Extending Explicit Shape Regression with Mixed Feature Channels and Pose Priors

Matthias Richter
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
matthias.richter@kit.edu

Hua Gao
École Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
hua.gao@epfl.ch

Hazım Kemal Ekenel
Istanbul Technical University (ITU)
Istanbul, Turkey
ekenel@itu.edu.tr

Abstract

Facial feature detection offers a wide range of applications, e.g. in facial image processing, human computer interaction, consumer electronics, and the entertainment industry. These applications impose two antagonistic key requirements: high processing speed and high detection accuracy. We address both by expanding upon the recently proposed explicit shape regression [1] to (a) allow usage and mixture of different feature channels, and (b) include head pose information to improve detection performance in non-cooperative environments. Using the publicly available "wild" datasets LFW [10] and AFLW [11], we show that using these extensions outperforms the baseline (up to 10% *gain in accuracy at* 8% *IOD) as well as other state-of-the-art methods.*

1. Introduction

Facial feature detection, the task of finding the locations of facial landmarks (that collectively define a *shape*) in images or videos, has a broad range of applications. In identity recognition and facial expression analysis, performance is improved by image normalization based on the detected landmarks. The landmarks can also be used to accurately estimate the head pose to guide visual gaze estimation. In itself, facial feature detection is used as novel interface in human computer interaction and offers unique opportunities for modern artists and the entertainment industry.

This rich potential has lead to a long history of research on this subject, but despite to this day, facial feature detection under real-world conditions still remains a difficult problem. Major challenges include variation in lighting conditions, poor image quality, different head pose and facial structure, and partial occlusion of the face. Additionally, the aforementioned use cases dictate two antagonistic requirements: algorithms must (a) provide accurate results in (b) a short amount of time. Usually there is a trade-off: More time can be devoted to the fitting process to achieve a more accurate result (e.g. in image normalization), or accuracy can be sacrificed to achieve real time processing (for example when used in an interactive art installation).

1.1. Related Work

One of the earliest works on the subject is Cootes *et al.*'s *active shape models* (ASM) approach [2], which was later extended to *active appearance models* (AAM) [3]. ASMs learn a generative model of landmark positions (point distribution model, PDM) that captures shape variation in very few parameters. By evaluating the local appearance around the landmarks, the PDM is used to iteratively refine a rough initial estimate. AAMs extend the ASM approach by additionally learning a generative model that parametrizes variation of texture akin to the PDM. Shape fitting is achieved by jointly optimizing both model parameters to match the input image. AAMs converge much faster than ASMs, but tend to fail on unseen faces that can not, or only poorly be represented within the appearance model. Part of this problem may be attributed to the implicit assumption of statistical dependence of the local appearance of two different landmarks in the same face.

The *constrained local model* (CLM) approach proposed by Christinacce and Cootes breaks this assumption by modeling appearance only in patches around landmarks of the current shape estimate [5]. The patch models are used to generate a set of *response maps*, which in turn guide the parameter optimization of the PDM. CLM methods often generalize better and converge faster than AAMs.

Apart from model-fitting approaches, regression methods have been applied with increasing success. In an early study, Cristinacce and Cootes combined the ASM approach with GentleBoost regression [6]. Instead of using patch models to predict landmark positions, the boosted regressor directly estimates landmark-displacement. The new posi-

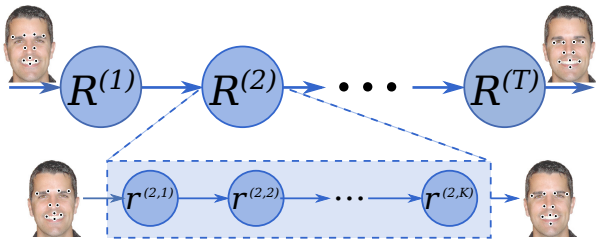


Figure 1. Schema of the two-level cascaded regression.

tions are then constrained to match the PDM and the process is repeated until convergence. In a similar method, Valstar *et al*. used support vector regression to estimate the landmark displacements and a markov random field to enforce the shape constraints [14].

