Sampling strategies for performance improvement in cascaded face regression

Romuald Perrot, Pascal Bourdon, David Helbert

Sampling strategies for performance improvement in cascaded face regression

Romuald Perrot[a], Pascal Bourdon[a], David Helbert[a]

[a] *XLIM-ASALI University of Poitiers, UMR CNRS 7252*

Abstract

Automatic face landmarking has received a lot of attention in the past decades. It is now mature enough to be implemented in fully autonomous video systems. As cascade-of-regression based algorithms have become state of the art in such systems, two major (and still relevant) sources of interest have slowly faded away: the need for semantic-driven learning beyond ground truth annotation, and full video chain performance *i.e.* tracking efficiency, which in the case of said methods strongly relates to their robustness towards shape initialization before fitting. In this paper, we investigate how data sampling using face priors can affect their performance in terms of convergence and robustness. We propose new strategies based on said priors to overcome inconsistencies observed during cascade-of-regression learning on purely random sampling-based stages. We will show that simple choices can be easily integrated within regression-based face tracking systems to increase accuracy and robustness.

Keywords: Face Landmarking, Regression, Sampling, Data augmentation

1. Introduction

Face tracking or face landmarking is an active topic and also a key tool for image and video analysis: authentification, emotion detection or face transfer are some of the applications relying on face landmarking.

The model used in cascaded face landmarking is a tree built with a random process. Nodes of the tree are created upon a set of features, which is inherently a limited set. As a result, a sampling scheme should again be employed to define this set. Like data augmentation, authors only rely on uniform sampling and the resulting trained models may not be optimal, with final step decisions leading to regression directions cancelling one another out, as we will illustrate later on.

Email addresses: romuald.perrot@univ-poitiers.fr (Romuald Perrot), pascal.bourdon@univ-poitiers.fr (Pascal Bourdon), david.helbert@univ-poitiers.fr (David Helbert)

Our contributions in this paper concern the study of sampling strategies used during two steps of the regression learning scheme: data augmentation and feature sampling. For both steps, we investigate several sampling methods found in literature as well as new propositions and investigate their impact on face fitting quality. Readers should keep in mind that while our field of investigation is primarily cascade-of-regression alignment, most other methods can or already benefit from such sampling strategies. In details, we propose:

1. Four sampling schemes for data augmentation, taking into account common knowledge about face geometry and dynamics ;
2. Two sampling schemes for features generation, with two opposite directions: better space coverage and landmark importance ;
3. An analysis of semantic-driven sampling strategies compared to conventional blind sampling.

The paper is structured as follows. Section 2 discusses previous work on feature sampling and data augmentation for face landmarking. Section 3 presents regression-based methods. In section 4 we expose various sampling schemes for building augmented sets of groundtruth shapes. Section 5 details sampling schemes for building feature sets. In section 6 we study the results of each sampling scheme in the context of face landmarking, and set a performance benchmark of our algorithm using a challenging faces-in-the-wild dataset, namely the 300W competition test-set [8, 9]. Finally section 7 concludes the paper.

2. Related work

In this section we only reference papers that are fundamental to understand our work. A full survey of face landmarking is beyond the scope of this paper. Readers can refer to recent surveys such as [10] or [11] for a detailed overview.

2.1. Face alignment

Historically, authors categorize face landmarking into three main methods: Active Shape and Active Appearance Models (ASM/AAM) [12, 13], Constrained Local Models (CLM) [14] and regression models [15]. AAM conjointly learn global texture and shape models, with face fitting consisting in minimizing the difference between a target face image and a deformable parametric texture model. Unfortunately, the idea of a global texture model is a major issue since it tends to drive the fitting process as a whole, turning common occurrences such as occlusion or light changes into a threat to result quality. To increase robustness against occlusions, CLM methods employ a local texture model around each landmark. While some CLM implementations allow interactive frame rate [16], they are usually computationally expensive and require high-performance hardware.

Regression-based methods have been claimed to enable both high-performance and high-robustness in face landmarking, even on limited hardware such as smart devices. Dollar *et al.* [15] introduce the cascaded pose regression method,

where a shape is progressively refined to a target shape. Cao *et al.* [1] enhance regression using a new shape indexation scheme and a boosted two-level cascade of regression. Kazemi *et al.* [2] provide a high-performance regression system with a simplified initialization stage and gradient boosted cascade building. At the same time, Ren *et al.* [3] announce three times speed-up using customized local binary features. While impressive performances are reported, almost every method suffers from the same issues. Yang *et al.* [17] show that such methods are highly sensitive to prior face detection performance, said detection being a mandatory step for initialization. As a result, despite solution proposals such as combined detection/regression [18], the issue of robustness towards initialization is still considered an open issue.

2.2. Features sampling

Most studies on features used in face landmarking have been done with robustness against face transformations in mind (mainly rotation and perspective transformation). Cao *el al.* [1] use shape-indexation, Burgos *et al.* [21] introduce interpolated shape features, which is later enhanced by Cao [22] using barycentric coordinates. Unfortunately, feature pool selection has not been well investigated. Dollar *et al.* [15], use uniform sampling; reference methods [1, 21, 2, 3] are based on Dollar's work thus use the same sampling strategy. To our concern, the only reference method that employs another method is the work of Cao *et al.* [22] where a Gaussian distribution over the unit square is used, although no specific justification or comparison with uniform sampling is provided by the authors. Kazemi *et al.* [2] observed that feature selection using distance priors leads to better fitting performance but feature generation is still based on uniform sampling.

2.3. Data augmentation sampling

Some authors have studied the impact of data augmentation on classification performance [23]. As an example, Krizhevsky *et al.* [24] perform Principal Component Analysis (PCA) on Red-Green-Blue (RGB) pixel values in a deep learning architecture to achieve the best results (at the time of publication) on the famous ImageNet classification challenge. De Vries *et al.* [25] also used data space to perform data augmentation. To our knowledge, such complex data-space augmentation methods have not been applied to regression-based face landmarking, and only blind, uniform selection of shapes is used during data augmentation. As an example, as PCA modelling has been criticized as the cause of ASM/AAM/CLM's failure to fit in-the-wild face shapes, anything PCA-related has been seemingly discarded from cascade-of-regression landmarking, including their use for data augmentation documented in [13, 26].

It is interesting to note that in the context of deep learning, where the number of training samples is often much higher, some works [27, 28] have been conducted in the opposite direction: sampling the training set to generate a smaller set that leads to the same fitting/classification error. The main aim is to reduce computational training cost.

In this paper, we propose new strategies for both data augmentation and feature sampling, where face semantic integration is induced implicitly by priors regarding data representation models and space partitioning. We show that these new strategies result in better fitting performance. We also show how rigid transformation strategies applied to data augmentation help increasing robustness to poor initialization, hence limiting the dependency and sensibility to face poor detection outputs.

3. Regression based face landmarking

In this section, we review the regression method used for face landmarking.

