Generalizing to Unseen Head Poses in Facial Expression Recognition and Action Unit Intensity Estimation

Philipp Werner[1], Frerk Saxen[1], Ayoub Al-Hamadi[1], and Hui Yu[2]
[1] Neuro-Information Technology Group, Otto-von-Guericke University Magdeburg, Germany
[2] Visual Computing Group, University of Portsmouth, UK

***Abstract*— Facial expression analysis is challenged by the numerous degrees of freedom regarding head pose, identity, illumination, occlusions, and the expressions itself. It currently seems hardly possible to densely cover this enormous space with data for training a universal well-performing expression recognition system. In this paper we address the sub-challenge of generalizing to head poses that were not seen in the training data, aiming at getting along with sparse coverage of the pose subspace. For this purpose we (1) propose a novel face normalization method called FaNC that massively reduces pose-induced image variance; (2) we compare the impact of the proposed and other normalization methods on (a) action unit intensity estimation with the FERA 2017 challenge data (achieving new state of the art) and (b) facial expression recognition with the Multi-PIE dataset; and (3) we discuss the head pose distribution needed to train a pose-invariant CNN-based recognition system. The proposed FaNC method normalizes pose and facial proportions while retaining expression information and runs in less than 2 ms. When comparing results achieved by training a CNN on the output images of FaNC and other normalization methods, FaNC generalizes significantly better than others to unseen poses if they deviate more than 20° from the poses available during training. Code and data are available.**

I. Introduction

Face normalization has been proven to be beneficial across several domains of face analysis including facial expression recognition [29], [9], face recognition [17], [52], [45], or gender recognition [17]. In its simplest form, face normalization (also called face registration or frontalization) compensates variation in face position, scale, and in-plane rotation. More advanced methods aim to remove the effects caused by out-of-plane rotations (head turned away), different facial proportions, expression [52], illumination [54], [45], occlusion [32], or background. The basic idea is to gain invariance regarding such nuisance factors by reducing their influence o n the extracted features; this can improve discriminative power for the recognition task at hand. In facial expression analysis both head pose and individual differences in facial shape and texture are a challenge [11]; normalizing these factors is beneficial i f t he e xpression i nformation i s p reserved, as it reduces within-class variance. Previous face normalization approaches, which we discuss in Sec. III, have at least one of the following limitations: (1) they do not frontalize out-of-plane poses, (2) they lose expression information or introduce artifacts, (3) they require training data covering all degrees of freedom (see Abstract), (4) they are too slow for real-time expression recognition or require heavy GPU computation.

This work was funded by the German Federal Ministry of Education and Research (BMBF), grants 03ZZ0470 and 03ZZ0443G. The sole responsibility for the content lies with the authors.

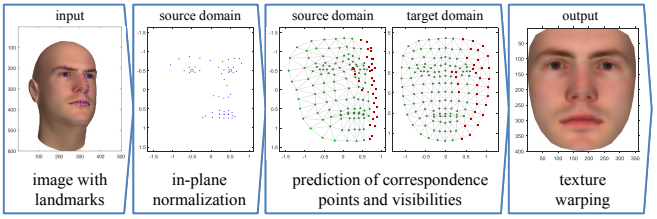

Fig. 1. Processing chain of the proposed method FaNC.

Aiming at head-pose-invariant real-time facial expression recognition systems, **we contribute** a **novel face normalization method** called FaNC (Sec. II). It learns to predict coordinates and visibilities of correspondence points from facial landmarks. The predicted information is used to generate a face image that is normalized regarding pose and facial proportions. FaNC can be learned and applied on top of any landmark localizer, also without facial contour landmarks, and runs in less than 2 ms even on cheap on-board GPUs. We review related work (Sec. III) and **compare normalization methods' impact** on deep learning based facial action unit intensity estimation and expression recognition (Sec. IV). To the authors best knowledge, we present (1) the first extensive analysis of **generalization to unseen head poses** and individuals and (2) the first **cross database evaluation** in which frontalization was developed and trained completely on another dataset than the datasets used for evaluation. We conduct experiments on the FERA 2017 challenge dataset and the Multi-PIE dataset, in which our proposed FaNC normalization method outperforms others on previously unseen head poses and individuals. Further, we discuss **which poses are needed in training data** to perform well across others. Data and code are available for research at `http://iikt.ovgu.de/FaNC.html`.

II. Face Normalization based on Learning Correspondences

In this Section we propose **Fa**ce **N**ormalization based on learning **C**orrespondences (**FaNC**), a method that can be applied on top of any facial landmark localizer. The core component is the prediction of correspondence point coordinates and visibilities from automatically detected landmarks. This mapping can be learned to handle different

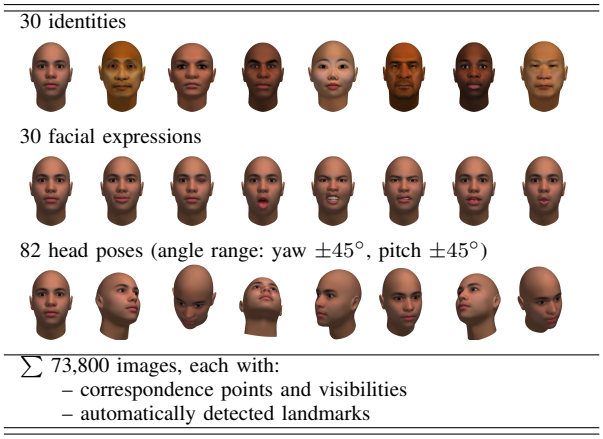

Fig. 2. SyLaFaN database: 3 degrees of freedom are varied systematically.

face normalization tasks, such as pure frontalization (pose compensation), normalization of pose and expression, or normalization of pose and identity-related factors (facial proportions). In this paper, we target the latter, since both pose and identity can be considered nuisance factors for recognizing facial expression. Fig. 1 gives an overview of the method. An arbitrary image $\mathbf{F}$ with facial landmarks $\mathbf{l}$ is the input of the algorithm. Landmarks are normalized through an in-plane transformation (Sec. II-B), followed by prediction of correspondence point coordinates in both source domain (arbitrary image) and target domain (frontal image), see Sec. II-C, and by prediction of the correspondence points' visibility (Sec. II-D). Finally, the normalized image is created from the input image by piecewise affine warping based on the predicted coordinates, whereas disocclusion is handled by blending and mirroring (Sec. II-E). For training the method, we create a synthetic dataset, which is described in the following section.

