Face alignment using a three layer predictor

Eugene Fox

Abstract

Face alignment is an important feature for most facial images related algorithms such as expression analysis, face recognition or detection etc. Also, some images lose information due to factors such as occlusion and lighting and it is important to obtain those lost features. This paper proposes an innovative method for automatic face alignment by utilizing deep learning. First, we use second order gaussian derivatives along with RBF-SVM and Adaboost to classify a first layer of landmark points. Next, we use branching based cascaded regression to obtain a second layer of points which is further used as input to a parallel and multi-scale CNN that gives us the complete output. Results showed the algorithm gave excellent results in comparison to state-of-the-art algorithms.

1 Introduction

Face alignment refers to the process of automatically detecting landmark points on a face regardless of it's orientation in the image. It is a fundamental part of tasks such as attribute inference on a face as shown by Kumar et al [1], for the verification of a face as shown by Lu et al [2], for the task of recognition of a face as shown by Huang et al [3] among others. Recently, research work in the field of face alignment has shown a great amount of success. However, there has been problems when detecting problems related to occlusions, lighting problems etc. Also, a less amount of work has been done to obtain landmarks with regards to different poses of the face. The large majority of face alignment work is done with regards to frontal poses of the people in the images. It is hard to compute the landmarks when the face is rotated or at an angle as many of the landmarks become self occluded.

The general approach is done by employing view-based models and choose results that give a best match as shown by Cootes et al [4], Zhu et al [5]. Non-linear statistical methods have been demonstrated as late as shown by Kanaujia et al [5]. These, however, are slow to execute and cannot be used for real-time applications. Landmark detection had been viewed as an independent problem before. Popular methods involved template fitting approaches as proposed by Zhu et al [5],Yu et al [7], Tzimiropoulos et al [8]. Some other approaches involved regression problems as shown by et al Burgos-Artizzu et al [9], Cao et al [10], Yang et al[11]. Off late however, deep learning has come into the picture. Sun et al [12] proposed the use of deep CNNs for landmark detection. This approach showed great accuracy in comparison to the older models. It is for this reason we consider the use of deep learning to solve this problem.

A shape constraint is essential in all method proposed. A few salient landmarks are used in all cases for example eye centers, mouth corners etc. These are considered to be salient landmarks. The points along the contour of the face are considered to be non-salient landmarks. Parametric models were used to enforce the correlation between the landmarks as shown by Cootes et al [4] when they used an active appearance model (AAM). These parametric models achieved good success, however the model flexibility was a heuristic approach. Also, the shape chosen in the initial stages is far from equal to the target image and it is difficult to use these models to extract the landmarks when the poses in the image are rotated or not frontal.

For the proposed method, we do the following, 1) We do a basic reconstruction of the facial images using two autoencoders that capture the main factors of inputs. The inputs are projected into a higher dimensional feature space by utilizing hidden layers. This is a basic reconstruction. 2) We use second order Gaussian derivatives along with the model proposed by Gowda et al [13] to obtain a basic image consisting of the marker or fiducial points. 3) Next, we use a branching based cascaded regression algorithm to obtain a second set of landmark points based on the already located first set. 4) Finally, we use a parallel and multi-scale CNN to obtain the required output.

2 Related Work

Teodoro et al [14] proposed an approach to reconstruct images in general. They used variable splitting and class adapted image priors to aid with their aim. Results were promising. A deep learning method was proposed by Hayet et al [15]. Though the method gave excellent results, it could not be used for our proposed approach as we needed the algorithm to spend as little time as possible on each step of the algorithm. We used a basic reconstruction algorithm which will be explained in the later sections. Shape regression ap-

proaches have been dominating the face alignment research work recently as can be seen by work proposed by Cao et al [16] who used regression for real-time facial animation, Dollar et al [17] who used cascaded pose regression, Jourabloo et al [18] who developed a regression approach for pose-invariant aligment, Kazemi et al [19] who used regression trees to aid the process of face alignment, Lee et al [20] who used cascade gaussian process regression trees.

