# Face Alignment by Explicit Shape Regression

Xudong Cao Yichen Wei Fang Wen Jian Sun
Microsoft Research Asia
{xudongca,yichenw,fangwen,jiansun}@microsoft.com

**Abstract.** We present a very efficient, highly accurate, "Explicit Shape Regression" approach for face alignment. Unlike previous regression-based approaches, we directly learn a vectorial regression function to infer the whole facial shape (a set of facial landmarks) from the image and *explicitly* minimize the alignment errors over the training data. The inherent shape constraint is naturally encoded into the regressor in a cascaded learning framework and applied from coarse to fine during the test, without using a fixed parametric shape model as in most previous methods.

To make the regression more effective and efficient, we design a two-level boosted regression, shape-indexed features and a correlation-based feature selection method. This combination enables us to learn accurate models from large training data in a short time (20 minutes for 2,000 training images), and run regression extremely fast in test (15 ms for a 87 landmarks shape). Experiments on challenging data show that our approach significantly outperforms the state-of-the-art in terms of both accuracy and efficiency.

## 1. Introduction

Face alignment or locating semantic *facial landmarks* such as eyes, nose, mouth and chin, is essential for tasks like face recognition, face tracking, face animation and 3D face modeling. With the explosive increase in personal and web photos nowadays, a fully automatic, highly efficient and robust face alignment method is in demand. Such requirements are still challenging for current approaches in unconstrained environments, due to large variations on facial appearance, illumination, and partial occlusions.

A face shape $S = [x_1, y_1, ..., x_{N_{\text{fp}}}, y_{N_{\text{fp}}}]^T$ consists of $N_{\text{fp}}$ facial landmarks. Given a face image, the goal of face alignment is to estimate a shape $S$ that is as close as possible to the true shape $\widehat{S}$, *i.e.*, minimizing

$$||S - \widehat{S}||_2. \quad (1)$$

The alignment error in Eq.(1) is usually used to guide the training and evaluate the performance. However, during testing, we cannot directly minimize it as $\widehat{S}$ is unknown. According to how $S$ is estimated, most alignment approaches can be classified into two categories: *optimization-based* and *regression-based*.

*Optimization-based* methods minimize another error function that is correlated to (1) instead. Such methods depend on the goodness of the error function and whether it can be optimized well. For example, the AAM approach [13, 16, 17, 3] reconstructs the entire face using an appearance model and estimates the shape by minimizing the texture residual. Because the learned appearance models have limited expressive power to capture complex and subtle face image variations in pose, expression, and illumination, it may not work well on unseen faces. It is also well known that AAM is sensitive to the initialization due to the gradient descent optimization.

*Regression-based* methods learn a regression function that directly maps image appearance to the target output. The complex variations are learnt from large training data and testing is usually efficient. However, previous such methods [6, 19, 7, 16, 17] have certain drawbacks in attaining the goal of minimizing Eq. (1). Approaches in [7, 16, 17] rely on a parametric model (*e.g.*, AAM) and minimize model parameter errors in the training. This is indirect and sub-optimal because smaller parameter errors are not necessarily equivalent to smaller alignment errors. Approaches in [6, 19] learn regressors for individual landmarks, effectively using (1) as their loss functions. However, because only local image patches are used in training and appearance correlation between landmarks is not exploited, such learned regressors are usually weak and cannot handle large pose variation and partial occlusion.

We notice that the *shape constraint* is essential in all methods. Only a few salient landmarks (*e.g.*, eye centers, mouth corners) can be reliably characterized by their image appearances. Many other non-salient landmarks (*e.g.*, points along face contour) need help from the shape constraint - the correlation between landmarks. Most previous works use a parametric shape model to enforce such a constraint, such as PCA model in AAM [3, 13] and ASM [4, 6].

Despite of the success of parametric shape models, the model flexibility (*e.g.*, PCA dimension) is often heuristical-

ly determined. Furthermore, using a fixed shape model in an iterative alignment process (as most methods do) may also be suboptimal. For example, in initial stages (the shape is far from the true target), it is favorable to use a restricted model for fast convergence and better regularization; in late stages (the shape has been roughly aligned), we may want to use a more flexible shape model with more subtle variations for refinement. To our knowledge, adapting such shape model flexibility is rarely exploited in the literature.

In this paper, we present a novel regression-based approach without using any parametric shape models. The regressor is trained by explicitly minimizing the alignment error over training data in a holistic manner - all facial landmarks are regressed jointly in a vectorial output. Our regressor realizes the shape constraint in an non-parametric manner: *the regressed shape is always a linear combination of all training shapes*. Also, using features across the image for all landmarks is more discriminative than using only local patches for individual landmarks. These properties enable us to learn a flexible model with strong expressive power from large training data. We call our approach "Explicit Shape Regression".

