Robust Face Frontalization For Visual Speech Recognition
Zhiqi Kang, Radu Horaud, Mostafa Sadeghi

To cite this version:
Zhiqi Kang, Radu Horaud, Mostafa Sadeghi. Robust Face Frontalization For Visual Speech Recognition. ICCVW 2021 - International Conference on Computer Vision Workshops, IEEE, Oct 2021, Montreal - Virtual, Canada. pp.2485-2495, 10.1109/ICCVW54120.2021.00281 . hal-03326002v3

HAL Id: hal-03326002
https://inria.hal.science/hal-03326002v3
Submitted on 8 Nov 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Robust Face Frontalization For Visual Speech Recognition

Zhiqi Kang & Radu Horaud
Inria & Univ. Grenoble Alpes
Montbonnot Saint-Martin, France

Mostafa Sadeghi
Inria Nancy Grand-Est
Nancy, France

Abstract

Face frontalization consists of synthesizing a frontally-viewed face from an arbitrarily-viewed one. The main contribution of this paper is a robust frontalization method that preserves non-rigid facial deformations, i.e. expressions, to perform lip reading. The method iteratively estimates the rigid transformation (scale, rotation, and translation) and the non-rigid deformation between 3D landmarks extracted from an arbitrarily-viewed face, and 3D vertices parameterized by a deformable shape model. An important merit of the method is its ability to deal with large Gaussian and non-Gaussian errors in the data. For that purpose, we use the generalized Student-t distribution. The associated EM algorithm assigns a weight to each observed landmark, the higher the weight the more important the landmark, thus favoring landmarks that are only affected by rigid head movements. We propose to use the zero-mean normalized cross-correlation (ZNCC) score to evaluate the ability to preserve facial expressions. We show that the method, when incorporated into a deep lip-reading pipeline, considerably improves the word classification score on an in-the-wild benchmark. An extended version with supplemental materials can be found at https://team.inria.fr/robotlearn/rff-vsr/

1. Introduction

Face frontalization is the problem of synthesizing a frontally-viewed face from an arbitrary view. Recent research has shown that face frontalization consistently boosts the performance of face recognition [53, 63, 4, 59, 60, 61]. A common feature of these methods is that they encourage expression-free face frontalization. In contrast, visual speech recognition (or lip reading) [17, 1, 35, 10] belongs to the larger class of facial expression analysis methods, e.g. [41], that require expression-preserving face frontalization. Indeed, lip and jaw motions are controlled by speech production as they are correlated with phonemes and with word pronunciation [45]. Nevertheless, lip and jaw motion analysis is perturbed by inherent rigid head movements. It is therefore important to discriminate between rigid head movements and non-rigid facial/lip movements. Moreover, the capacity to properly analyse rigid and non-rigid movements separately plays a crucial role in audio-visual speech technologies, e.g. [14, 42, 52, 44, 48]. Visual speech is particularly useful when audio signals are corrupted by ambient noise, by competing speech sources, or by acoustic perturbations.

In this paper we address face frontalization as the problem of independently estimating rigid 3D pose and non-rigid 3D deformations of an arbitrarily-viewed face, Figure 1. In more detail, (i) we estimate the scale, rotation, and translation between a set of 3D facial landmarks ex-
We empirically evaluate the performance of the proposed method using the zero-mean normalized cross-correlation (ZNCC) coefficient between a frontalized face and the ground-truth frontal face. We also consider the task of isolated word recognition (IWR) based on lip movements. We embed the proposed face frontalization method into a state-of-the-art lip-reading deep model [35] and we show that IWR performance is improved by a considerable margin. These evaluations, based on direct comparison between predictions and ground-truth videos of faces, and on visual speech recognition, stand in contrast with evaluation based on face recognition, e.g. [53, 4, 59] that require as-neutral-as-possible frontal faces.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 describes in detail the proposed frontalization method, analyses its performance and provides implementation details. Section 4 describes the proposed evaluation protocols and shows results obtained with our method and with several state-of-the-art methods. Section 5 draws some conclusions.

2. Related Work

Recently, a successful face frontalization approach has been to train deep neural networks (DNNs) to learn non-linear 2D-to-2D mappings between an arbitrary view and a frontal view. Some of the best performing DNN-based frontalization methods use CNN/GAN architectures, e.g. [54, 27, 49, 59, 57, 43, 58, 55], which outperform CNN-only ones, e.g. [53]. These methods necessitate large collections of input/output pairs of face images. For that purpose [57, 55, 58] use two datasets that contain multiple-camera recordings in a controlled setup, i.e. [20, 21]. [57] proposed to learn dense pixel-to-pixel correspondences between the two faces. Subsequently, [58] proposed a semi-supervised GAN-based method that augments the paired face images of [21] with unpaired in-the-wild faces with large variations in identity, e.g. [26]; their adversarial and identity-preserving losses enhance face recognition performance. [55] proposed a dual-attention GAN architecture that captures long-term dependencies in image space, thus providing a means to preserve identity. Another way to estimate the non-linear 2D-to-2D mapping between a profile image and a canonical frontal image of a face is to use a rectification network that learns local homographies between a deformed grid, that corresponds to a profile view, and a regular grid, that corresponds to a frontal view [60]. These DNN-based methods are designed to predict as-neutral-as-possible frontal faces, i.e. expression-free faces, in order to improve the performance of face recognition. The main drawback of these methods is that they cannot guarantee that they preserve rigidity – a necessary feature for preserving facial expressions.