A complex relationship exists between face shape (poses) and image appearance of the image and this makes it difficult to obtainto determine the true shape of the face. Many methods have been suggested to overcome this problem. One of these was proposed by Cao et al [10] where they tried multiple initializations and picked the best one. Another approach was proposed by utilizing a coarse-fine search as shown by Zhu et al [21]. We however, propose the use of multiple iterations by utilizing a cascaded shape regressor and to re-compute the shape-indexed features. Conventionally cascaded shape regressors progress from one level to another in a straight line. Each regressor attempts to fit the entire data-set. The setback of this approach is that the regressor function includes many gradient directions that are often conflicting. Xiong et al [22] proposed a global supervised descent algorithm which was an update on the supervised descent method proposed by the same authors [23]. The global supervised descent method (GSDM) splits the regressor function into different regions. Each region consisted of similar gradient directions. Each region had one regressor and this resulted in multiple, independent regressors to solve the problem in hand. They used the method for face alignment. We look at their method and propose an approach to use a branching based CSR.

CSR methods in general can be divided into 2 categories: using off-the-shelf mapping functions (SIFT as proposed by Lowe[24]) such as the method proposed by Tzimiropoulos [25], and methods that learn feature mapping functions such as using a combination of regression trees as proposed by Ren et al [26]. We utilize a combination of regression trees in our proposed approach, the difference in our approach is the use of point distribution model coefficients instead of the 2-D offsets used in [26]. Head-pose variation is one of the biggest problems with regards to face alignment related work. View-based models, non-linear statistical models and 3-D shape models have been used to address the issue. Some examples of view-based models are work proposed in [4,5]. Examples of non-linear statistical methods include kernel methods as proposed by Romdhani et al [27] and mixture models as proposed by Zhou et al [28]. Examples of 3-D shaped models include work proposed by Yu et al [29].

View-based models require separate models each viewpoint, however, they are quick and straightforward. They generally partition the training set in a discontinued ad-hoc way. Nonlinear statistical models on the other hand tend to be too slow for practical applications. The 3-D models however lack from availability of good data. Holi et al [30] showed that high precision parameters are not always fundamental to high accuracy of image classification. Krizhevsky et al [31] used this to show a negligible drop in performance when 16-bit or 8-bit quantization was used. Recently, Soudry et al [32] showed that extreme quantization (binarization) of a network was not impractical and could be used. They obtained good results after qunaitizing to (-1,1). CNNs could be trained using binary weights for both passes i.e forward and backward as shown by Courbariaux et al [33]. They also binarized the activation function and got great results. We use the binarization of a CNN to obtain heatmaps. We use these heat maps to predict landmarks on the facial images.

We follow a three step process, first we obtain a basis for fiducial points, then we use a branching based cascaded regression to obtain a second layer of landmark points. Finally, this is used as input to our binarized CNN that gives us the complete output.

3 Proposed Approach

3.1 Dataset

We test our algorithm on three state-of-the-art databases. First, the Helen database is considered. It was proposed by Le et al [33]. It consists of 2330 images and most faces are near-frontal in the dataset. Next, we used the 300W dataset as proposed by Sagonas et al [34]. This is a collection of in-the-wild data-sets that were annotated with landmarks. Another data-set exists, the FERET dataset which was proposed by Phillips et al [35]. This dataset included neutral expression faces and these had a uniform background. We, however, have not used this.

3.2 First set of fiducial points

First, we detect a set of fiducial points using RBF-SVM and Adaboost classification as shown by Gowda et al [13]. We consider obtaining the local maximum in second-order derivatives of the image after conversion to gray scale. To do this we introduce the use of gaussian derivatives concept. We use three kernels G_{xx}, G_{yy} and G_{xy}. Using these kernels we obtain second-order gaussian derivatives by convoluting the image with these kernels as shown in equation (1-4).

$$G^{\sigma}_{xx}(x,y) = \frac{x^2-\sigma^2}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

$$G^{\sigma}_{yy}(x,y) = \frac{y^2-\sigma^2}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2}$$

$$G^{\sigma}_{xy}(x,y) = \frac{xy}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3}$$

$$I^{\sigma}_{ij}(x,y) = I(x,y) * G^{\sigma}_{ij}(x,y) \tag{4}$$

Figure 1 shows the result obtained after execution of this step.