A. SyLaFaN Database

We introduce the **Sy**nthetic dataset for **La**ndmark based **Fa**ce **N**ormalization (**SyLaFaN**). It contains 73,800 images rendered using the FaceGen 3D morphable model (3D-MM similar to [8], `https://facegen.com/`). Identity, facial expression, and head poses are varied systematically (see Table 2). Illumination and occlusion, which are other challenging factors for face normalization, are not varied in the dataset, since they are handled better and better with new landmark localizers and do not change facial shape.

Each of 30 subjects (with varying ethnicity, age, and gender) is combined with 30 facial expressions (including basic emotions and phonemes), resulting in 900 meshes (all created from 3D-MM). Each mesh is rendered in 82 different head poses, including the frontal pose (0° rotation angles) and 81 other poses covering the angle range of ±45° in yaw (turn right/left) and pitch (turn up/down). The roll angle is not varied in the dataset, as it can be compensated by in-plane rotation. For each image, a previously defined subset of the 3D-MM mesh points were projected to the image coordinate system yielding a set of correspondence points. Due to self-occlusions in out-of-plane head poses several of them might be invisible. So, along with the coordinates we provide a binary visibility flag for each point.

Formally, the database comprises N samples with index $i \in \mathcal{I} = \{1, 2, \ldots, N\}$, each with an image frame $\mathbf{F}_i \in \mathbb{R}^{a_1 \times a_2 \times c}$ with $a_1 \times a_2$ being the number of pixels and c the number of channels. For each sample i we have a set of $M_\mathbf{p}$ correspondence points $\mathbf{p}_{i,j} \in \mathbb{R}^2$ with $j = 1, \ldots, M_\mathbf{p}$, which can be summarized in a vector $\mathbf{p}_i \in \mathbb{R}^{2M_\mathbf{p}}$. Each point j is semantically equivalent throughout all samples i. For each correspondence point there is an associated binary visibility $v_{i,j} \in \{0, 1\}$. The visibilities of sample i are summarized in vector $\mathbf{v}_i \in \{0, 1\}^{M_\mathbf{p}}$. Further, we have $M_\mathbf{l}$ facial landmark points $\mathbf{l}_{i,j} \in \mathbb{R}^2$ with $j = 1, \ldots, M_\mathbf{l}$, which can be summarized in a vector $\mathbf{l}_i \in \mathbb{R}^{2M_\mathbf{l}}$.

The landmarks can be automatically localized with one of numerous methods, but we include our automatically detected landmarks in the dataset. See Sec. IV for more details on the landmarks.

We decided to use an own synthetic database instead of Multi-PIE [16], FERA17 [39], or BP4D [47] due to the following reasons: (1) accurate correspondence point coordinates and visibilities are easy to obtain when rendering from a 3D-MM, (2) we can generate more head pose variation, (3) we are mainly interested in landmarks and correspondence points; so low detail in texture, lack of occlusions, and low variability in lighting are no problem, because those are handled well by landmark detectors.

B. In-Plane Point Normalization

We register the facial landmarks and correspondence points with a non-reflective similarity transformation to compensate for in-plane rotation, translation, and scale. The eye center points of the landmarks $\mathbf{l}$, which we calculate from the eye corners, are used to estimate the transformation $s(\mathbf{x})$. It is applied to all $\mathbf{p}$ and $\mathbf{l}$ coordinates, $\hat{\mathbf{p}} = s(\mathbf{p})$ and $\hat{\mathbf{l}} = s(\mathbf{l})$. Sec. II-C and II-D only work in this normalized coordinate system.

C. Correspondence Point Prediction

The task of mapping arbitrary faces (source domain) to the desired normalized faces (target domain) is defined by an index mapping function $t(i) : \mathbb{N} \mapsto \mathbb{N}$ that associates each sample in our dataset with a corresponding frontal target sample. The image $\mathbf{F}_i$ is associated with the frontal image $\mathbf{F}_{t(i)}$ and the correspondence points $\mathbf{p}_i$ with the frontal correspondence points $\mathbf{p}_{t(i)}$. For the task of facial expression recognition, $t(i)$ selects the sample with frontal pose, same expression, but from an average identity. I.e. it aims to normalize geometric differences between individuals, such as facial proportions, and reduces inter-person variability, which is beneficial for facial expression analysis.

We learn to predict correspondence point coordinates $\hat{\mathbf{p}}$ from the normalized landmarks $\hat{\mathbf{l}}$. More precisely, the ground truth response vector $\mathbf{y}_i$ of sample i is constructed by concatenating the correspondence points from the source domain $\hat{\mathbf{p}}_i$ (arbitrary pose) with those from the target domain $\hat{\mathbf{p}}_{t(i)}$ (associated frontal pose), i.e. $\mathbf{y}_i = [\hat{\mathbf{p}}_i, \hat{\mathbf{p}}_{t(i)}]$.

We use a linear model $\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{b}$ to learn the mapping, because it facilitates very fast prediction and has lower potential for overfitting to our synthetic training dataset. To cope with non-linearity of the problem, we use non-linear features. Next to the normalized landmarks $\hat{\mathbf{l}}$ we also use the landmarks $\check{\mathbf{l}}$ after being aligned based on the mouth corner points (instead of eye center). Further, we include the element-wise squares $\hat{\mathbf{l}}^2$ and $\check{\mathbf{l}}^2$, i.e. $\mathbf{x}_i = [\hat{\mathbf{l}}_i, \check{\mathbf{l}}_i, \hat{\mathbf{l}}_i^2, \check{\mathbf{l}}_i^2]$. For training we decompose $\mathbf{W} \in \mathbb{R}^{4M_\mathbf{p} \times 8M_\mathbf{l}}$ and $\mathbf{b} \in \mathbb{R}^{4M_\mathbf{p}}$ into $4M_\mathbf{p}$ models (one for each response dimension). The model parameters are selected by optimizing the L2-regularized L2-loss for support vector regression with LIBLINEAR [15]. We standardize the feature vectors $\mathbf{x}$ before training. The source domain coordinate regressors are only trained with those images in which the respective correspondence point is visible, since our warping only uses the coordinates of visible points.