Other methods estimate the pose of an input face with respect to a frontal 3D face model, then use the pose parameters to warp the facial pixels from the input image onto a frontal one. This amounts to pose estimation based on 2D-to-3D fitting, e.g. [63, 22, 18, 4]. [22] used 2D facial landmarks extracted from the input face that are in correspondence with 3D model landmarks. This corresponds to the estimation of intrinsic and extrinsic camera parameters, followed by image warping. Similar methods were proposed in [63] and [4] where 2D landmarks are extracted from the input face and associated with 3D landmarks of a generic 3D model. The pose estimators of [63, 22, 18, 4] lack a built-in mechanism capable of making them robust against the mixed presence of large errors in landmark localization and of non-rigid facial deformations. Moreover, [63] purposely removes expressions in order to favor identity features. Recently, [61] proposed to combine the 2D-to-3D geometric fitting method of [64] with a state-of-the-art GAN-based style transfer model to fill in the occluded regions caused by 3D face rotation. This method yields state-of-the-art results for the task of face recognition, but there is
no theoretical guarantee that non-rigid facial deformations are preserved by the GAN-based transfer process.

A fundamental building block of the proposed method is to embed face frontalization into a robust estimator. Robustness refers to the capacity of the estimator to be weakly affected by outliers. For that purpose, we cast the problem at hand in the framework of maximum likelihood estimation (MLE). In MLE, the choice of the pdf is crucial. As already mentioned above, we opt for the generalized Student t-distribution which belongs to the larger class of heavy-tailed distributions that are well known to be robust to outliers [36, 47, 19]. The associated EM procedure evaluates the posteriors of the weights – there is a weight associated with each point pair. The weights are treated as random variables drawn from a gamma distribution. The realizations of these variables vary from zero to an arbitrary high number. Therefore, they are an absolute measure of the importance of the data pairs.

In addition to estimating the weights, the GStudent EM algorithm estimates a posterior covariance matrix. The evaluation of a full covariance is fundamental for taking into account an anisotropic error distribution of the landmarks’ 3D coordinates, and hence for assessing how much one can trust them. This stays in strong contrast with the traditional methods that are used in computer vision to estimate the rigid transformation between two point sets, e.g. [24, 25, 15, 3, 51]. These methods assume an isotropic covariance – spherical shaped, with the advantage of yielding closed-form solutions for estimating the rotation parameters. Nevertheless, it has been shown that these estimators are biased if there are measurement errors in both point sets [31], and unbiased estimators based on convex optimization and on branch-and-bound algorithms were proposed [39]. Nevertheless, [31, 39] used an isotropic covariance and did not incorporate a robust pdf that ultimately weights the absolute importance of the point pairs. The fact that GStudent estimates a full covariance matrix – ellipsoidal shaped, introduces an additional complexity, namely a non-linear solver is required to estimate the rotation matrix. In [23] this was addressed via convex relaxation. In this paper, we simplify the optimization problem by using quaternions and propose to use sequential quadratic programming [6].

3. Robust Face Frontalization

The core idea of the method, Figure 1, is to estimate the 3D pose (scale, rotation, and translation) and 3D shape of an input face and to warp it onto a frontal view. The pose is robustly estimated from a set of 3D landmarks, associated with a set of 3D vertices of a deformable geometric model, i.e. a triangulated mesh. This allows to rigidly frontalize the input landmarks, hence to preserve facial expression, i.e. (7), then to fit the deformable model to these landmarks, thus yielding a frontalized and deformed 3D model of the input face, i.e. (11). Next, vertex interpolation enables to compute a dense depth map of the frontalized face, i.e. (12) and (13). Finally, the input face is warped onto a frontal view using the dense depth map, i.e. (40) and (42) of Appendix E.

