Robust and Accurate Shape Model Fitting using Random Forest Regression Voting

T.F.Cootes, M.C.Ionita, C.Lindner and P.Sauer

The University of Manchester, UK

Abstract. A widely used approach for locating points on deformable objects is to generate feature response images for each point, then to fit a shape model to the response images. We demonstrate that Random Forest regression can be used to generate high quality response images quickly. Rather than using a generative or a discriminative model to evaluate each pixel, a regressor is used to cast votes for the optimal position. We show this leads to fast and accurate matching when combined with a statistical shape model. We evaluate the technique in detail, and compare with a range of commonly used alternatives on several different datasets. We show that the random forest regression method is significantly faster and more accurate than equivalent discriminative, or boosted regression based methods trained on the same data.

1 Introduction

The ability to accurately detect features of deformable models is important for a wide range of algorithms and applications. A widely used approach is to use a statistical shape model to regularise the output of independent feature detectors trained to locate each model point. Examples include Active Shape Models (ASMs) [1, 2], Pictoral Structures [3] and Constrained Local Models (CLMs) [4, 5], though there are many others.

The task of the feature detector is to compute a (pseudo) probability that the target point occurs at a particular position, given the image information $p(\mathbf{x}|I)$[1]. Local peaks in this correspond to candidate positions (eg in ASMs) or the probabilities for each point are combined with the shape model information to find the best overall match (eg CLMs and Pictoral Structures). A wide variety of feature detectors have been used in such frameworks which can be broadly classified into three types:

Generative in which generative models are used, so $p(\mathbf{x}|I) \propto p(I|\mathbf{x})$.
Discriminative in which classifiers are trained to estimate $p(\mathbf{x}|I)$ directly.
Regression-Voting in which $p(\mathbf{x}|I)$ is estimated from accumulating votes for the position of the point given information in nearby regions.

[1] Where techniques return a quality of fit measure, C, we assume these can be converted to a pseudo-probability with a suitable transformation.

Although there has been a great deal of work matching deformable models using the first two approaches, the Regression-Voting approach has only recently begun to be explored in this context.

Recent work on Hough Forests [6] has shown that objects can be effectively located by pooling votes from Random Forest regressors. Valstar *et al.*[7] have shown that facial feature points can be accurately located using a similar approach, but using kernel SVM based regressors.

In the following we show that Regression-Voting is a powerful technique, and that using Random Forest voting leads to fast, accurate and robust results when used in the Constrained Local Model framework.

We demonstrate the performance on a range of datasets, both images of faces and radiographs of the hand, and show the effect of different choices of parameters. We show that voting using Random Forest regression outperforms classification based methods and boosted regression when trained on the same data, leading to state of the art performance.

1.1 Related Work

Shape Model Matching: There is a wide range of literature on matching statistical shape models to images, starting with Active Shape Models [1] in which the shape model is fit to the results of searching around each model point with a suitably trained detector. Active Appearance Models (AAMs) [8] match combined models of shape and texture using an efficient parameter update scheme. Pictoral Structures [3] introduced an efficient method of matching part-based models to images, in which shape is encoded in the geometric relationships between pairs of parts. Constrained Local Models [4, 5] build on a framework in which response images are computed estimating the quality of fit of each model point at each point in the target image, then a shape model is matched to the data, selecting the overall best combination of points.

Belhumeur *et al.* [9] have shown impressive facial feature detection results using sliding window detectors (SVM classifiers trained on SIFT features) combined with a RANSAC approach to selecting good combinations of feature points.

Regression based matching: One of the earliest examples of regression based matching techniques was the work of Covell [10] who used linear regression to predict the positions of points on the face. The original AAM [11] algorithm used linear regression to predict the updates to model parameters. Non-linear extensions include the use of Boosted Regression [12, 13] and Random Forest Regression [14]. The Shape Regression Machine [15] uses boosted regression to predict shape model parameters directly from the image (rather than the iterative approach used in AAMs). Zimmerman and Matas [16] used sets of linear predictors to estimate positions locally, an approach used for facial feature tracking by Ong and Bowden [17]. Dollár *et al.*[18] use sequences of Random Fern predictors to estimate the pose of an object or part.

Regression based voting: Since the introduction of the Generalised Hough Transform [19] voting based methods have been shown to be effective for locating

shapes in images, and there have been many variants. For instance, the Implicit Shape Model [20] uses local patches located on an object to vote for the object position, and Poselets [21] match patches to detect human body parts.

Hough Forests [6] use Random Forest regression from multiple sub-regions to vote for the position of an object. This includes an innovative training approach, in which regression and classification training are interleaved to deal with arbitrary backgrounds and where only votes believed to be coming from regions inside the object are counted.