3.3 Branching Cascaded Regression

The input to this is the output from the first step. The fiducial points show a region of interest and this reduces the computation cost for the branching cascaded regression. We use the model constructed by Smith et al [36] as the inspiration for the branching cascaded regression algorithm. We first build a point distribution model. These model a set of 'n' shapes

Figure 1: Execution after step 1

S=[s_1,...,s_n] by utilizing a linear combination of bases B_s and the mean shape as shown in (5).

$$S' = \mu_s + B_s p_s \tag{5}$$

Here, all the parameters are computed from S via Principal component analysis (PCA). The shapes are aligned using the method proposed by Kunert et al [35]. This helps to remove variations due to rotation, scaling and translation. We however use additional vectors to incorporate these changes back into the model. We then develop a model that gives weightage to both shape and visibility of the landmark points as p=Cq where C is the correlation between shape and visibility and q is the new parameter space. We construct a binary tree to be used as our branching cascade regressor. The update at a cascade level 't' in a BCR node k is executed by R(t,k). Each node in the BCR will have its shape regressor and a point distributed model. Every BCR node models around a subset of training faces that are similar and this helps to decrease the complexity involved for the regressor and point distribution model as there will be fewer parameters. The input to a regressor t is a feature-descriptor d(I,s) that uses the relative shape of an image I with respect to a shape s and captures features out of the shape difference. An updated shape estimate is obtained as the output. Equations (6-10) represent the process explained.

$$\Delta q^t = R^t d^t(I, s^{t-1}) \tag{6}$$

$$q^t = q^{t-1} + \Delta q^t \tag{7}$$

$$\begin{bmatrix} p_s^t \\ p_v^t \end{bmatrix} = C^t q^t \tag{8}$$

$$s^t = \mu_s^t + B_s^t p_s^t \tag{9}$$

$$v^t = \mu_v^t + B_v^t p_v^t \tag{10}$$

Each R is obtained by solving a ridge regression problem. Here q' is the ideal parameter update for face i and N_t refers to the training faces that are part of the BCR node that is currently being tested. After learning 'R' for each node, the training set is partitioned into two sets each overlapping with the other and one set for each child node. This gives each child node a simpler objective function with regards to regression. Finally, we train a RBF-SVM to predict the partition labels from 'd' which refers to the feature descriptors. The SVM output is shown in (11).

$$y^t = w^{t^T} d^t(I, s^{t-1}) + b^t \tag{11}$$

If the output value is negative then the left child node is taken else the right one is taken.

3.4 Parallel and multi-scale CNN

We use the model proposed by Bulat et al as inspiration to construct our parallel and multiscale CNN. Figure 2 shows how an example of how a multiscale and parallel network looks.

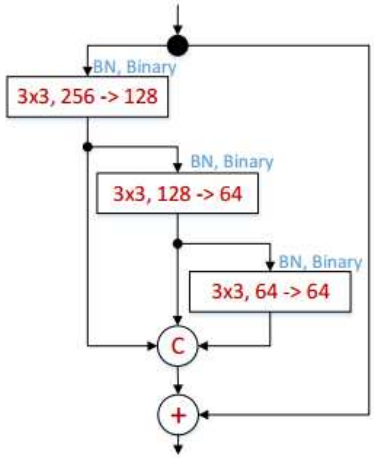

Figure 2: Sample multi-scale and parallel network

The binarization of the network layer is done as shown in (12).

$$I * W \approx (sign(I) \oplus sign(W)) * \alpha \tag{12}$$

I is the input tensor, W is the layer weights and alpha is a scaling factor. The first and last layer of the CNN is kept real whereas the rest are binarized. Our CNN consists of 8 layers followed by 2 fully connected layers to learn global features. The convolution operation is performed as shown in (13).

$$y^l = \sum_n k^{nl} * x^n + b^l \tag{13}$$

Here, x corresponds to the input map and y to the output map. 'k_{ij}' corresponds to the convolution kernel between i-th input map and j-th output map. 'b' corresponds to the bias of the output map and '*' is used to represent convolution. We need to minimize the loss function represented in (14).

$$L = \frac{1}{2}(f - f')^2 \tag{14}$$

Here 'f' corresponds to the ground truth and f' corresponds to the predicted landmark locations. The gradient of loss is back-propogated to update the CNN to minimize the error rate.

3.5 Architecture

The entire architecture of the system is shown in figure 3.

