Face Parts Localization Using Structured-Output Regression Forests

Heng Yang and Ioannis Patras

School of EECS, Queen Mary University of London
{heng.yang,i.patras}@eecs.qmul.ac.uk

Abstract. In this paper, we propose a method for face parts localization called Structured-Output Regression Forests (SO-RF). We assume that the spatial graph of face parts structure can be partitioned into star graphs associated with individual parts. At each leaf, a regression model for an individual part as well as an interdependency model between parts in the star graph is learned. During testing, individual part positions are determined by the product of two voting maps, corresponding to two different models . The part regression model captures local feature evidence while the interdependency model captures the structure configuration. Our method has shown state of the art results on the publicly available BioID dataset and competitive results on a more challenging dataset, namely Labeled Face Parts in the Wild.

1 Introduction

Accurate detection and localization of face parts, e.g. eyebrow corners, eye corners, mouth corners, tip of the chin, tip of the nose, is often the first step for many applications such as face recognition and facial expression analysis. It has been a very active field in computer vision in the past decade [15,3]. Most of the proposed algorithms report their high-accuracy performance on images from constrained environments and only recently, a few works such as [2,1,4,8,16] have tested their methods on face images "in the wild". Localizing a large collection of pre-specified parts in face images taken under various conditions remains challenging.

Random forests, particularly the regression forests, have emerged as a powerful and versatile method which achieves state-of-the-art results on various high-level computer vision tasks [12,13,5], due to their simplicity and relatively low computational complexity at test time. Recently, it has been appeared to the problem of facial feature detection [1]. However, in existing implementations for regression, the individual parts of an object, e.g. the facial points on a face, are assumed independent.

Inspired by the success of [2] in detecting local parts by imposing global spatial constraints, we propose imposing structural constraints, or shape information, on detection results from various local detectors. Unlike the traditional methods in which one or several shape models are learned from the whole training

K.M. Lee et al. (Eds.): ACCV 2012, Part II, LNCS 7725, pp. 667–679, 2013.

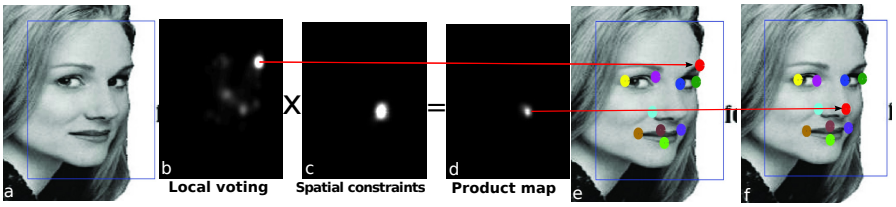

Fig. 1. SO-RF vs. RF. (a)-(f) are respectively test image, probabilistic map for an individual part (left external nostril) from **RF**, constraint map from its neighbors, product map (**SO-RF**), detection result from **RF** and the proposed **SO-RF**. Face parts differ by color.

set with parametric representations, our key contribution in this paper is to incorporate the structure information of an object within the regression forests framework. During forests training, we learn not only the regression model for the localization of individual parts, but also the dependencies between the parts. During testing, we combine the results from interdependent constraint and part detection returned from the same forest to find the optimal configuration for all parts. An outline of the proposed method is shown in Fig. 1. In our experiments, we demonstrate the benefit of the proposed SO-RF and validate our model on datasets recorded in both controlled (BioID) and uncontrolled (LFPW) environments. Structured-output regression forests are able to improve the accuracy as well as the detection rate at a very small additional computational cost.

2 Related Work

Face parts localization, or facial feature points detection, is a well-studied problem in computer vision. Earlier works can be classified into three categories: holistic shape-based, local feature-based and the combination of these two.

A typical holistic method is Active Appearance Model (AAM) [15,11] approach. The entire face shape is reconstructed using appearance model by minimizing the texture residual. However, such methods have difficulties with large variations in facial appearance due to head pose, illumination or expression. Also their localization accuracy degrades drastically on unseen faces [19] and low-resolution images.