Valstar *et al.* [7] showed that facial feature points can be accurately located using kernel SVM based regressors to vote for each point position combined with pair-wise constraints on feature positions. We show that using Random Forest regression, together with a global shape model, leads to significantly faster and more accurate results.

Girshick *et al.* [22] showed that Random Forests can be used to vote for the position of joint centres when matching a human body model to a depth image. Criminisi *et al.*[26] use Random Forest regression to vote for the positions of the sides of bounding boxes around organs in CT images.

Recently Dantone *et al.* [23] have used conditional random forests to find facial features. Our method differs from this in that we use an explicit shape model to find the best combination of points.

2 Constrained Local Models

The CLM is a method for matching the points of a statistical shape model to an image. Here we summarise the key points of the approach - for details see [4, 5]. We use a linear model of shape variation [1] which represents the position of each point using

$$\mathbf{x}_i = T(\bar{\mathbf{x}}_i + \mathbf{P}_i\mathbf{b}; \mathbf{t}) \tag{1}$$

where $\bar{\mathbf{x}}_i$ is the mean position of the point in a suitable reference frame, $\mathbf{P}_i$ a set of modes of variation and $T(\cdot; \mathbf{t})$ applies a global transformation (eg. similarity) with parameters $\mathbf{t}$.

To match the model to a new image, I, we seek the points, $\mathbf{x} = \{\mathbf{x}_i\}$, which optimise the overall quality of fit of the model to the image.

More formally, we seek parameters $\mathbf{p} = \{\mathbf{b}, \mathbf{t}\}$ which minimise

$$Q(\mathbf{p}) = -\log p(\mathbf{b}, \mathbf{t}|I) = -\log p(\mathbf{b}) - \alpha \sum_{i=1}^{N} \log p(\mathbf{x}_i|I) \tag{2}$$

where the scaling factor α is included to take account of the fact that the conditional probabilities for each point, $p(\mathbf{x}_i|I)$, are not strictly independent, and we have assumed that all poses are equally likely, so $p(\mathbf{b}, \mathbf{t}) = p(\mathbf{b})$.

Given an estimate of the scale and orientation, we can scan the target image to compute quality of fit $C_i(\mathbf{x}_i) = -\log p_i(\mathbf{x}_i|I)$. The objective function is then

$$Q(\mathbf{p}) = -\log p(\mathbf{b}) + \alpha \sum_{i=1}^{N} C_i(\mathbf{x}_i) \tag{3}$$

The first term encodes the shape constraints, the second the image matching information. The cost function $Q(\mathbf{p})$ can then be optimised either using a general purpose optimiser [4], or using the mean-shift approach advocated by Saragih *et al.*[5].

Examples of quality of fit functions, $C(\mathbf{x})$, for each point include using normalised correlation with a globally constrained patch model [4] or sliding window search with a range of classifiers [5, 9]. Figure 2 gives an example of a set of such fit functions.

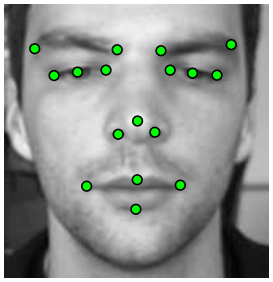

Fig. 1. 17 facial points used in experiments

Fig. 2. Superposition of vote accumulation images for 17 point face model. Note that there is a separate image for each point.

In this work we show that Random Forest regression can be used to produce effective cost maps, leading to fast, robust and accurate performance.

3 Voting with Random Forest Regressors

In the Regression-Voting approach, we evaluate a set of points in a grid over a region of interest. At each point, $\mathbf{z}$, a set of features, $\mathbf{f}(\mathbf{z})$, are sampled. A regressor, $\mathbf{d} = R(\mathbf{f}(\mathbf{z}))$, is trained to predict the most likely position of the target point relative to $\mathbf{z}$. The predictions are used to vote for the best position in an accumulator array V, so $V(\mathbf{z}+\mathbf{d}) \rightarrow V(\mathbf{z}+\mathbf{d}) + v$ where v expresses the degree of confidence in the prediction (Figure 3). After scanning the whole region of interest, V can be smoothed to allow for uncertainty in the predictions. For instance, [7] uses an SVM regressor, with each sample voting for one nearby point.

One advantage of the regression approach is that it avoids the somewhat arbitrary cut-off radius sometimes required when selecting positive and negative examples for the training. It also allows integration of evidence from regions which may not even overlap the target point.

