Facial Feature Points Detection Using Cascaded Regression Tree

Pushkal Thakur[1], Govind Wadajkar[2]
[1,2]Student, Department of Computer Engineering, MGM College, Navi Mumbai, India

***Abstract*— Detecting facial feature points, or Face Alignment, are the problem of detecting semantic facial points in an image or a video, such as points around eyes, nose, mouth or jaw. Cascaded-regression methods for Face Alignment are regression-based methods that work in a stage-by-stage cascade, iteratively improving an initial shape estimate in a coarse-to-fine manner. The goal of this work is to provide an open source implementation of various cascaded regression-based Face Alignment methods.**

This can be achieved by sparse subset of pixel intensity is high quality predictions and real-time performance. this paper has a General Framework based on gradient boosting for learning and n symbol of regression tree that optimizes the sum of square error at naturally handles missing data with the help of image data. The implementations achieved superior performance on the LFPW and Helen datasets compared to implementations of two other state-of-the-art techniques, namely an Active Appearance Model-based approach and the Supervised Descent Method. Analysis and comparison of these algorithms are provided in this report.

***Index Terms*— Facial Feature, Cascaded-Regression Tree**

I. Introduction

The problem of face alignment concerns localising facial feature points (also called facial landmarks) in an image or a video. Typically, 17, 29 or 68 such points are elected to be searched for. Examples of such landmarks are points located around eyes, nose, lips or the jaw. These areas carry the most amount of semantic information for discriminative and generative purposes. The sought-after facial feature points are typically represented as a shape vector $s = (x_1, y_1, x_2, y_2, \dots x_n, y_n)$ where (x_i, y_i) is the position of the i-th landmark within an image and n is the number of landmarks that we wish to detect. The objective of face alignment is to produce such a shape vector from a given image.

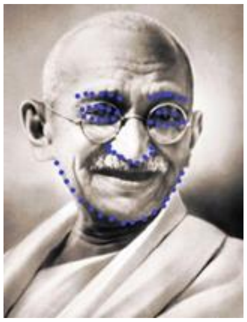

Fig. 1. The famous picture of Mr. Gandhi annotated with facial landmarks. Image source: built-in menpo assets

Typically the problem of face alignment assumes an image with an annotated bounding box which has been detected to contain a face. These can be found using an off-the-shelf face detector, such as the one implemented in OpenCV based on a HOG-based (Histogram of Oriented Gradient) detector found in the dlib library.

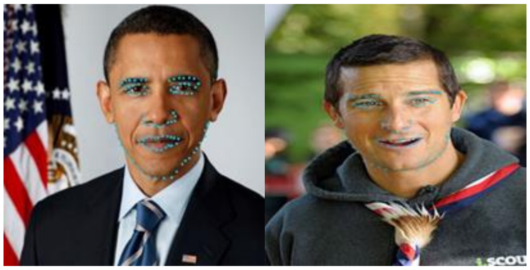

Fig. 2. Picture annotated with facial landmarks

Thus, a face alignment pipeline normally starts with a face detector, which takes the input image and outputs bounding boxes containing faces. The image together with bounding boxes is then fed to the face alignment component that returns a facial shape.

The problem of face alignment is typically approached by supervised machine learning, whereby a model is trained from a large amount of human-labelled images and can then be used for facial shape estimation on unseen images.

There are popular state-of-the-art approaches for face alignment currently studied. In sections, we will describe some of the most used ones, but we shall only have a closer look at regression-based methods, namely those that use a cascaded shape regression framework first proposed by. As opposed to other methods, these progressively refine an initial shape estimate in several stages directly from appearance, without learning any parametric shape or appearance models.

II. Method

A. Regression Based Method

Regression-based methods do not build any parametric models of shape/appearance, but merely study the correlations between image features to infer a facial shape. These methods directly learn a regression function from image features to the target facial shape:

$$M : \varphi(Image) - > s \in R^{2N}$$

Where *M* is the model, *φ*(*Image*) is a function which extracts features from an image instance and *s* is the resultant facial shape. Examples of generally used features include pairwise pixel differences, Haar-like, SIFT or HOG features. There are various face alignment methods based on regression. As the 3 methods studied in this thesis are all based on cascaded shape regression, we shall study this framework more closely in the following section.