3.1. Random tree model

Regression-based face landmarking relies on supervised machine learning, using a *random tree* model [29] most of the time. Random trees are decision trees built using a random process. Internal nodes contain decisions that split samples into two disjoint subsets, and leaves containing a displacement vector. Usually, the depth of one tree is set and remains constant for all trees. Our method uses true random trees (*i.e.* each node has its own splitting decision).

A splitting function indicates, for a node, to which subtree a sample should belong. For any given sample $\mathbf{x}_i$, we define $k = \phi(\mathbf{x}_i)$ as the result of the splitting decision, where $k \in [l, r]$; l and r respectively for the left and right subtree. Function $\phi(.)$ behaves as a similarity measure between two descriptions (*i.e. features*) $\mathcal{F}_1$ and $\mathcal{F}_2$ using a threshold κ. It is defined as:

$$\phi(\mathbf{x}) = \begin{cases} l & \text{if} \quad d(\mathcal{F}_1, \mathcal{F}_2) > \kappa \\ r & \text{else.} \end{cases}$$

where $d(.)$ is a similarity distance operator *e.g.* subtraction, Euclidean norm, *etc.* Note that any feature description could be used, in face fitting or detection advanced features such as LBP [30, 31] or SIFT [32] where employed. In cascaded face regression, simple pixel intensity difference is used for efficiency reasons and we will keep this strategy throughout the paper.

3.2. Training set

A training set $\mathcal{T}$ is required to serve as a reference for landmarking performance and to build the model. Such a set is composed of images and their ground truth annotations (*i.e.* shapes). Creating a training set is usually a manual and long process where one has to indicate on each image where fiducial points or landmarks composing the shape are. There are some publicly available datasets using a various number of landmarks per images. Our method can work on any of them [9, 33, 34], as long as the number and order of the landmarks are consistent across all images within the dataset.

Let $\mathbf{x}_i = (\mathcal{I}_i, \mathbf{y}_i)$ be the i-th sample of a training set, with $\mathcal{I}_i$ the image and $\mathbf{y}_i$ the ground truth annotation. On top of a training set $\mathcal{T}$, an *augmented set*

$\mathcal{T}' = \{\mathbf{x}'_i\}$ is built using synthesized perturbations of annotations $\mathbf{y}'_i$. We define the residual $\mathbf{r}_i$ of sample $\mathbf{x}_i$ with respect to an augmented sample $\mathbf{x}'_i = (\mathcal{I}, \mathbf{y}'_i)$ as the difference between ground truth and noisy annotations:

$$\mathbf{r}_i = \mathbf{y}_i - \mathbf{y}'_i.$$

3.3. Face alignment

The key idea of face landmarking using regression is to iteratively refine a shape $\mathcal{S}^t$ to an optimal target shape accordingly with an input image $\mathcal{I}$, a single regression model $\mathcal{R}_t$, and a previous state $\mathcal{S}^{t-1}$. This process can be written as follows:

$$\mathcal{S}^t \leftarrow \mathcal{S}^{t-1} + \mathcal{R}_t(\mathcal{I}, \mathcal{S}^{t-1}).$$

The complete regression model $\mathcal{R}$ is the accumulation of all single regression models. With an abuse of mathematical notation, it can be written as:

$$\mathcal{R} = \sum_{t=1}^{T} \mathcal{R}_t.$$

Note that all single regression models $\mathcal{R}_t$ can differ from one another. In our case, a single regression model is a tree: given an input image and shape, it will compute an increment for the refinement process.

Current regression approaches rely on cascades of regressions. It is a two-level variation of the regression method: a single cascade c_i is composed of K trees t^i_j (with $j \in [1; K]$), where all trees within this same cascade share a common set of P features (see Figure 1). Hence, given T cascades, the total number of iterations in the regression process equals $T \times K$.

Regression starts with an initial shape $\mathcal{S}^0$ which can be, as suggested by Kazemi [2], the mean shape of the training set. This mean shape should be computed using a Generalized Procrustes Analysis [35] in order to compensate the rotational and scaling factors of the training samples.

While our fitting process could be combined using a coarse to fine approach such as the one proposed by Zhu *et al* [36], we decide in our implementation to keep fitting process as close as possible to the Kazemi work, since it makes our contributions more interpretable without any side effect.

3.4. Training process

The key to success with regression methods is precise and robust tree model building. The gradient boosting approach described in this section is the one proposed by Kazemi *et al.* [2]. It is simple to understand, easy to implement and very efficient. Each tree in the cascade is built using a recursive top-down approach using estimation residuals to drive construction phases.

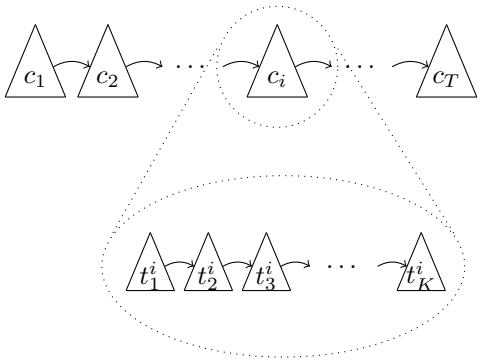


Figure 1: Regression Cascade. A cascade c_i is a sequence of regression trees t_j^i.

3.4.1. Node training

Given a set of splitting criteria Φ, the goal is to find the best ϕ_{opt} that divides the training set into two subsets $\mathcal{T}_l$ and $\mathcal{T}_r$ such that residuals of samples belonging to the same subtree leads to a unique value. Intuitively, this means that samples sharing similar face characteristics should belong to the same subtree. This corresponds to minimizing the following energy:

$$E(\phi, \mathcal{T}') = \sum_i \sum_{k \in \{l,r\}} \sum_{\phi \in \Phi} \|\mathbf{r}_i - \mu_{\phi,k}\|^2$$

where $\mu_{\phi,k}$ is the mean value of residuals for all samples i belonging to the same subtree k:

$$\mu_{\phi,k} = \frac{1}{|\mathcal{K}|} \sum_{i \in \mathcal{K}} \mathbf{r}_i \quad \text{with} \quad \mathcal{K} = \{i / \phi(\mathbf{x}_i) = k\}.$$

A straightforward implementation can be done as follows: pick R splitting criteria $\phi_j \in \Phi$, then select the one that minimizes energy E:

$$\phi_{opt} = \arg\min_{\phi_j} E(\phi_j, \mathcal{T}').$$

When a splitting criterion is selected, the training set is split into two disjoint sets. Each subtree of a newly built node is trained using the same optimization process with its corresponding subset.

3.4.2. Leaf value:

Tree construction ends when a maximum depth criterion is met. In that case, a leaf is created and its value is the mean of residuals for samples reaching that leaf. Authors usually employ a shrinking approach in order to avoid overfitting issues and reduce the influence of noisy data. In that case, the value of leaf $\mathcal{L}_k$ is computed as:

$$\mathcal{L}_k = \frac{\beta}{|\mathcal{K}|} \sum_{i \in \mathcal{K}} \mathbf{r}_i$$

where β is a shrinking factor which can be either defined as a constant [2] or computed for normalization and stability purposes [1].