Random Forests [27] have been shown to be effective in a wide range of classification and regression problems. They consist of a set of binary trees, each stochastically trained on random subsets of the data. Although any one tree may

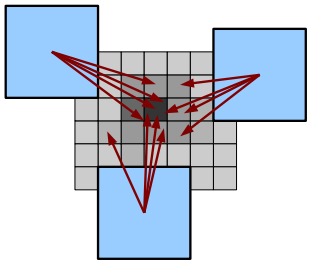

Fig. 3. During search each rectangular patch predicts (one or more) positions for the target point. Votes are accumulated in a grid.

be somewhat overtrained, the randomness in the training process encourages the trees to give independent estimates, which can be combined to achieve an accurate and robust result.

A natural extension of Regression-Voting is to use multiple (independent) regressors, or to have each regressor predict multiple positions. Both ideas are combined in Hough Forests [6] which use sets of random trees whose leaves store multiple training samples. Thus each sample produces multiple weighted votes, allowing for arbitrary distributions to be encoded. In related work, [22] and [26] produce votes in higher dimensional spaces (3D or 6D), but work directly with the vector votes rather than accumulator arrays.

In the following we use Random Forest regression, accumulating votes in a 2D array (Figure 3). A key advantage of decision trees is that each leaf can store arbitrary information derived from the training samples which arrived at that leaf, $\{\mathbf{d}_k\}$. For instance, this could be the mean, $\bar{\mathbf{d}}$, and covariance $\mathbf{S}_d$ of these samples, or the full set of samples.

When scanning the target region, a range of styles of voting can be used:

1. A single, unit vote per tree at the mean offset.
2. A single, weighted vote per tree, using a weight of $|\mathbf{S}_d|^{-0.5}$. This penalises uncertain predictions.
3. A Gaussian spread of votes, $N(\bar{\mathbf{d}},\mathbf{S}_d)$.
4. Multiple votes from the training samples [6].

In the experiments below we compare these different approaches, and show that using a single vote per tree gives the best performance, both in terms of accuracy and speed in our applications.

If the number of votes cast for the point to be at pixel $\mathbf{x}$ is $V(\mathbf{x})$, then we set the cost map image to be given as $C(\mathbf{x}) = -\log(\max(V(\mathbf{x}), v_0))$, where $v_0 > 0$ introduces robustness to occlusion by allowing points to have a non-zero probability of occuring anywhere.

An advantage of using regression voting, rather then classification, is that good results can be obtained by evaluating on a sparse grid, rather than at every pixel. Sampling every third pixel, for instance, speeds up the process by a factor of nine, with minimal loss of accuracy (see results below).

3.1 Training

We train the models from sets of images, each of which is annotated with the feature points of interest on the object, $\mathbf{x}$. A statistical shape model is trained by applying PCA to the aligned shapes [1]. The model is scaled so that the width of the bounding box of the mean shape is a given value, w_{ref} (typically in the range 50-150 pixels).

The shape model is used to assess the global pose, $\mathbf{t}$, of the object in each image, by minimising $|T(\bar{\mathbf{x}};\mathbf{t})-\mathbf{x}|^2$. Each image is resampled into a standardized reference frame by applying the inverse of the pose, $I_{ref}(i,j) = I(T(i,j;\mathbf{t}))$.

To train the detector for a single feature point we generate samples by extracting features $\mathbf{f}_j$ at a set of random displacements $\mathbf{d}_j$ from the true position in the reference frame, $T^{-1}(\mathbf{x}_i;\mathbf{t})$, where $\mathbf{x}_i$ is the position of the point in the image. Displacements are drawn from a flat distribution in the range $[-d_{max}, +d_{max}]$ in x and y. To allow for inaccurate initial estimates of the pose, we repeat this process with random perturbations in scale and orientation of the estimate of the pose. We then train a set of randomised decision trees [27] on the N_s pairs $\{\mathbf{f}_j, \mathbf{d}_j\}$. To train one tree we take a bootstrap sample (drawing N_s examples with replacement) of the training set, then use a standard, greedy approach to construct the tree, recursively splitting the data at each node. Given the samples at a particular node, we seek to select a feature and threshold to best split the data into two compact groups. Let f_i be the value of one feature associated with sample i. The best threshold, t, for this feature at this node is that which minimises

$$G_T(t) = G(\{\mathbf{d}_i : f_i < t\}) + G(\{\mathbf{d}_i : f_i \geq t\}) \tag{4}$$

where $G(S)$ is a function evaluating the set of vectors S. In the following we use an entropy measure $G(\{\mathbf{d}_i\}) = Nlog|\Sigma|$ where Σ is the covariance matrix of the N samples.