1) *Cascaded shaped regression*

Many face alignment methods work in a cascaded framework whereby an ensemble of *N* regressors works in a stage-by-stage manner, which is referred to as stage regressors. This approach was first explored by. At test time, the input to a regressor R_t at stage *t* is a tuple (I,S_{t-1}) where *I* is an image and S_{t-1} is the shape estimate from the previous stage (the initial shape S_0 is typically the mean shape of the training set). The stage regressor extracts features w.r.t to the current shape estimate and regresses a vectorial shape increment: $S_t = S_{t-1} + R_t(\varphi_t(I,S_{t-1}))$

Where $\varphi_t(I,S_{t-1})$ are referred to as shape-indexed features, i.e. they depend on the current shape estimate.The cascade progressively infers the shape in a coarse-to-fine manner - the early regressors handle large variations in shape, while the later ones ensure small refinements. After each stage, the shape estimate resembles the true shape closer and closer.

B. *Face Alignment by Explicit Shape Regression*

The Explicit Shape Regression method by Cao et al uses a cascade of regressors to infer the shape as a whole and explicitly minimises the alignment error over the training data.Each regressor in the cascade returns a vector which is used to update the current shape estimate in an additive manner. To achieve invariance to scale, the shape increment is returned normalised and has to be first transformed before the current shape estimate is updated.

1) *Training a stage regressor*

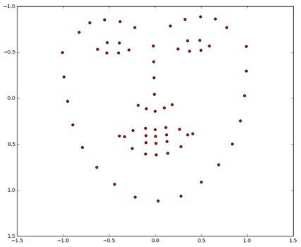

Fig. 3. The average shape

Before the training takes place, the mean shape is calculated out of all training shapes, which is rescaled and centred at the origin. To train a regressor at one stage, each ground truth shape in the training set is first centred at the origin and then aligned with the mean shape using a similarity alignment $M_{t,i}$ (consisting of rotation and scaling only) that minimises the point-to-point alignment error between the two. Such a similarity transformation can be found by Generalised Procrustes Analysis.Operating in this "mean shape frame" is necessary to ensure scale-invariance. For a regressor at stage *t*, the (normalised) target shape increment is

$$yt,i = Mt,i \cdot (Si - St,i)$$

Where S_i is the ground truth shape of training image *i* and $S_{t,i}$ is the estimate at stage *t*.

The average shape calculated from all training ground truth shapes, normalised and centred at origin. Thus, each stage regressor is trained using tuples (I_i,$S_{t,i}$,$y_{t,i}$) where the target variable is the normalised shape difference $y_{t,i}$. The objective of the training is to explicitly minimise the L2 alignment error, which is the same objective that we have at testing:

R_t=argmin

$$R \qquad \sum ||y_i - R(I_i, S_{i-1})||_2^2 \quad i=0$$

To ensure better generalization, the whole training dataset is augmented by perturbing the initial estimates of the initial stage regressor. In my implementations, I performed 20 perturbations of each image. When testing, the current shape estimate is updated at each stage by the regressed normalised shape increment, which is transformed to the global coordinates using the corresponding inverse similarity alignment

$$M_{t,i}^{-1}: St,i = St,i-1 + Mt,i-1 \cdot Rt(Ii,Si-1)$$

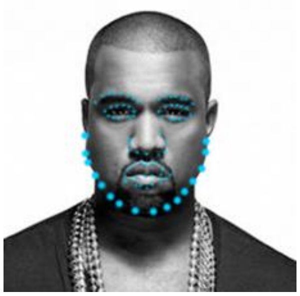

Fig. 4. A pass through one stage regressor.

A pass through one stage regressor. Each stage regressor extracts shape-indexed pixel difference features from the given image and returns a normalised shape increment. The current shape estimate is updated with the regressed shape increment transformed to the frame of current shape using $M_{t,i}^{-1}$. Source of included photo: LFPW dataset.

2) *Shape indexed local features*

To ensure invariance to illumination conditions, the features used in each regressor are differences in pixel intensities, extracted from the image based on the current shape estimate at each stage.

The pixel-difference features are extracted locally w.r.t the nearest landmark on the mean shape. This makes these features invariant to pose and expression variations. At training time, each stage regressor generates a random set of normalised pixel coordinates indexed relative to the nearest landmark on the mean shape. To extract features from a given image I_i with current shape estimate $S_{t,i}$, each of the local pixel coordinates ($x_{l,i}$,$y_{l,i}$) (where l is an index of the nearest landmark), is transformed by $M_{t,i}^{-1}$ to global coordinates of the image. The way feature extraction in machine learning problems is done significantly impacts the predictive power of a constructed model. In this case, the reasons for choosing the aforementioned features are as follows:

- Local features close to facial landmarks are more discriminative than global ones.
- Generating pixel coordinates with respect to the mean shape achieves geometric invariance.

Using differences in pixel intensities rather than absolute values achieves invariance to illumination conditions.