3.1. Robust Pose Estimation

Let $I_p$ be an observed image of a face in an unknown pose. A set $X$ of $J = 68$ 3D landmarks is extracted from $I_p$ with image-centered coordinates $X_{1:j} = \{X_j\}_{j=1}^J \subset \mathbb{R}^3$. Throughout the paper we adopt the notation $X_j = (X_{j1}, X_{j2}, X_{j3})$ to designate the three coordinates of a point in $\mathbb{R}^3$. Let $Z_{1:j} = \{Z_j\}_{j=1}^J \subset \mathbb{R}^3$ be the 3D coordinates of a set of vertices, $Z$, that correspond to the frontal view of a 3D deformable geometric model with a neutral expression, and let $I_f$ be the frontal image to be predicted. Pose estimation consists of finding the rigid transformation that best maps $X_{1:j}$ onto $Z_{1:j}$, namely

$$Z_j = \rho RX_j + t + e_j, \quad \forall j \in \{1 \ldots J\},$$

where $\rho \in \mathbb{R}^+ \times \mathbb{R}^3 \times \mathbb{R}^3$ are the scale, rotation matrix and translation vector, respectively, associated with the unknown pose. Because the landmark locations $X_{1:j}$ are inherently affected by detection errors as well as by non-rigid facial deformations, it is suitable to use a robust rigid-parameter estimation technique. For this purpose, we assume that the residuals $e_{1:j}$ are samples of a random variable $e$ drawn from a robust probability pdf, $P(e; \theta)$, where $\theta$ are the parameters. Then, the problem is cast into maximum likelihood estimation (MLE) or, equivalently into the minimization of the following negative log-likelihood function:

$$L(\theta | X_{1:j}, Z_{1:j}) = - \sum_{j=1}^J \log P(e_j; \theta),$$

The generalized Student-t distribution [19] writes:

$$P(e; \theta) = \int_0^\infty N(e; 0, w^{-1} \Sigma) \mathcal{G}(w; \mu, \nu) dw,$$

where $\mathcal{N}(\cdot; 0, \Sigma)$ is the zero-mean normal distribution with covariance $\Sigma \in \mathbb{R}^{3 \times 3}$. The random latent variable $w \in \mathbb{R}^+$ is drawn from a gamma distribution $\mathcal{G}(\cdot; \mu, \nu)$, and it plays the role of a precision. Therefore, the variables $w_{1:j}$ (one for each data point) characterize the landmark-vertex pair: the higher the more reliable. The model’s rigid and statistical parameters are gathered in the parameter vector $\theta = \{\rho, R, t, \Sigma, \mu, \nu\}$. Direct minimization of (2) is intractable. Expectation-maximization (EM) is therefore
adopted, namely the negative log-likelihood (2) is replaced with the expected complete-data negative log-likelihood:
\[
E_{W}[-\log P(X_{1:j}, Z_{1:j}, w_{1:j}|X_{1:j}, Z_{1:j}; \theta)].
\] (4)

In practice, EM alternates between the estimation of the weight posteriors – the means \(\pi_{1:j}\) and the estimation of the parameters \(\theta\) via minimization of (4):
\[
\theta^{*} = \text{argmin}_{\theta} \sum_{j=1}^{J} (\pi_{j}|Z_{j} - \rhoRX_{j} - t|^{2} + \log |\Sigma|). \tag{5}
\]

The standard solution to (5) is to assume an isotropic covariance, \(\Sigma = \sigma I_{3}\), which yields various closed-form solutions, e.g. [24, 51]. Indeed, after developing (5), one may verify that the term \(X_{j}R\Sigma^{-1}R^{T}X_{j}^{T}\) is proportional to \(X_{j}^{T}X_{j}^{T}\), where centered landmark coordinates were used. Nevertheless, the isotropic assumption cannot deal with anisotropic error distribution of the 3D landmark coordinates. In the case of a full covariance, the optimization of (5) with respect to the rotation parameters yields the following non-linear optimization problem:
\[
\begin{align*}
\min_{q} & \quad \text{trace} \left( \Sigma^{-1}(\rho^{2}R(q)AR(q)^{T} - 2sR(q)B) \right) \\
\text{s.t.} & \quad q^{*}q = 1,
\end{align*}
\] (6)

where \(q\) is a unit quaternion, thus reducing the number of parameters from nine to four. The six quadratic constraints and the quartic constraint needed to guarantee a rotation are replaced with one quadratic constraint, i.e. Appendix A.