In the experiments below we use Haar-like features [28] sampled in a box around the current point, as they have been found to be effective for a range of applications and can be calculated efficiently from integral images.

Thus to select the feature and threshold at each node, we choose a random set of features from all possible Haar features, then choose the feature and associated optimal threshold which minimises G_T. Following [29] we speed up the training by only using a random subset of the available data at each node to select the feature and threshold. In the following we use random subsets of size 400 when there are more than 400 samples to be processed at the node.

3.2 Shape Model Matching

Given an initial estimate of the pose of the model (either from a detector or from an earlier model), we seek to find the model parameters which optimise $Q(\mathbf{p})$ (Eq.3). In order to focus on the effect of the methods used to compute the quality of fit images, $C_i(\mathbf{x}_i)$ in the experiments below, we follow [1] in assuming

a flat distribution for the model parameters, $\mathbf{b}$ within hyper-ellipsoidal bounds:

$$p(\mathbf{b}) = \begin{cases} p_0 & \text{if } \mathbf{b}^T \mathbf{S}_b^{-1} \mathbf{b} \leq M_t \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where $\mathbf{S}_b$ is the covariance matrix of the parameters and M_t is a threshold on the Mahalabanobis distance. M_t is chosen using the CDF of the χ^2 distribution so that 99% of samples from a multivariate Gaussian of the appropriate dimension would fall within it. In this case, optimising $Q(\mathbf{p})$ is equivalent to optimising

$$Q_0(\mathbf{b}, \mathbf{t}) = \sum_{i=1}^{N} C_i(\mathbf{x}_i) \qquad \text{subject to } \mathbf{b}^T \mathbf{S}_b^{-1} \mathbf{b} \leq M_t \tag{6}$$

This has the advantage that it avoids having to choose the value of α.

Given initial estimates of $\mathbf{b}$ and $\mathbf{t}$, we first transform the image into the reference frame by resampling using the current pose: $I_{ref}(i,j) = I(T(i,j;\mathbf{t}))$. We compute the cost images, $C_i(\cdot)$, by searching in I_{ref} around the current estimate of each point in the reference frame.

We then estimate the shape model parameters using the following simple but robust model fitting approach in the reference frame:

1. $r = r_{max}$, $\mathbf{t}_{ref} = \text{Identity}$, $\mathbf{x}_i = \bar{\mathbf{x}}_i + \mathbf{P}_i \mathbf{b}$
2. while $r \geq r_{min}$
   (a) Search in disk of radius r for best points:
   $\mathbf{x}_i \rightarrow \arg\min_{\mathbf{y}:|\mathbf{y}-\mathbf{x}_i|<r} C_i(\mathbf{y})$
   (b) Estimate the shape and pose parameters, $\mathbf{b}$, $\mathbf{t}_{ref}$, to fit to the points
   (c) If $\mathbf{b}^T \mathbf{S}_b^{-1} \mathbf{b} > M_t$, move $\mathbf{b}$ to nearest valid point on limiting ellipsoid
   (d) Regularise the points: $\mathbf{x}_i \rightarrow T(\bar{\mathbf{x}}_i + \mathbf{P}_i \mathbf{b}; \mathbf{t}_{ref})$
   (e) $r \rightarrow k_r r$
3. Map results to the image frame: $\mathbf{t} \rightarrow \mathbf{t} \circ \mathbf{t}_{ref}$, $\mathbf{x}_i \rightarrow T(\mathbf{x}_i; \mathbf{t})$.

If the pose changes significantly, we resample the image at the new pose, recompute the cost images and repeat the above steps. In the experiments below, we set r_{max} to approximately 25% of the object width, r_{min} to 3 pixels (in the reference image) and $k_r = 0.7$.

4 Experiments on Faces

To compare the performance of the approach with alternatives, and to evaluate the effects of choices of parameters, we train a 17 point model using 1182 images of 105 different people, including a range of head orientations and expressions. Figure 4 shows some of the modes of the shape model. The reference frame image is 120 pixels wide, and Haar-like features are sampled from patches of size 24 x 24. The regression functions are trained using random displacements of up to 8 pixels in x and y in the reference image, as well as random changes in scale of up to 5% and rotations of up to 3^o. Each random forest has 10 trees.