D. Visibility Prediction

Similar to the previous Section, we learn to predict correspondence point *visibilities* $\mathbf{v}$ from the normalized landmarks $\hat{\mathbf{l}}$, respectively the features $\mathbf{x}$ described in the previous section. Again we use a linear model; this time the parameter matrices are $\mathbf{W} \in \mathbb{R}^{M_\mathbf{p} \times 8M_\mathbf{l}}$ and $\mathbf{b} \in \mathbb{R}^{M_\mathbf{p}}$, since we have only one response per correspondence point. Further, the visibility is binary, so we threshold the responses to get the final predictions. We optimize the parameters by learning $M_\mathbf{p}$ support vector classifier models with LIBLINEAR [15] using L2-regularized L2-loss. To avoid the imbalanced data problem [26], we apply random undersampling to balance the class distributions before training.

E. Texture Warping

Basically, we apply piecewise affine warping based on a triangle mesh to create the output image. The mesh (see Fig. 1) has been obtained once by Delaunay triangulation of the correspondence points from a frontal pose image of the SyLaFaN database. In contrast to typical piecewise affine warping, the vertex coordinates not only vary for the input, but also for the output image space. Further, we use the predicted correspondence points instead of landmarks. Disocclusion is handled by blending and mirroring from the visible facial side.

To warp an image, the predicted source domain correspondence points (see Sec. II-C) are transformed back to the input image space; the target domain points are transformed to the output image space. The predicted binary visibilities (see Sec. II-D) are post-processed as follows: (1) In triangles with one or two invisible vertices ($v_{i,j} = 0$), all vertices are set invisible ($v_{i,j} := 0$). (2) In the neighboring triangles, visible vertices ($v_{i,j} = 1$) are set to be half-visible ($v_{i,j} := 0.5$). (3) In the side of the face that has more visible vertices, all vertices are set visible. After that we warp the texture. In the first run, the input coordinates of each triangle with any vertex visibility $v_{i,j} < 1$ are set to the coordinates of the corresponding triangle from the other side, i.e. the texture is mirrored from the other facial side for those triangles. We do a second run with alpha blending to avoid strong edges at boundaries of the mirrored triangles. Each triangle with any vertex $0 < v_{i,j} < 1$ is blended on top of the first run image with $\alpha_{i,j} = 1 - v_{i,j}$. The blending factor α is linearly interpolated, eliminating strong edges between visible and mirrored parts.

III. Related Work

Facial expression recognition has been surveyed recently by Sariyanidis et al. [34]. A variety of methods are used for normalization. The simplest form is cropping the face bounding box obtained by face detection and rescaling it to a canonical size [19], [5] (we later refer to this as FaceDet). When landmarks are known, another easy option is to only scale the image [7], which may be sufficient for using local descriptors around the landmarks. More advanced landmark based normalization methods are summarized in Table I. They are based on different landmarks, such as only eye landmarks, inner landmarks (excluding the facial contour), or landmarks with facial contour. For some landmark localizers, facial contour landmarks are not available; further, they are often less accurate than the inner landmarks. Our proposed FaNC method can be trained on top of any number of landmarks. Most methods register the landmarks with a static reference shape (usually an average face), but they differ regarding the used transformation: non-reflective similarity and affine transformations are very common choices.

The first five methods in Table I create the normalized image by warping with a single transformation, which registers the images to a certain degree, but does not generate a frontal view. In contrast, the other methods in the table use piecewise warping or 3D rendering to synthesize a frontal view. Piecewise affine warping (PieceAff) to a reference shape is widely used for frontalization. It offers accurate registration for a wide range of poses, but has the following limitations: (1) It removes facial shape information, i.e. differences in facial proportions and deformations due to expression are lost, (2) the warping might also drop relevant texture information or fill large areas from a few pixels, and (3) the method does not handle occlusions, which leads to artifacts for extreme poses (see Fig. 4 for examples). Hassner et al. [17] (3dStatic) use a static 3D model with corresponding 3D landmark positions. They assume the intrinsic camera parameters to be known and estimate the extrinsic camera parameters to find the head pose. Next, they texturize the model with the input image and render it in frontal pose. Occlusions are handled by blending with the mirrored version of the model. Wang et al. [40] learn to map the detected landmarks from arbitrary views to the frontal view, apply piecewise affine warping to generate a frontal texture, and handle disocclusions and other artifacts by synthesizing an appearance image from a pre-defined Eigen-face space by minimizing the pixel-wise mean squared error. The first part is similar to our approach, but we not only map detected landmarks to the target domain, but predict a denser set of correspondence points in both source and target domain. Further, our method is fully discriminative and does not require an optimization for a query image, making it

TABLE I

OVERVIEW OF STATE-OF-THE-ART LANDMARK BASED REAL-TIME CAPABLE FACE NORMALIZATION METHODS.

Abbreviation	Registration input	Registration target	Texture warping	OH	Applications
SimEye	eye landmarks	reference shape	NR similarity transf.	×	[38], [25], [36], [50], [12]
SimInner	inner landmarks	reference shape	NR similarity transf.	×	[48], [13], [49], [27]
SimStable	landm. stable under expression [3]	reference shape	NR similarity transf.	×	[3], [4], [28]
AffInner	inner landmarks	reference shape	affine transf.	×	[44], [30], [14], [2], [35]
AffStable	stable inner landmarks (eye/nose)	reference shape	affine transf.	×	[37], [23], [1]
PieceAff	landmarks with facial contour	reference shape	piecewise affine transf.	×	[9], [41], [20], [42]
3dStatic [17]	inner landmarks	static 3D model	3D rendering	✓	[17]
FaNC (ours)	predicted corresp. points	predicted corresp. points	piecewise affine + blending	✓	Sec. IV-B and IV-C

OH: occlusion handling NR: non-reflective

usable for online expression analysis at high frame rates – in contrast, the optimization part of Wang [40] runs for more than one minute per image. Next to the landmark-based methods, there are purely texture-based approaches to normalize faces [54], [45], [46], which are not in the focus here. They require expensive hardware to run at high frame-rates (if possible at all) and huge training datasets with variation in all degrees of freedom (for generalizing well across datasets).