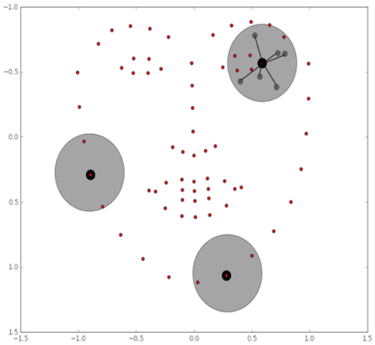

Fig. 5. The pixel-difference features

3) *Correlation based features selection*

Ferns are trained using merely a subset of pixel-difference features extracted in the preceding regression level. In fact, ferns use only $F = 5$ out of P^2 ($P = 400$ in the implementation) features. There are two requirements for the F features selected:

- Features carry as much discriminative information as possible
- Features are as independent to one another as possible

The suggested method is based on calculating the correlation between each feature and the regression targets (ground truth shapes). This is achieved by generating a random unit vector, projecting each target onto it and finding the Pearson correlation coefficient between feature values and lengths of projections.

C. One Millisecond Face Alignment with an Ensemble of Regression Trees

This method outlined in [10] uses a very similar approach to the previous one [9] with a notable difference of using a decision tree as a primitive regressor instead of a random fern.

Thus, in this method, each stage regressor in the cascaded shape regression framework is an ensemble of regression trees (also called a Random Forest). The objective of building a decision tree is explicitly minimising the alignment error in the least squares sense, which is the same goal as in testing. The training of the decision tree is governed by three rules:

1) The optimal split in each internal node of the decision tree is chosen from a random pool of candidate splits s.t. it maximizes the variance reduction in the child nodes.
2) Each leaf node contains the mean of all training samples falling into the leaf multiplied by a regularization parameter $\lambda = 0.1$ in a multiplicative manner.

When choosing splits at internal nodes, rather than performing correlation-based feature selection, a pool of features are selected at random with an exponential prior distribution, biased towards pixel-pairs that are closer together. From this pool, features are further selected to maximize the variance reduction, as stated in point 1.

1) *Exponential prior distribution of selected features*

As mentioned in the previous section, the pool of features θ_{pool} is selected at random. However, as features consisting of pixel pairs that are closer together tend to be more discriminative as of those that are further away, the pixel pairs are sampled from an exponential distribution that favours closer pixels: $P(p,p0) = ke-\lambda||p-p0||$.

III. EXPERIMENT

In our experiments on facial feature detection we used two datasets with annotated faces – the LFPW dataset and the Helen dataset.

LFPW: The Labelled Face Parts in-the-wild (LFPW) dataset consists of 1,287 images collected from the internet. The images contain faces with large variations of facial expressions, illumination, head pose, and occlusions.

HELEN: The Helen dataset contains 2,330 annotated images downloaded from flickr.com website. The face images are of a high resolution, and the provided annotations are very detailed.

We split the LFPW dataset into two parts – one for training and the other for validation. The Helen dataset was used only for testing of the results.

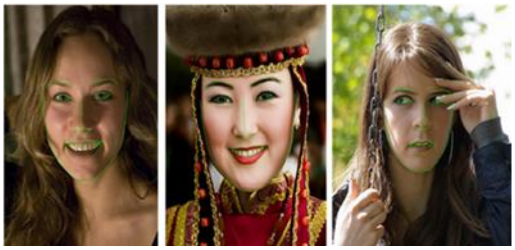

Fig. 6. Facial landmark detection using ensemble of cascaded regressions

To evaluate the accuracy of our method, we used as error measure the point-to-point Euclidean distance, normalized by the distance between the outer corners of the eyes. Facial landmark detection performance was assessed on the 68 landmark point's mark-up scheme. Some images with detected landmarks are shown. Finally, the cumulative error rates were calculated for the Helen dataset.

A. Alignment Accuracy Analysis

We trained all regression methods on a combined dataset consisting of training images from both LFPW and Helen. We performed tests on the corresponding testing datasets and measured point-to-point alignment errors normalised by the diagonal size of each shape. Normalisation of errors is important for consistency - without normalisation, large images would have inherently fairly large errors compared to smaller ones, even though the actual alignment might be reasonable. The SDM implementation comes from the menpofit library. The AAM based method builds a HOG-based Active Appearance Model and use the Alternating Inverse-Compositional algorithm to perform fitting. This set up performed the best among the ones experimented with in. In my experiments, I used the reference implementation of HOG-AIC provided by. As can be seen from figure NAME, the three methods give comparable performance when tested on LFPW and Helen with ERT slightly outperforming the other two. These findings are consistent with the results from the original papers.