Recently, Dantone *et al*. applied regression forests to facial feature detection [7]. In this method each tree casts a vote on a landmark's position. The votes are based on randomly sampled patches in the image. They furthermore train multiple regression forests for different head poses and find the final shape estimate by collecting the votes of a selection of trees in the different forests. This *conditional regression forest* approach outperforms regular regression forests, while achieving real-time performance and close-to-human accuracy. However, due to the lack of a shape constraints, the estimates are sometimes incoherent. In a similar approach, Cootes *et al*. incorporated random forest regression in their CLM framework [4]. The votes are collected based on Haar-like features at randomly sampled positions around the landmarks. They showed that this approach outperforms their previous GentleBoost method, but did not compare their approach with the one proposed in [7].

In the same year, Cao *et al*. published their *explicit shape regression* (ESR) [1] approach. Instead of optimizing a "surrogate"-criterion to find the landmarks, the alignment error is explicitly minimized using ensembles of random fern regressors. No additional shape model is needed to enforce shape constraints, since they are implicitly encoded in the ensembles. The method shows both remarkable accuracy and speed and as such is suited for real-time feature detection systems.

1.2. Contributions

The main contributions of this work are twofold: Firstly, the ESR algorithm is formulated in a generalized fashion. This allows to use different types of feature descriptors as well as selecting mixtures of features. In particular, we show that, compared to pixel difference features, a mixture of intensity difference and gradient features significantly improves performance (10% improvement in accuracy at 8% interocular distance (IOD) on AFLW), while having only a minor impact on processing speed ($\sim 4ms$ difference per image in testing). Secondly, we develop a method to utilize head pose information in the fitting process. Several "pose experts" are trained on subsets of the training data. Based on a rough pose estimate, only one of the experts is used to find the landmarks. This approach achieves $\sim 95\%$ accuracy at 8%IOD, and additionally allows to use different shape descriptors for different head poses.

2. Method

In this section, the ideas behind ESR are briefly introduced[1]. We then extend the baseline method to allow different feature channels and usage of head pose information.

2.1. Problem Formulation

The location $\mathbf{m}_i = (x_i, y_i)^T$ of n two dimensional landmarks is fully described by collecting the $\mathbf{m}_i$ into a shape vector $\mathbf{S} \in \mathbb{R}^{2n}$. The order of landmark coordinates within the vector is arbitrary, but must remain the same for all shapes. Given an image $\mathcal{I}$, the task is to estimate the shape $\mathbf{S}$ closest to the true shape $\mathbf{S}^*$ shown in the image, i.e.

$$\mathbf{S} = \arg\min_{\mathbf{S}} \|\mathbf{S}^* - \mathbf{S}\|_2. \tag{1}$$

2.2. Explicit Shape Regression

Many shape-fitting approaches achieve the goal in equation (1) using a surrogate criterion, e.g. maximum likelihood on model parameters. In ESR, it is instead explicitly minimized by learning a regressor $\mathbf{S} = R(\mathcal{I})$. The regressor R is itself modeled as a cascade of T weak regressors $R^{(t)}$, where, starting from a rough initial shape estimate $\mathbf{S}^{(0)}$, each $R^{(t)}$ contributes a displacement towards the true shape $\mathbf{S}^*$,

$$\mathbf{S}^{(t)} = \mathbf{S}^{(t-1)} + R^{(t)}(\mathcal{I}, \mathbf{S}^{(t-1)}), \quad t = 1, \ldots, T. \tag{2}$$

Using N training samples $(\mathcal{I}_i, \mathcal{S}_i^*, \mathbf{S}_i^{(0)})$, each weak regressor is learned to minimize the sum of alignment errors,

$$R^{(t)} = \arg\min_{R} \sum_{i=1}^{N} \|\mathbf{S}_i^* - R(\mathcal{I}_i, \mathbf{S}_i^{(t-1)})\|. \tag{3}$$

To overcome weak generalization in early stages, a second level regression is introduced, where each $R^{(t)}$ is a cascade of K primitive regressors $r^{(t,k)}$ similar to equation (2). Figure 1 illustrates this two-level cascade. To avoid degeneration to a one-level cascade, the shape estimates are fixed when training the $r^{(t,k)}$ in a given $R^{(t)}$ and only updated once the second-level cascade is learnt.