While it iterates, Algorithm 1 (Appendix A) estimates (i) the posterior weight means, (ii) the pose parameters, (iii) the covariance, and (iv) the parameters of the gamma pdf. At convergence, the optimal parameters (denoted with an \(^{*}\)) are applied to \(X_{1:j}\) in order to obtain expression-preserving frontalized landmarks, whose coordinates in \(I_{f}\) are denoted \(Y_{1:j} \subset \mathbb{R}^{3}\), namely:
\[
Y_{j} = \rho^{*}R^{*}X_{j} + t^{*}, \quad \forall j \in \{1 \ldots J\}. \tag{7}
\]

### 3.2. Robust Deformable Shape Fitting

The next step is to fit a deformable 3D geometric model to these landmarks in order to obtain a frontal dense depth map of the face. For that purpose and without loss of generality, we consider a linear deformation model, e.g. the Basel Face Model (BFM) [40]. The latter consists of a 3D mesh with a set \(\hat{V}\) of \(N\) vertices, whose coordinates \(\hat{V}_{1:N} = \{\hat{V}_{n}\}_{n=1}^{N} \subset \mathbb{R}^{3}\) are parameterized by a statistical linear shape-model in the following way, i.e. Appendix C:
\[
\hat{V}_{n} = \overline{V}_{n} + W_{n}s, \quad \forall n \in \{1 \ldots N\}, \tag{8}
\]
where \(\overline{V}_{1:N} \subset \mathbb{R}^{3}\) are the vertices of a mean (neutral) shape, \(W_{1:N} \subset \mathbb{R}^{3 \times K}\) are reconstruction matrices, and \(s \in \mathbb{R}^{K}\) is a low dimensional embedding of the vertex set, with \(K \ll 3N\) (Appendix C). In order to fit this model to the frontalized landmarks \(Y_{1:j}\), we consider a subset of \(J = 68\) (\(\ll N\)) vertices with coordinates \(\hat{V}_{1:j}\), associated one-to-one to the frontalized landmarks.\(^{1}\) For that purpose, the statistical shape model must be deformed, such that the vertices \(\hat{V}_{1:j}\) are optimally aligned with the landmarks \(Y_{1:j}\). We thus obtain the residuals:
\[
r_{j} = Y_{j} - Q(\hat{V}_{j} + W_{j}s) - d, \quad \forall j \in \{1 \ldots J\}, \tag{9}
\]
where the rotation \(Q\) and the translation \(d\) align the image-centered and model-centered coordinate frames. As above, one can use the generalized Student-t distribution (3) and EM to robustly estimate the model parameters \(s\) and the weights \(\pi_{1:j}\). The shape parameters yield a closed-form expression, i.e. Appendix D:
\[
s^{*} = \left( \sum_{j=1}^{J} \pi_{j}A_{j}^{T}\Gamma^{-1}A_{j} + \kappa\Lambda^{-1} \right)^{-1} \left( \sum_{j=1}^{J} \pi_{j}A_{j}^{T}\Gamma^{-1}b_{j} \right), \tag{10}
\]
where \(\Gamma\) is a covariance, i.e. (37), and with the notations \(A_{j} = QW_{j}\) and \(b_{j} = Y_{j} - Q\hat{V}_{j} - d; \Lambda\) is a diagonal matrix containing the \(K\) principal eigenvalues of the shape embedding, and \(\kappa \in \mathbb{R}^{+}\). All the \(N\) vertices of this deformed shape can now be mapped onto the frontal view, namely \(\hat{V}_{n} = (\hat{V}_{n1}, \hat{V}_{n2}, \hat{V}_{n3})\), with:
\[
\hat{V}_{n} = Q^{*}(\overline{V}_{n} + W_{n}s^{*}) + d^{*}, \quad \forall n \in \{1 \ldots N\}. \tag{11}
\]

### 3.3. Synthesizing a Frontal Face Image

A frontal dense depth map is then computed in the following way. Remember that the shape vertices form a 3D triangulated mesh; therefore the projection of \(\hat{V}_{1:N}\) onto the image plane \(I_{f}\) form a 2D triangulated mesh whose vertices have \((\hat{V}_{n1}, \hat{V}_{n2})^{T}_{1:N}\) as image coordinates. Let \(n_{1}, n_{2}\) and \(n_{3}\) be the indexes of the vertices of a mesh triangle. We now compute the barycentric coordinates, \(\{\alpha_{1}, \alpha_{2}, \alpha_{3}\} \subset \mathbb{R}^{+}\) of a pixel \((a_{1}, a_{2})^{T}\) \(\in \mathbb{N}^{2}\) that lies inside that triangle, i.e. \(0 \leq \alpha_{1}, \alpha_{2}, \alpha_{3} \leq 1\). These barycentric coordinates are estimated by solving the following system of linear equations:
\[
\begin{align*}
\alpha_{1} &= \alpha_{1}\hat{V}_{n11} + \alpha_{2}\hat{V}_{n12} + \alpha_{3}\hat{V}_{n13} \\
\alpha_{2} &= \alpha_{1}\hat{V}_{n21} + \alpha_{2}\hat{V}_{n22} + \alpha_{3}\hat{V}_{n23} \\
\alpha_{3} &= \alpha_{1}\hat{V}_{n31} + \alpha_{2}\hat{V}_{n32} + \alpha_{3}\hat{V}_{n33},
\end{align*}
\] (12)

with the constraint \(\alpha_{1} + \alpha_{2} + \alpha_{3} = 1\). Once the barycentric coordinates are computed, the depth \(A_{3} \in \mathbb{R}\) associated with pixel \((a_{1}, a_{2})^{T}\) is computed by linear interpolation, namely:
\[
A_{3} = \alpha_{1}\hat{V}_{n13} + \alpha_{2}\hat{V}_{n23} + \alpha_{3}\hat{V}_{n33}. \tag{13}
\]