Jointly regressing the entire shape is challenging in the presence of large image appearance variations. We design a boosted regressor to *progressively* infer the shape - the early regressors handle large shape variations and guarantee robustness, while the later regressors handle small shape variations and ensure accuracy. Thus, the shape constraint is adaptively enforced from coarse to fine, in an automatic manner. This is illustrated in Figure 1 and elaborated in Section 2.2.

In the explicit shape regression framework, we further design *a two-level boosted regression*, effective *shape-indexed features*, and a fast *correlation-based feature selection method* so that: 1) we can quickly learn accurate models from large training data (20 mins on 2,000 training samples); 2) the resulting regressor is extremely efficient in the test (15 ms for 87 facial landmarks). We show superior results on several challenging datasets.

## 2. Face Alignment by Shape Regression

In this section, we introduce our basic shape regression framework and how to fit it to the face alignment problem.

We use boosted regression [9, 8] to combine $T$ weak regressors $(R^1, ...R^t, ..., R^T)$ in an additive manner. Given a facial image $I$ and an initial[1] face shape $S^0$, each regressor computes a shape increment $\delta S$ from image features and then updates the face shape, in a cascaded manner:

$$S^t = S^{t-1} + R^t(I, S^{t-1}), \quad t = 1, ..., T, \qquad (2)$$

where the $t$th weak regressor $R^t$ updates the previous shape $S^{t-1}$ to the new shape $S^t$.

Notice that the regressor $R^t$ depends on both image $I$ and previous estimated shape $S^{t-1}$. As will be described later, we use *shape indexed (image) features* that are relative to previous shape to learn each $R^t$. Such features can greatly improve the boosted regression by achieving better geometric invariance. The similar idea is also used in [7].

Given $N$ training examples $\{(I_i, \widehat{S}_i)\}_{i=1}^N$, the regressors $(R^1, ...R^t, ..., R^T)$ are sequentially learnt until the training error no longer decreases. Each regressor $R^t$ is learnt by explicitly minimizing the sum of alignment errors (1) till then,

$$R^t = \arg\min_R \sum_{i=1}^{N} ||\widehat{S}_i - (S_i^{t-1} + R(I_i, S_i^{t-1}))||, \qquad (3)$$

where $S_i^{t-1}$ is the estimated shape in previous stage.

### 2.1. Two-level cascaded regression

Previous methods use simple weak regressors such as a decision stump [6] or a fern [7] in a similar boosted regression manner. However, in our early experiments, we found that such regressors are too weak and result in very slow convergence in training and poor performance in the testing. We conjecture this is due to the extraordinary difficulty of the problem: regressing the entire shape (as large as dozens of landmarks) is too difficult, in the presence of large image appearance variations and rough shape initializations. A simple weak regressor can only decrease the error very little and cannot generalize well.

It is crucial to learn a good weak regressor that can rapidly reduce the error. We propose to learn each weak regressor $R^t$ by a second level boosted regression, *i.e.*, $R^t = (r^1, ...r^k, ..., r^K)$. The problem is similar as in (2)(3), but the key difference is that the shape-indexed image features are fixed in the second level, *i.e.*, they are indexed only relative to $S^{t-1}$ and no longer change when those $r$'s are learnt[2]. This is important, as each $r$ is rather weak and allowing feature indexing to change frequently is unstable. Also the fixed features can lead to much faster training, as will be described later. In our experiments, we found using two-level boosted regression is more accurate than one level under the same training effort, *e.g.*, $T = 10, K = 500$ is better than one level of $T = 5000$, as shown in Table 3.

Below we describe how to learn each weak regressor $r^k$. For notation clarity, we call it a *primitive regressor* and drop the index $k$.

### 2.2. Primitive regressor

We use a *fern* as our primitive regressor $r$. The fern was firstly introduced for classification [15] and later used for

[1]The initial shape can be simply a mean shape. More details of initialization are discussed in Section 3.

[2]Otherwise this degenerates to a one level boosted regression.

regression [7]. A fern is a composition of $F$ (5 in our implementation) features and thresholds that divide the feature space (and all training samples) into $2^F$ bins. Each bin $b$ is associated with a regression output $\delta S_b$ that minimizes the alignment error of training samples $\Omega_b$ falling into the bin:

$$\delta S_b = \arg\min_{\delta S} \sum_{i \in \Omega_b} ||\widehat{S}_i - (S_i + \delta S)||, \tag{4}$$

where $S_i$ denotes the estimated shape in the previous step. The solution for (4) is the mean of shape differences,

$$\delta S_b = \frac{\sum_{i \in \Omega_b} (\widehat{S}_i - S_i)}{|\Omega_b|}. \tag{5}$$

To overcome over-fitting in the case of insufficient training data in the bin, a shrinkage is performed [9, 15] as

$$\delta S_b = \frac{1}{1 + \beta/|\Omega_b|} \frac{\sum_{i \in \Omega_b} (\widehat{S}_i - S_i)}{|\Omega_b|}, \tag{6}$$

where $\beta$ is a free shrinkage parameter. When the bin has sufficient training samples, $\beta$ makes little effect; otherwise, it adaptively reduces the estimation.