Random Fern Regressors. Motivated as a form of semi-naive Bayesian method for classification, random Ferns provide a faster and simpler alternative to random

[1] A more thorough discussion can be found in [1].

forests [13]. As such, they fit well in the role of the primitive regressor. Briefly, a random fern can be thought of as the collection of F features f_i and thresholds τ_i to obtain binary attributes $y_i = [f_i < \tau_i]$.

In the context of shape regression, a method similar to Friedmans gradient boosting trees [8] is applied: The S binary attributes of a given fern divide the joint feature space (and thus training samples) into 2^S disjoint regions. The regression output associated with each region should move shape estimates towards their true shapes, i.e. satisfy

$$\delta\mathbf{S}_b = \arg\min_{\delta\mathbf{S}} \sum_{\mathbf{S}_i \in \Omega_b} \|\mathbf{S}_i^* - (\mathbf{S}_i + \delta\mathbf{S})\|, \qquad (4)$$

where Ω_b denotes the set of current shape estimates $\mathbf{S}_i$ whose features fall into the b-th region.

The exact solution to equation (4) is the mean of target displacement shapes, $\mathbf{S}_i^* - \mathbf{S}_i$. However, this solution may lead to over-fitting in small bins with only a few shapes. Therefore, bin-size dependent shrinkage is applied,

$$\delta\mathbf{S}_b = \left(\frac{1}{|\Omega_b|} \sum_{\mathbf{S}_i \in \Omega_b} (\mathbf{S}_i^* - \mathbf{S}_i) \right) \cdot \frac{|\Omega_b|}{|\Omega_b| + \beta}. \qquad (5)$$

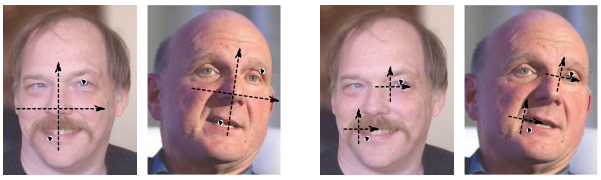
Here, β is a user-defined parameter that governs the influence of the shrinkage term. For small bins and large β, the shape update will vanish, but for large $|\Omega_b|$ equation (5) will approach the true mean.

Local Coordinates. Indexing features relative to a global coordinate system does not reliably describe the same facial structures (see Figure 2(a)). Cao *et al*. solve this problem using shape indexed features [1]: The global coordinates of a feature can be derived by applying an offset $\boldsymbol{\delta}$ to a landmark $\mathbf{m}$ in the current shape estimate. To achieve invariance against face scale and rotation, the offset is applied in relation to the reference shape, and the result is transformed back to the original reference frame, i.e.

$$\mathbf{x} = T^{-1}(T(\mathbf{m}) + \boldsymbol{\delta}). \qquad (6)$$

Correlation Based Feature Selection. The quality of a random fern regressor is mostly dependent on its features. For efficiency reasons, Cao *et al*. use the intensity difference at two local coordinates [1]. These features are very cheap to compute, but surprisingly effective at the same time.

To find good features, a pool of p^2 possible features is created by randomly sampling p local coordinates. These feature candidates remain fixed for all ferns in a first level regressor. Since exhaustive evaluation of all possible random ferns is not feasible and randomized selection does not scale well [1], a subspace projection method is employed. The target shape displacements $(\mathbf{S}_i^* - \mathbf{S}^{(t-1)})$ are projected on a random direction $\mathbf{D}$ to produce N scalars λ_i.



(a) Global coordinate system. (b) Local coordinates.

Figure 2. Different methods to describe feature locations. Locations in (a) differ significantly, while the local coordinates in (b) describe similar facial structures.

The feature candidate f_i with the highest correlation to the λ_i is selected. The process is repeated F times to obtain F features and a fern is constructed with random thresholds.

2.3. Multiple Feature Channels

In the original formulation, ESR uses the intensity difference of two pixels in the image as features to guide the regression [1]. In large quantities, these features are very powerful, but somewhat susceptible to noise and sudden illumination changes. Therefore, it might be useful to use other descriptors as well.

To allow additional channels, the features f_i are formulated in terms of a function $\Phi(\mathbf{x}, \mathcal{I})$, that extracts a descriptor (scalar, vectorial, nominal, etc.) at the pixel $\mathbf{x}$.