3.4.3. Cascade training:

Training of the regression cascade is simply a sequence of tree trainings with an update on residuals before computing a next tree. Every training sample $\mathbf{x}'_i$ is updated such that:

$$\mathbf{y}'_i \leftarrow \mathbf{y}'_i + \mathcal{R}_t(\mathcal{I}_i, \mathbf{y}'_i).$$

In this section, we have reviewed how regression model is built. We identify two source of randomness: (i) the generation of the augmented data set $\mathcal{T}'$ (section 3.2), (ii) how features ϕ_j are producted to elaborate the feature set Φ. In the remaining of the paper, we propose various strategies for both problems.

4. Augmented shape sampling

In this section, we detail the sampling strategies used to generate augmented samples $\mathbf{y}'_i$. We assume that the training set has n elements.

Ground truth data augmentation strategies can be divided into two complementary sampling schemes: nonrigid face-space sampling and rigid position-space sampling. The first one tries to increase the generalization of the facial expressions while the second one tries to increase the robustness to incorrect bounding box initializations.

4.1. Non rigid face-space sampling (N-Rig)

4.1.1. Uniform sampling

A first strategy used in literature is to randomly select a shape within the training set as the noisy shape using a uniform distribution law:

$$\mathbf{y}'_i \leftarrow \mathbf{y}_\xi, \xi \in [1; n], \xi \neq i.$$

4.1.2. Linear interpolation (LI)

As noted by Cao *et al.* [1], the output of a trained regressor lies in a linear interpolation of all training shapes as long as the initial shape belongs to the training set. We propose a new data augmentation scheme based on that idea. We extend the sampling space through the introduction of new shapes which, despite not being part of the training set, can be considered *face like* nevertheless. This strategy consists in linearly interpolating two randomly selected training shapes $\mathbf{y}_{\xi_p}$ and $\mathbf{y}_{\xi_q}$ with a random interpolation factor α:

$$\mathbf{y}_{i\prime} \leftarrow (1-\alpha)\mathbf{y}_{\xi_p} + \alpha\mathbf{y}_{\xi_q},$$

with $\alpha \in [0; 1]$, $(\xi_p, \xi_q) \in [0; n]^2$ and $p \neq q$.

We argue that coherent, natural-looking face shapes can be obtained this way, thus reducing the risk of over-generalization during the training process. This can be observed on Figure 2 which presents some examples of faces generated using such an interpolation.

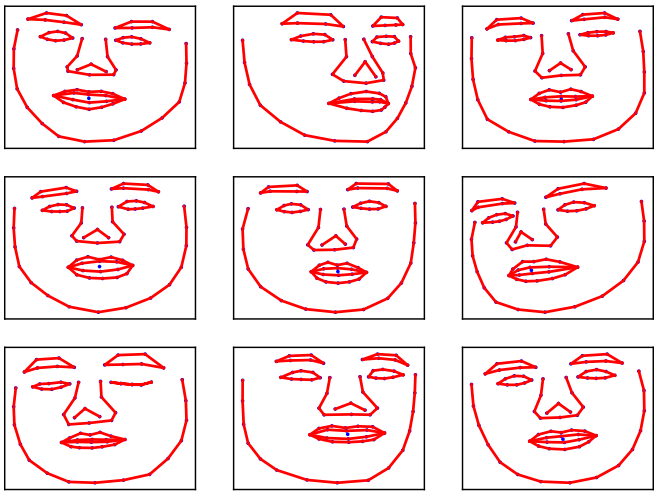


Figure 2: Example of shapes generated using linear interpolation sampling.

4.1.3. Flipping of shapes (FL)

Image flipping for data augmentation is a common practice in classification or indexation-related deep learning frameworks [37, 38, 39], where invariance towards global orientation is often sought after. It also sounds natural to extend this idea to faces, as an example to synthesize left eye winking out of right eye winking:

$$\mathbf{y}_i\prime \leftarrow \mathcal{M}(\mathbf{y}_i),$$

where $\mathcal{M}$ is the mirror operation (with respect to the y axis)

4.1.4. Model-Free Noise (MFN)

The classical freeform, random pertubation of landmark positions using simple Gaussian noise is studied as well. Let $\mathcal{L}(\mathbf{y}, k)$ be function which returns the k-th landmark position from any given shape $\mathbf{y}$, new augmented data can be written as:

$$\mathcal{L}(\mathbf{y}_i\prime, k), \leftarrow \mathcal{L}(\mathbf{y}_i, k) + \delta_{\mathbf{u}}$$

where $\delta_{\mathbf{u}}$ is a centered random vector which variance is defined with respect to a metric on the shape (ex: 5% of the shape size).

Note that this sampling strategy does not ensure realistic face outputs (see Figure 3). Facial expressions remain unchanged for the most part, while landmarks positions are being altered on an individual basis. The aim is to deal with small variations on the training set annotations which could arise from faulty manual annotations as an example.

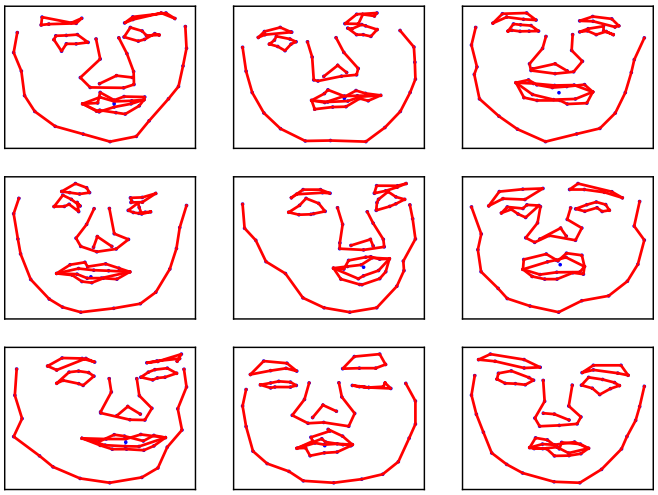


Figure 3: Shapes generated using perturbation of 50% of the landmarks with maximum magnitude of 5% of the shape size.

4.1.5. Model Noise (MN)

This approach is a simplified version of DeVries' strategy *et al.*[25], where a recurrent deep network is used to augment data. They propose a two step strategy:

1. First, build a feature space representation of the training set using an autoencoder ;
2. Second, compute an augmented sample by projecting a sample into the feature space, apply perturbation in the feature space, then project the augmented sample back into the primal space.

Instead of relying on a complex, deep learning network architecture, we build the feature space using PCA. This results in a simpler implementation one can relate to either Krizhevsky's use of PCA for texture modeling [24] or previous works on face alignment using Cootes' ASM or AAM [12, 13].