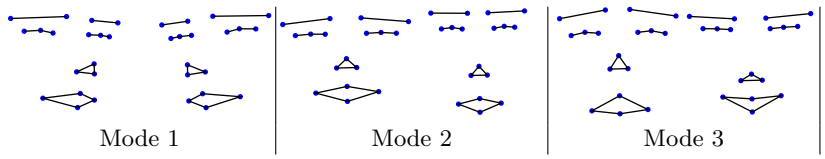

Fig. 4. First three modes of 17 point shape model

We test the model on the images from Session 1 of the XM2VTS database [30]. The model begins with the mean shape at a position displaced randomly by up to 15 pixels in x and y and 5% in scale and 3^o orientation. Following common practice [4], the error is recorded as the mean error per point as a percentage of the inter-occular distance, d_{eyes}. Thus we use

$$m_{17} = \frac{1}{17 d_{eyes}} \sum_i |\mathbf{x}_i - \hat{\mathbf{x}}_i| \tag{7}$$

where $\hat{\mathbf{x}}_i$ is the position of the manual annotation for the point, and $d_{eyes} = |\hat{\mathbf{x}}_{lefteye} - \hat{\mathbf{x}}_{righteye}|$.

4.1 Comparison with other Classifiers/Regressors

We apply a single stage of the model search and evaluate the performance when using different methods of generating the cost images, all trained on the same data. These are (a) the proposed RF Regression, (b) Boosted regressors (trained on the same features) and (c) RF classification (in which samples within 5% of inter-occular distance of the point are classed as positive examples, the rest as negative examples). Figure 5 shows the relative performance. The RF regression significantly outperforms the other methods, and is considerably faster (31ms compared to 140ms for boosting and 1130ms for the classifier, when running on a single core).

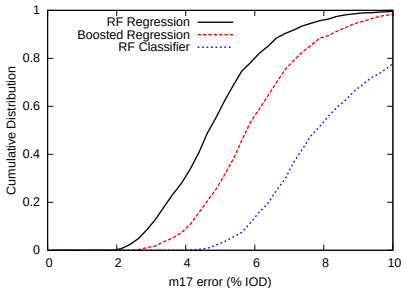

Fig. 5. Effect on performance of different local models

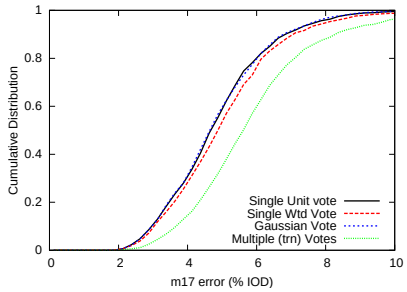

Fig. 6. Effect on performance of different voting schemes

4.2 Effect of Voting Style

We apply a single stage of the model search and evaluate the performance when using different methods of voting. The error is recorded as the mean error per point as a percentage of the inter-occular distance (m_{17}). Figure 6 compares the performance. It shows that using a single vote at the mean position for each tree leads to the best overall performance. The only competing method is casting Gaussian votes, however this is significantly slower and gives no significant difference in performance.

4.3 Effect of Step Size

Rather than sample every pixel, we need only sample the image on a sparse grid. Since each tree casts votes over a region, we can achieve good accuracy without having to sample everywhere.

The table below shows the performance as a function of step size (where step size of s means only take one sample in each s x s square. It shows that significant subsampling can be used without compromising accuracy.

Step	Search time (ms)	Median error %IOD	90%ile %IOD
1	193ms	4.8%	7.0%
2	58ms	4.8%	6.9%
3	31ms	4.8%	7.0%
4	21ms	4.9%	7.0%

4.4 Test on BioID Data

We test for the generalisation performance of our proposed approach in a naturally unconstrained environment by considering the widely used BioID database. The database consists of 1521 images of several individuals captured in an arbitrary office space scenario using a webcam device. It exhibits natural variations in lighting, background, facial expression or pose, and has been used to test a range of different algorithms. In order to compare with other reported results, we focus on localising the 17 facial features shown in Fig. 1. We initialise the model using a Viola and Jones face detector [28].

The final result is presented as the CDF of m_{17} in Fig. 7, where it is compared to the other published results on the same images. Note that Luo *et al.* [24] and Cao *et al.* [25] have recently published results which are slightly better than [9], but follow the same form. Discussions with the authors of [7] suggest the curves are not directly comparable as they re-annotated some of the data.