Two descriptors d_1 and d_2 of the same type are compared using a suitable comparison function $\Delta(d_1, d_2)$ to obtain the feature

$$f_i = \Delta_i(\Phi_i(\mathbf{x}_{1i}, \mathcal{I}), \Phi_i(\mathbf{x}_{2i}, \mathcal{I})) \in [-1, 1]. \qquad (7)$$

To preserve some sense of direction in the feature, the comparison function should hold $\Delta(a, a) = 0$ and $\Delta(a, b) = -\Delta(b, a)$, but is otherwise unrestricted.

This formulation also allows for ferns with mixed features. For example, f_i could be based on the texture in the neighborhood of $\mathbf{x}_{1i}$ and $\mathbf{x}_{2i}$, while f_{i+1} could be a pixel difference feature.

In our experiments we restricted the feature extraction to simple methods that fast to compute in order to preserve the real-time characteristics of the original formulation. However, more sophisticated descriptors (e.g. SIFT [12]) can also be used within this framework.

Feature Descriptors. Equation (7) describes features that can be expressed as comparison of two locations in the image. The pixel difference features used in [1] can be retrofitted using $\Phi^{pd}(\mathbf{x}, \mathcal{I}) = \mathcal{I}[\mathbf{x}]$ and $\Delta^i(a, b) = a - b$.

To become less susceptible to high frequency noise, one can consider the mean intensity in regions R around $\mathbf{x}$,

$$\Phi_\omega^{rd}(\mathbf{x}, \mathcal{I}) = \frac{1}{|R_\omega(\mathbf{x})|} \sum_{\mathbf{y} \in R_\omega(\mathbf{x})} \mathcal{I}[\mathbf{y}], \qquad (8)$$

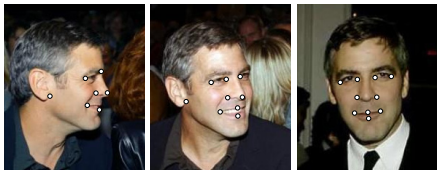


Figure 3. Pose-dependent shape descriptors.

where ω parametrizes the shape of the region[2]. The region difference feature acts as mean-filter on the input image, but is different from pre-processing in that for two given regions R_{ω_1} and R_{ω_2} it is not necessarily true that $\omega_1 = \omega_2$.

Robustness against shadows and other low-frequency noise can be obtained by considering the gradient magnitude at location $\mathbf{x}$,

$$\Phi^{gm}(\mathbf{x}, \mathcal{I}) = \|\nabla \mathcal{I}\|, \tag{9}$$

and choosing the comparison function as the scaled difference $\Delta^g(a, b) = \frac{1}{2}(a - b)$.

Features based on texture codes might also introduce valuable cues, but we have found it to be difficult to find comparison functions that go well with the characteristics of the descriptors while offering a high degree of granularity. For this reason we exclude those methods from our discussion.

Fast Correlation Computation. Calculating the correlation coefficients for the feature selection process has a run-time complexity quadratic in the number of local coordinates. However, if $\Delta(d_j, d_k)$ is a linear combination of its arguments, the correlation coefficient can be expressed in terms of $cov(\lambda, d_j)$, $cov(\lambda, d_k)$ and $cov(d_j, d_k)$. Since the third term does not depend on the λ_i, it can be precomputed and each feature can be selected in $O(p)$ time. This optimization is also stated in [1].

2.4. Head Pose

In [7], Dantone *et al*. demonstrated that conditioning feature detection on rough head pose information can have a positive impact on the overall accuracy. Since their method is tightly integrated in the random forest framework, it is not directly applicable in ESR. The key idea, however, can be adopted in a simple manner:

Instead of learning one regressor R on all training samples, several pose experts R_θ are trained on (possibly overlapping) sub-sets of the training data, where each set corresponds to a head-pose θ. In the fitting stage, a rough pose estimator is used to determine the expert, which is best suited for the image and only this regressor is used to detect the facial landmarks. Similar to the method of Dantone *et al*. [7], this technique can be used to integrate other cues, such as age or ethnicity, as well. However, our approach goes further and allows to use different shape descriptors for different head poses. For example, a frontal pose shape might contain a landmark describing the left mouth corner, while a profile shape might not (see Figure 3).

[2] Parametrization can be arbitrary. Here, we chose to use axis aligned rectangular regions to enable efficient computation using integral images.