Assuming PCA vector bases are sorted according to their eigenvalues, we perform a classical dimension reduction scheme for the PCA by keeping only the first m vectors of the PCA basis as feature (*i.e.* projection) space. Augmented samples are drawn by generating a random vector $\mathbf{u} = (\xi_1, \xi_2, ..., \xi_m)$ in the feature space, then projecting it back to the original space. With an abuse of notation, augmented samples could be generated as:

$$\mathbf{y}'_i \leftarrow \bar{\mathbf{y}} + PCA^{-1}(\mathbf{u})$$

where PCA is the PCA projection of the training set, PCA^{-1} is the back projection function, and $\bar{\mathbf{y}}$ is the mean shape.

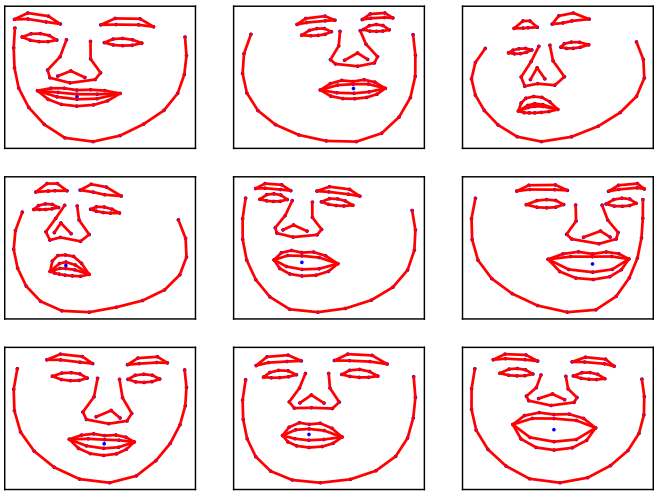


Figure 4: Example of shapes generated using PCA sampling

Because this strategy is based on the training set, it will generate faces that are coherent with respect to the training set. Unlike the linear interpolation that tends to generate faces that are near the mean face, we think this strategy generate faces that are more general and with more variation. Figure 4 shows some example of PCA sampled faces, note that mouth variation seems to highlight more variation than with linear interpolation.

4.2. Rigid transformation sampling

A major threat to regression-based face landmark alignment is its need for coarse yet rather precise face position, scale and orientation initializations. Rough bounding box estimations or different face detection algorithms have a strong influence on robustness [17].

In order to deal with approximative bounding box detections, we define a 4-parameters sampling scheme that simulates rigid transformation on the shape. Let parameters ξ_s, ξ_x, ξ_y, ξ_θ be respectively a random scale, random displacement on x and y, as well as a random rotation of angle θ. We define the rigid augmentation strategy as:

$$\mathbf{y}'_i \leftarrow \xi_s R_{\xi_\theta} \mathbf{y}_j + (\xi_x, \xi_y)$$

where $\mathbf{y}_j$ is a shape. Note that this strategy is used in combination with the face space sampling, in that case, $\mathbf{y}_j$ is the result of any facial-sampling scheme previously presented.

5. Feature sampling

In this section, we present several strategies used for feature sampling within cascade trees. Features are 2d positions in the image plane and P is the number of features sampled within each cascade. We define $\mathcal{F}_i$ as the position of the i-th feature .

In common regression-based face landmarking implementations, samples are drawn uniformly inside the bounding box of the mean shape. We define this box using parameters (x_0, y_0), w and h, respectively the top left corner of the bounding box, its width and height. Note that we use a coordinate system where $\vec{y}$ axis is pointing down.

For uniform sampling schemes, features are drawn as:

$$F_i \leftarrow (x_0, y_0) + (\xi_1 \cdot w, \xi_2 \cdot h)$$

where ξ_1 and ξ_2 are random real values taken within the range $[0; 1]$.

A slightly modified version of this algorithm relies on an extended bounding box, in order to make sure samples can be generated in every landmark surrounding.

5.1. Analysis of trained cascades

We train a regression model using $P = 500$ features with $T = 10$ and $K = 500$, then we extract the position of the features in all trees and finally we compute density of the tree feature positions. Figure 5 shows the resulting density.

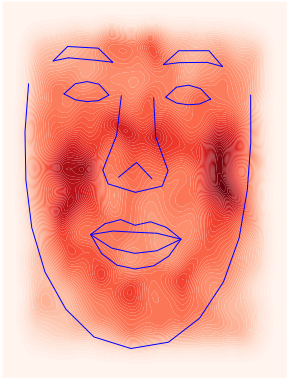


Figure 5: Density of trained features using uniform sampling.

The first observation we can make is that the position of the features is not uniform over the sampling space. Some areas present high densities, especially

in the upper cheeks. Other areas have surprisingly low densities (*e.g.* in the center of the mouth). Note that features are mainly located inside the convex hull of the face; this would suggest that extended bounding box sampling (as discussed at the end of the preceding section) may not be necessary.

We now analyse the position of features with respect to the depth of the features inside the tree. All features located at depth=0 are extracted, then those at depth=1 and so on. Figure 6 shows the density for each depth ranging from 0 to 4.

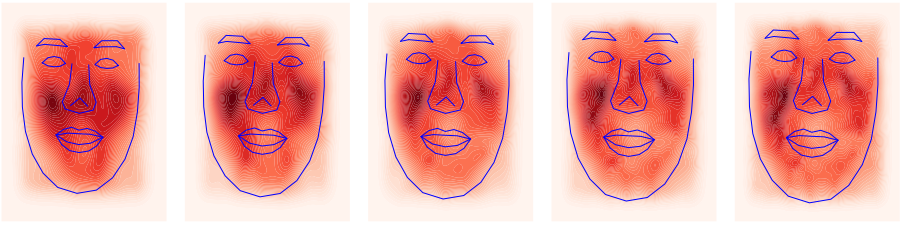


Figure 6: Density of the features with respect to the depth of the tree. From left to right, density at depth from 0 to 4

We can observe that the density of features located at the top of the trees presents a large horizontal symmetry, while features located lower in the trees seem to be more specialized, with very low-density areas such as the nose area.

In order to make sure that sample distribution is not biased by the random generator being used, we first derive a sampling scheme that ensures a better spatial distribution of features over the sample space. Then we analyze the resulting feature densities with respect to this new sampling scheme.

