Guided Unsupervised Learning of Mode Specific Models for Facial Point Detection in the Wild

Shashank Jaiswal Timur R. Almaev Michel F. Valstar
School of Computer Science, The University of Nottingham
{psxsj3,psxta4,michel.valstar}@nottingham.ac.uk

Abstract

Facial landmark detection in real world images is a difficult problem due to the high degree of variation in pose, facial expression and illumination, and the presence of occlusions and background clutter. We propose a system that addresses the problem of head pose and facial expressions in a guided unsupervised learning approach to establish mode specific models. To detect 68 fiducial facial points we employ Local Evidence Aggregated Regression, in which local patches provide evidence of the location of the target facial point using Support Vector Regressors. We improve an earlier version of this approach by employing mode specific models and substituting the original Local Binary Pattern features with Local Gabor Binary Patterns. We show that by using specialised model selection we are capable of dealing with various head poses and facial expressions occurring in the wild without the need for manual annotation of pose and expression, and that our proposed detector performs significantly better than the current state of the art.

1. Introduction

Automatic face analysis is an important area of computer vision due to the potential groundbreaking applications in emotion recognition, face recognition, (mental) health assessment, etc. One aspect of analysing the properties of a face is by detecting certain unique fiducial facial landmarks (see Fig.1) and using their locations and displacements over time to infer higher level semantics. Hence algorithms which can accurately detect such facial landmarks are of great significance as they can improve the facial analysis in general. Detecting such points in "faces in the wild", that is faces in images captured in situations that would be encountered by a real application, is particularly challenging because of the high variation in the appearance of facial points caused by different head poses and facial expressions. These variations cause non-linear changes in the appearance of the area immediately surrounding a facial point, making it difficult to learn facial point detectors.

There are generally two approaches to dealing with head pose and facial expressions in facial point detection: parametric or mode specific. In a parametric approach, the machine learning model utilised is supposed to learn the different modes of the facial point appearance and shapes from a sufficiently rich dataset. However, even with training sets containing many thousands of images these different modes do not emerge [13]. On the other hand, mode specific approaches aim to explicitly separate the training data into groups in which facial points have a significantly different shape and/or appearance. Good examples of this are Cootes et al. [7] and Zhu and Ramanan [22].

However, all existing methods for mode-specific facial point detection use supervised learning to create the mode specific models (MSMs). This is problematic, because both pose estimation and facial expression recognition are notoriously time consuming and have relatively low inter-rater reliability (i.e. noisy labels). This makes it hard to create large datasets to train each separate MSM. In addition, supervised labelling of the data is done under the assumption that the different modes are known *a priori*, in our case the head poses and facial expressions causing significant appearance and shape changes. Yet it is not at all evident how to segment the space of head poses and expressions that best separates the appearance and shape variation.

In the light of the shortcomings of existing approaches to create MSMs, we propose to use guided unsupervised learning, in which we employ unsupervised learning on a different sets of points, depending on the type of mode we aim to find (e.g. head pose, or facial expression). The unsupervised learning is applied to the ground truth of the facial points, and the goal is to reduce the shape variation in the obtained modes as much as possible.

Our system builds upon the earlier works of [13] and [19] but is extended to be able to cope with appearance variation caused by non-frontal head poses and facial expressions by using MSMs learned through guided unsupervised learning. We also extend [13] to be able to detect 68 points, rather than 20. This extension required us to formulate a hierarchical Markov Random Field (MRF) shape model to ensure the computation of the MRFs remains tractable. We compared our new point detector to the current state of the art, and evaluated it on the 300W dataset that forms the basis of the 300 Faces in-the-Wild Challenge (300-W).

In summary, our main contributions are:

- Guided unsupervised learning of mode specific models (MSM), where each MSM corresponds to particular head pose and facial expression
- Learning a hierarchical shape model based on Markov Random Field for increased run-time efficiency
- State of the art accuracy in facial point localisation

The remainder of the paper is as follows. Section 2 presents an overview of the related work. Section 3 describes our strategy to learn MSMs of head poses and facial expressions, while section 4 details our full point detection algorithm. Section 5 provides the details for our specific 68 point detection algorithm, which is evaluated in section 6. Finally, we present our closing remarks in section 7.

2. Related work

Face shapes are typically modelled using a statistical shape model [6]. Variations in face shape depend on two different sets of parameters: rigid shape transformations are parameterised using a Procrustes transformation, i.e. using in-plane rotation, translation and uniform scaling. Non-rigid transformations are those that cannot be eliminated through Procrustes analysis, and include transformations caused by facial expressions and out-of-plane head rotations.