Instead of learning a holistic model for all the parts of a face, local feature-based method attempts to learn descriptions for individual parts. In [6], an independent GentleBoost classifier for an individual facial point is leaned separately based on Gabor filter's response. In [7], a coupled prediction classification framework is proposed for visual tracking and applied on facial feature tracking under large motions. Very recently, [1] proposed regression forests for detecting individual parts in real-time, which has reported close-to-human accuracy. The success is due to the power of regression forests to tackle large variances such as appearance changes. In their work, regression forests conditioned on head pose

were proposed in order to deal with the difficulties caused by head pose variance. However, the assumption of independence of parts without any structural constrains can lead to non-plausible facial configuration especially in the presence of partial occlusions (see failure cases of [1]).

Recent works focus on global spatial models built on top of local part detection. Depending on how shape information is modeled, these methods can be classified into two categories. The methods in the first category model the shape information as a prior learned from all training samples while methods in the second type always model the shape information in a non-parametric way. BoRMaN point detector proposed by Valstar et al. [3] is a typical first-category method. It combines the Support Vector Regressors (for local part detection) with Markov Random Fields (for global shape constrains). Among the second category, the work proposed by [8] combines the output of local detectors with a non-parametric set of global models for part locations in a Bayesian framework. This algorithm has been tested on face images recorded in uncontrolled conditions from their Labeled Face Parts in the Wild (LFPW) dataset [8]. The success of this work is mainly due to its Bayesian objective function which implicitly models the anatomical and geometric constrains between facial points. However, this method requires the availability of large number of exemplars and it is computational expensive to choose the global models for each test image. In a very recent work [4], a model based on a mixture of trees with shared pool of parts was proposed for face detection, pose estimation and landmark localization. Every facial landmark is modeled as a part and global mixtures of trees are used to capture topological changes due to viewpoint. Another recent work [2] proposed an "Explicit Shape Regression" approach for face alignment. Instead of using a fixed shape model, the shape constraint is encoded into two-level boosted regression based on fern regressor [9]. This method has been validated on most publicly available dataset and achieved state-of-the-art performance.

3 Structured-Output Forests

In this section we present our SO-RF and its application on face parts localization. In previous literature, a structure prior is usually learned as constraints to guarantee shape consensus of multiple outputs. This prior model is general but it is very hard to capture the informative variances using only one model. Therefore we propose learning the structure constraints in a semi-parametric way within regression forests. The *relative* position between parts is modeled as a Gaussian, but the parameters for each Gaussian model are conditioned on the data. More specifically, a Gaussian is learned for each leaf. Clearly it depends on the data that arrived at the leaf in question during training. In this way, instead of using the average structural prior from the whole training set, in our method the interdependency between parts is conditioned on the test data itself.

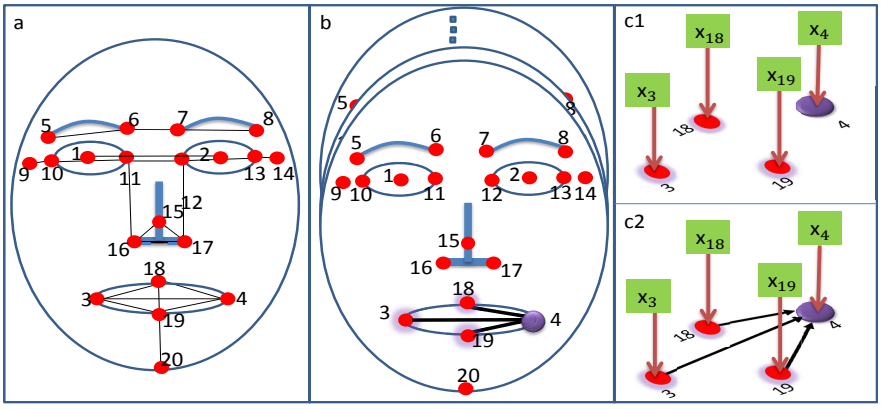

Fig. 2. Structure graphical model. (a) shows manually defined sparse spatial relations of parts on face based on their physical locations. 20 selected face parts (red dots) are displayed and their relations are represented by dark lines. (b) displays partitioned local structure graph associated with individual part. The frontal one shows left mouth corner (purple) and its neighbor nodes (purple shadow). (c1) illustrates an example of the independence assumption between parts used in previous regression forests methods. x_i represents the voting element x that is able to vote for part i. (c2) shows the spatial interdependency model of our method, of which the 4^{th} part not only depends on its voting elements x_4 but also its neighbor parts in the structure graph.