**Non-parametric shape constraint** By learning a vector regressor and explicitly minimizing the shape alignment error (1), the correlation between the shape coordinates is preserved. Because each shape update is additive as in Eq. (2), and each shape increment is the linear combination of certain training shapes $\{\widehat{S}_i\}$ as in Eq. (5) or (6), it is easy to see that the final regressed shape $S$ can be expressed as the initial shape $S^0$ plus the linear combination of all training shapes:

$$S = S^0 + \sum_{i=1}^{N} w_i \widehat{S}_i. \tag{7}$$

Therefore, as long as the initial shape $S^0$ satisfies the shape constraint, *the regressed shape is always constrained to reside in the linear subspace constructed by all training shapes*. In fact, any intermediate shape in the regression also satisfies the constraint. Compare to the pre-fixed PCA shape model, the non-parametric shape constraint is adaptively determined during the learning.

To illustrate the adaptive shape constraint, we perform PCA on all the shape increments stored in all primitive fern regressors ($2^F \times K$ in total) for each first level regressor $R^t$. As shown in Figure 1, the intrinsic dimension (by retaining $95\%$ energy) of such shape spaces increases during the learning. Therefore, *the shape constraint is automatically encoded in the regressors in a coarse to fine manner*. Figure 1 also shows the first three principal components of the learnt shape increments (plus a mean shape) in first and final stage. As shown in Figure 1(c)(d), the shape updates

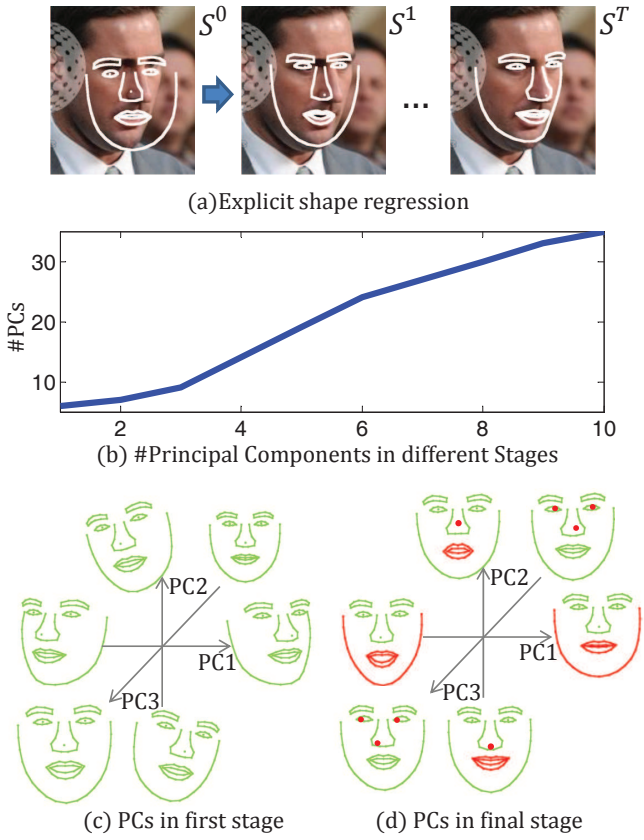


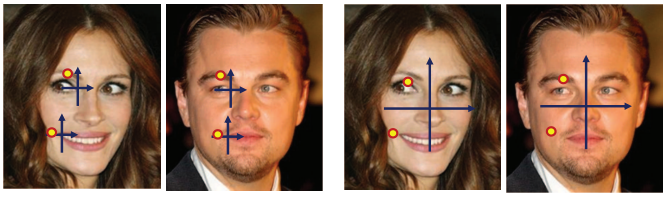

Figure 1. Shape constraint is preserved and adaptively learned in a coarse to fine manner in our boosted regressor. (a) The shape is progressively refined by the shape increments learnt by the boosted regressors in different stages. (b) Intrinsic dimensions of learnt shape increments in a 10-stage boosted regressor, using 87 facial landmarks. (c)(d) The first three principal components (PCs) of shape increments in the first and final stage, respectively.

learned by the first stage regressor are dominated by global rough shape changes such as yaw, roll and scaling. In contrast, the shape updates of the final stage regressor are dominated by the subtle variations such as face contour, and motions in the mouth, nose and eyes.