Other shape models include graphical models, where facial point detection is posed as a problem of minimising the graph energy. For example, [22] use a tree to model the relative position between connected points. Here convergence to the global maximum is guaranteed due to the absence of loops in the graph. Similarly, a MRF-based shape model was proposed in [13, 19], where the relative angle and length ratio of the segments connecting pairs of points are modelled, making it invariant to both scale and rotation.

A linear model might not be enough to approximate the space of all 2D shapes in the presence of head pose and expression variations. Both Cootes et al. [7] and Zhu & Ramanan [22] propose pose-wise models to handle out-of-plane head poses. Unlike our proposal, the poses are manually annotated, making it hard to collect a large set of training data. In addition, these approaches do not have MSMs for facial expressions.

When it comes to the modelling of appearance, approaches vary significantly. The most common trends with respect to the way texture information is used include Active Appearance Models (AAMs), Active Shape Models (ASM)/Constrained Local Models (CLMs) [1], and regression-based algorithms.

AAMs [14] try to match the whole face appearance with a reference face model. To this end, the facial points are used to define a mesh, and the appearance variations of each triangle within the mesh is modelled using PCA. Face alignment consists on finding the optimal shape and texture parameters so that the reconstruction error is minimised. The appearance models trained for AAMs are often incapable of reconstructing generic faces. Furthermore, the error of the reconstruction is typically measured using the L_2 norm, which is not a robust error measure. Therefore, reconstruction errors dominate alignment errors, resulting in a poor performance. As a consequence, it is common practise to apply AAMs in person-specific scenarios.

In the ASM framework, the face appearance is represented as a constellation of patches local to the facial points. That is, face locations are represented by extracting a representation over a local patch centred at it. A classifier is trained per point to distinguish between the true target location and surrounding locations. An example of a well-optimised ASM is the work by Milborrow and Nicolls [15].

Alternatively, Saragih et al. [17] proposed the Constrained Local Models (CLM), where the authors use a non-parametric distribution to approximate the response map. Accordingly, the resulting gradient ascent shape fitting is substituted by a mean-shift algorithm. It is therefore an efficient algorithm that can run in real time. Although the fitting offered is not very precise, it can offer a good trade-off as it can run in real time and offers high robustness. An extension of the CLM was presented in [2], which substitutes the Mean-Shift fitting by a discriminative shape fitting strategy in order to avoid the convergence to local maxima.

The work by Zhu and Ramanan [22] can be categorised within the ASM/CLM methodology as it uses local appearance models. The authors use a tree-based shape model so that the maximum *a posteriori* likelihood can be attained without using an iterative procedure, and trained a large number of pose-specific experts. This results in a very robust algorithm, capable of performing facial point detection on faces with up to 90 degrees of jaw rotation. However, the precision of the algorithm is often limited and, in particular, it is usually unable to adapt to the presence of expressions.

In regression-based methods the local appearance is analysed by a regressor instead of a classifier. More specifically, given a feature vector, regressors are trained to directly infer the displacement from the test location to the facial point location. Although regression-based models are very recent, they are one of the dominating trends nowadays and yield the best results to date [4, 5, 9, 13, 19].

A popular option is to use of random forests regression and fern features to obtain shape estimates (e.g.[9, 4, 5]).

[1] CLMs can be considered a generalisation of ASM [17]

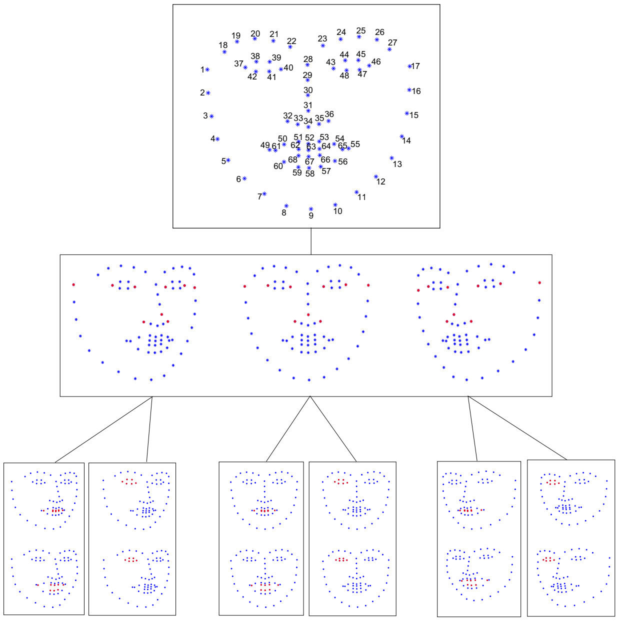

Figure 1. Average shape of face in each cluster obtained for head poses (second row) and facial expressions (third row) due to mouth and eyes. The red coloured dots indicates the facial point which were used to obtain that cluster.