3.1 Spatial Structure Model

Let us assume that the dependencies between the location of face parts can be expressed using a graph, $G = (V, E)$, with V and E are the sets of nodes and edges. The nodes $i = 1, ..., N$ correspond to face parts, and edges $(i, j) \in E$ capture their spatial relations. We further assume the graph structure is already known and what needs to be done is to learn to parameters of joint probabilities of the cliques in the graph. Let $Y = \{y_1, ...y_i, ...y_N\}$ denote the set of random variables associated with the nodes. y_i is the variable associated with the location of part i. Let X be the observed image and x denotes the voting element, i.e. feature descriptor of the local image patch. Each voting element is fed to each tree and will arrive at a leaf node. If x arrives at a leaf node that is able to vote for the i-th part (the voting possibility of one leaf will be discussed later in 3.3), we say $\mathrm{x} \in \hat{\mathrm{X}}_i$. Let $p(y_i|\hat{\mathrm{X}}_i)$ denote the posterior distribution over each part given the image evidence $\hat{\mathrm{X}}_i$. $p(y_i|\hat{\mathrm{X}}_i) = \sum_{\mathrm{x}\in\hat{\mathrm{X}}_i} p(y_i|\mathrm{x})$. Past works on regression forests for localization of multiple parts, e.g. [1,20], assume that locations of different parts are independent, i.e.,

$$p(Y|\mathrm{X}) = \prod_{i=1}^{N} p(y_i|\hat{\mathrm{X}}_i) \tag{1}$$

The independence assumption is shown in Fig. 2-c1. The structure graph in this work is manually defined and shown in Fig. 2. Readers who are interested in learning the structure are referred to [21]. In order to approximate the marginals for random variables in an undirected graphical model, iterative methods such as LBP [22] are often applied. But such iterative method are not easily applicable in regression forests. In many applications, it is of interest to model Y given the input random vector X instead of modeling the dependency structure in a single graph G. The problem of estimating the graph $G(X)$ of Y conditioned on $X = \mathcal{X}$ is called "graph-valued regression" in [18] where "Graph-optimized CART" is proposed. The idea of partitioning the graph based on a greedy splitting procedure is adapted in our problem and we partition the graph into local star graph model to each part. Let $G = \{G(\mathcal{X}_i), i = 1, ..., N\}$ be a set of partitioned directed graphs, one part associated with one directed graph. Let $\mathcal{X}_i = \hat{\mathrm{X}}_i \bigcup \hat{\mathrm{X}}_j|_{j \in Ne(i)}$ be the support feature for the i-th part which is used to determine the model of $G(\mathcal{X}_i)$. This procedure is illustrated in Fig. 3. In this way, each part is associated with a directed graph and is independent from other parts given the support feature for the graph. Thus we have:

$$p(Y|\mathrm{X}, G) = \prod_{i=1}^{N} p(y_i|\mathcal{X}_i) \tag{2}$$

Based on the partitioned directed graph $G(\mathcal{X}_i)$ as shown in Fig. 2-c2, we can calculate $p(y_i|\mathcal{X}_i)$ by gathering the messages from all parent nodes $Ne(i)$ of i and the local image evidence $\mathrm{x} \in \hat{X}_i$, i.e.,

$$p(y_i|\mathcal{X}_i) \propto \left(\sum_{\mathrm{x} \in \hat{X}_i} p(y_i|\mathrm{x}) \right) \cdot \left(\prod_{j \in Ne(i)} \sum_{\mathrm{x} \in \hat{X}_j} p(y_i|y_j, \mathrm{x}) \right) \tag{3}$$

It can also be interpreted intuitively that the localization of one part is not only determined by local image evidence but also by the constrains from the locations of its neighbor parts. Comparing with Eq. (1) in regression forests, our method has an additional term $\prod_{j \in Ne(i)} \sum_{\mathrm{x} \in \hat{X}_j} p(y_i|y_j, \mathrm{x})$ for enforcing the structure constraints. We will illustrate how these two terms $p(y_i|\mathrm{x})$ and $p(y_i|y_j, \mathrm{x})$ are obtained using regression forests in the following section.