(a) Samples from the LFW dataset.

(b) Samples from the AFLW dataset.

Figure 4. Selected images from the AFLW and LFW datasets.

2.5. Super- and Sub-sampling

To obtain robust estimators even with a relatively small training set, Cao *et al*. suggest to super-sample the training data [1]. Existing training data is re-used by drawing the initial shapes $\mathbf{S}_i^{(0)}$ from all other *target* shapes $\mathbf{S}_{k \neq i}^*$.

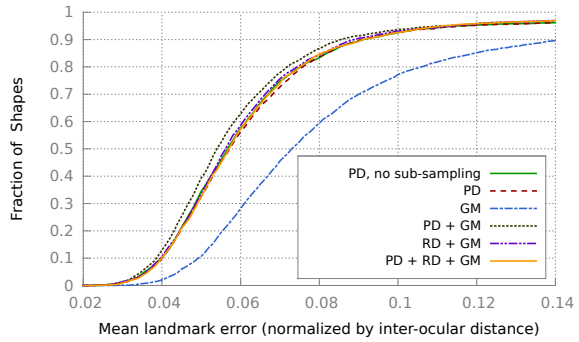
Since the features depend on the current shape estimate as well as on the image, the super-sampling initially corresponds to a much larger training set. However, as the shape estimates converge on their true shapes, this method could lead to over-fitting in later stages. To account for this issue, we randomly sub-sample the training data when generating the feature candidates for each first-level regressor. A similar idea, although in a different problem domain, is found in Friedman's stochastic gradient boosting, where he showed that sub-sampling can improve the overall performance [9].

3. Experiments

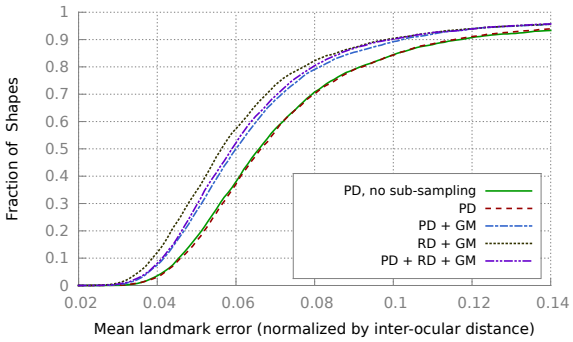
We performed experiments using the publicly available "real-world" datasets *Labeled Faces in the Wild* (LFW) [10] and *Annotated Facial Landmarks in the Wild* (AFLW) [11].

LFW consists of color images collected from news articles on the web, and shows large variation in facial expression, gender, ethnicity, and image quality, but only a moderate amount of variation in lighting conditions and head pose. Faces are often partially occluded and the background is often noisy and may contain other faces. Annotations of 10 facial landmarks were provided by Dantone *et al*. [7]. All faces are fully labeled, even if a landmark is not visible, e.g. in the middle picture of Figure 4(a).

AFLW was specifically designed to evaluate facial feature detection algorithms in a real-world setting. The im-



(a) LFW Dataset.



(b) AFLW Dataset.

Figure 5. Cumulative error curves for different features combinations, with and without sub-sampling.

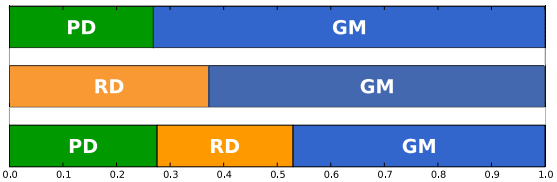


Figure 6. Composition of selected feature mixtures. First row: PD+GM (LFW). Second row: RD+GM (AFLW). Third row: PD+RD+GM (AFLW).

Mixture	Training		Testing	
PD	1.25s	1.56s	5.8ms	8.1ms
PD + GM	2.45s	2.51s	7.5ms	8.2ms
RD + GM	3.48s	4.14s	10.6ms	12.2ms
PD + RD + GM	4.29s	3.96s	13.7ms	14.4ms
	LFW	AFLW	LFW	AFLW

Table 1. Average time required to process one image.

ages were collected from the photo sharing service *flickr* and show even more variation than LFW, including large head rotation and strong shadows. Since not all images show all 21 labeled landmarks, we restricted evaluation to the samples in which at least 19 of landmarks were visible. Samples images are shown in Figure 4(b).