This results in very fast algorithms, ideal for low computational cost requirements. Among them, [9] uses conditional random forests to perform regression conditioned to the current face shape. [5] uses random forest voting to generate a response map in combination with the shape alignment strategy of [17]. [4] uses random forests in a cascade regression strategy [10], and they directly regress the full shape, avoiding the shape alignment step. Alternatively, [19] and [13] use Support Vector Regression to obtain point location estimates from stochastically selected local appearance, and aggregate them into a final prediction.

3. Learning Modes

In order to learn relevant MSMs, modes related to pose and expression were found by guided unsupervised learning and pose and expression detectors were built from those data. These detectors generated subsets of the training data to learn the actual MSMs used in detecting facial landmarks in a particular combination of head pose and expression. Partitioning the training data into these mode specific clusters reduces the variance in the appearance and the relative locations of the facial points. This makes the learning process more efficient and increases the point detection accuracy.

3.1. Learning head poses

We assume that the training images are only labelled with facial points without any labels for head pose or facial expression. Since the head poses are not explicitly labelled, a guided unsupervised approach is applied to learn a head pose detector using the labelled facial points. It is guided in the sense that only those points whose location depend on the head pose, but not facial expression, are used to find the top level modes of head poses. The x and y coordinates of these facial points are concatenated to form a feature vector. The feature vectors from all the face images in the training set are clustered using Ward's minimum variance algorithm [20]. These clusters represent the modes in the data corresponding to head poses.

Since the clustering is done on the basis of facial point locations in the training set, it is not possible to classify a test image into one these clusters as the facial points locations will be unknown in the test images. For this reason, a mapping is learnt from the appearance features of a face image to the modes of head poses obtained from clustering. This mapping is learnt using a multi-class Support Vector Machine (SVM).

3.2. Learning facial expressions

To further reduce the variance in the shape and appearance of the facial points, the face images present in each head pose cluster obtained in section 3.1 are clustered again to learn the top level modes of facial expressions. Clustering for the expressions is done independently for different expressive regions of the face, e.g. the mouth and the eyes. In order to guide the expression clustering process , only the points located in that specific region are used. As features the concatenated pairwise distances between the points from a region are calculated, and used again for clustering using Ward's method.

As with pose estimation, a mapping is learnt from the appearance descriptors of a face image to the facial expression modes determined from the clusters. Sets of multiclass SVMs are learnt separately for each set of clusters obtained for a specific head pose mode, and separate SVMs are learned for each expressive region. This results in learning MSMs which are specialised in estimating facial expressions for a specific head pose.

4. Facial Point Detection

The facial point detection algorithm used here is the regression based Local Evidence Aggregation (LEAR) [13]. This algorithm learns separate regressors for each facial point to estimate the target point location. The output from these regressors are used as evidences and an aggregation of these evidences is used to detect the facial points. These regressors are used in combination with a shape model to localise the the facial points in a test image. The shape model is used to avoid infeasible relative arrangements of facial points.

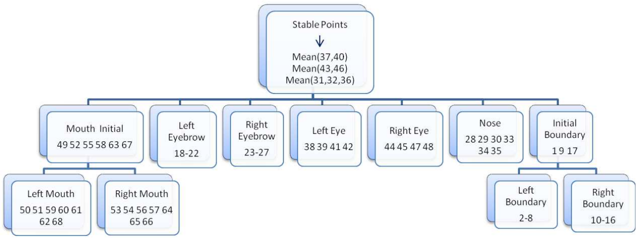

Figure 2. Hierarchical grouping of facial points for computing shape models.