3.2 Structured-Output Forests Training

A regression forest $\mathcal{T} = \{T_t\}$ is an ensemble of regression trees T_t. Each regression tree is built based on a randomly selected subset of training images. Here we adapt standard procedure to grow trees that is used in [20], [13] and [1]. From each training image, several square patches are randomly extracted, denoted by x. At training stage $\mathrm{x} = (x^1, x^2, ..., x^F, d)$ which consists of image feature vectors as well as offset vector to each part, $d = (d^1, ..., d^n, ...d^N)$. At testing stage, x only consists of image features. A binary test in relation to the image features is assigned to each non-leaf node of a tree during training. The binary

test function is defined as a comparison of feature values in feature channel f from two positions or regions, R_1 and R_2, with a threshold τ, which is similar as [12]:

$$t_{f,R_1,R_2,\tau}(\mathrm{x}) = \begin{cases} 0 \text{ if } & x^f(R_1) < x^f(R_2) + \tau \\ 1 \text{ otherwise} \end{cases} \tag{4}$$

The value $x^f(R_i), i = \{1, 2\}$ from region R_1 and R_2 can be either a pixel value (R_1 and R_2 are pixel locations) or a mean value of region. Firstly, many splitting candidates $\Phi = \{(R_1, R_2, f, \tau)\}$ are randomly generated. For each candidate, according to the splitting function defined by Eq. (4), the patches S can be divided into the left S^L and right S^R subsets. We use the classification-like method proposed in [1] to calculate the entropy of a data set:

$$H(S) = -\sum_{n=1}^{N} \frac{\sum_{\mathrm{x}\in S} p(c_n|\mathrm{x})}{|S|} log(\frac{\sum_{\mathrm{x}\in S} p(c_n|\mathrm{x})}{|S|}) \tag{5}$$

with

$$p(c_n|\mathrm{x}) \propto exp(-\frac{d_{\mathrm{x}}^{c_n}}{\lambda}) \tag{6}$$

where $p(c_n|\mathrm{x})$ is a soft assignment of x to the the face part n, indicating the probability that the training patch x belongs to the target. This class affiliation is based on the distance between x and part n which is measured on 2D image as in [1]. The factor λ controls the steepness of this function. the best splitting candidate at node j is selected by maximizing the information gain: $I_j = H(S_j) - \sum_{i\in(L,R)} \omega_i H(S_j^i)$. $\omega_i = \frac{|S_j^i|}{|S_j|}$ is the ratio between the number of patches in i and in its parent node. At each subsequent child node, the same procedure continues recursively with each node being designated as a non-leaf node until the stop criteria is met, i.e., the I_j is below a fixed threshold, there are less than a minimum number of patches remaining or a maximum tree depth is reached.

At each leaf node l, we identify the *base part* i at leaf l as that holds *relative offset* within a threshold, i.e. patches arrived this leaf are most likely surrounding this part. In the testing stage, at each leaf only the base part will be voted to avoid a bias towards an average face configuration. At some leaf node, there might be more than one base part holding relative vote within the threshold. We will treat them in the same way so we just use one base part to illustrate our method. We store a distribution over the *relative offset* to each face part y_i of interest. Instead of the Gaussian distribution used in [1], we use the same algorithm proposed in [5] and represent the distribution using a few *relative offset* vectors. Specifically, a mean-shift algorithm with a Gaussian kernel of fixed bandwidth is used to cluster the relative votes. The largest K clusters are stored. Each cluster is represented by relative vote Δ_{lik} (indexed by k), given by the mean-shift mode. We also assign a confidence weight ω_{lik} to each vote, given by the size of its cluster. We refer below to the set of relative votes for part i at each leaf node l, i.e. $p(y_i|l)$ as:

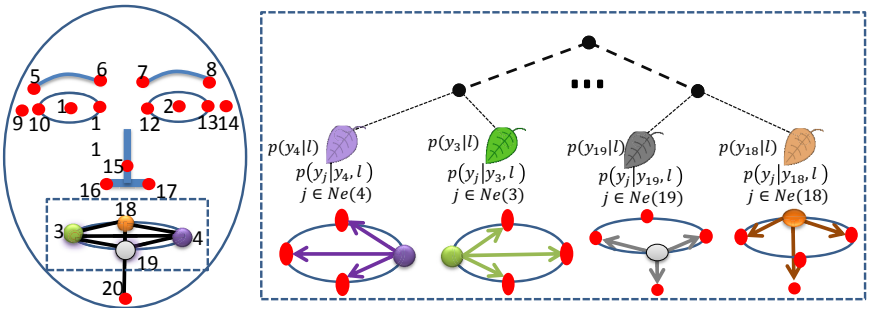

Fig. 3. Structured-output regression forests learning. The left is our predefined simple sparse graphical model of face parts associated with mouth parts. The right shows an example of learning the interdependency of local graph for mouth parts. At leaf node, the base part i is identified (can be more than one). We learn the regression model ($p(y_i|l)$) for base part as well as the conditional distribution of its neighbor nodes $p(y_j|y_i, l)$. Note that the leafs are not real neighbors in a tree.