5.2. Stratified sampling

In blind uniform sampling, especially with a small number of samples P, the sampling space may be under-sampled. This can result in parts of the sampling space where no features are drawn (Figure 7). We suspect this may have an impact on regression quality, since large parts of the sampling space are omitted. To overcome this issue, we propose to use stratified sampling (also known as *stratified-jittered* or *jittered* sampling). In this strategy, the sampling space is divided in a regular grid of size $m \times m$, with:

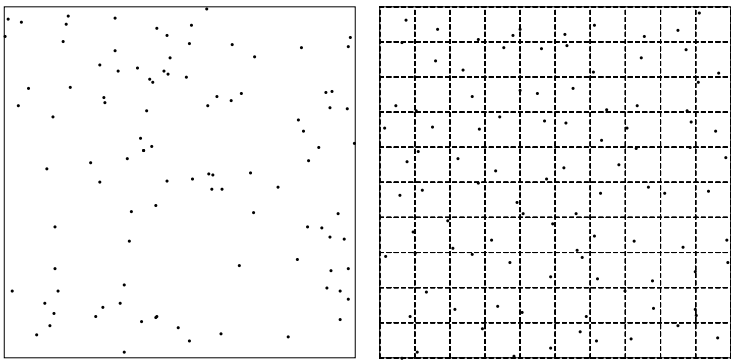
$$m = \lfloor \sqrt{P} \rfloor$$

where $\lfloor x \rfloor$ is the floor function:

$$\lfloor x \rfloor = \max \{m \in \mathbb{Z} | m \leq x\}$$

In each cell of the grid, a sample is drawn using uniform sampling (Figure 7b)

To be consistent with the other methods, where the number of samples equals P, the remaining samples $m' = P - m^2$ samples are drawn using uniform sampling over the whole sampling space.



(a) (b)

Figure 7: Sampling schemes for 100 points. (a) Uniform sampling (b) Stratified sampling.

5.3. Importance sampling

In [1] Cao *et al.* mention future works about clever sampling strategies that could exploit salient parts of faces (eye, mouth, ...), without providing more details. Here we extend this idea through a sampling scheme that facilitates Region-Of-Interest (ROI) sampling. We first analyze the density of features trained with stratified sampling on Figure 8.

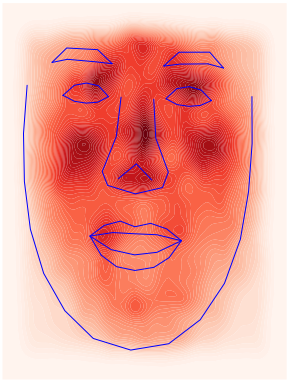


Figure 8: Density of the selected features using stratified sampling.

We can observe on Figure 8 how features are still locating around specific areas, yet how density distribution is now different compared to uniform sampling. Especially, we can notice that areas close to the eyes and nose have more importance in stratified sampling than they have in uniform sampling. We argue that this could be a hint on how samples should be drawn. Stratified sampling

appears to force samples to be selected near areas where salient information can be found (eyes and nose). This will drive our final sampling scheme.

In this scheme, we draw more samples next to fiducial points, since they are by definition points of importance. We use a sampling grid that is similar to the one used in stratified sampling, only with a fixed number of cells which does not dependent on the number of samples.

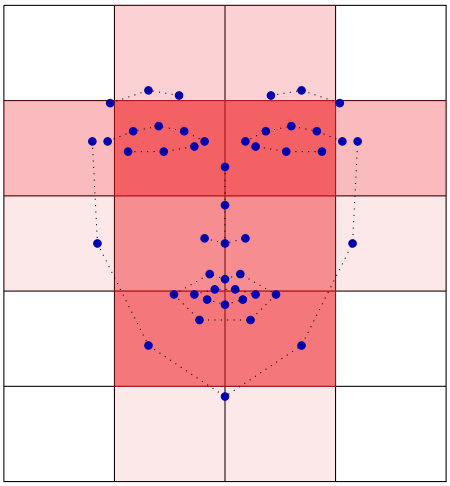


Figure 9: Sampling scheme using importance sampling and grid sampling. For each cell, the number of samples is directly tied to the number of landmarks (the darker, the higher). Note that this partition is not the actual subdivision, it is just an illustration.

Assuming a grid is composed of cells $\mathcal{C}_j$, the sampling process is a two-phase one. In a first step, importance is computed using the mean shape: for each cell, importance is increased given the number of landmarks it contains (Figure 9). In a second step, samples are selected given cell importance: the more important a cell is, the more samples are drawn. Samples within a cell are generated using uniform sampling, as they were in stratified sampling.

Formally, inside a cell $\mathcal{C}_i$, the number of samples n_i is equal to:

$$n_i = \frac{k_i}{m}$$

where k_i is the number of landmarks that fall into the cell $\mathcal{C}_i$, and m is the total number of landmarks.

6. Experimental results

This section analyzes the sampling strategies proposed in this paper, namely data augmentation and feature sampling. We also provide experimental results for comparison with other face landmarking methods.

6.1. Methodology

Sampling strategy performance assessment is made using two well-known face datasets to generate the training set: LFW [40] and FaceWarehouse [41]. While LFW is made of *in the wild* situations, FaceWarehouse present controlled strict face acquisitions. The training set is composed of 6400 random samples taken from the two sets. The testing set is obtained in a similar fashion using 1600 samples in the remaining set of images (*i.e.* no overlap between the training and testing sets). We use the publicly available databases provided by Cao *et al.* (`http://gaps-zju.org/DDE/`)

Landmark position errors are computed as the median distances between fitted shapes and their ground truth counterparts. To limit the impact of spatial resolution differences between testing samples, all results are normalized with respect to the intra-ocular distance, as literature usually suggests. During training, the bounding box used for initialization is the minimal one that encompasses ground truth shapes *i.e.* it is not the result of a face detection algorithm.

In order to evaluate the performance of our algorithm not just within the realm of cascade-of-regression-based methods but also with regard to current state-of-art face landmarking methods, at the end of this section we provide additional results obtained on the challenging test-set of 300W competition [8, 9] in terms of cumulative error curves, Area-Under-Curve (AUC) values, failure rate and computational time.

6.2. Data augmentation

In this section, we review the performance of the data augmentation strategies. We explore all combinations using all sampling schemes presented in the previous sections, resulting in 31 different situations. We also test two initialization strategies: the theoretical one, using ground truth bounding boxes, and a more realistic one, using random perturbations of said theoretical bounding boxes. While the former method can be used to benchmark the fitting power of the model, the later is headed towards real field application and robustness analysis. This leads to a total of 62 combinations. In all graphics, "Non Rig" means regression model with only non-rigid sampling scheme, "Non Rig+Rig" means nonrigid data augmentation followed by rigid sampling, and "ξ_{init}" means initial bounding box perturbed. Note that initial bounding box perturbation consists in random deformations of $\pm 25\%$ in scale and random $\pm 25\%$ move in translation.

Each regressor is trained using $T = 10$ cascades, $K = 500$ trees per cascade, $P = 500$ features per cascade, $S = 300$ augmented samples, and the splitting set Φ is composed of $R = 50$ functions at each node building stage. We empirically chose the first 6 vectors of the PCA base when building the PCA sampling space, which corresponds to the vectors holding most of the variation among all faces. When more than one sampling scheme is used, the number of augmented samples drawn for each scheme is S/n, with n the number of splitting schemes.