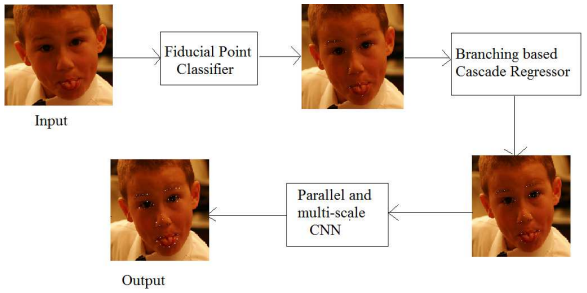

Figure 3: Sample multi-scale and parallel network

First, the input image is sent through a fiducial point classifier. The output from there is passed as input to a branching cascade classifier. This is done so that the regions of interest are predetermined and this reduces the computational cost for the next stage, the branching cascade regressor. The output from here is sent as input to a parallel and multi-scale CNN whose layers are binarized to reduce computational cost.

4 Experimental Results and Analysis

We compare our outputs based on the mean error percentage with outputs from Kazemi et al [19], Xiong et al [23], Yu et al [37], Zhu et al[21] and Ren et al [26]. Table 1 corresponds to the comparison for 300-W dataset for the 68 landmarks based images.

Method	Common Subset	Challenging subset	Full set
Kazemi et al [19]	-	-	6.40
Xiong et al [23]	5.59	15.38	7.51
Yu et al [37]	10.11	19.57	11.96
Zhu et al [21]	**4.75**	9.98	5.78
Ren et al [26]	4.93	11.96	6.31
Proposed	4.78	**8.74**	**5.60**

Table 1: Accuracy comparison using 300-W

Table 2 corresponds to the results obtained for Helen dataset.

Method	194 landmarks	68 landmarks
Kazemi et al [19]	4.91	-
Xiong et al [23]	5.89	5.48
Yu et al [37]	-	9.89
Zhu et al [21]	4.75	4.65
Ren et al [26]	5.43	-
Proposed	**4.67**	**4.62**

Table 2: Accuracy comparison using Helen data-set

5 Conclusion

The proposed algorithm follows three stages. First, a simple fiducial point detection algorithm is used to obtain regions of interest. Next, we use a branching cascaded regression to obtain a second layer of output. Finally, a multi-scale and parallel CNN is used to obtain the final output. Novelty in the approach includes detecting of regions of interest which helped to reduce computational cost for the cascading regression algorithm, using a branching cascaded regression algorithm and finally using a multi-scale CNN which has not been used to the best of our knowledge with regards to face alignment. Also, such ensembles have never been tried before. The results of the algorithm showed that the algorithm performed better than recent state-of-the-art approaches.

6 References

[1] Kumar, N., Belhumeur, P. and Nayar, S., 2008, October. Facetracer: A search engine for large collections of images with faces. In European conference on computer vision (pp. 340-353). Springer Berlin Heidelberg.

[2] Gowda, S.N., 2017. Human activity recognition using combinatorial deep belief networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 1-6).

[3] Huang, Z., Zhao, X., Shan, S., Wang, R. and Chen, X., 2013. Coupling alignments with recognition for still-to-video face recognition. In Proceedings of the IEEE International Conference on Computer Vision (pp. 3296-3303).

[4] Cootes, T.F., Wheeler, G.V., Walker, K.N. and Taylor, C.J., 2002. View-based active appearance models. Image and vision computing, 20(9), pp.657-664.

[5] Zhu, X. and Ramanan, D., 2012, June. Face detection, pose estimation, and landmark localization in the wild. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on (pp. 2879-2886). IEEE.

[6] Gowda, S.N., 2016, December. Age estimation by LS-SVM regression on facial images. In International Symposium on Visual Computing (pp. 370-379). Springer, Cham.

[7] Yu, X., Huang, J., Zhang, S., Yan, W. and Metaxas, D.N., 2013. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1944-1951).

[8] Tzimiropoulos, G. and Pantic, M., 2014. Gauss-newton deformable part models for face alignment in-the-wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1851-1858).

[9] Burgos-Artizzu, X.P., Perona, P. and Dollár, P., 2013. Robust face landmark estimation under occlusion. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1513-1520).

[10] Gowda, S.N., 2016, October. Face verification across age progression using facial feature extraction. In 2016 International Conference on Signal and Information Processing (IConSIP) (pp. 1-5). IEEE.

[11] Yang, H. and Patras, I., 2013. Sieving regression forest votes for facial feature detection in the wild. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1936-1943).