$$V_{li} = \{(\Delta_{lik}, \omega_{lik})\}_{k=1}^{K} \tag{7}$$

For computational efficiency we use a small K. The maximum of K is two and at some leaf, only one center is stored if data points in the second largest cluster is less than a threshold (10 in our experiments).

To capture spatial relations between the base part and its neighbor parts, we model the relative offset of a part y_j to y_i in Gaussian distribution if and only if j is the neighbor node of i in a predefined structure graph G,

$$p(y_j|y_i, l) = \mathcal{N}(d_l^j - d_l^i | \Delta_{i,l}^j, \Lambda_{i,l}^j) \tag{8}$$

where d_l^i and d_l^j are respectively the offset vectors to i and j at leaf l from all training patches arrived this leaf.

3.3 Structural Inference

During testing, a voting element (local image patch) x located at y^0 , densely or sparsely sampled from image X, is fed to all the trees in the regression forests. At each node of a tree, it is evaluated according to the stored binary test function and passed either to the left or right child until a leaf node l is reached. The leaf node provides the relative votes for each part as defined in Eq. (7), then the absolute vote casted by voting element x for part i is:

$$\bar{\mathrm{y}}_{kli} = \mathrm{y}^0 + \Delta_{kli} \tag{9}$$

To aggregate the absolute votes, we define per part a continuous distribution over world space using a Gaussian Parzen density estimator so the probability of part i at the pixel location y_i is:

$$p(\mathrm{y}_i|l) \propto \sum_{k \in K} \omega_{kli} \cdot exp(-||\frac{\mathrm{y}_i - \bar{\mathrm{y}}_{kli}}{h_i}||_2^2) \cdot \delta(||\Delta_{kli}||_2^2 \leq \alpha) \tag{10}$$

where h_i is an empirical bandwidth. Note that only those relative votes that fulfill a distance threshold α are used, i.e. i should be a *base part* of this leaf as defined before. Another way of stating this is $\mathrm{x} \in \hat{X}_i$. While Eq. (10) models the probability for a voting element arriving at the leaf l of a single tree, the probability of the forest is calculated by averaging over all trees,

$$p(\mathrm{y}_i|\mathrm{x}) = \frac{1}{T} \sum_t p(y_i|l_t(\mathrm{x})) \tag{11}$$

where l_t is the corresponding leaf of tree T_t in the forest.

If the i-th part is the base part at leaf l, we also need to calculate $p(y_j|y_i, \mathrm{x})$. Since the absolute location of y_i is already estimated by x in Eq. (10), we have the conditional probability: $p(y_j|y_i, \mathrm{x}) = p(y_j|\bar{\mathrm{y}}_i)$. Again to aggregate the votes, given the Gaussian Kernel $\mathcal{K}$ from Eq. (8) with bandwidth h_i^j, the absolute location of j-th part, y_j, conditioned on $\bar{\mathrm{y}}_{kli}$ is

$$p(\mathrm{y}_j|\bar{\mathrm{y}}_i, l) \propto \sum_k \mathcal{K}(\frac{\mathrm{y}_j - (\bar{\mathrm{y}}_{kli} + \Delta_{i,l}^j)}{h_i^j}) \tag{12}$$

Similarly, the forest posterior is simply the average of all tree posteriors:

$$p(\mathrm{y}_j|\bar{\mathrm{y}}_i, \mathrm{x}) = \frac{1}{T} \sum_t p(\mathrm{y}_j|\bar{\mathrm{y}}_i, l_t(\mathrm{x})) \tag{13}$$

For each individual part, the local voting term is obtained by gathering the results from all the voting elements, e.g. for part i, it is $\sum_{\mathrm{x} \in \mathcal{X}_i} p(\mathrm{y}_i|\mathrm{x})$. The structure constraint term is obtained by collecting constraints from all its parent nodes, i.e., $\prod_{j \in Ne(i)} \sum_{\mathrm{x} \in \mathcal{X}_i} p(\mathrm{y}_i|\bar{\mathrm{y}}_j, \mathrm{x})$. Therefore, for the i-th part, we get two voting maps. The location of an individual part is obtained by applying a mean-shift algorithm on the product map, as illustrated in Fig. 1.