\(^{1}\)By abuse of notation, we use \(\{1 \ldots J\} \subset \{1 \ldots N\}\).
The above procedure is repeated for all the triangles and for all the points inside each triangle, thus obtaining a frontal dense depth map for each face pixel $A = (a_1, a_2, A_3)\top$. The final face frontalization step consists of synthesizing a frontal image of the input face. For this purpose we use a standard 3D shape visualization method, i.e., Appendix E.

3.4. Performance Analysis

In order to assess the performance of GStudent-EM, we compared it with the use of a Gaussian distribution and with a Gaussian-uniform mixture (GUM) distribution with a single Gaussian component [56]. The use of a uniform component in addition to Gaussian components in a mixture was initially proposed in [5]. In [37] it was proposed to be used in the context of point-set registration with the rationale of eliminating points in one set that don’t have a match in the other set. One should note that in this paper we want to weight the relative importance of already registered point pairs and to reduce the influence of those point pairs that cannot be rigidly aligned.

In more detail, the likelihood $P(e; \theta)$ in (2) is replaced with: (i) a Gaussian distribution with isotropic covariance, i.e., $[24]$, (ii) a Gaussian distribution with anisotropic (full) covariance, and (iii) a Gaussian-uniform distribution. The associated algorithms are referred to as Horn, Gen-Horn and GUM-EM, respectively. Starting with a frontal set of landmarks $Z_{i,j}$ with image coordinates normalized in the interval $[0, 1]$, we randomly generated $P = 500$ poses (rotation, translation and scale). A landmark set $X^p_{1:j}$ associated with a pose $p$ is therefore simulated with $X^p_j = s^pR^pZ_j + t^p + e^p(b)$, where $b > 0$ is a scalar that controls the level of noise and $p$ is the trial (pose) index. $b$ can be the variance associated with isotropic Gaussian noise $e \sim N(0, \Sigma)$, or the volume associated with uniform noise $e \sim U(-b/2, b/2)^3$. For each trial we randomly split the samples into an equal number of inliers and outliers. Inliers are corrupted by anisotropic Gaussian noise with a fixed variance, while outliers are corrupted with either (i) uniform errors of increasing amplitude or (ii) anisotropic Gaussian errors of increasing total variance. The evaluation is based on the root mean square error (RMSE) between the estimated pose and the ground-truth pose, namely $\text{RMSE}(R) = 1/P\sum_{p=1}^{P}||R^p - \hat{R}^p||^2)^{1/2}$ as well as similar formulas for translation and scale. These RMSEs allow to directly compare robust and non-robust estimators. The RMSE curves are shown on Figure 2.

In the light of these experiments, one concludes that both GStudent-EM and GUM-EM yield robustness against outliers. GUM-EM performs better than GStudent-EM when the outliers are drawn from a uniform distribution. Figure 2(a): this can be easily explained since the simulated outliers follow the same statistical distribution as the one of the model. It is worth to be noticed that the posteriors estimated by GUM-EM are normalized between 0 and 1, hence their values are a relative measure of the quality of the point pairs. In contrast, the data weights estimated with GStudent-EM are random variables whose realizations can vary from zero to a very high number (higher the better). Therefore, the weight values are an absolute measure of the relevance of data points. While GUM-EM yields very good results, GStudent-EM offers better theoretical guarantees and a better way to weight the data; its behavior is easier to predict because all the parameters are estimated by the EM algorithm.

3.5. Implementation Details

The proposed method starts with 3D facial landmark extraction, or 3D face alignment (3DFA). Recently there has been a flourishing literature on this topic, yielding several DNN models and associated software packages, e.g., [7, 62, 8, 16, 13, 64, 30, 50, 38]. Moreover, these models were thoroughly trained, validated and benchmarked using very large datasets that are automatically annotated, thus limiting the inherent errors associated with manual intervention, e.g., [29, 28, 64, 12]. We thoroughly analysed and benchmarked four publicly available software packages associated with four 3DFA methods [8, 16, 64, 50]. We could not find a significant difference in performance between these four 3DFA methods. In practice, the results reported below were obtained with the method of [8].

All the computations inside Algorithm 1 and Algorithm 2 are in closed-form, with the notable exception of the estimation of the rotation. The minimization of (6) is carried out using the sequential least squares programming
The (SLSQP) solver of `scipy.optimize`, in combination with a root-finding software package [32]. The SLSQP minimizer found at the previous EM iteration is used to initialize the current EM iteration. At the start of EM, the closed-form method of [24] is used to initialize the rotation.