IV. EXPERIMENTS

In several experiments, we compare the proposed FaNC with other face normalization methods, analyze generalization to unseen poses (and individuals), and analyze the impact of the poses available in training data. Sec. IV-A compares qualitative results and runtime of face normalization methods. In Sec. IV-B we experiment with the FERA17 dataset [39] and compare the results we achieve in facial action unit intensity estimation when changing the normalization used for preprocessing the recognition CNN input. Similarly, Sec. IV-C addresses expression recognition on the Multi-PIE dataset [16].

Landmark Localization: To localize facial landmarks (68 points) across a wide range of poses, we train an ensemble of regression trees based on the method by Kazemi and Sullivan [21] using the implementation from dlib [22]. The model is trained on multiple datasets (Multi-PIE [16], afw [53], helen [24], ibug, 300-W [31], 300-VW [10], and lfpw [6]). The point annotations for ibug, afw, helen, 300-W, and lfpw are provided by Sagonas et al. [33]. From the 300-VW dataset we selected the hardest 10 frames of each video based on the point to point error (normalized by interocular distance) with a previously trained model. From the Multi-PIE dataset we used all fully annotated samples from the camera pose 080 and 190. The resulting model performed significantly better than the model coming with dlib [22]. An advantage of our method is that it can benefit from advances in landmark localization and that using a more recent approach may improve face normalization results.

SyLaFaN Dataset: Despite the improved model, there are still moderate to severe landmark localization errors, especially in extreme head poses. For our experiments we only use a subset of the SyLaFaN database with lower errors. To find this set, we calculated the mean distance of landmark points and associated correspondence points for each sample i, sort them by distance, and choose the 75% of samples with lowest error.

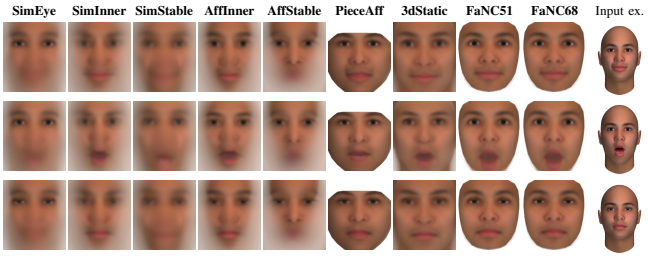

Fig. 3. Normalized images for facial expressions smile (top row), mouth open (middle), and half closed eyes (bottom). Columns: mean of results of different methods and one of the images before normalization.

FaNC Training: We trained FaNC with the $M_\mathbf{p} = 153$ correspondence points provided with the SyLaFaN dataset. Regarding landmarks, we use two variants: **FaNC68** with all $M_\mathbf{l} = 68$ landmarks and **FaNC51** with the 51 inner landmarks (excluding the facial contour points along the jaw and chin). The coordinate prediction is trained with $\epsilon = 0.005, C = 0.25$, the visibility prediction with $C = 1$. We render the normalized images to a resolution of 180×200 pixels for Sec. IV-A and 256×256 pixels for Sec. IV-B and IV-C (same for all other methods). For the cross-dataset experiments in Sec. IV-B and IV-C we augment the SyLaFaN training set by mirroring the asymmetric expressions and train on 30,000 randomly selected samples with $M_\mathbf{l} = 68$.

A. Face Normalization

We qualitatively compare face normalization results of the methods listed in Table I and shortly discuss runtime. The 3dStatic method was applied with the inner 51 landmarks, as this performed better than using all 68 landmarks.

Qualitative Results on SyLaFaN: We applied the normalization methods on all images of the SyLaFaN dataset and calculated the pixel-wise mean images for each expression (across all combinations of head poses and identities). Fig. 3 shows the resulting mean images for three facial expressions (rows). Blur indicates high within-class variation in the respective region, which is generally undesirable. The single-transformation methods (first five columns) can at most register parts of the images accurately – the parts around the used landmarks if they are few and planar as in SimEye and AffStable. PieceAff achieves an accurate

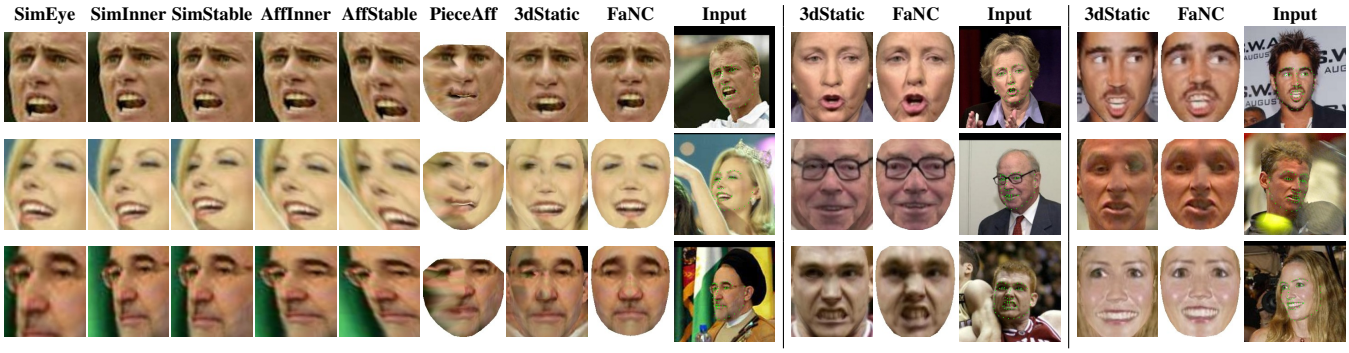

Fig. 4. Normalized face images and input images from LFW database. See Table I for acronyms.