4 Experiments

4.1 Datasets

We test our algorithm for localizing parts on face on two representative datasets, BioID database [17] and LFPW dataset [8]. There are some other similar datasets such as the LFW used in [1] and LFW87 [2], but they are not publicly available up to date.

BioID dataset consists of 1521 images, each showing a frontal view of face of one of 23 different subjects with various facial expressions captured in a lab environment. We use manual landmarks for this dataset from the FGNET project. Most of the previous methods have reported their results on this dataset which

allows us to compare our work with them. We randomly select 1000 images from the dataset for training and 400 from the remaining ones for testing.

LFPW (Labeled Face Parts in the Wild) is a more challenging dataset. The images are downloaded from internet under a variety of acquisition conditions, including large variability in pose, illumination, expression, partial-occlusion of the face. This dataset shares only image URLs on web but some of them are no longer valid. 821 of the 1132 training images and 214 of the 300 test images can be downloaded when we carried out the experiment. We only use this sub-dataset for training and testing the algorithm.

4.2 Implementation Details

Firstly, the Viola and Jones detector, the same as [3], is applied to find the face bounding box and it is enlarged by 40% in order to ensure all facial feature points are enclosed. Then the box is resized into 150×150 pixels. We extract 43 channels of feature, consisting of grey values, normalized grey values, nine HOG channels as in [12], a Gabor filter bank with eight different rotations and four different phase shifts. For each training image, 100 patches of size 20×20 pixels are extracted randomly, 80 from the inside of face box and 20 from outside. We empirically set the maximum depth of each tree to 15 and 10 trees are trained in a forest. The parameter that control the soft class assignment defined in Eq. (6) $\lambda = 8$ in experiments. We randomly select 600 images from training set to train each tree. At each node, 2000 random tests along with 20 random thresholds for each test are generated for finding the best split function. During testing, the per-part bandwidth $h_i = 5$ pixels and the bandwidth in Eq. (13) $h_i^j = 10$ pixels. The threshold α in Eq. (10) is empirically set to 10 pixels.

4.3 Results

An intuitive example for the out-performance of our **SO-RF** over regression forests (**RF**) is shown in Fig. 1. As shown in the figure, due to the visual similarity to facial points, or other conditions like occlusion, some false hypothesis yields

Table 1. Classification rate of SO-RF compared with reported result in [3] and RF

Part	Valstar[3]	RF	SO-RF	Part	Valstar[3]	RF	SO-RF
P1	94.75%	94.75%	98.25%	P12	92.25%	97.50%	100%
P2	94.75%	96.00%	98.50%	P13	90.50%	96.00%	99.75%
P3	93.50%	96.00%	98.50%	P15	96.25%	91.50%	95.50%
P4	92.50%	97.25%	99.00%	P16	93.50%	94.25%	97.75%
P5	89.00%	90.00%	95.25%	P17	93.25%	95.20%	97.25%
P6	90.25%	91.25%	97.25%	P18	95.00%	97.20%	98.50%
P7	91.25%	94.00%	96.50%	P19	89.50%	94.50%	97.25%
P8	81.00%	86.75%	96.50%	P20	19.25%	80.00%	95.25%
P10	92.25%	94.50%	97.50%	P9		94.00%	97.75%
P11	92.25%	97.50%	100%	P14		93.75%	97.00%

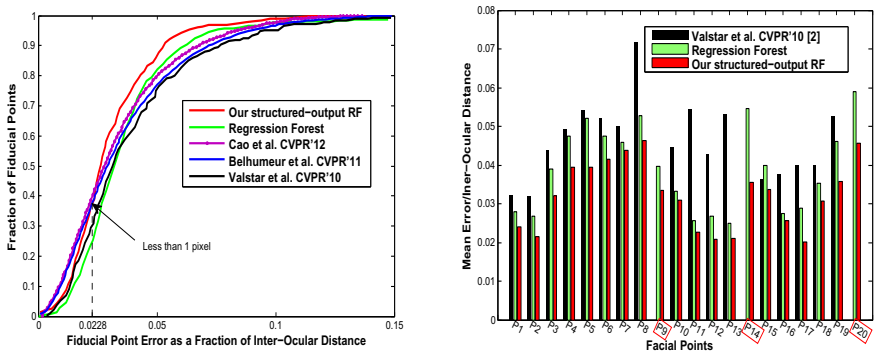

Fig. 4. Overall performance on BioID dataset. (left) Cumulative error distributions for m_{e17} of our proposed SO-RF, RF and the state of the arts. The comparative results are reported by [2], [8] and [3]. (right) Mean error of individual parts. The part ID is defined in Fig. 2. Mean error of P9, P14 is not reported in [3] and its error on P20 is too huge for nice show.