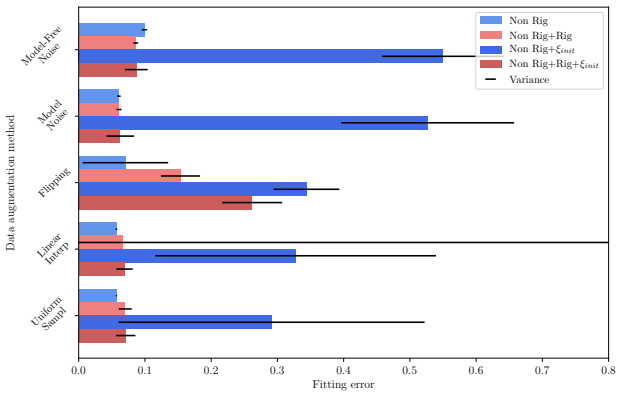


Figure 10: Comparison of data augmentation sampling schemes

Figure 10 presents the results obtained by comparing all methods. We can note that uniform selection leads to the best performance with the exact bounding box, while Model Noise augmentation gives the best performance with an initially perturbed bounding box. Regression models trained with rigid and non-rigid augmentation get consistent results with or without initialization noise, while regressors trained with only non-rigid sampling perform poorly with noisy initializations.

Several observations can be made about these experiments. The first one is about theoretical versus real-life contexts: if we have a precise, *learned regressor-friendly* position of the initial bounding box, data augmentation with non-rigid transformation performs always better than non-rigid combined with rigid. This can be explained by high variations in the augmented set increasing the risk of over-generalization (*i.e.* producing training samples that are unlikely to be encountered in the fitting context). In real-field situations *i.e.* when adding noise to the initial face bounding box, data augmentation with rigid transformations largely outperforms augmentation with only non-rigid transformations: it almost completely fixes poor initial bounding box positioning.

Another observation is that the random perturbation of landmarks doesn't affect or reduce testing errors. It is actually almost always the worst strategy. As discussed earlier, this can be explained by the fact that a freeform perturbation model is likely to generate sample shapes that do not look like faces, hence increasing the risk of over-generalization.

Overall, Model Noise (PCA sampling) appears to produce the best results. To highlight this, we produce a histogram of the occurrence of a sampling scheme in the best score (Figure 11). Model Noise, uniform selection and linear interpolation give best performance either globally (*i.e.* taking all possible combinations) or locally (*i.e.* counting with sampling is best using 1, 2, ..., 5 combinations). These results are confirmed by the mean rank of each sampling scheme, as exposed in Figure 12. In this figure, we determine the rank of each combina-

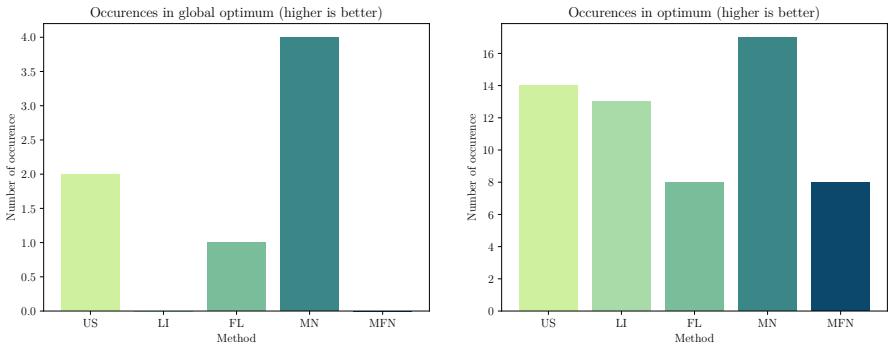


Figure 11: Comparison of the data augmentation methods. (a) Global optimum (b) Per number of combination optimum.

tion and compute the mean of the ranks where a specific sampling occurs. As suggested, Model Noise-based data augmentation has the lowest rank in almost all situations.

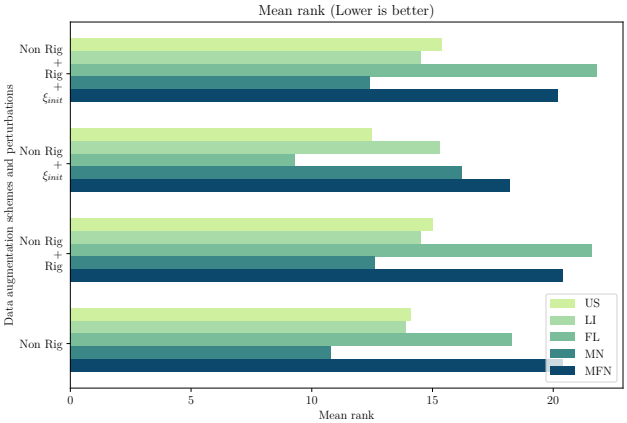


Figure 12: Mean rank of each sampling scheme for every configuration. Non Rig : Non rigid sampling scheme only, Non Rig+Rig : Non rigid and rigid sampling scheme ; ξ_{init} : Random perturbation of the initial bounding box.

The only situation where it is not the best solution is using nonrigid perturbation only for training and testing with a perturbed initial bounding box. Because this situation is unrealistic, it may be discarded from the interpretation of the results. With that consideration, rank ordering is consistent across all methods. This gives the following top-3 rank (from the best strategy to the worst one): Model Noise (MN), Linear Interpolation (LI) and Uniform Selection (US).

6.3. Feature sampling

In this experiment, we study the influence of feature sampling within tree cascades, based on median regression errors, for the three strategies presented in section 5: purely random sampling, stratified sampling and importance sampling. Regression models are built using $T = 10$, $K = 500$, $S = 300$, $R = 50$. For each method, we study the importance of the strategy regarding the number of features per cascade P. Data augmentation is made using uniform selection only and regression is performed with ground truth initial bounding box.

6.3.1. Stratified sampling

In a first step, we study the impact of stratified sampling to drive the generation of the P features. Figure 13 shows the difference between purely random sampling and stratified sampling. Interestingly, stratified sampling results in lesser errors with higher values of P. This can be explained by a better distribution of samples over space. Nevertheless, difference is not significant : less than 2% at maximum, as seen on Figure 13.

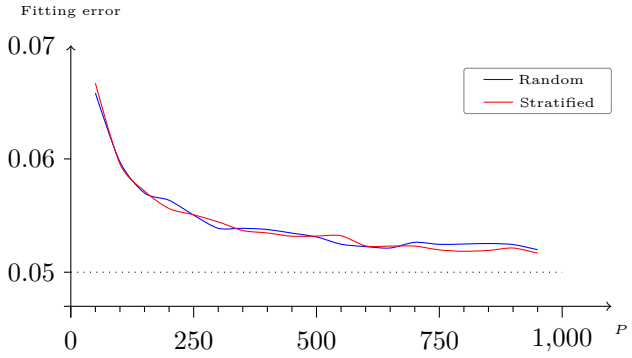


Figure 13: Difference between pure random sampling and stratified importance sampling of the P features. Blue line is the random sampling scheme, red is the stratified one.

6.3.2. Importance sampling

In this scheme, we first divide the sampling space into a 10×10 partition grid. Then, given landmark positions over the mean shape, we compute the importance of grid cells as explained in section 5.3. Finally we draw the feature using a two step approach: a cell is selected using importance sampling, then we uniformly sample a new feature within this selected cell. Figure 14 shows the difference between this strategy and the uniform one.