We perform an experiment to evaluate the effect of annotation noise on the shape of the CDFs. We add Gaussian noise of SD 0.5, 1 and 2 pixels in x and y to the manual annotations of BioID, then compare with the original positions. Figure 8 shows the CDFs of m_{17} for different noise levels, and our own result. The shape of our curve is consistent with noise of about 1.5 pixels. The lack of a

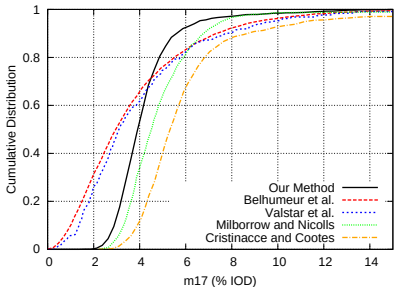

Fig. 7. CDF of the m_{17} measure on BioID, compared with $[9, 7, 2, 4]$

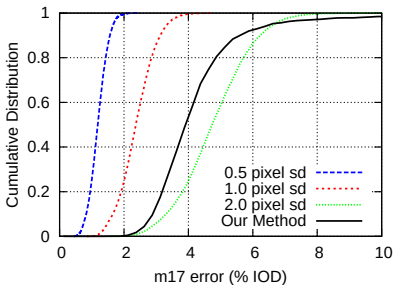

Fig. 8. CDF of the m_{17} measure for 'ideal' system with noise, and our results.

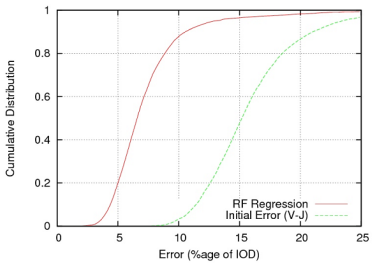

Fig. 9. Model fitting accuracy on AFLW

distinctive "S" shape to the results of [9, 7] suggests their point errors are more correllated.

In terms of computational efficiency we believe our method is is one of the fastest (it takes 27ms on a single core to perform the search given the result of the face detector, compared to many seconds for [9] - though that includes the global search).

4.5 Test on AFLW Data

We also test the system on a subset of images from the Annotated Facial Landmarks in the Wild dataset [31], which includes a wide range of face variation in images sampled from the internet. We selected all the images from the database for which all 15 frontal landmark points were annotated (about 6700 images).

We manually annotated an additional two points to obtain a 17-point markup. Our model was trained using 326 images plus the reflected pairs (652 in total). To test the model, we first applied an off-the-shelf face detector to obtain the initialisation data and gathered a total of 4755 images (those for which the face is detected correctly, and removing the images used for training). The model was trained using a slightly different markup (as in the Bioid data set) - the centre of the nostrils are annotated instead of the exterior. Figure 9 shows the CDF of

the m_{17} measure before and after running a two stage model from a range of perturbed positions on the set. Figure 10 shows example results on this data.

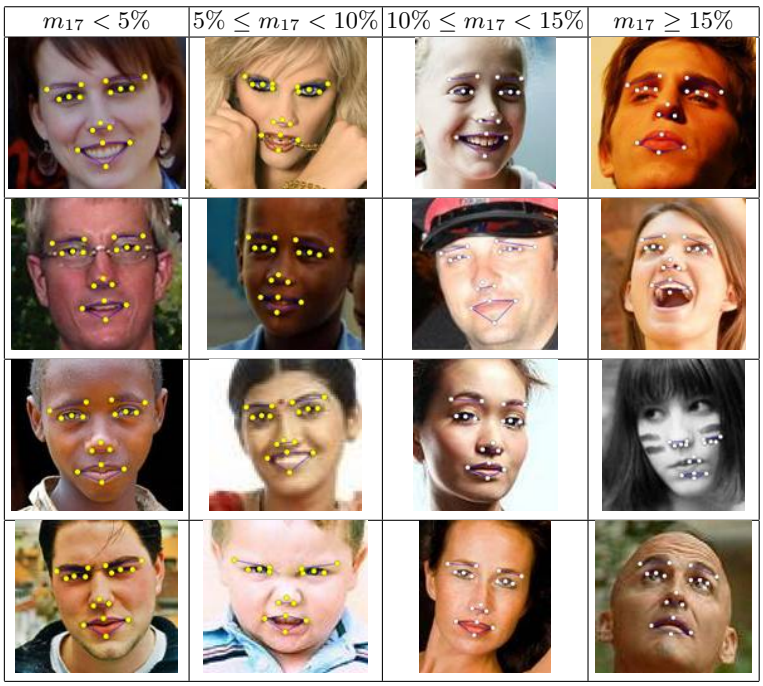

Fig. 10. Examples of results on image from AFLW

5 Performance on Hand Radiographs

To demonstrate that the approach generalises to other application areas, we tested the system on a set of 550 radiographs of hands, each annotated with 37 points. Figure 11 shows an annotation and two of the shape modes of the resulting model, indicating that there is significant shape variation.