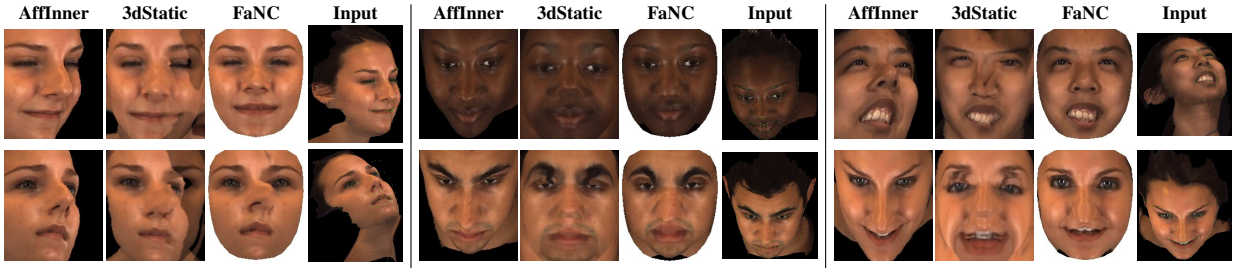

Fig. 5. Normalized face images and input images from FERA 2017 database. Bottom row are FaNC failure cases, see text.

registration, but most of the expression-induced shape deformation is lost. The more advanced methods, 3dStatic and FaNC yield accurate registration and retain the expression information at the same time. There is no qualitative difference between FaNC51, which only uses the 51 inner landmarks, and FaNC68, which also uses facial contour landmarks.

Qualitative Results on LFW and FERA: Fig. 4 depicts examples of the Labeled Faces in the Wild (LFW) database [18]. SimEye is sensitive to the foreshortening effect in out-of-plane poses, which may significantly alter scale as in the third row. SimInner and SimStable yield similar results, whereas SimInner tends to have higher registration accuracy at the landmarks and SimStable tends to yield more upright and centered faces. AffInner has more potential to compensate differences in facial proportions, but may cause unrealistic looking shearing of the image. The latter effect is even more pronounced in AffStable. PieceAff suffers from disocclusion artifacts and removes expression information. 3dStatic and FaNC both handle occlusions by exploiting symmetry, but FaNC causes less artifacts. Note that 3dStatic has been developed with the LFW database, so it is "optimized" for this database. Our FaNC method has been developed and trained on the SyLaFaN database and we did not optimize it towards any other database. Fig. 5 shows examples from the FERA 2017 challenge dataset [39]. If landmarks are localized well (see top row), FaNC is able to synthesize high quality frontal views in most of the cases. If landmarks are inaccurate (bottom row), FaNC's frontal images suffer from more artifacts. Further, FaNC is not able to recover occlusions by the nose in pitch angles (see bottom right) yielding a long nose and deformations at the lip. However, 3dStatic generally suffers from more artifacts (although we resized images to the expected resolution and tried some other adaptations for improvement). AffInner is not able to reduce variance between head poses, but does not cause any artifacts (except some shearing).

Runtime: The FaNC normalization method is designed to be fast. Essentially, it only needs two matrix multiplications and warping with blending, which can be efficiently done with any (even a very old) GPU. With our unoptimized OpenGL 2.0 implementation, warping into a 256×256 image takes about 1.5 ms with an Intel HD 4000 GPU (integrated in Intel i7-3770, launched 2012) including data transfers, similar to PieceAff and all single transformation methods. For a same sized image, 3dStatic runs for about 100 ms [17]. State of the art texture-based methods require heavy GPU computation and can achieve high frame rates only with expensive hardware (if at all).

B. Action Unit Intensity Estimation

We evaluate the effect of face normalization on facial action unit (AU) intensity estimation and the generalization to unseen poses with the FG 2017 Facial Expression Recognition and Analysis challenge (FERA 2017) dataset [39], which is intended to raise the bar for expression recognition for different view angles of the face. The dataset provides a training and validation set, with 41 and 20 different participants, respectively. Each participant was stimulated in 8 different scenarios and each scenario is captured from 9 different viewing angles (in total $2,952$ training and $1,431$ validation videos). 7 different Action Units (AUs) are manually labeled for each frame.

Training: We use the NASNet-A architecture [55] and fine-tune the pretrained NASNet-A_Mobile_224 model available with the tensorflow/slim implementation. Due to the limited variability in the data (compared to ImageNet), we

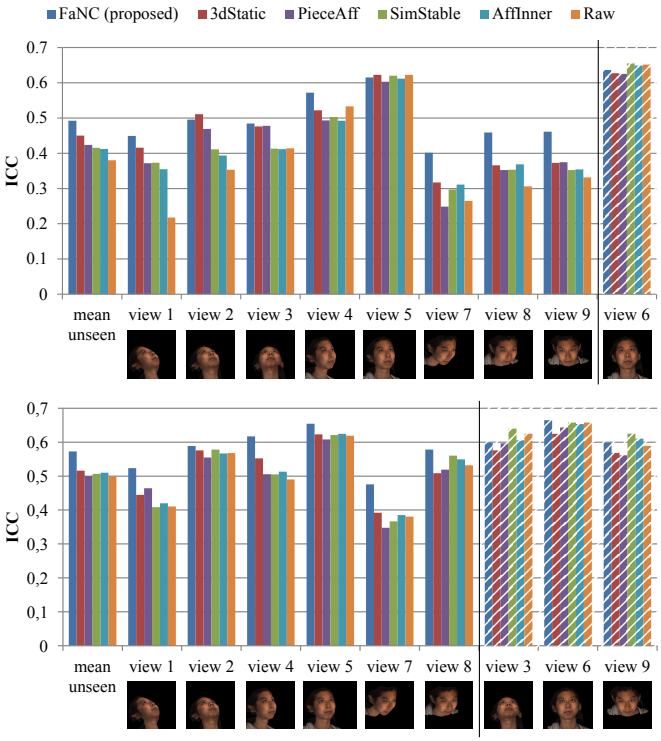

Fig. 6. AU intensity estimation results on unseen poses (solid). Training was done with view 6 only (top) and with views 3, 6, and 9 (bottom).