This strategy results in better results than pure random sampling or even stratified sampling ; in our experiments, only one result has a lower error with pure random sampling. This performance can by explained by the fact that importance sampling tends to introduce priors about face semantics *i.e.* fiducial points. As a result, whenever the last sets of cascade tree nodes start struggling between specialization and generalization over the full feature space, higher

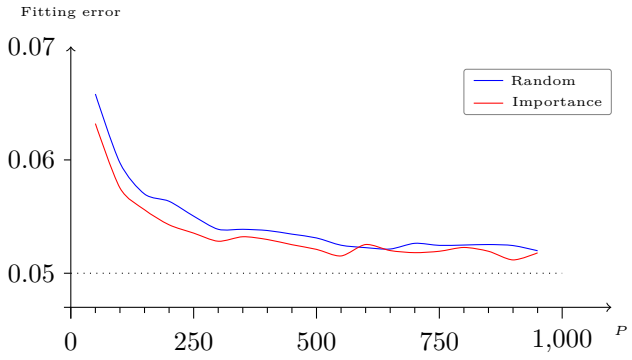
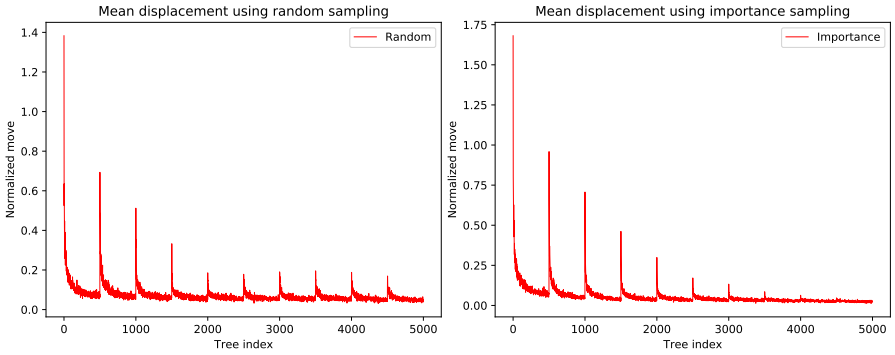


Figure 14: Difference between pure random sampling and importance sampling. Blue : random sampling. Red : importance sampling.

probability to focus on relevant areas such as eyes or mouths ensures feature reliability.

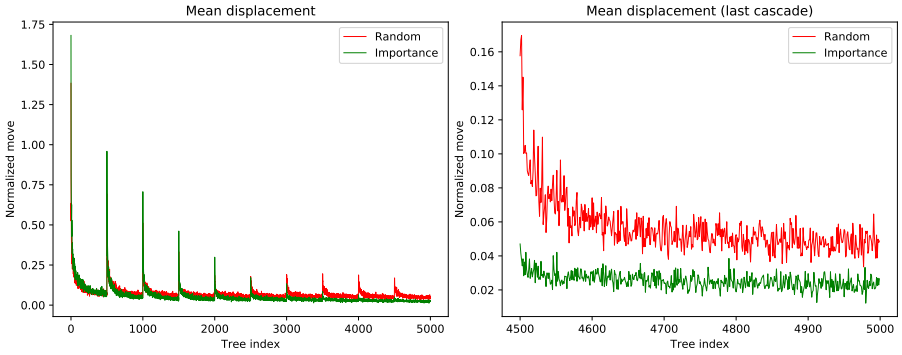
We lead further investigation in this analysis by claiming that importance sampling accelerates regression model convergence. In order to prove this, we study landmark displacement increments within tree leaves on trained regressors. Figure 15 presents mean 2D displacement values using random sampling and using importance sampling.



(a) (b)

Figure 15: Mean landmark displacement over regression tree leaves using (a) random sampling and (b) importance sampling.

Several observations can be made from this figure. First, importance sampling starts with a mean displacement that is higher than random sampling (1.75 compared to 1.4). Next, importance sampling displacement tends to converge nicely into a minimum when a number of about 4000 to 5000 cascades has been reached, while random sampling cascades seem to be locked into a periodic loop. We interpret this as non-convergence of the regression model.



(a) (b)

Figure 16: Comparison of the mean displacement of the leaves using random sampling and importance sampling. Using (a) all cascades ; (b) the last cascade.

The regression model with importance sampling has progressively less displacement compared to pure random sampling. This can be underlined by overlapping the displacement in the same plot. In figure 16 we overlap mean displacements of the two methods using all cascades and zooming over the last one. This last cascade clearly shows that landmark position increments are much higher with random sampling than they are with importance sampling.

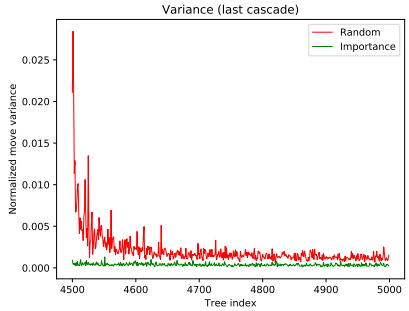


Figure 17: Variance of the mean displacement contained in the trees of the last cascade.

Variance is also highly reduced as shown in figure 17. Even at the end of the last cascade, variance values with importance sampling equal about half the values observed on random sampling. Thus confirming that convergence is better with importance sampling.

7. Conclusion

In this paper, we have investigated data sampling within state of art, cascade-of-regression based automatic face landmarking algorithms, and how semantic-

driven (e.g. basic knowledge about face geometry and dynamics) data modelling strategies can affect such systems in terms of convergence and robustness. We have proposed new sampling schemes in that respect to build augmented data sets before the learning phase, as well as sampling schemes for features generation in cascaded regression trees.

We have provided experimental results showing that simple data augmentation strategies, such as the proposed (PCA) model-based alteration of groundtruth face shapes, can increase the performance of the regression model. We have also demonstrated that the use of semantic priors during features generation has a significant positive effect on convergence, paving the way for another proposal called *importance sampling* which also improves face regression quality.

So far we have only studied shape-related sampling strategies. An additional study should be done in future works, where texture sampling is investigated as well for data augmentation. We expect to achieve better robustness to illumination and occlusion with this new study.

References

[1] X. Cao, Y. Wei, F. Wen, J. Sun, Face alignment by explicit shape regression, in: in CVPR, 2012.

[2] V. Kazemi, J. Sullivan, One millisecond face alignment with an ensemble of regression trees, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1867–1874.

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

[4] A. Jourabloo, M. Ye, X. Liu, L. Ren, Pose-invariant face alignment with a single cnn, in: 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, 2017, pp. 3219–3228.

[5] G. Trigeorgis, P. Snape, M. A. Nicolaou, E. Antonakos, S. Zafeiriou, Mnemonic descent method: A recurrent process applied for end-to-end face alignment, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4177–4187.