We trained the models on the first 200 images, then tested on the remainder. In the first experiment we use a model with a 70 pixel wide reference image, and 13 x 13 pixel patches. We demonstrate the effect of varying the magnitude of the maximum displacements used when training the regressors from 5 to 15 pixels. We measure the error as the mean error relative to the width of wrist (to provide scale invariance). Figure 12 shows the CDFs of the mean point error for a range of displacements. It shows that larger displacements lead to higher convergence rates, but at the cost of overall accuracy. Such models are useful in the early part of a multi-stage search. In Figure 13 we show that excellent results can be achieve with a three stage approach, fitting with a model with

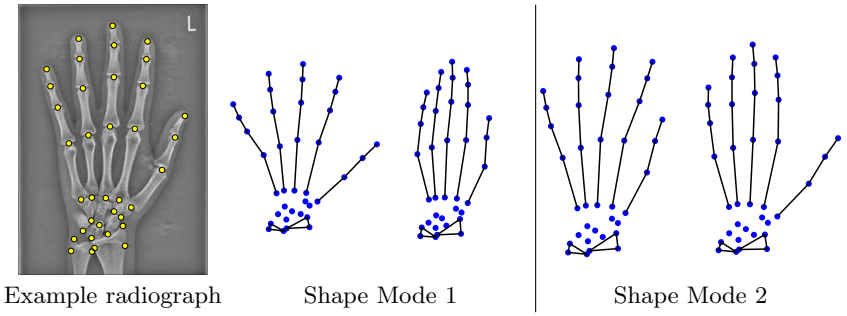

Fig. 11. Hand radiograph with 37 points, and first two modes of a shape model

(w_{ref} = 70,13 x 13 patches, d_{max} = 15), then (w_{ref} = 70,13 x 13 patches, $d_{max} = 7$) and finally ($w_{ref} = 210$,13 x 13 patches, $d_{max} = 7$). The early stages find the approximate positions, which are refined using the higher resolution model in the final stage. The graph also compares three different local models for the response images; (a) normalised correllation with the mean, (b) a random forest classifier (using the same Haar-like features) and (c) the proposed random forest regression voting approach. In each case reference image was 70 pixels wide, and each patch was 13 x 13 pixels within the reference frame. In each case the models were initialised with the mean shape but at randomly displaced positions from the correct pose. Figure 12 shows that the classifier significantly out-performs normalised correlation, but that the regression voting (c) is by far the best overall. Note that the initial error has a median of 17%, which is off the graph. Given that the mean wrist width is about 50mm, this suggests that more than 90% of the time the mean point error is less than 1mm.

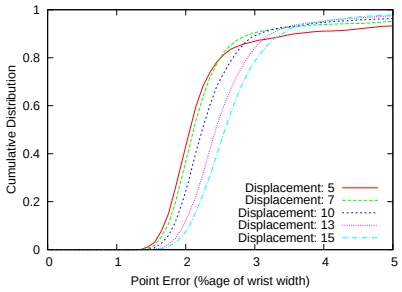

Fig. 12. Performance on hand radiographs

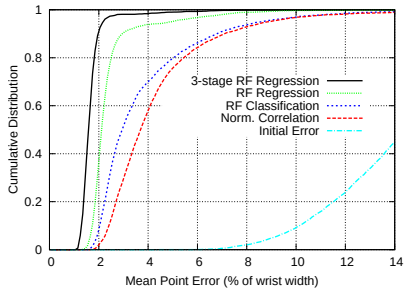

Fig. 13. Performance on hand radiographs

In [32] we demonstrate how the methods describe above form part of a completely automatic system for segmenting the femur in pelvic radiographs, achieving the best published results in that field.

6 Discussion and Conclusions

We show that voting with Random Forest regressors trained to predict the offset of points from the evaluated patch is an effective method of locating feature points. When incorporated into the Constrained Local Model framework, it achieves excellent performance on a range of facial and medical datasets. The approach is found to outperform alternative methods (classification and boosted regression) trained on the same data. We show that using a single vote per tree gives good results, and is significantly faster than alternative approaches. The coarseness of the sampling step can be adjusted to balance between speed and accuracy as required. Overall the method is fast, allowing tracking of faces at frame-rate. Note that although the approach has been demonstrated on frontal faces, it would be equally be applicable to feature detection for non-frontal faces, given a suitable training sets. We have focused on comparing the voting method with different feature detectors, rather than on producing the best facial feature finder. The approach could equally well be incorporated into a range of other model matching frameworks, such as Active Shape Models or Pictorial Structure Matching [3].

Acknowledgements

We thank K.Ward, R.Ashby, Z. Mughal and Prof.J.Adams for providing the hand radiographs and S. Adeshina for the annotations.