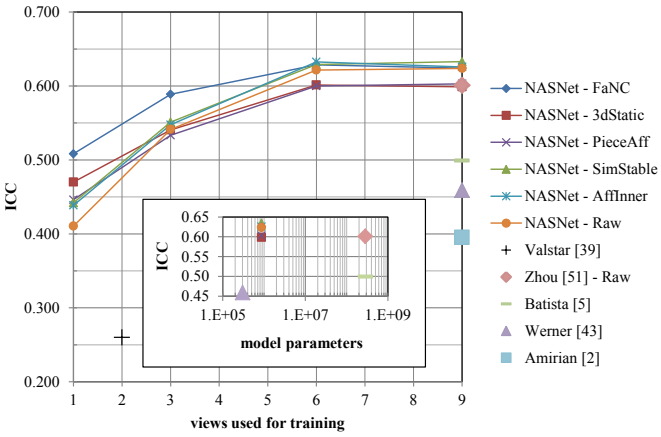

Fig. 7. AU intensity estimation results compared to state of the art. Mean ICC over all views and AUs depending on number of views used for training (outer plot) and number of model parameters (inner plot).

cut the network after the 6'th of 12 cells. We append a fully connected layer with 7 neurons, one for each AU (with linear activation). Further, the stem weights (first part of the network) are kept fixed to speed up the training. Standard gradient descent is used to minimize MSE loss for 50,000 iterations (with a mini-batch size of 32 samples). The initial learning rate is set to 0.1 and reduced according to the single period cosine decay [55] down to 10^{-8}. For regularization we set the drop path keep probability to 0.9 and L2 weight decay to $4 \cdot 10^{-5}$. To avoid divergence due to huge gradients, local gradient clipping is applied (max. L2 norm value of 5) during the first 2,000 iterations. Similar to Zhou et al. [51], we randomly under-sample the training set for each view by selecting 6,000 samples per AU (3k with intensity label 0 and 3k with label 1-5). We augment the training data (42,000 samples per view) by randomly changing brightness, contrast, and saturation and by randomly flipping the image. Each trained model is tested on every frame of the entire validation set to calculate the ICC(3,1) measure (per view and AU). Training and evaluation is repeated 5 times for each normalization method and results are averaged.

Generalization to unseen poses: We investigate to which degree different face normalization methods help with generalizing to unseen head poses and which views are needed for training to achieve good results. For this purpose, we vary the subset of views used for training. The compared methods include our FaNC, 3dStatic, PieceAff, SimStable, AffInner, and Raw (using original images depicted in Fig. 6). Fig. 6 (top) shows the results (mean ICC across all AU) of training with only the frontal samples (view 6). We can observe that all methods generalize well to view 5, which differs 20° from the training view in the yaw angle. However, performance drops significantly for all other views. On average and in most cases, our proposed FaNC methods facilitates best generalization to unseen poses. Changes in pitch ($\pm 40°$) yield lowest performance due to the change in appearance (e.g. occlusion by nose) that cannot be fully compensated by any of the methods, but our FaNC method outperforms the others clearly in all top views (7, 8, and 9). In Fig. 6 (bottom) we show the results of training with one view per pitch angle (3, 6, and 9). Compared to training with the frontal view only, the overall performance improves due to more training samples and more variability. But enormous performance drops remain for views that differ 40° from training data in yaw angle (view 1, 4, and 7). Our FaNC method still outperforms the others on those and the other unseen views.

Comparison with state of the art: In Fig. 7 we compare the results we obtain with NASNet to those reported in other works that address AU intensity estimation on the FERA 2017 dataset. Valstar et al. [39] are the only who tried to generalize to unseen views (they trained on view 5 and 6), but their simple challenge baseline system performed poorly compared to all other works. Amirian et al. [2] and Werner et al. [43] both greatly outperform the baseline while training with all views, but the deep learning based approaches by Batista et al. [5] and Zhou et al. [51] perform significantly better. Batista et al. [5] fed the cropped face bounding boxes to a custom network architecture. The FERA 2017 challenge winners Zhou et al. [51] used the original images (as our "Nasnet - Raw") and fine-tuned one VGG16-based network per action unit. Our NASNet yields similar results with 3dStatic and PieceAff face normalization, but outperforms all related works for the other face normalization methods. Even with same performance, NASNet has the advantage of less model parameters (and memory footprint); both other networks [5], [51] have 300 times more parameters than NASNet (see inner plot in Fig. 7). Fig. 7 also shows the overall results we obtain with different number of views

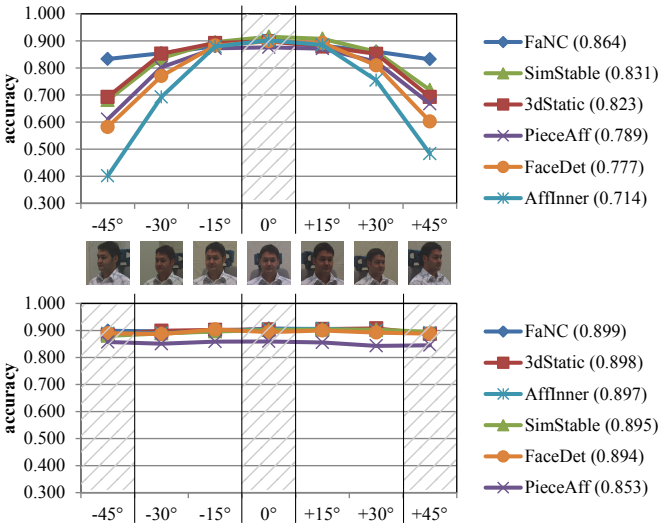

Fig. 8. Facial expression recognition results on unseen poses (without shading). Training was done with view $0°$ only (top) and with views $-45°$, $0°$, and $+45°$ (bottom). Mean accuracies across all views in brackets. Trivial classifier achieves 0.277.

used for training (view 6; views 3, 6, 9; views 1, 3, 4, 6, 7, 9; all views). Our proposed FaNC with NASNet trained on only frontal images performs better than Batista et al. [5] (challenge's second place), who trained on all views. Further, we observe that there is no improvement between training with all nine view and six views (combinations of yaw $\in \{-40°, 0°\}$ and pitch $\in \{-40°, 0°, +40°\}$. To analyze if results with all views would benefit from longer training, we tried to train for 100k instead of 50k iterations, but found no significant difference. So we conclude that the additional views with intermediate yaw angles do not add much and the model already generalizes well to the intermediate views. See supplementary material for detailed result tables.