4.1. Appearance model

The local appearance features x_l at a test location l close to the target location t are used to estimate the location of a target facial point. For this purpose, Support Vector Regression (SVR) is used to learn 3 different regressors for each facial point. One regressor r_x is trained for predicting the distance in the horizontal direction ($\Delta x = t_x - l_x$) and one regressor r_y for predicting the distance in the vertical direction ($\Delta y = t_y - l_y$). Hence the predicted target location is given as $\hat{t} = l + \hat{v}$, where $\hat{v} = (\Delta x, \Delta y)$. A third regressor r_d is trained for assessing the quality of the prediction by estimating the distance $\hat{d} = r_d(x_{\hat{t}})$ between the estimated target location $\hat{t}$ and the true target location. The estimated distance $\hat{d}$ is used to calculate a likelihood given by:

$$f_{lik}(\hat{t}) = e^{-\hat{d}/\sigma^2_{lik}} \tag{1}$$

where the variance σ^2_{lik} is a fixed parameter.

4.2. Hierarchical shape model

A probabilistic graphical shape model is used to capture the spatial relationship between the facial points. This shape model is used to avoid searching for points in impossible spatial configurations. It detects if the constellation of a subset of facial points is possible, and if not, suggests a maximum a posteriori probability (MAP) location estimate as a solution to the inconsistency. A spatial relation $r_{i,j}$ between 2 points is defined as the line segment joining the 2 points expressed in polar coordinates as,

$$r_{i,j} = (\alpha_{i,j}, \rho_{i,j}) \tag{2}$$

A probabilistic network of these spatial relations is built using Markov Random Fields (MRF) to encode their interactions. The MRF is constructed with binary states $s_{i,j}$ indicating whether the relation $r_{i,j}$ is a valid shape or not. Each node of the network represents a spatial relation and so the probability of the network is decomposed as the pairwise interaction between the nodes given by:

$$p(\{s_{i,j}\}) = \frac{1}{Z} \prod \varphi_{i,j,k,l}(s_{i,j}, s_{k,l}) \prod \psi_{i,j}(s_{i,j}) \tag{3}$$

where Z is a normalisation factor, $\varphi_{i,j,k,l}$ is a function which encodes the compatibility of $s_{i,j}$ and $s_{k,l}$ depending upon the configuration of points in the training images, and $\psi_{i,j}$ denotes the likelihood of $s_{i,j}$ being 0/1 before considering other nodes. For more details please refer to [13].

The joint MRF is maximised using the Belief Propagation (BP) algorithm, to test a configuration of facial points. The complexity of a fully connected network considering all possible relations, increases quadratically with the number of nodes and hence becomes infeasible if the number of facial points considered is large. In order to make the algorithm more efficient, a hierarchical approach is used to learn the shape model. The facial points are split into smaller groups and a hierarchy of these groups is constructed. A shape model is learnt from the points in a particular group combined with all the points in its parent group. This results in a hierarchy of shape models which is used in combination with the appearance model to detect the facial points in a test image (see Fig. 2).

The points are detected following the same hierarchy, i.e. the points in a particular group I are detected after the detection of all the points in its parent group. The points at the upper levels of the hierarchy seeks to preserve the global shape of the face (for e.g. location of eyes and mouth w.r.t. each other), while the points at the lower levels seeks to preserve the local shape of a smaller region of the face (for e.g. the location of points in the mouth region w.r.t each other). This hierarchical shape model allows an efficient modelling of the shape by keeping the number of points used in constructing any Markov Network, considerably low.

4.3. Detection by Specialised Model Selection (SMS)

Prior to the actual facial point detection process, the head pose and facial expressions are estimated for a face image. The SVM learnt for head pose estimation is used to classify the face image into one of the head pose modes. Depending on the estimated pose, we apply the appropriate SVMs for estimating facial expression. Specialised shape and appearance models are selected depending on the predicted head pose and facial expression. The regressors trained on the partition corresponding to the estimated head pose and expressions were employed in detecting the facial points. Similarly, shape models computed specifically for the particular head pose and facial expressions estimated from the test image were applied during the detection process. This specialised model selection (SMS) procedure ensures that appropriate models are applied to a face image having a specific pose and expression.

The facial point detection process starts with the initialisation of a sampling region for each facial point. Points sampled from the region are used for evaluating the regressors to obtain target estimates (local evidences). The sampling region for a particular facial point is initialised from a Gaussian fitted to the prior distribution of the location of that point in the training images.