Evaluation was performed using 5-fold cross validation on a subset of 5000 images on either dataset, unless otherwise noted. The initial shapes were placed on the center of the annotated facial bounding boxes and randomly shifted by ± 2 pixels and scaled to 95% to 105% of the original size to simulate usage of a face detector. Parameters were chosen as suggested in [1], i.e. $F = 5$, $\beta = 1000$, $T = 10$, $K = 500$, and $p = 400$. Training data was super-sampled 20 times, and sub-sampling was performed using 60% of the data.

3.1. Feature Mixtures

Figure 5 shows the cumulative distribution of the mean normalized landmark error. On both datasets, a mixture of pixel or region difference (PD, RD), and gradient magnitude (GM) features improves detection accuracy, but the effect is much stronger on the AFLW dataset. This result may be explained by strong variations of lighting conditions in AFLW, where gradient magnitude offers a more robust descriptor.

The relative composition of selected mixtures is shown in Figure 6. In each mixture, the gradient magnitude descriptor provides the bulk of selected features. This indicates the relative importance of gradient information: More than 60% of the features in the best performing mixture (RD+GM) on AFLW are gradient features. On LFW, the effect is even stronger, as more than 70% of the features in a PD+GM mixture are derived from gradient descriptors. This is a somewhat intriguing result: on their own, gradient magnitude descriptors are not nearly as expressive as intensity descriptors (Figure 5(a)). Contrary to our expectation, sub-sampling does not have a noticeable impact on the overall performance. However, this technique greatly reduces training time, so its usage can still be justified.

Table 1 shows the average processing time per image on a 32 Core 2.2GHz Intel Xeon CPU with 64 GB RAM. The algorithm is very fast: Fitting 10 (LFW) or 19 (AFLW) landmarks takes just a few milliseconds. The training algorithm utilizes all available cores, but the cascaded regression prevents parallelization of the fitting process. Therefore, similar execution times can be expected on current generation consumer hardware.

3.2. Head Pose

We used the same pose-detector as Dantone *et al.* [7] to partition the LFW dataset into two balanced sets corresponding to a frontal and a side pose. Each set contained 2915 samples. In a 5-fold cross validation, the 5830 samples were either used to train two independent pose experts (each using half of the available data) or one monolithic regressor (using all available data).

Figure 7 shows the cumulative distribution of normal-

ized landmark error for both approaches. The multi-pose approach outperforms the monolithic regressor. The work of Dantone *et al*. suggests that using more poses would further improve recognition performance [7].

3.3. Comparison with related work

We now turn our attention to comparison with similar methods that were evaluated using the same datasets. Results are shown in Figure 8.

On LFW, the method of Dantone *et al*. [7] achieves better results when considering a mean landmark error below 5% inter-ocular distance (IOD), but is outperformed by our single-pose approach thereon. In particular, Dantone *et al*. report 86.1% of the shapes have a landmark error $\leq 10\%$ IOD [7], where our method achieves 93.6% accuracy. The lower accuracy on smaller IOD-thresholds may be explained by the lack of shape constraints in the model in [7]. This allows to correctly estimate the position of landmarks, where our method would fail, e.g. the center image in Figure 4(a). However, these results are not strictly comparable, since Dantone *et al*. evaluated their system on all 13233 images and used 10-fold cross validation [7], while we did 5-fold cross validation on a subset of 5000 images. In addition, it seems that some of the labels we used were corrupted (see third row of Figure 9(a)).

Similar limitations have to be considered when comparing our results on AFLW to the results of Cootes *et al*. [4]: In their analysis, only the subset of samples in which 15 frontal landmarks were visible were used. Furthermore, they augmented the labels to obtain a 17-point markup scheme, which was unavailable to us. Their system was trained on 326 images and their reflected pairs, and evaluated using 4755 images in which a face could be detected. In particular, the upside-down faces in Figure 9(b) were likely not included in their analysis. To our knowledge, cross-validation was not performed. Still, the comparison in Figure 8 suggests that our system outperforms their random regression voting scheme on this challenging dataset.