The proposed method also requires a deformable shape model (Appendix C). We consider a set \( S^I = \{ S^I_1, ..., S^I_{M} \} \subset \mathbb{R}^{3N} \) of \( M \) shapes, where \( S^I_m \) is a concatenation of \( N \) 3D coordinates that corresponds to face identity \( m \), and where each face was scanned in a frontal view and with a neutral expression. We also consider a set \( S^E = \{ S^E_1, ..., S^E_{M}, S^E_M \} \subset \mathbb{R}^{3N} \) that contains \( M \) scans with facial expressions, associated one-to-one, with the \( M \) identities. Let \( S^\Delta_m = S^E_m - S^I_m \) be the expressive-neutral difference of face identity \( m \), namely the expressive offset. A face \( S \) can be reconstructed from its identity and expression embeddings (Appendix C):

\[
\hat{S} = \bar{S}^I + \bar{S}^\Delta + \bar{U}^I s^I + \bar{U}^\Delta s^\Delta,
\]

where \( \bar{S}^I \) and \( \bar{S}^\Delta \) are the means associated with the identity- and the expressive-neutral difference, matrices \( \bar{U}^I \) and \( \bar{U}^\Delta \) contain the \( K \) principal eigenvectors, and \( s^I \) and \( s^\Delta \) are the corresponding embeddings. Note that we use the notation \( \hat{S} \) to emphasize that the reconstructed shape is an approximation of \( S \).

In practice we use the publicly available Basel Shape Model (BSM) [40] augmented with [9]. This provides registered face scans: \( M = 200 \) in a frontal view and with neutral expressions, corresponding to \( M \) different identities, and an additional set of \( M \) expressive scans of the same identities. A scan consists of a triangulated mesh with \( N = 53490 \) vertices. All the scans have the same number of vertices that are registered. The dimension of the embedding is \( K = 200 \), hence \( K \ll 3N \). The landmarks \( Z_{1:1}^I \) correspond to the vertices of the mean identity \( \bar{S}^I \).

The processing time for a \( 256 \times 256 \) face image is of 1.11 seconds on an Intel(R), Xeon(R) W-2145, 3.70GHz CPU equipped with a Quadro RTX 4000 GPU. This time decomposes as follows: 3D landmark extraction (0.48 s), pose estimation (0.02 s), model fitting (0.23 s), depth map interpolation and face warping (0.38 s).

### 4. Experiments

We now evaluate the performance of face frontalization for the task of lip reading. The evaluation is twofold. First, we use the OuluVS2 dataset [2] that contains pairs of frontal and profile videos of speaking participants for a large number of subjects. The evaluation consists of computing a metric between an image obtained by face frontalization of a profile view of a speaker, with an image containing a frontally-viewed face of the same speaker. It is important that the profile and frontal images are recorded with synchronized cameras in order to capture the same expression. Consequently, the proposed evaluation is based on image-to-image comparison. Several metrics were developed in the past for comparing two images, e.g. feature-based and pixel-based metrics. In this work we use the zero-mean normalized cross correlation (ZNCC) coefficient between two image regions, a measure that has successfully been used for stereo matching, e.g. [46]. ZNCC is invariant to differences in brightness and contrast between the two images, due to the normalization with respect to mean and standard deviation. Second we use a lip-reading network in conjunction with a dataset that contains short videos of speakers that utter a single word, together with the ground-truth annotations (word labels). We devise an experimental protocol that measures the effect of face frontalization on the word classification score.

Let \( R_f(h, v) \subset I_f \) be a region of size \( H \times V \) whose center coincides with pixel location \((h, v)\) of a frontalized image \( I_f \). Similarly, let \( R_e(h, v) \subset I_e \) be a region of the same size and whose center coincides with pixel location \((h, v)\) of a ground-truth image \( I_t \). The ZNCC coefficient between these two regions writes:

\[
\text{ZNCC}(h, v, \delta h', \delta v') = \max_{\delta h, \delta v} \left\{ \frac{\text{Cov} [R_f(h, v), R_e(h + \delta h, v + \delta v)] \cdot \text{Var} [R_f(h, v)]^{1/2} \cdot \text{Var} [R_e(h + \delta h, v + \delta v)]^{1/2}}{\text{Var} [R_f(h, v)]^{1/2} \cdot \text{Var} [R_e(h + \delta h, v + \delta v)]^{1/2}} \right\},
\]

where \( \text{Cov} [\cdot, \cdot] \) is the centered covariance between the two regions, \( \text{Var} [\cdot] \) is the centered variance of a region, \( \delta h \) and \( \delta v \) are horizontal and vertical shifts, and \( \delta h' \) and \( \delta v' \) are the horizontal and vertical shifts that maximize the ZNCC coefficient. ZNCC lies in the interval \([0, 1]\).