The local evidences obtained from the regressor estimates are used to update the evidence distribution and the sampling region in an iterative manner. The shape model checks the correctness of the point configuration at each iteration and provides a new sampling region if the shape gets violated. The evidence distribution is modelled as a mixture of Gaussian distributions where each evidence for the facial point i adds a component $S_k^i(x)$ at the iteration k. The estimate of the target location at the iteration k is given as,

$$\hat{T}_k^i = \arg\max_x S_k^i(x) \tag{4}$$

The confidence on the estimated target location $\hat{T}_k^i$ is given as:

$$p(\hat{T}_k^i) = \max(S_k^i)/\theta_{acc} \tag{5}$$

where θ_{acc} is a predefined acceptance threshold. This process is repeated iteratively until $p(\hat{T}_k^i) > \theta_{acc}$ or a predefined maximum number of iterations has passed.

5. Methodology

The training of our models was based on the 68 points markup definition of Multi-PIE [11] data set (see Fig. 1). We first describe the training of our head pose detector and head pose specific facial expression detectors which were used to partition the training data. We then give details of our mode specific point detection models.

5.1. Head pose estimation

In order to find the modes of head poses in our training data we clustered the face images using the concatenated coordinates of selected facial points. For this purpose, we selected facial points that do not move due to expressions and hence can be used to get clusters corresponding to different head pose modes. We also wanted to keep the dimensionality of our feature vectors to be low and therefore we used only a subset of all such so-called 'stable' points for this purpose, i.e. the points numbered 1,17,31,32,36,37,40,43 and 46.

The average facial point locations in each of the 3 clusters obtained at the top level of the cluster hierarchy are shown in Fig. 1. In this figure, one can clearly see that each of the 3 clusters correspond to a particular head pose (out of plane rotations of the face). The average variance of the location of facial points within the clusters are shown in table 1. From this table, we can clearly see that the average variance within the clusters is significantly lower compared to the entire training data, indicating that the shapes are more similar and thus should be easier to detect.

We trained our head pose detector using appearance descriptors extracted from the face images. We used Local Gabor Binary Pattern (LGBP) [21] to extract the appearance features from face images normalized to 200 × 200 pixels. LGBP features are extracted by applying Gabor filters of various frequencies and orientations on an image before applying the LBP transform and computing histograms.

	All points	Mouth points	Eye points
Entire data	2.88	0.93	0.31
After clustering for head poses	1.36	0.44	0.15
After clustering for expressions	-	0.40	0.14

Table 1. Variances in the locations of facial points in the entire dataset, in head pose clusters and in clusters obtained for facial expressions in the eyes/mouth region.

5.2. Facial expression estimation

We restricted ourselves to expressions involving eyes and mouth region because for these points the facial point location variation due to facial expressions is largest. For expressions involving the mouth region, the pairwise distances between the points from the inner mouth region (labelled 61-68) were calculated in each face image. The concatenated pairwise distances were used as features for clustering the faces into 2 groups, using Ward's method. A similar method was applied for clustering expressions related to the eyes. Assuming symmetrical expressions, the pairwise distances between the points from the left eye (labelled 37-42), were concatenated to form feature vectors, and clustered into 2 groups using Ward's method.

Fig. 1 shows the average shape of face in each of the clusters obtained for facial expressions due to eyes and mouth. The reduced variance for the points belonging to eyes and mouth region are also shown in table 1.

As for head pose estimation, the eyes/mouth expression detectors were trained by extracting LGBP features from face images in each cluster and learning a 2 class SVM. This procedure was repeated for each of the 3 groups of training images obtained from head pose clustering as described in section 3.1. Hence a total 3 SVMs was learnt, each specialised in detecting expressions for a particular head pose.

5.3. Mode specific appearance modelling

The SVMs learnt for estimating the head pose and eye/mouth expressions were used for partitioning the training data. The regressors r_x, r_y and r_d for each facial point (described in section 4.1) were trained on each partition separately. The partitioning was done separately for each facial point. Regressors for points which do not depend on the eye/mouth expressions were trained on 3 separate partitions belonging to the different head poses. Regressors for other points were trained on 3 × 2 partitions, each partition belonging to a particular head pose and eye/mouth expression.