stronger response than the true position. Therefore, detection returned by the largest mean-shift mode of RF might be at the location of false hypothesis. Our SO-RF is able to avoid this local optimization and return the best configuration of parts on face.

Furthermore, besides eliminating the false hypotheses and finding the best configuration, we want to show the performance on improving the detection accuracy. We evaluate the the results of each location by measuring the Euclidean distance between the detected result y_i and its manually labeled ground truth $\hat{y}_i$. As in [3], the error is defined with respect to the Inter-Ocular Distance D_{IOD}, i.e., $e_i = \frac{||y_i - \hat{y}_i||}{D_{IOD}}$. We evaluate the results of all 20 points provided by the BioID dataset. The results including the cumulative error distribution of m_{e17} defined in [14] and the mean relative error are shown in Fig. 4. Though our method does not give better result than state of the art methods at the range of the fraction of inter-ocular distance is less than 0.02, which translates to less than one pixels per point, after that, our method shows significant improvement. Since the ground truth is obtained by hand, we believe the accuracy less than 1 pixel does not make much sense. The mean error of individual parts is shown in the right figure and our method has achieved the state-of-the-art performance. Furthermore, the result validates our SO-RF greatly outperforms RF in detection accuracy. The classification rate $C_i = \frac{\sum_{j=1}^{n} e_i^j < 0.1}{n}$ [3] is also calculated and the comparison of performance is shown in Table 1. Our method has achieved promising performance and some parts, e.g. inner corners of eyes have achieved 100% classification rate.

In figure 5 we compare our LFPW results to human labeling performance and the state of the art methods including [8] [2] and [1]. Note that [1] is not evaluated on this dataset but on LFW dataset with similar challenges. We only

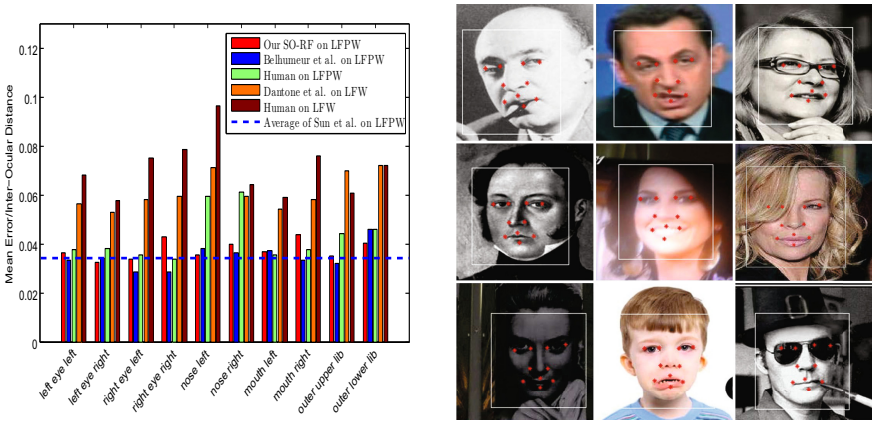

Fig. 5. Performance on LFPW dataset. (left) Mean error of individual parts of our method and the counterparts. (right) Images from LFPW along with part located by our SO-RF. Note that, the number of training images of our method, Belhumeur et al. [8] and Sun et al. [2] (only average error is reported) is respectively 821, 1130, 2000. The Conditional RF [1] of Dantone et al. is trained on LFW in which more than tenfold images are provided for training.

evaluate the results for 10 key points on face, the same as in [1]. According to the mean error comparison, though much less training samples are available to train our detector, it has achieved competitive performance. From the detection results on representative images, our detector has shown the ability to deal with images with low-quality (Col.1), head pose (Row.1) and facial expression variance (Col.2) and partial occlusion (Col.3). Moreover, our SO-RF method does not require much additional time (20% more) comparing with RF and the computational complexity is less than conditional RF in [1]. This quality meets the needs of real-time application.