[6] A. Bulat, G. Tzimiropoulos, How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks), in: International Conference on Computer Vision, Vol. 1, 2017, p. 8.

[7] J. Deng, Q. Liu, J. Yang, D. Tao, M3 csr: Multi-view, multi-scale and multi-component cascade shape regression, Image and Vision Computing 47 (2016) 19–26.

[8] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, M. Pantic, 300 faces in-the-wild challenge: Database and results, Image and Vision Computing 47 (2016) 3–18.

[9] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, M. Pantic, 300 faces in-the-wild challenge: The first facial landmark localization challenge, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 397–403.

[10] N. Wang, X. Gao, D. Tao, X. Li, Facial feature point detection: A comprehensive survey, CoRR abs/1410.1037.
URL http://arxiv.org/abs/1410.1037

[11] X. Jin, X. Tan, Face alignment in-the-wild: A survey, CoRR abs/1608.04188.
URL http://arxiv.org/abs/1608.04188

[12] T. F. Cootes, C. J. Taylor, D. H. Cooper, J. Graham, Active shape models-their training and application, Computer vision and image understanding 61 (1) (1995) 38–59.

[13] T. F. Cootes, G. J. Edwards, C. J. Taylor, Active appearance models, in: European conference on computer vision, Springer, 1998, pp. 484–498.

[14] D. Cristinacce, T. F. Cootes, Feature detection and tracking with constrained local models., in: BMVC, Vol. 1, 2006, p. 3.

[15] P. Dollár, P. Welinder, P. Perona, Cascaded pose regression, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 1078–1085.

[16] T. Baltrusaitis, P. Robinson, L.-P. Morency, Constrained local neural fields for robust facial landmark detection in the wild, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 354–361.

[17] H. Yang, X. Jia, C. C. Loy, P. Robinson, An empirical study of recent face alignment methods, CoRR abs/1511.05049.
URL http://arxiv.org/abs/1511.05049

[18] D. Chen, S. Ren, Y. Wei, X. Cao, J. Sun, Joint cascade face detection and alignment, in: European Conference on Computer Vision, Springer, 2014, pp. 109–122.

[19] H. Liu, J. Lu, J. Feng, J. Zhou, Learning deep sharable and structural detectors for face alignment, IEEE Transactions on Image Processing 26 (4) (2017) 1666–1678.

[20] H. Liu, J. Lu, J. Feng, J. Zhou, Two-stream transformer networks for video-based face alignment, IEEE transactions on pattern analysis and machine intelligence.

[21] X. P. Burgos-Artizzu, P. Perona, P. Dollár, Robust face landmark estimation under occlusion, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1513–1520.

[22] C. Cao, Q. Hou, K. Zhou, Displaced dynamic expression regression for real-time facial tracking and animation, ACM Transactions on Graphics (TOG) 33 (4) (2014) 43.

[23] S. C. Wong, A. Gatt, V. Stamatescu, M. D. McDonnell, Understanding data augmentation for classification: when to warp?, CoRR abs/1609.08764. URL http://arxiv.org/abs/1609.08764

[24] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012.

[25] T. DeVries, G. W. Taylor, Dataset Augmentation in Feature Space, ArXiv e-printsarXiv:1702.05538.

[26] J. Saragih, R. Göcke, Learning aam fitting through simulation, Pattern Recognition 42 (11) (2009) 2628–2636.

[27] B. Harwood, V. K. BG, G. Carneiro, I. Reid, T. Drummond, Smart mining for deep metric learning, space 9 (13) (2017) 22.

[28] R. Manmatha, C.-Y. Wu, A. J. Smola, P. Krahenbuhl, Sampling matters in deep embedding learning, in: 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, 2017, pp. 2859–2867.

[29] L. Breiman, Random forests, Machine learning 45 (1) (2001) 5–32.

[30] Y. Duan, J. Lu, J. Feng, J. Zhou, Learning rotation-invariant local binary descriptor, IEEE Transactions on Image Processing 26 (8) (2017) 3636–3651.

[31] Y. Duan, J. Lu, J. Feng, J. Zhou, Context-aware local binary feature learning for face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[32] X. Xiong, F. De la Torre, Supervised descent method and its applications to face alignment, in: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE, 2013, pp. 532–539.

[33] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, N. Kumar, Localizing parts of faces using a consensus of exemplars, IEEE transactions on pattern analysis and machine intelligence 35 (12) (2013) 2930–2940.

[34] V. Le, J. Brandt, Z. Lin, L. Bourdev, T. S. Huang, Interactive facial feature localization, in: European Conference on Computer Vision, Springer, 2012, pp. 679–692.

[35] J. C. Gower, Generalized procrustes analysis, Psychometrika 40 (1) (1975) 33–51.

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

[37] F. Chollet, et al., Keras, `https://github.com/fchollet/keras` (2015).

[38] R. Collobert, S. Bengio, J. Marithoz, Torch: A modular machine learning software library (2002).

[39] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems, software available from tensorflow.org (2015).
URL `http://tensorflow.org/`

[40] G. B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, Tech. rep., Technical Report 07-49, University of Massachusetts, Amherst (2007).

[41] C. Cao, Y. Weng, S. Zhou, Y. Tong, K. Zhou, Facewarehouse: A 3d facial expression database for visual computing, IEEE Transactions on Visualization and Computer Graphics 20 (3) (2014) 413–425.

[42] R. Gross, I. Matthews, J. Cohn, T. Kanade, S. Baker, Multi-pie, Image and Vision Computing 28 (5) (2010) 807–813.

[43] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, W. Worek, Overview of the face recognition grand challenge, in: Computer vision and pattern recognition, 2005. CVPR 2005. IEEE computer society conference on, Vol. 1, IEEE, 2005, pp. 947–954.

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

[45] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, M. Pantic, A semi-automatic methodology for facial landmark annotation, in: Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on, IEEE, 2013, pp. 896–903.

[46] S. Milborrow, T. Bishop, F. Nicolls, Multiview active shape models with sift descriptors for the 300-w face landmark challenge, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 378–385.

[47] S. Jaiswal, T. Almaev, M. Valstar, Guided unsupervised learning of mode specific models for facial point detection in the wild, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 370–377.

[48] E. Zhou, H. Fan, Z. Cao, Y. Jiang, Q. Yin, Extensive facial landmark localization with coarse-to-fine convolutional network cascade, in: Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on, IEEE, 2013, pp. 386–391.

[49] M. Hasan, C. Pal, S. Moalem, Localizing facial keypoints with global descriptor search, neighbour alignment and locally linear models, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2013, pp. 362–369.

[50] J. Yan, Z. Lei, D. Yi, S. Li, Learn to combine multiple hypotheses for face alignment, in: Computer Vision Workshops (ICCV-W), Sydney, Australia, 2013 IEEE Conference On. IEEE, 2013.

[51] H. Fan, E. Zhou, Approaching human level facial landmark localization by deep learning, Image and Vision Computing 47 (2016) 27–35.