### 2.3. Shape-indexed (image) features

For efficient regression, we use simple pixel-difference features, *i.e.*, the intensity difference of two pixels in the image. Such features are extremely cheap to compute and powerful enough given sufficient training data [15, 18, 7]. A pixel is indexed relative to the currently estimated shape rather than the original image coordinates. The similar idea can also be found in [7]. This achieves better geometric invariance and in turn leads to easier regression problems and faster convergence in boosted learning.

To achieve feature invariance against face scales and ro-



(a) (b)

Figure 2. Pixels indexed by the same local coordinates have the same semantic meaning (a), but pixels indexed by the same global coordinates have different semantic meanings due to the face shape variation (b).

tations, we first compute a similarity transform to normalize the current shape to a mean shape, which is estimated by least squares fitting of all facial landmarks. Previous works [6, 19, 16] need to transform the image correspondingly to compute Harr like features. In our case, we instead transform the pixel coordinates back to the original image to compute pixel-difference features, which is much more efficient.

A simple way to index a pixel is to use its *global coordinates* $(x, y)$ in the canonical shape. This is good for simple shapes like ellipses, but it is insufficient for non-rigid face shapes. Because most useful features are distributed around salient landmarks such as eyes, nose and mouth (*e.g.*, a good pixel difference feature could be "eye center is darker than nose tip" or "two eye centers are similar"), and landmarks locations can vary for different faces 3d-poses/expressions/identities. In this work, we suggest to index a pixel by its *local coordinates* $(\delta x, \delta y)$ with respect to its nearest landmark. As Figure 2 shows, such indexing holds invariance against the variations mentioned above and make the algorithm robust.

For each weak regressor $R^t$ in the first level, we randomly sample[3] $P$ pixels. In total $P^2$ pixel-difference features are generated. Now, the new challenge is how to quickly select effective features from such a large pool.

### 2.4. Correlation-based feature selection

To form a good fern regressor, $F$ out of $P^2$ features are selected. Usually, this is done by randomly generating a pool of ferns and selecting the one with minimum regression error as in (4) [15, 7]. We denote this method as *n-Best*, where $n$ is the size of the pool. Due to the combinatorial explosion, it is unfeasible to evaluate (4) for all of the compositional features. As illustrated in Table 4, the error is only slightly reduced by increasing $n$ from 1 to 1024, but the training time is significantly longer.

To better explore the huge feature space in a short time and generate good candidate ferns, we exploit the *correlation* between features and the regression target. The target is vectorial delta shape which is the difference between the groundtruth shape and current estimated shape. We expect that a good fern should satisfy two properties: (1) each feature in the fern should be highly discriminative to the regression target; (2) correlation between features should be low so they are complementary when composed.

To find features satisfying such properties, we propose a correlation-based feature selection method:

1. Project the regression target(vectorial delta shape) to a random direction to produce a scalar.
2. Among $P^2$ features, select a feature with highest correlation to the scalar.
3. Repeat steps 1. and 2. $F$ times to obtain $F$ features.
4. Construct a fern by $F$ features with random thresholds.

The random projection serves two purposes: it can preserve proximity [2] such that the features correlated to the projection are also discriminative to delta shape; the multiple projections have low correlations with a high probability and the selected features are likely to be complementary. As shown in Table 4, the proposed correlation based method can select good features in a short time and is much better than the n-Best method.

**Fast correlation computation** At first glance, we need to compute the correlation of $P^2$ features with a scalar in step 2, which is still expensive. Fortunately the computational complexity can be reduced from $O(P^2)$ to $O(P)$ by the following facts: The correlation between a scalar $y$ and a pixel-difference feature $(f_i - f_j)$ can be represented as the function of three terms: **cov**$(f_i, f_j)$, **cov**$(y, f_i)$, and **cov**$(y, f_j)$. As all shape indexed pixels are fixed for the first-level regressor $R^t$, the first term **cov**$(f_i, f_j)$ can be reused for all primitive regressors under the same $R^t$. Therefore, the feature correlation computation time is reduced to that of computing the covariances between a scalar and $P$ different pixels, which is $O(P)$.

## 3. Implementation details

We discuss more implementation details, including the shape initialization in training and testing, parameter setting and running performance.

**Training data augmentation** Each training sample consists of a training image, an initial shape and a ground truth shape. To achieve better generalization ability, we augment the training data by randomly sampling multiple (20 in our implementation) shapes of other annotated images as the initial shapes of each training image. This is found to be very effective in obtaining robustness against large pose variation and rough initial shapes during the testing.

**Multiple initializations in testing** The regressor can give reasonable results with different initial shapes for a test

[3] We left for future work how to exploit a prior distribution that favors salient regions (*e.g.*, eyes or mouth) for more effective feature generation.