[12] Sun, Y., Wang, X. and Tang, X., 2013. Deep convolutional network cascade for facial point detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3476-3483).

[13] Gowda, S.N., 2016, November. Fiducial Points Detection of a Face Using RBF-SVM and Adaboost Classification. In Asian Conference on Computer Vision (pp. 590-598). Springer, Cham.

[14] Teodoro, A.M., Bioucas-Dias, J.M. and Figueiredo, M.A., 2016, September. Image restoration and reconstruction using vari-

able splitting and class-adapted image priors. In Image Processing (ICIP), 2016 IEEE International Conference on (pp. 3518-3522). IEEE.

[15] Hayat, M., Bennamoun, M. and An, S., 2015. Deep reconstruction models for image set classification. IEEE transactions on pattern analysis and machine intelligence, 37(4), pp.713-727.

[16] Cao, C., Weng, Y., Lin, S. and Zhou, K., 2013. 3D shape regression for real-time facial animation. ACM Transactions on Graphics (TOG), 32(4), p.41.

[17] Dollár, P., Welinder, P. and Perona, P., 2010, June. Cascaded pose regression. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on (pp. 1078-1085). IEEE.

[18] Jourabloo, A. and Liu, X., 2015. Pose-invariant 3D face alignment. In Proceedings of the IEEE International Conference on Computer Vision (pp. 3694-3702).

[19] Gowda, S.N. and Yuan, C., 2018, December. ColorNet: Investigating the importance of color spaces for image classification. In Asian Conference on Computer Vision (pp. 581-596). Springer, Cham.

[20] Lee, D., Park, H. and Yoo, C.D., 2015. Face alignment using cascade gaussian process regression trees. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4204-4212).

[21] Zhu, S., Li, C., Change Loy, C. and Tang, X., 2015. Face alignment by coarse-to-fine shape searching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4998-5006).

[22] Xiong, X. and De la Torre, F., 2015. Global supervised descent method. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2664-2673).

[23] Xiong, X. and De la Torre, F., 2013. Supervised descent method and its applications to face alignment. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 532-539).

[24] Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), pp.91-110.

[25] Tzimiropoulos, G., 2015. Project-out cascaded regression with an application to face alignment. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3659-3667).

[26] Ren, S., Cao, X., Wei, Y. and Sun, J., 2014. Face alignment at 3000 fps via regressing local binary features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1685-1692).

[27] Romdhani, S., Gong, S. and Psarrou, A., 1999, September. A Multi-View Nonlinear Active Shape Model Using Kernel PCA. In BMVC (Vol. 10, pp. 483-492).

[28] Zhou, Y., Zhang, W., Tang, X. and Shum, H., 2005, June. A bayesian mixture model for multi-view face alignment. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on (Vol. 2, pp. 741-746). IEEE.

[29] Yu, X., Huang, J., Zhang, S., Yan, W. and Metaxas, D.N., 2013. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1944-1951).

[30] Holi, J.L. and Hwang, J.N., 1993. Finite precision error analysis of neural network hardware implementations. IEEE Transactions on Computers, 42(3), pp.281-290.

[31] Krizhevsky, A. and Hinton, G., 2009. Learning multiple layers of features from tiny images.

[32] Soudry, D., Hubara, I. and Meir, R., 2014. Expectation backpropagation: Parameter-free training of multilayer neural networks with continuous or discrete weights. In Advances in Neural Information Processing Systems (pp. 963-971).

[33] Le, V., Brandt, J., Lin, Z., Bourdev, L. and Huang, T., 2012. Interactive facial feature localization. Computer Vision–ECCV 2012, pp.679-692. Vancouver

[34] Sagonas, C., Tzimiropoulos, G., Zafeiriou, S. and Pantic, M., 2013. A semi-automatic methodology for facial landmark annotation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 896-903).

[35] Kunert, J. and Qannari, E.M., 1999. A simple alternative to generalized procrustes analysis: application to sensory profiling data. Journal of sensory studies, 14(2), pp.197-208.

[36] Smith, B.M. and Dyer, C.R., 2016. Efficient Branching Cascaded Regression for Face Alignment under Significant Head Rotation. arXiv preprint arXiv:1611.01584.

[37] Yu, X., Huang, J., Zhang, S., Yan, W. and Metaxas, D.N., 2013. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1944-1951).