C. Facial Expression Recognition

To further evaluate the effect of face normalization, we conduct experiments on the Multi-PIE dataset [16]. We use the data of all 337 subjects in homogeneous illumination (no. 00) recorded from seven views provided in the dataset (yaw angle $0°$, $\pm15°$, $\pm30°$, $\pm45°$). In total these are about 18k images of the following six facial expressions, which we aim to recognize: neutral expression, smile, surprise, squint, disgust, and scream. We train NASNet as described in the previous section. The only differences are the number of outputs (6, one per class), the loss function (soft-max cross entropy), the number of iterations (20k), and the initial learning rate (0.01). We run 5-fold cross validation without subject overlap between training and test sets and the results are averaged. The normalization methods are the same as above, except that we use the face detection bounding box (FaceDet) instead of full database images (Raw).

Fig. 8 (top) shows the results of training with only frontal faces. Similar to the results on the FERA dataset, performance drops significantly if the pose deviates $30°$ or more from the data seen during training. But again, FaNC generalizes best to those unseen poses. If we additionally include $\pm45°$ to the training set, the network is able to generalize to the intermediate views without significant performance drops, see Fig. 8 (bottom). In this case, the face normalization has minor influence on the performance. Only PieceAff performs significantly worse, probably because it suffers from artifacts due to a lack of occlusion handling.

V. Discussion and Conclusion

Due to the advent of deep learning, limited amount of data with high-quality annotations is one of the major issues now. The previous sections addressed the question how to achieve head pose invariance with limited training data. For this purpose we developed the FaNC method to normalize arbitrary faces to frontal views. In contrast to most other works in face normalization [17], [52], [45], [54], [40], we tested our method cross-database, i.e. FaNC was evaluated on data that was completely unseen during the development of the method. Normalization of those data shows that FaNC generalizes well to new data generating realistic frontal images without significant artifacts in most of the cases. Based on our experiments on the FERA 2017 and Multi-PIE database, we can clearly recommend to use FaNC if most of the available training data for the task at hand is frontal, because it generalizes best to unseen views. We observed that AU intensity estimation and expression recognition performance degrades if the tested poses deviate more than $20°$ from the poses available during training, but less so with our proposed FaNC method.

The experiments indicated that CNNs are able to generalize well to unseen poses without sophisticated face normalization methods if training data is available that covers the pose space in steps of about $40°$. We expect that a less systematic, high variance coverage of the pose space would have a similar or even better effect on generalization. However, for training a robust universal expression recognition system, pose is not the only nuisance factor we need to vary (but also identity, illumination, occlusion, background, resolution, sharpness, noise etc.). Generalizing across head poses with less need for variation in the training data may help to also address the other factors. Gathering huge amounts of suitable data for expression recognition is still challenging, because (1) annotation with high quality labels is expensive and (2) it is hard to avoid dataset biases and cover rare events/conditions sufficiently. Gathering 3D data and rendering in several poses seem to be an alternative to gathering multiple views, but 3D data are usually incomplete (e.g. occluded part of head is missing) and inaccurate (at least in convex parts), impairing realism in out-of-plane views. Further, in contrast to face normalization the pose augmentation approach cannot benefit from existing 2D data. So we belief that improving face normalization is still promising. A direction to advance FaNC is training with more realistic 3D morphable models and/or arbitrary 3D datasets that cover more variation of identity and expression. Further, there is room for improvement in handling pitch variation, in which symmetry does not help for filling disocclusions.

REFERENCES

[1] T. Almaev, B. Martinez, and M. Valstar. Learning to transfer: transferring latent task structures and its application to person-specific facial action unit detection. In *ICCV*, 2015.

[2] M. Amirian, M. Kächele, G. Palm, and F. Schwenker. Support vector regression of sparse dictionary-based features for view-independent action unit intensity estimation. In *FG*, 2017.

[3] T. Baltrusaitis, M. Mahmoud, and P. Robinson. Cross-dataset learning and person-specific normalisation for automatic action unit detection. In *FG*, 2015.

[4] T. Baltrusaitis, P. Robinson, and L.-P. Morency. OpenFace: an open source facial behavior analysis toolkit. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016.

[5] J. C. Batista, V. Albiero, O. R. P. Bellon, and L. Silva. Aumpnet: Simultaneous action units detection and intensity estimation on multipose facial images using a single convolutional neural network. In *FG*, 2017.

[6] P. Belhumeur, D. Jacobs, D. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *CVPR*, 2011.

[7] C. F. Benitez-Quiroz, R. Srinivasan, and A. M. Martinez. EmotioNet: An Accurate, Real-Time Algorithm for the Automatic Annotation of a Million Facial Expressions in the Wild. In *CVPR*, 2016.

[8] V. Blanz and T. Vetter. A Morphable Model for the Synthesis of 3d Faces. In *Proc. of the 26th Annual Conf. on Computer Graphics and Interactive Techniques*, SIGGRAPH, 1999.

[9] S. W. Chew, P. Lucey, S. Lucey, J. Saragih, J. F. Cohn, I. Matthews, and S. Sridharan. In the pursuit of effective affective computing: The relationship between features and registration. *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, 42(4):1006–1016, 2012.

[10] G. G. Chrysos, E. Antonakos, S. Zafeiriou, and P. Snape. Offline deformable face tracking in arbitrary videos. In *ICCVW*, 2015.

[11] W.-S. Chu, F. D. L. Torre, and J. F. Cohfcn. Selective transfer machine for personalized facial action unit detection. In *CVPR*, 2013.

[12] A. Dapogny, K. Bailly, and S. Dubuisson. Pairwise conditional random forests for facial expression recognition. In *ICCV*, 2015.

[13] X. Ding, W.-S. Chu, F. De la Torre, J. F. Cohn, and Q. Wang. Facial action unit event detection by cascade of tasks. In *ICCV*, 2013.

[14] S. Eleftheriadis, O. Rudovic, and M. Pantic. Multi-conditional latent variable model for joint facial action unit detection. In *ICCV*, 2015.

