgeburg, Germany*, pages 297–304, 2003.

[18] B. Lucas and T. Kanade. An iterative technique of image registration and its application to stereo. In *Proc. 7th International Joint Conference on Artificial Intelligence, Vancouver, Canada*, pages 674–679, 1981.

[19] I. Matthews and S. Baker. Active appearance models revisited. *Int. J. Computer Vision*, 60(2):135–164, 2004.

[20] R. Meir and G. Raetsch. *An introduction to boosting and leveraging*. S. Mendelson and A. Smola, Editors, Advanced Lectures on Machine Learning, LNAI 2600. Springer, 2003.

[21] C. P. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *Proc. 6th Int. Conf. on Computer Vision, Bombay, India*, pages 555–562, 1998.

[22] P. J. Phillips, H. Moon, P. J. Rauss, and S. Rizvi. The FERET evaluation methodology for face recognition algorithms. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, 2000.

[23] H.-Y. Shum and R. Szeliski. Construction of panoramic image mosaics with global and local alignment. *Int. J. Computer Vision*, 36(2):101–130, 2000.

[24] M. B. Stegmann, B. K. Ersboll, and R. Larsen. FAME - A flexible appearance modeling environment. *IEEE Trans. Medical Imaging*, 22(10):1319–1331, 2003.

[25] P. Viola and M. Jones. Robust real-time face detection. *Int. J. Computer Vision*, 57(2):137–154, 2004.

[26] O. Williams, A. Blake, and R. Cipolla. Sparse Bayesian learning for efficient visual tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(8):1292–1304, 2005.

[27] S. Yan, C. Liu, S. Z. Li, H. Zhang, H.-Y. Shum, and Q. Cheng. Face alignment using texture-constrained active shape models. *Image and Vision Computing*, 21(1):69–75, 2003.

[28] M. Zhao, C. Chen, S. Z. Li, and J. Bu. Subspace analysis and optimization for AAM based face alignment. In *Proc. 6th Int. Conf. on Automatic Face and Gesture Recognition, Seoul, Korea*, pages 290–295, 2004.