5 Conclusion

In this paper, we have presented a novel structured-output model based on regression forests which shows competitive results in face parts localization. Our primary innovation is incorporating structure information within the regression forests. We demonstrate state-of-the-art performance on the BioID dataset. On a more challenging dataset (LFPW) that contains images that exhibit greater variability, our method performs in par with the state of the art methods despite the fact that we use much fewer training images (less than a half of [3]).

We have identified that on test faces with large pose variance from the frontal view, our detector does not work very well. In future work we will consider to deal with the problem of the localization of facial features under large pose changes. Furthermore, this structured-output regression forests framework is general and can also be applied to other structural vision problems like human pose estimation.

Acknowledgement. This research is partially supported by an EPSRC grant 'Recognition and Localization of Human Actions in Image Sequences' (EP/G033935/1) and a CSC/Queen Mary Joint PhD scholarship to Heng Yang.

References

1. Dantone, M., Gall, J.: Real-time facial feature detection using conditional regression forests. In: CVPR (2012)
2. Cao, X., Wei, Y., Wen, F., Sun, J.: Face Alignment by Explicit Shape Regression. In: CVPR (2012)
3. Valstar, M., Martinez, B., Binefa, X., Pantic, M.: Facial point detection using boosted regression and graph models. In: CVPR (2010)
4. Zhu, X., Ramanan, D.: Face detection, pose estimation and landmark localization in the wild. In: CVPR (2012)
5. Girshick, R., Shotton, J., Kohli, P., Criminisi, A., Fitzgibbon, A.: Efficient regression of general-activity human poses from depth images. In: ICCV (2011)
6. Vukadinovic, D., Pantic, M.: Fully automatic facial feature point detection using Gabor feature based boosted classifiers. In: IEEE International Conference on Systems, Man and Cybernetics (2005)
7. Patras, I., Hancock, E.R.: Coupled prediction classification for robust visual tracking. T-PAMI (2010)
8. Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J., Kumar, N.: Localizing parts of faces using a consensus of exemplars. In: CVPR (2011)
9. Dollar, P., Welinder, P., Perona, P.: Cascaded pose regression. In: CVPR (2010)
10. Saragih, J.M., Lucey, S., Cohn, J.F.: Face alignment through subspace constrained mean-shifts. In: ICCV (2009)
11. Cootes, T.F., Wheeler, G.V., Walker, K.N., Taylor, C.J.: View-based active appearance models. Image and Vision Computing 20 (2002)
12. Gall, J., Yao, A., Razavi, N., Van Gool, L., Lempitsky, V.: Hough forests for object detection, tracking, and action recognition. T-PAMI (2011)
13. Fanelli, G., Gall, J., Van Gool, L.: Real time head pose estimation with random regression forests. In: CVPR (2011)
14. Cristinacce, D., Cootes, T.: Feature detection and tracking with constrained local models. In: BMVC (2006)
15. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. T-PAMI (2001)
16. Efraty, B., Huang, C., Shah, S.K., Kakadiaris, I.A.: Facial landmark detection in uncontrolled conditions. In: International Joint Conference on Biometrics (2011)
17. Jesorsky, O., Kirchberg, K.J., Frischholz, R.W.: Robust Face Detection Using the Hausdorff Distance. In: Bigun, J., Smeraldi, F. (eds.) AVBPA 2001. LNCS, vol. 2091, pp. 90–95. Springer, Heidelberg (2001)
18. Liu, H., Chen, X., Lafferty, J., Wasserman, L.: Graph-valued regression. In: NIPS (2010)
19. Gross, R., Matthews, I., Baker, S.: Generic vs. person specific active appearance models. Image and Vision Computing (2005)

20. Criminisi, A., Shotton, J., Robertson, D., Konukoglu, E.: Regression Forests for Efficient Anatomy Detection and Localization in CT Studies. In: Menze, B., Langs, G., Tu, Z., Criminisi, A. (eds.) MICCAI 2010. LNCS, vol. 6533, pp. 106–117. Springer, Heidelberg (2011)
21. Schmidt, M.: Graphical Model Structure Learning with l1-Regularization. University of British Columbia (2010)
22. Weiss, Yair.: Correctness of Local Probability Propagation in Graphical Models with Loops. Neural Comput. (2000)