The original algorithm uses LBP features as the local appearance descriptor for training the regressors. In this work, we have used LGBP features to extract the local ap-

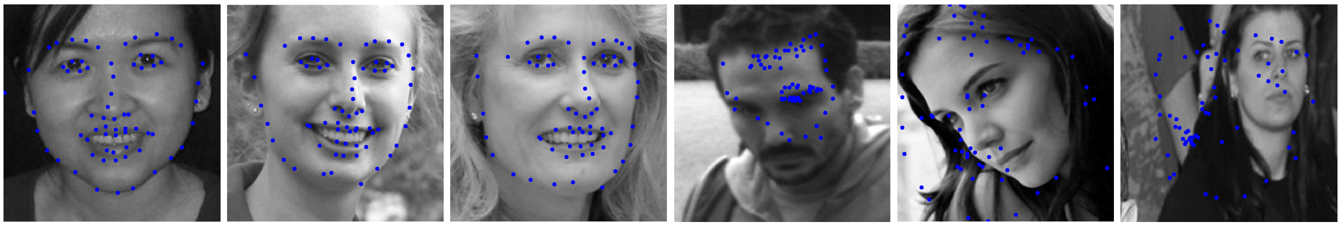

Figure 3. Example results from our method on our internal evaluation test set. The first 3 images shows the best detections measured by normalised mean error. The last 3 are the worst images measured by normalised mean error.

pearance. The LGBP features have been found to be more robust to noise and lighting variation [21] and have been shown to perform better than LBP features for facial Action Units (AUs) recognition [1].

5.4. Mode specific hierarchical shape modelling

As is the case for appearance modelling, the shape models were computed on each training partition separately resulting in shape models which are specialized for a particular head pose and facial expression. For points which depend only on head pose, shape models were computed on 3 different partitions, while for points whose location depends on both head pose and facial expression, shape models were computed from 3 × 2 partitions.

As explained in section 4.2, facial points are split into a hierarchy of smaller group of points and the shape model for each group is computed following that hierarchy. At the top of the hierarchy are the stable points. Stable points are those points which are easy to detect because of their unique local appearance and their invariance to facial expressions. These points are detected first in a test image followed by an affine image registration step. We decided to classify the points 31, 32, 36, 37, 40, 43 and 46 as stable. All other points were classified as unstable.

We selected only these points as stable because firstly, we wanted to keep the number of stable points to be as low as possible as they are detected before the image registration step and hence chances of error are high. Secondly, some points are difficult to detect because the appearance of the area immediately surrounding them may not be unique (e.g. points 28-30). Since the stable points are used for registering the image, any error in detecting them may lead to errors in image registration.

Stable points are followed by all other points as we move down the hierarchy. In order to further reduce the complexity, a set of composite points are computed from the detected stable points using the mean of left eye points (37,40), right eye points (43,46) and nose points (31,32,36). These composite points are used for computing the shape models for points further down the hierarchy. As discussed in section 4.2, the shape model for a group is computed using all the points in that group and all the points in its parent group. In case of groups at the second level of the hierarchy, only the composite points from their parent group (root node) are used in computing the shape model.

The complete hierarchy of the facial point groups can be seen in Fig. 2. Facial points were split into smaller groups, each group belonging to smaller part of the face namely left eyebrow, right eyebrow, left eye, right eye, nose , mouth and face boundary. Since the mouth and face boundary consists of many points, they were further split into smaller groups. For e.g. the mouth region was split into mouth initial, left mouth and right mouth. Similarly, the face boundary region was split into initial boundary, left boundary and right boundary. This hierarchical grouping of points limits the maximum number of points used in any Markov network to 13 and hence makes the algorithm more efficient.

6. Evaluation

We trained our model using approximately 3300 face images from LFPW [3] and HELEN [12] datasets which were re-annotated with facial landmarks [16] using the Multi-PIE [11] 68 points mark-up (see Fig. 1) . The evaluation of our trained model was done in 2 separate ways. One was an internal evaluation in which we tested our model on a test set selected by the authors. The other was an external evaluation in which our model was tested by the organisers of the 300-W Challenge, on the 300-W testset (unknown to us). In both evaluations, the error for a facial point was calculated as the Euclidean distance between the detected location $\hat{T}_i$ of the point and the ground truth location T_i, normalised by the inter-ocular distance d_{IOD} :

$$e_i = \frac{||T_i - \hat{T}_i||}{d_{IOD}} \tag{6}$$

Here the inter-ocular distance d_{IOD} was defined as the distance between the outer corner of the eyes i.e. the distance between the points 37 and 46. The mean error from each image was used for computing the Cumulative error distribution (CED) curves for performance evaluation. These mean errors were computed separately for 51 points (points on the face boundary excluded) and 68 points in each image.