The OuluVS2 dataset [2] targets the understanding of visual speech perception – the analysis of non-rigid lip motions that are associated with speech production. The dataset was recorded in an office with ordinary (artificial and natural) lighting conditions. The recording setup consists of five synchronized cameras (2 MP, 30 FPS) placed in different points of view: \(0^\circ, 30^\circ, 45^\circ, 60^\circ, 90^\circ\). The dataset contains 5 \times 780 videos recorded with 53 participants. Each participant was instructed to read loudly several text sequences displayed on a computer monitor placed slightly to the left and behind the \(0^\circ \) (frontal) camera. While participants were asked to keep the head still, natural uncontrolled head movements and body position changes were inevitable. As a consequence the actual head pose varies from one participant to another and there is no exact match between the head and the camera orientations.
Table 1: Mean ZNCC coefficients for 15 participants from the OuluVS2 dataset. ZNCC lies in the interval [0, 1].

<table>
<thead>
<tr>
<th>Method, Reference</th>
<th>Principle</th>
<th>ZNCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hassner et al. [22]</td>
<td>2D-to-3D fitting+symmetry</td>
<td>0.780</td>
</tr>
<tr>
<td>Banerjee et al. [4]</td>
<td>2D-to-3D fitting+symmetry</td>
<td>0.739</td>
</tr>
<tr>
<td>Zhou et al. [61]</td>
<td>2D-to-3D fitting + GAN</td>
<td>0.801</td>
</tr>
<tr>
<td>Yin et al. [55]</td>
<td>2D-to-2D GAN</td>
<td>0.773</td>
</tr>
<tr>
<td>Proposed</td>
<td>3D-to-3D robust fitting</td>
<td>0.841</td>
</tr>
</tbody>
</table>

Table 2: ZNCC scores for nine participants as a function of estimated yaw angle (in degrees) that corresponds to the horizontal head orientation computed with the proposed 3D head-pose estimator. For each participant, the best scores are in **bold** and the second best are in *slanted bold*.

<table>
<thead>
<tr>
<th>Part.</th>
<th>Yaw</th>
<th>[22]</th>
<th>[4]</th>
<th>[61]</th>
<th>[55]</th>
<th>Prop.</th>
</tr>
</thead>
<tbody>
<tr>
<td>#31</td>
<td>19.1</td>
<td>0.905</td>
<td>0.856</td>
<td>0.822</td>
<td>0.875</td>
<td>0.927</td>
</tr>
<tr>
<td>#01</td>
<td>23.5</td>
<td>0.915</td>
<td>0.893</td>
<td>0.884</td>
<td>0.921</td>
<td>0.909</td>
</tr>
<tr>
<td>#02</td>
<td>24.9</td>
<td>0.888</td>
<td>0.878</td>
<td><strong>0.929</strong></td>
<td>0.881</td>
<td><strong>0.956</strong></td>
</tr>
<tr>
<td>#10</td>
<td>29.0</td>
<td>0.805</td>
<td><strong>0.812</strong></td>
<td>0.873</td>
<td>0.792</td>
<td><strong>0.812</strong></td>
</tr>
<tr>
<td>#23</td>
<td>30.0</td>
<td>0.810</td>
<td>0.857</td>
<td>0.819</td>
<td>0.817</td>
<td><strong>0.847</strong></td>
</tr>
<tr>
<td>#27</td>
<td>32.9</td>
<td>0.685</td>
<td>0.852</td>
<td>0.824</td>
<td>0.772</td>
<td>0.787</td>
</tr>
<tr>
<td>#19</td>
<td>37.8</td>
<td>0.752</td>
<td>0.650</td>
<td>0.662</td>
<td>0.677</td>
<td><strong>0.755</strong></td>
</tr>
<tr>
<td>#12</td>
<td>38.5</td>
<td>0.731</td>
<td>0.713</td>
<td><strong>0.755</strong></td>
<td>0.683</td>
<td><strong>0.770</strong></td>
</tr>
<tr>
<td>#21</td>
<td>40.6</td>
<td>0.632</td>
<td><strong>0.743</strong></td>
<td>0.653</td>
<td>0.673</td>
<td><strong>0.766</strong></td>
</tr>
<tr>
<td>Mean</td>
<td>0.791</td>
<td>0.801</td>
<td><strong>0.802</strong></td>
<td>0.787</td>
<td>0.836</td>
<td></td>
</tr>
</tbody>
</table>