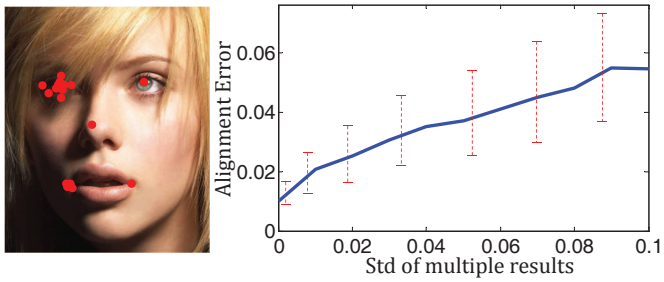



Figure 3. Left: results of 5 facial landmarks from multiple runs with different initial shapes. The distribution indicates the estimation confidence: left eye and left mouth corner estimations are widely scattered and less stable, due to the local appearance noises. Right: the average alignment error increases as the standard deviation of multiple results increases.

image and the distribution of multiple results indicates the confidence of estimation. As shown in Figure 3, when multiple landmark estimations are tightly clustered, the result is accurate, and vice versa. In the test, we run the regressor several times (5 in our implementation) and take the median result[4] as the final estimation. Each time the initial shape is randomly sampled from the training shapes. This further improves the accuracy.

**Running time performance** Table 1 summarizes the computational time of training (with $2,000$ training images) and testing for different number of landmarks. Our training is very efficient due to the fast feature selection method. It takes minutes with $40,000$ training samples (20 initial shapes per image), The shape regression in the test is extremely efficient because most computation is pixel comparison, table look up and vector addition. It takes only $15$ ms for $87$ landmarks (3 ms $\times$ 5 initializations).

| Landmarks | 5 | 29 | 87 |
|---|---|---|---|
| Training (mins) | 5 | 10 | 21 |
| Testing (ms) | 0.32 | 0.91 | 2.9 |

Table 1. Training and testing times of our approach, measured on an Intel Core i7 2.93GHz CPU with C++ implementation.

**Parameter settings** The number of features in a fern $F$ and the shrinkage parameter $\beta$ adjust the trade off between fitting power in training and generalization ability in testing. They are set as $F = 5$, $\beta = 1000$ by cross validation.

Algorithm accuracy consistently increases as the number of stages in the two-level boosted regression ($T$,$K$) and number of candidate features $P^2$ increases. Such parameters are empirically chosen as $T = 10, K = 500, P = 400$ for a good tradeoff between computational cost and accuracy.

[4]The median operation is performed on x and y coordinates of all landmarks individually. Although this may violate the shape constraint mentioned before, the resulting median shape is mostly correct as in most cases the multiple results are tightly clustered. We found such a simple median based fusion is comparable to more sophisticated strategies such as weighted combination of input shapes.

## 4. Experiments

The experiments are performed in two parts. The first part compares our approach with previous works. The second part validates the proposed approach and presents some interesting discussions.

We briefly introduce the three datasets used in the experiments. They present different challenges, due to different numbers of annotated landmarks and image variations.

**BioID**[11] dataset is widely used by previous methods. It consists of 1,521 near frontal face images captured in a lab environment, and is therefore less challenging. We report our result on it for completeness.

**LFPW** (Labeled Face Parts in the Wild) was created in [1]. Its images are downloaded from internet and contain large variations in pose, illumination, expression and occlusion. It is intended to test the face alignment methods in unconstraint conditions. This dataset shares only web image URLs, but some URLs are no longer valid. We only downloaded 812 of the 1,100 training images and 249 of the 300 test images. To acquire enough training data, we augment the training images to 2,000 in the same way as in [1] and use the available test images.

**LFW87** was created in [12]. The images mainly come from the LFW(Labeled Face in the Wild) dataset[10], which is acquired from wild conditions and is widely used in face recognition. In addition, it has 87 annotated landmarks, much more than that in BioID and LFPW, therefore, the performance of an algorithm relies more on its shape constraint. We use the same 4,002 training and 1,716 testing images as in [12].

### 4.1. Comparison with previous work

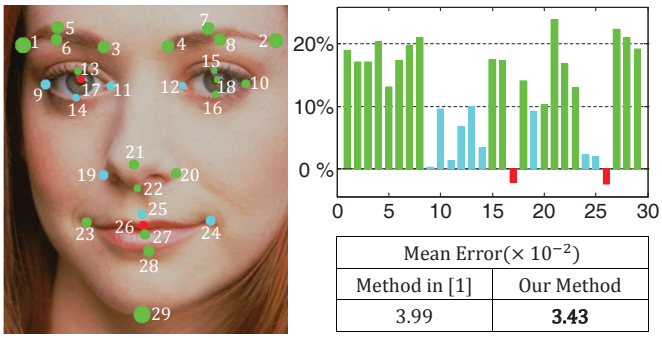
For comparisons, we use the alignment error in Eq.(1) as the evaluation metric. To make it invariant to face size, the error is not in pixels but normalized by the distance between the two pupils, similar to most previous works.