[15] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of machine learning research*, 9(Aug):1871–1874, 2008.

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

[17] T. Hassner, S. Harel, E. Paz, and R. Enbar. Effective Face Frontalization in Unconstrained Images. In *CVPR*, 2015.

[18] 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.

[19] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim. Joint fine-tuning in deep neural networks for facial expression recognition. In *ICCV*, 2015.

[20] S. Kaltwang, S. Todorovic, and M. Pantic. Latent Trees for Estimating Intensity of Facial Action Units. In *CVPR*, 2015.

[21] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *CVPR*, 2014.

[22] D. E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[23] S. Koelstra, M. Pantic, and I. Y. Patras. A Dynamic Texture-Based Approach to Recognition of Facial Actions and Their Temporal Models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(11):1940–1954, 2010.

[24] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *ECCV*, 2012.

[25] M. Liu, S. Shan, R. Wang, and X. Chen. Learning Expressionlets on Spatio-temporal Manifold for Dynamic Facial Expression Recognition. In *CVPR*, 2014.

[26] V. Lpez, A. Fernndez, S. Garca, V. Palade, and F. Herrera. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, 250:113–141, Nov. 2013.

[27] I. Masi, S. Rawls, G. Medioni, and P. Natarajan. Pose-aware face recognition in the wild. In *CVPR*, 2016.

[28] F. Ringeval, M. Pantic, et al. AVEC 2017: Real-life Depression, and Affect Recognition Workshop and Challenge. In *Proc. Workshop on Audio/Visual Emotion Challenge (AVEC)*, 2017.

[29] O. Rudovic, M. Pantic, and I. Patras. Coupled Gaussian processes for pose-invariant facial expression recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(6):1357–1369, 2013.

[30] O. Rudovic, V. Pavlovic, and M. Pantic. Context-Sensitive Dynamic Ordinal Regression for Intensity Estimation of Facial Action Units. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(5):944–958, 2015.

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

[32] C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic. Robust statistical face frontalization. In *ICCV*, 2015.

[33] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. A semi-automatic methodology for facial landmark annotation. In *CVPRW*, 2013.

[34] E. Sariyanidi, H. Gunes, and A. Cavallaro. Automatic Analysis of Facial Affect: A Survey of Registration, Representation, and Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(6):1113–1133, 2015.

[35] F. Saxen, P. Werner, and A. Al-Hamadi. Real vs. Fake Emotion Challenge: Learning to Rank Authenticity from Facial Activity Descriptors. In *ICCVW*, 2017.

[36] T. Senechal, V. Rapp, H. Salam, R. Seguier, K. Bailly, and L. Prevost. Facial action recognition combining heterogeneous features via multikernel learning. *IEEE Trans. Systems, Man, and Cybernetics, Part B*, 42(4):993–1005, 2012.

[37] M. Valstar, J. Girard, T. Almaev, G. McKeown, M. Mehu, L. Yin, M. Pantic, and J. Cohn. Fera 2015 - second facial expression recognition and analysis challenge. In *FG*, 2015.

[38] M. F. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. Scherer. The first facial expression recognition and analysis challenge. In *FG*, 2011.

[39] M. F. Valstar, E. Snchez-Lozano, J. F. Cohn, L. A. Jeni, J. M. Girard, Z. Zhang, L. Yin, and M. Pantic. FERA 2017 - Addressing Head Pose in the Third Facial Expression Recognition and Analysis Challenge. In *FG*, 2017.

[40] Y. Wang, H. Yu, J. Dong, M. Jian, and H. Liu. Cascade support vector regression-based facial expression-aware face frontalization. In *IEEE Int. Conf. on Image Processing (ICIP)*, 2017.

[41] Z. Wang, Y. Li, S. Wang, and Q. Ji. Capturing global semantic relationships for facial action unit recognition. In *ICCV*, 2013.

[42] P. Werner, A. Al-Hamadi, K. Limbrecht-Ecklundt, S. Walter, S. Gruss, and H. Traue. Automatic Pain Assessment with Facial Activity Descriptors. *IEEE Trans. on Affective Computing*, 8(3):286–299, 2017.

[43] P. Werner, S. Handrich, and A. Al-Hamadi. Facial action unit intensity estimation and feature relevance visualization with random regression forests. In *Int. Conf. Affective Computing Intelligent Interaction*, 2017.

[44] P. Werner, F. Saxen, and A. Al-Hamadi. Handling Data Imbalance in Automatic Facial Action Intensity Estimation. In *British Machine Vision Conf. (BMVC)*, 2015.

[45] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, and J. Kim. Rotating Your Face Using Multi-Task Deep Neural Network. In *CVPR*, 2015.

[46] X. Yin, X. Yu, K. Sohn, X. Liu, and M. Chandraker. Towards large-pose face frontalization in the wild. In *ICCV*, 2017.

[47] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu, and J. M. Girard. BP4d-Spontaneous: a high-resolution spontaneous 3d dynamic facial expression database. *Image and Vision Computing*, 32(10):692–706, Oct. 2014.

[48] K. Zhao, W.-S. Chu, F. De la Torre, J. F. Cohn, and H. Zhang. Joint Patch and Multi-Label Learning for Facial Action Unit Detection. In *CVPR*, 2015.

[49] K. Zhao, W.-S. Chu, and H. Zhang. Deep Region and Multi-Label Learning for Facial Action Unit Detection. In *CVPR*, 2016.

[50] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, and D. N. Metaxas. Learning active facial patches for expression analysis. In *CVPR*, 2012.

[51] Y. Zhou, J. Pi, and B. E. Shi. Pose-independent facial action unit intensity regression based on multi-task deep transfer learning. In *FG*, 2017.

[52] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li. High-Fidelity Pose and Expression Normalization for Face Recognition in the Wild. In *CVPR*, 2015.

[53] X. Zhu and D. Ramanan. Face detection, pose estimation and landmark localization in the wild. In *CVPR*, 2012.

[54] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning identity-preserving face space. In *ICCV*, 2013.

[55] B. Zoph, V. Vasudevan, J. Shlens, and Q. Le. Learning Transferable Architectures for Scalable Image Recognition. In *CVPR*, 2018.