B. Alignment Accuracy Analysis

As the only difference between ERT and ESR is using a random forest instead of an ensemble of ferns, the increase in accuracy (around 6% and 2% lower mean error on LFPW and HELEN respectively) must be from a better generalisation ability of decision trees. This corresponds to intuition - the decision trees pick different features in different split nodes and explicitly maximise variance reduction at each split. On the contrary, ferns compare the same features across each level, which might not necessarily maximise variance reduction (note as each leaf node outputs the mean of all training shapes falling into that leaf, maximising variance reduction is equivalent to minimising sum of squares of alignment errors at training. Minimising square of alignment error is also our objective at test time.)Although LBF also uses a random forest (albeit per single landmark and combined with a linear regression matrix), its accuracy is slightly lower than the one of ERT. This is because in my tests, I opted for the faster and less accurate version of LBF that is referred to as LBF-fast in the original paper and consists of 5 stages of regression, 300 decision trees per stage, each of depth 5. The more accurate (but less efficient) version has 5 stages and 1200 decision trees of depth 7 per each stage. Unfortunately, I did not have enough computational resources to run this version and thus cannot comment on the accuracy. However, I expect the accuracy to be comparable to ERT - as there will be a greater number of local binary features; they will be a more discriminative descriptor of each shape. Also, the global regression matrix will have larger dimensions, thus will be capable of capturing more variation.

IV. Conclusion

From above work we understand that how and ensemble of regression tree is important and can be used to regress the location of landmarks from face of spare subset of intensity values extracted from input image this way is very efficient and fast way.

References

[1] Belhumeur, P. N., D. W. Jacobs, D. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in Computer Vision and Pattern Recognition (CVPR), 2011 *IEEE Conference on*, 2011, pp. 545-552.

[2] Breiman, L., "Random forests," Machine Learning, vol. 45, pp. 5-32, 2001/10/01 2001.

[3] Burgos-Artizzu, X. P., P. Perona, and P. Dollar, "Robust face landmark estimation under occlusion," Proceedings of the *IEEE International Conference on Computer Vision*, pp. 1513– 1520, 2013.

[4] Burl, M., M. Weber, and P. Perona, "A probabilistic approach to object recognition using local photometry and global geometry," in Computer Vision — ECCV'98. vol. 1407, H. Burkhardt and B. Neumann, Eds., Ed: Springer Berlin Heidelberg, 1998, pp. 628-641.

[5] Cao, X., Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," International Journal of Computer Vision, vol. 107, pp. 177-190, 2014/04/01 2014.

[6] Cevikalp, H., B. Triggs, and V. Franc, "Face and landmark detection by using cascade of classifiers," in Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on, 2013, pp. 1-7.

[7] L. Ding and A. M. Mart´ınez. Precise detailed detection of faces and facial features. In CVPR, 2008. 1

[8] P. Dollar, P. Welinder, and P. Perona. Cascaded pose regres- ´ sion. In CVPR, pages 1078–1085, 2010. 1, 2, 6

[9] G. J. Edwards, T. F. Cootes, and C. J. Taylor. Advances in active appearance models. In ICCV, pages 137–142, 1999. 1, 2

[10] T. Hastie, R. Tibshirani, and J. H. Friedman. The elements of statistical learning: data mining, inference, and prediction. New York: Springer-Verlag, 2001. 2, 3

[11] V. Kazemi and J. Sullivan. Face alignment with part-based modeling. In BMVC, pages 27.1–27.10, 2011. 2

[12] V. Le, J. Brandt, Z. Lin, L. D. Bourdev, and T. S. Huang. Interactive facial feature localization. In ECCV, pages 679– 692, 2012. 5 [13] L. Liang, R. Xiao, F. Wen, and J. Sun. Face alignment via component-based discriminative search. In ECCV, pages 72–85, 2008. 1

[13] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. In ECCV, pages 504–513, 2008. 5

[14] J. Saragih, S. Lucey, and J. Cohn. Deformable model fitting by regularized landmark mean-shifts. Internation Journal of Computer Vision, 91:200–215, 2010. 1

[15] B. M. Smith and L. Zhang. Joint face alignment with nonparametric shape models. In ECCV, pages 43–56, 2012. 1

[16] P. A. Viola and M. J. Jones. Robust real-time face detection. In ICCV, page 747, 2001. 5

[17] X. Zhao, X. Chai, and S. Shan. Joint face alignment: Rescue bad alignments with good ones by regularized re-fitting. In ECCV, 2012. 1

[18] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In CVPR, pages 2879– 2886, 2012.