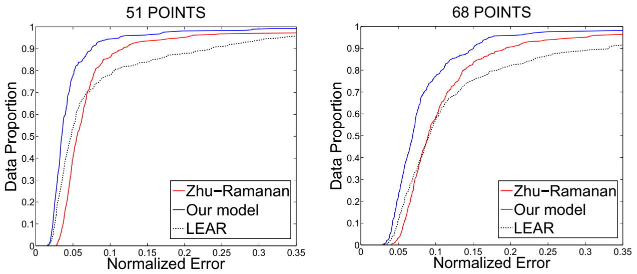

Figure 4. CED curves for 51 landmarks (left) and 68 landmarks (right) computed on our internal evaluation test set.

6.1. Internal evaluation

In our own internal evaluation of our model , we compared the performance of our model with other existing approaches on a test set consisting of 370 face images from AFW [22] and IBUG [18] datasets. These datasets consist of real world images of faces in various head poses and facial expressions. The images also have a wide variation in illumination and image quality and many of the them contains occlusions (e.g. sunglasses, hair, etc.). Overall, the images are quite challenging and can be considered a good test set for benchmarking facial point detection algorithms. We compared the performance of our model with 2 other approaches. The first approach is the regression based local evidence aggregation (LEAR) [13], the second is the part based approach of Zhu and Ramanan [22], which uses a mixture of trees to detect face, pose and facial landmarks. The third method we compared against was the CLM of Saragih et al. [17].

It should be noted that although the AFW and IBUG datasets contains a total of 472 face images annotated with facial landmarks, we had to remove 102 images because in those images the implementation from [22] either doesn't detects any face or detects only 39 facial landmarks due to incorrect head pose estimation corresponding to 90 degree out of plane rotation of the face. Hence, in order to have a fair comparison we prepared a common ground by selecting only those images in which [22] detects a face with all 68 facial landmarks.

In addition, we found that the CLM was often unable to initialize properly. Because the face locations were given for this test set but CLM detects faces internally, we presented the CLM with the image patch surrounding and including the face by growing the face region by 50% in all directions. However, even with this intervention the average point detection error of the CLM was 1.64, and the cumulative error graph was off the scale of Fig. 4. It is entirely possible that this is a problem with the initialization of the CLM rather than the point detection quality though.

Fig. 4 shows the CED curves from the 3 approaches on our test set. The CED curves were computed separately for 51 facial landmarks (labelled 18-68) and the 68 facial landmarks. Both the plots clearly show that our model outperforms the other 2 approaches. The mean errors on the test set for all the 3 approaches can be seen in table 2, which shows that our model is performing significantly better than the other 2 approaches. Fig. 3 shows the 3 best and worst images measured by normalized mean error. It shows that most errors are caused due to poor face detection.

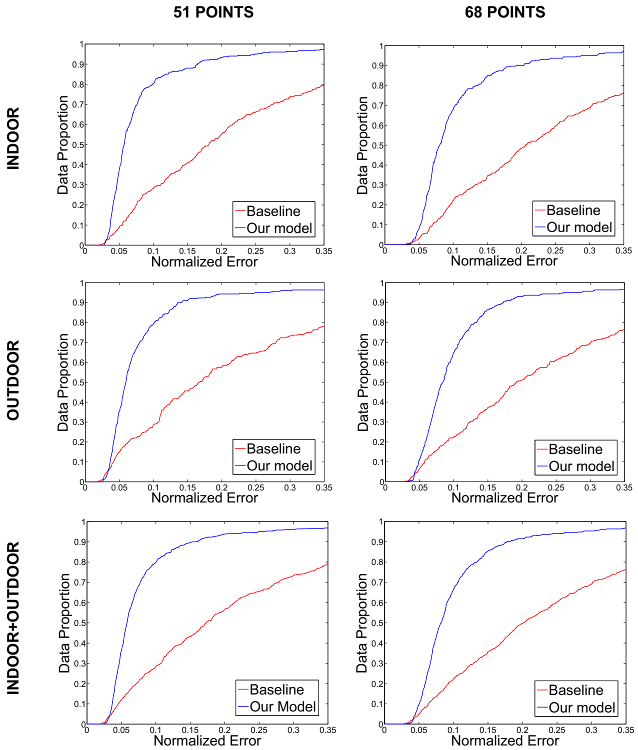

Figure 5. CED curves for landmark detection on the 300-W test set. The first column shows the CED curves for 51 landmarks and the second column shows the CED curves for 68 landmarks. The baseline model corresponds to the approach in [14, 8].

	LEAR [13]	Zhu-Ramanan [22]	Our model
51 points	0.0883	0.0790	0.0494
68 points	0.1379	0.1223	0.0886

Table 2. Comparison of the mean errors on our test set from the 3 approaches.