4. Conclusion

We have formulated a generalized version of explicit shape regression for facial feature detection, which allows to use and mix complementary feature channels. We have shown that such mixtures outperform the original ESR method on the "wild" LFW and AFLW datasets, while still allow real-time processing. We furthermore developed a simple method to utilize head-pose information in the shape fitting process. This method is not specific to the ESR algorithm and can also be used to integrate other clues. Comparison with related methods shows that the generalized ESR is capable of outperforming these methods on challenging real-world datasets.

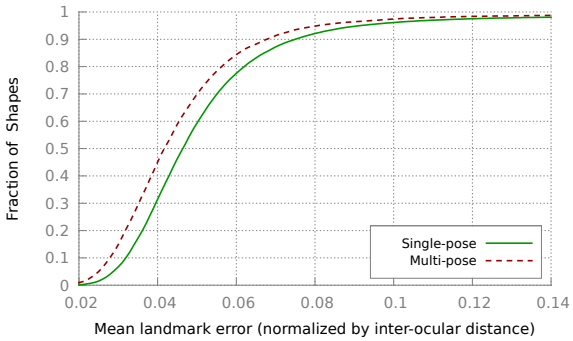


Figure 7. Mean landmark error of single- and multi-pose regression on the LFW dataset.

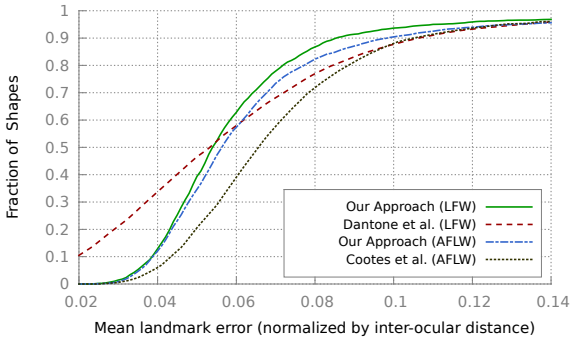


Figure 8. Comparison with other methods on the same datasets. Results of Dantone *et al*. and Cootes *et al*. were estimated from the corresponding publications [7] and [4]. See text for further limiting factors on the analysis of these graphs.

Acknowledgement. This work was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) within the CHIST-ERA project titled "Collaborative Annotation of Multimodal, Multi-lingual and Multimedia Documents (CAMOMILE)" project no. 112E176 and by the Concept for the Future of Karlsruhe Institute of Technology within the framework of the German Excellence Initiative.

References

[1] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by Explicit Shape Regression. In *CVPR*, pages 2887–2894, Jun 2012. 1, 2, 3, 4, 5

[2] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active Shape Models – Their Training and Application. *Computer Vision and Image Understanding*, 61(1):38–59, Jan 1995. 1

[3] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active Appearance Models. In *ECCV*, pages 484–498. Springer, 1998. 1

[4] T. F. Cootes, M. C. Ionita, C. Lindner, and P. Sauer. Robust and Accurate Shape Model Fitting Using Random Forest Regression Voting. In *ECCV*, pages 278–291. Springer, 2012. 2, 6

[5] D. Cristinacce and T. Cootes. Feature Detection and Tracking with Constrained Local Models. In *BMVC*, 2006. 1

(a) LFW Dataset.

(b) AFLW Dataset.

Figure 9. Selected Results. Detected landmarks are shown as black-and-white circles, ground-truth labels as cyan crosses.

[6] D. Cristinacce and T. F. Cootes. Boosted Regression Active Shape Models. In *BMVC*, pages 79.1 – 79.10, 2007. 1

[7] M. Dantone, J. Gall, G. Fanelli, and L. Van Gool. Real-time Facial Feature Detection using Conditional Regression Forests. In *CVPR*, pages 2578–2585. IEEE, 2012. 2, 4, 5, 6

[8] J. H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistic*, 29(5):1189–1232, 2001. 3

[9] J. H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, 2002. 4

[10] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007. 1, 4

[11] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization. In *BeFIT*, 2011. 1, 4

[12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 3

[13] M. Özuysal, M. Calonder, V. Lepetit, and P. Fua. Fast Keypoint Recognition Using Random Ferns. *Transactions on Pattern Analysis and Machine Intelligence*, 32(3):448–461, 2010. 3

[14] M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial Point Detection using Boosted Regression and Graph Models. In *CVPR*, pages 2729–2736. IEEE, 2010. 2