References

1. Cootes, T.F., Taylor, C.J., Cooper, D., Graham, J.: Active shape models - their training and application. Computer Vision and Image Understanding **61** (1995) 38–59
2. Milborrow, S., Nicolls, F.: Locating facial features with an extended active shape model. In: ECCV. (2008) http://www.milbo.users.sonic.net/stasm.
3. Felzenszwalb, P., D.P.Huttenlocher: Pictorial structures for object recognition. International Journal of Computer Vision **61** (2005) 55–79
4. Cristinacce, D., T.F.Cootes: Automatic feature localisation with constrained local models. Pattern Recognition **41** (2008) 3054–3067
5. J.M.Saragih, S.Lucey, J.F.Cohn: Deformable model fitting by regularized mean-shifts. International Journal of Computer Vision **200-215** (2011)
6. J.Gall, V.Lempitsky: Class-specic hough forests for object detection. In: CVPR. (2009)
7. Valstar, M., Martinez, B., Binefa, X., Pantic, M.: Facial point detection using boosted regression and graph models. In: CVPR. (2010)
8. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE Trans. Pattern Analysis and Machine Intelligence **23** (2001) 681–685
9. Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J., Kumar, N.: Localizing parts of faces using a consensus of exemplars. In: CVPR. (2011)
10. Covell, M.: Eigen-points: Control-point location using principal component analysis. In: International Conference on Automatic Face and Gesture Recognition, Killington, USA (1996) 122–127

11. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. In: ECCV. Volume 2. (1998) 484–498
12. Saragih, J., Goecke, R.: A non-linear discriminative approach to AAM fitting. In: Proc. ICCV. (2007)
13. Tresadern, P., Sauer, P., Cootes, T.: Additive update predictors in active appearance models. In: British Machine Vision Conference, BMVA Press (2010)
14. P.Sauer, T.Cootes, C.Taylor: Accurate regression procedures for active appearance models. In: BMVC. (2011)
15. Zhou, S., Comaniciu, D.: Shape regression machine. In: Information Processing in Medical Imaging, IEEE Computer Society Press (2007) 13–25
16. K.Zimmermann, J.Matas, T.Svoboda: Tracking by an optimal sequence of linear predictors. IEEE Trans. PAMI **30** (2009) 677–692
17. Ong, E., Bowden, R.: Robust facial feature tracking using shape-constrained multiresolution-selected linear predictors. IEEE PAMI **33** (2004) 1844–1859
18. Dollár, P., Welinder, P., Perona, P.: Cascaded pose regression. In: CVPR. (2010) 1078 1085
19. D.H.Ballard: Generalizing the hough transform to detect arbitrary shapes. Pattern Recognition **13** (1981) 111–122
20. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: ECCV Workshop on Statistical Learning in Computer Vision. (2004)
21. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3d human pose annotations. In: ICCV. (2009)
22. Girshick, R., Shotton, J., Kohli, P., Criminisi, A., Fitzgibbon, A.: Efficient regression of general-activity human poses from depth images. In: ICCV. (2011)
23. Dantone M., Gall J., Fanelli G., Van Gool L.: Real-time Facial Feature Detection using Conditional Regression Forests. In: CVPR. (2012)
24. Luo P. and Wang X. and Tang X.: Hierarchical Face Parsing vis Deep Learning. In: CVPR. (2012)
25. Cao X. and Wei Y. and Wen F. and Sun J.: Face Alignment by Explicit Shape Regression. In: CVPR. (2012)
26. A.Criminisi, J.Shotton, D.Robertson, Konukoglu, E.: Regression forests for efficient anatomy detection and localisation in CT studies. In: Medical Computer Vision Workshop. (2010) 106–117
27. L.Breiman: Random forests. Machine Learning **45** (2001) 5–32
28. P.Viola, M.Jones: Rapid object detection using a boosted cascade of simple features. In: CVPR. Volume 1. (2001) 511–518
29. Schulter, S., Leistner, C., P.M.Roth, Gool, L.V., H.Bischof: On-line hough forests. In: BMVC. (2011)
30. Messer, K., Matas, J., Kittler, J., Luettin, J., Maitre, G.: XM2VTSdb: The extended m2vts database. In: Proc. 2nd Conf. on Audio and Video-based Biometric Personal Verification, Springer Verlag (1999) 72–77
31. Kostinger, M., Wohlhart, P., Roth, P.M., Bischof, H.: Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In: ICCV Workshops, IEEE (2011) 2144–2151
32. C.Lindner, Thiagarajah, S., Wilkinson, J., arcOGEN Consortium, Wallis, G., T.F.Cootes: Accurate fully automatic femur segmentation in pelvic radiographs using regression voting. In: MICCAI. (2012)