The following comparison shows that our approach outperforms the state of the art methods in both accuracy and efficiency, especially on the challenging LFPW and LFW87 datasets. Figure 7, 8, and 9 show our results on challenging examples with large variations in pose, expression, illumination and occlusion from the three datasets.

**Comparison to [1] on LFPW** The consensus exemplar approach [1] is one of the state of the art methods. It was the best on BioID when published, and obtained good results on LFPW.

Comparison in Figure 4 shows that most landmarks estimated by our approach are more than $10\%$ accurate[5]

[5]The relative improvement is the ratio between the error reduction by




| Mean Error($\times 10^{-2}$) | |
|---|---|
| Method in [1] | Our Method |
| 3.99 | **3.43** |

Figure 4. Results on the LFPW dataset. Left: 29 facial landmarks. The circle radius is the average error of our approach. Point color represents relative accuracy improvement over [1]. Green: more than $10\%$ more accurate. Cyan: $0\%$ to $10\%$ more accurate. Red: less accurate. Right top: relative accuracy improvement of all landmarks over [1]. Right bottom: average error of all landmarks.

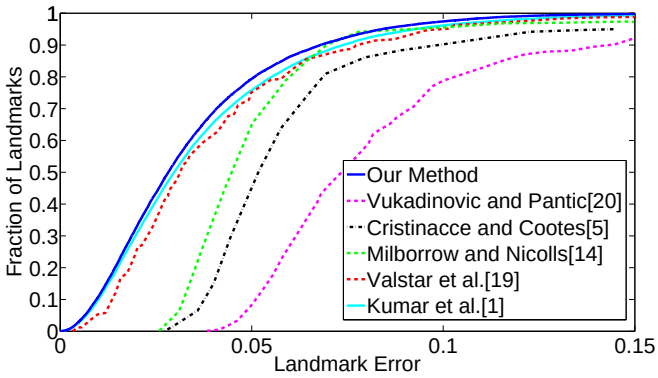



Figure 5. Cumulative error curves on the BioID dataset. For comparison with previous results, only 17 landmarks are used [5]. As our model is trained on LFPW images, for those landmarks with different definitions between the two datasets, a fixed offset is applied in the same way as in [1].

than [1] and our overall error is smaller.

In addition, our method is *thousands of times faster*. It takes around 5ms per image ($0.91 \times 5$ initializations for 29 landmarks). The method in [1] uses expensive local landmark detectors (SIFT+SVM) and it takes more than 10 seconds[6] to run 29 detectors over the entire image.

**Comparison to [12] on LFW87** Liang et al.[12] train a set of direction classifiers for pre-defined facial components to guide the ASM search direction. Their algorithm outperform previous ASM and AAM based works by a large margin.

We use the same RMSE (Root Mean Square Error) in [12] as the evaluation metric. Table 2 shows our method is significantly better. For the strict error threshold (5 pixels), the error rate is reduced nearly by half, from $25.3\%$ to $13.9\%$. The superior performance on a large number of landmarks verifies the effectiveness of proposed holistic shape regression and the encoded adaptive shape constraint.

| RMSE | $<5$ pixels | $<7.5$ pixels | $<10$ pixels |
|---|---|---|---|
| Method in [12] | 74.7% | 93.5% | 97.8% |
| Our Method | **86.1**% | **95.2**% | **98.2**% |

Table 2. Percentages of test images with RMSE(Root Mean Square Error) less than given thresholds on the LFW87 dataset.

**Comparison to previous methods on BioID** Our model is trained on augmented LPFW training set and tested on the entire BioID dataset.

Figure 5 compares our method with previous methods [20, 5, 14, 19, 1]. Our result is the best but the improvement is marginal. We believe this is because the performance on BioID is nearly maximized due to its simplicity. Note that our method is thousands of times faster than the second best method in [1].

our method and the original error.

[6]It is discussed in [1] as: "The localizer requires less than one second per fiducial on an Intel Core i7 3.06GHz machine". We conjecture that it takes more than 10 seconds to locate 29 landmarks.

## 4.2. Algorithm validation and discussions

We verify the effectiveness of different components of the proposed approach. Such experiments are performed on the augmented LPFW dataset, using 1,500 images for training and 500 for testing. Parameters are fixed as in Section 3, unless otherwise noted.

**Two-level cascaded regression** As discussed in Section 2, the first level regression exploits shape indexed features to obtain geometric invariance and decompose the original difficult problem into easier sub-tasks. The second level regression inhibits such features to avoid instability.