6.2. External evaluation

Our model was also evaluated independently by the organizers of the 300-W challenge on their own 300-W testset which was not disclosed to any of the participants of the challenge [18]. Their test set was divided into 3 subsets, one consisting of indoor images, another one consisting of outdoor images and the third one consisting of a mixture of indoor and outdoor images. Our model was compared to their baseline model which was based on the project-out inverse compositional AAM algorithm [14] implemented using the edge-structure features described in [8]. The CED curves comparing the performance of our model with the baseline, for each subset of the test set are shown in Fig. 5. The curves were plotted separately for 51 and 68 facial landmarks. From the curves one can clearly see that the performance from our model is much higher than the baseline performance.

7. Conclusion

We presented a novel facial point detection approach using mode specific models, which were found using clustering by guided unsupervised learning. Experts defined the facial points that would result in either clusters corresponding to head pose variations or facial expressions. This approach allows the creation of mode specific facial point detection models without the need for manual annotation of head pose or facial expression. Our approach was applied to the Local Evidence Aggregated Regression framework, and showed significant improvements both over the current state of the art in facial point detection as well as compared to the baseline results of the 300-W facial point detection challenge.

Acknowledgments

The research was in part funded by the National Institute for Health MindTech Healthcare Technology Co-operative. The work of Michel Valstar is also supported by Horizon Digital Economy Research, RCUK grant EP/G065802/1.

References

[1] T. R. Almaev and M. F. Valstar. Local gabor binary patterns from three orthogonal planes for automatic facial expression recognition. In *Proc. Affective Computing and Intelligent Interaction*, 2013. 6

[2] A. Asthana, S. Cheng, S. Zafeiriou, and M. Pantic. Robust discriminative response map fitting with constrained local models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 2

[3] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, pages 545–552, Washington, DC, USA, 2011. IEEE Computer Society. 6

[4] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2887–2894, 2012. 2, 3

[5] T. Cootes, M. Ionita, C. Lindner, and P. Sauer. Robust and accurate shape model fitting using random forest regression voting. In *European Conf. on Computer Vision*, 2012. 2, 3

[6] T. Cootes and C. Taylor. Statistical models of appearance for computer vision. Technical report, University of Manchester, 2004. 2

[7] T. Cootes, G. Wheeler, K. Walker, and C. Taylor. View-based active appearance models. *Image and Vision Computing*, 20(910):657 – 664, 2002. 1, 2

[8] T. F. Cootes and C. Taylor. On representing edge structure for model matching. In *CVPR*, pages 1114–1119, 2001. 7, 8

[9] M. Dantone, J. Gall, G. Fanelli, and L. J. V. Gool. Real-time facial feature detection using conditional regression forests. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2578–2585, 2012. 2, 3

[10] P. Dollár, P. Welinder, and P. Perona. Cascaded pose regression. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010. 3

[11] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image Vision Comput.*, 28(5):807–813, May 2010. 5, 6

[12] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *Proc. European conference on Computer Vision*, pages 679–692, 2012. 6

[13] B. Martinez, M. F. Valstar, X. Binefa, and M. Pantic. Local evidence aggregation for regression based facial point detection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 35(5), pages 1149–1163, 2013. 1, 2, 3, 4, 7

[14] I. Matthews and S. Baker. Active appearance models revisited. *Int. J. Comput. Vision*, 60(2):135–164, Nov. 2004. 2, 7, 8

[15] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. *European Conf. on Computer Vision*, pages 504–513, 2008. 2

[16] Z. S. Sagonas C., Tzimiropoulos G. and M. Pantic. A semi-automatic methodology for facial landmark annotation. In *CVPR*, 2013. 6

[17] J. M. Saragih, S. Lucey, and J. F. Cohn. Face alignment through subspace constrained mean-shifts. In *proc. ACMInt'l Conf. ICCV*, pages 1034–1041. IEEE, 2009. 2, 3, 7

[18] Y. Tzimiropoulos. 300w facial point detection challenge, Sept. 2013. 7

[19] M. F. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using boosted regression and graph models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2729–2736, June 2010. 1, 2, 3

[20] J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244, 1963. 3

[21] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang. Local gabor binary pattern histogram sequence (lgbphs): a novel non-statistical model for face representation and recognition. In *ICCV*, volume 1, pages 786–791 Vol. 1, 2005. 5, 6

[22] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 2879–2886, 2012. 1, 2, 7