Different tradeoffs between two-level cascaded regression are presented in Table 3, using the same number of primitive regressors. On one extreme, not using shape indexed features ($T = 1, K = 5000$) is clearly the worst. On the other extreme, using such features for every primitive regressor ($T = 5000, K = 1$) also has poor generalization ability in the test. The optimal tradeoff ($T = 10, K = 500$) is found in between via cross validation.

| #stages in level 1 (T) | 1 | 5 | 10 | 100 | 5000 |
|---|---|---|---|---|---|
| #stages in level 2 (K) | 5000 | 1000 | 500 | 50 | 1 |
| Mean Error ($\times 10^{-2}$) | 15 | 6.2 | **3.3** | 4.5 | 5.2 |

Table 3. Tradeoffs between two levels cascaded regression.

**Shape indexed feature** We compare the global and local methods of shape indexed features. The mean error of local index method is 0.033, which is much smaller than the mean error of global index method 0.059. The superior accuracy supports the proposed local index method.

**Feature selection** The proposed correlation based feature selection method (CBFS) is compared with the commonly used *n-best* method [15, 7] in Table 4. CBFS can select good features rapidly and this is crucial to learn good models from large training data.

| | 1-Best | 32-Best | 1024-Best | CBFS |
|---|---|---|---|---|
| Error ($\times 10^{-2}$) | 5.01 | 4.92 | 4.83 | **3.32** |
| Time (s) | 0.1 | 3.0 | 100.3 | 0.12 |

Table 4. Comparison between correlation based feature selection(CBFS) method and n-Best feature selection methods. The training time is for one primitive regressor.

**Feature range** The *range* of a feature is the distance between the pair of pixels normalized by the distance between the two pupils. Figure 6 shows the average ranges of selected features in the 10 stages of the first level regressors. As observed, the selected features are adaptive to the different regression tasks. At first, long range features (*e.g.*, one pixel on the mouth and the other on the nose) are often selected for rough shape adjustment. Later, short range features (*e.g.*, pixels around the eye center) are often selected for fine tuning.

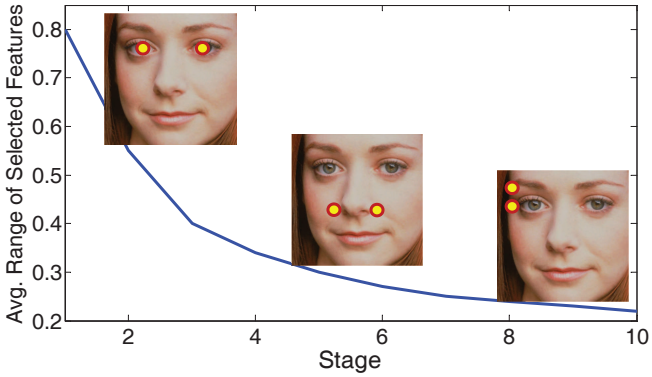



Figure 6. Average ranges of selected features in different stages. In stage 1, 5 and 10, an exemplar feature (a pixel pair) is displayed on an image.

## 5. Discussion and Conclusion

We have presented the explicit shape regression method for face alignment. By jointly regressing the entire shape and minimizing the alignment error, the shape constraint is automatically encoded. The resulting method is highly accurate, efficient, and can be used in real time applications such as face tracking. The explicit shape regression framework can also be applied to other problems like articulated object pose estimation and anatomic structure segmentation in medical images.

## References


[1] P. Belhumeur, D. Jacobs, D. Kriegman, and N. Kumar. Localizing parts of faces using a concensus of exemplars. In *CVPR*, 2011.

[2] E. Bingham and H. Mannila. Random projection in dimensionality reduction: Applications to image and text data. In *KDD*, 2001.

[3] T. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *ECCV*, 1998.

[4] T. Cootes and C. J. Taylor. Active shape models. In *BMVC*, 1992.

[5] D. Cristinacce and T. Cootes. Feature detection and tracking with constrained local models. In *BMVC*, 2006.

[6] D. Cristinacce and T. Cootes. Boosted regression active shape models. In *BMVC*, 2007.

[7] P. Dollar, P. Welinder, and P. Perona. Cascaded pose regression. In *CVPR*, 2010.

[8] N. Duffy and D. P. Helmbold. Boosting methods for regression. *Machine Learning*, 47(2-3):153–200, 2002.

[9] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

[10] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[11] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz. Robust face detection using the hausdorff distance. pages 90–95. Springer, 2001.

[12] L. Liang, R. Xiao, F. Wen, and J. Sun. Face alignment via component-based discriminative search. In *ECCV*, 2008.

[13] I. Matthews and S. Baker. Active appearance models revisited. *IJCV*, 60:135–164, 2004.

[14] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. In *ECCV*, 2008.

[15] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *PAMI*, 2010.

[16] C. T. P. Sauer, T. Cootes. Accurate regression procedures for active appearance models. In *BMVC*, 2011.

[17] J. Saragih and R. Goecke. A nonlinear discriminative approach to aam fitting. In *ICCV*, 2007.

[18] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.

[19] M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using boosted regression and graph models. In *CVPR*, 2010.

[20] D. Vukadinovic and M. Pantic. Fully automatic facial feature point detection using gabor feature based boosted classifiers. *Int. Conf. on Systems, Man and Cybernetics*, 2:1692–1698, 2005.

Figure 7. Selected results from LFPW.

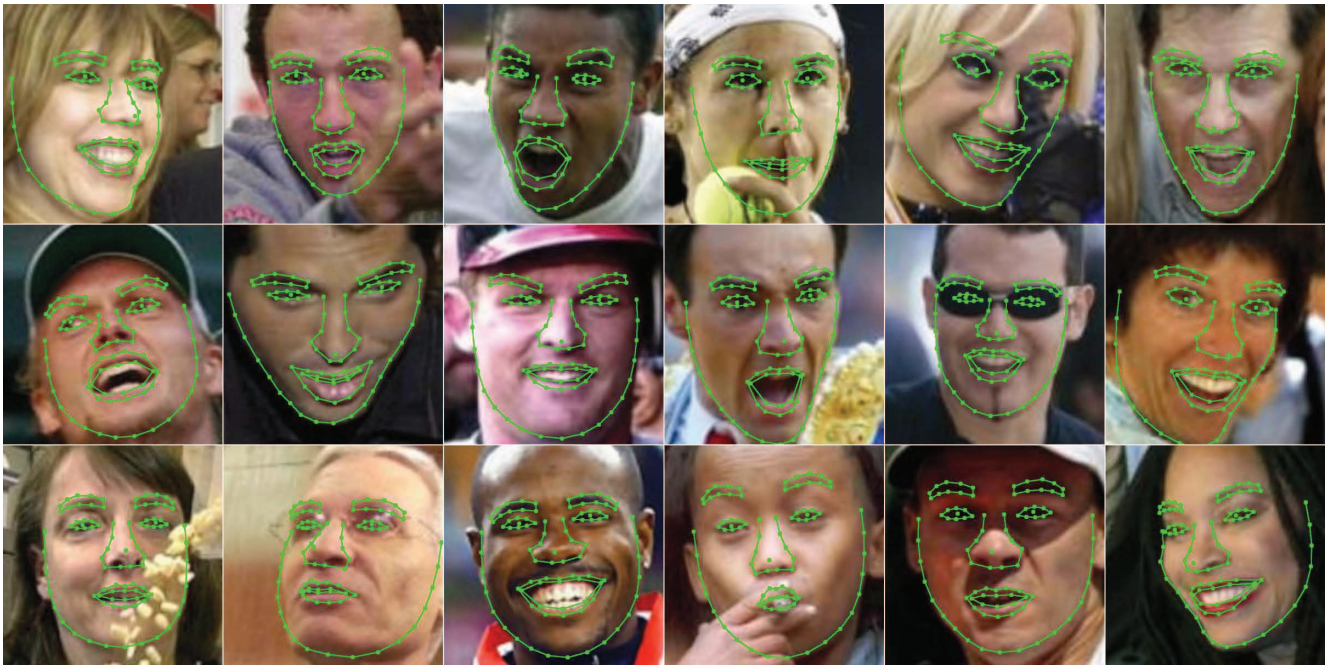


Figure 8. Selected results from LFW87.

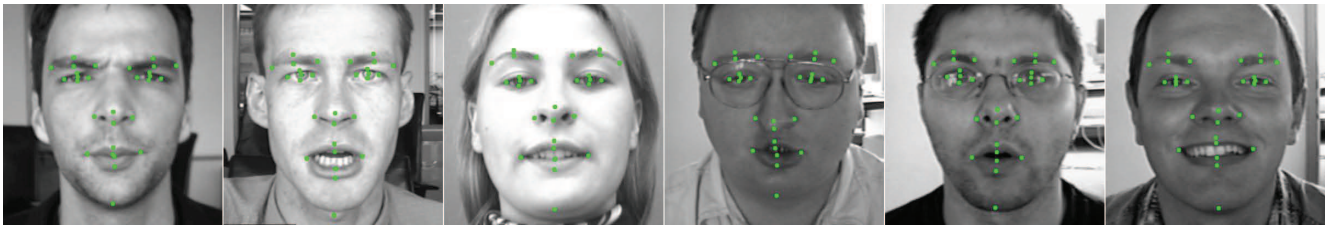


Figure 9. Selected results from BioID.