We compared the proposed methods with four state-of-the-art methods for which the code is publicly available, [22, 4, 61, 55]. We applied the frontalization to images extracted from the videos recorded with the 30° camera ($I_p$) and compared the results with the "ground-truth", namely the corresponding images extracted from the videos recorded with the 0° camera ($I_t$). Notice that videos recorded with higher viewing angles, i.e. 45°, 60° and 90°, can be hardly exploited for lip reading. For each frontalized image $I_f$ we extract the mouth region $R_f$ and we search in the associated ground-truth image $I_t$ for the best-matching region $R_t$. This provides a ZNCC coefficient (15) for each query image $I_p$. Notice that (15) only cares about the horizontal and vertical shifts in the image plane and assumes that the frontalized face and the corresponding ground-truth frontal face share the same scale. In practice, different frontalization algorithms output faces at different scales. For this reason and for the sake of fairness, prior to applying (15), we extract facial landmarks from both the frontalized and ground-truth faces and we use a subset of this set of landmarks to estimate the scale factor between the two faces.

We randomly selected 30 video pairs, recorded with the 30° and 0° cameras, respectively, associated with 15 participants from the OuluVS2 dataset. Each video contains 160 images, hence there are $30 \times 160 = 4800$ image pairs in our benchmark. The mean ZNCC coefficients obtained with four state-of-the-art methods and with the proposed method are displayed in Table 1.

We noticed that there were important discrepancies in
We performed two sets of experiments. The first experiment replaces their frontalization with ours and fine-tunes the lip reading model. For each one of the 500 words, we used 200 videos for training and 20 videos for validation. We tested the pre-trained and fine-tuned models with the same test set that consists of 20 videos for each one of the 500 word vocabulary. As for test, we replaced the built-in frontalization with ours as well as with the other frontalizations under comparison. Frontalization examples are shown on Figure 7 of Appendix F. The results summarized in Table 3 show an increase in performance when our frontalization method is used to fine tune the model, instead of the one used in [11].

5. Conclusions

In this paper we proposed a face frontalization method that preserves non-rigid facial deformations. This stays in contrast with several state-of-the-art frontalization methods that are designed to boost the performance of face recognition by predicting as-neutral-as-possible frontal faces. We conducted a series of experiments in order to analyze the effect of frontalization on the task of visual speech recognition, whose success heavily relies on the analysis of non-rigid mouth motions, i.e. lip reading. For this purpose, we used two datasets.

We proposed an evaluation pipeline that consists of measuring the ZNCC score between a frontalized face and a frontal view of the same face. We compared our method with four state-of-the-art methods that use various geometric and DNN models. This benchmark reveals that the proposed method better preserves the shape of the mouth by a significant margin, and this without making recourse to facial symmetry or to DNN-based inpainting techniques to fill in the occluded areas.

The LRW dataset contains videos of persons uttering speech. Unlike the OuluVS2 participants who keep their heads still, the LRW participants perform large and unexpected head motions. We plugged our frontalization model into a DNN-based lip reading model and we thoroughly analyzed its effect on the word classification score. These experiments empirically demonstrate that the proposed robust frontalization improves these scores significantly, Table 3.

As already outlined, face frontalization may well be viewed as a process of discriminating between rigid head movements and non-rigid facial deformations, hence it can be used to eliminate head motions that naturally accompany speech production. The preliminary lip reading experiments described above are clear evidence that expression-preserving frontalization boosts the performance of visually-augmented speech technologies.

<table>
<thead>
<tr>
<th>Training</th>
<th>[22]</th>
<th>[61]</th>
<th>[55]</th>
<th>[33]</th>
<th>Prop.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-trained [33]</td>
<td>60</td>
<td>59</td>
<td>20</td>
<td>87</td>
<td>82</td>
</tr>
<tr>
<td>Fine-tuned [33]</td>
<td>64</td>
<td>66</td>
<td>24</td>
<td>88</td>
<td>85</td>
</tr>
</tbody>
</table>

Table 3: Word classification scores (WCSs) in %. First row: Pre-trained lip reading model [33]. Second row: Fine tuned lip reading model, where our frontalization replaces the built-in frontalization [35]. The pre-trained and fine-tuned models were tested using several frontalization methods.
A. Robust Estimation of the Rigid Alignment Between Two 3D Point Sets

In this appendix, we address the problem of robust alignment between two 3D point sets, \( X \) with coordinates \( X_{1:J} = \{ X_j \}_{j=1}^J \subset \mathbb{R}^{3 \times J} \) and \( Z \) with coordinates \( Z_{1:J} = \{ Z_j \}_{j=1}^J \subset \mathbb{R}^{3 \times J} \), respectively. As mentioned in Section 3 the residuals (1) are samples of a random variable drawn from the generalized Student t-(GStudent) distribution [19]:

\[
P(e; \Sigma, \mu, \nu) = \frac{\Gamma\left(\frac{\mu + 3}{2}\right)}{\Gamma\left(\frac{\mu}{2}\right)2^{\frac{\mu}{2}}\pi^{\frac{3}{2}}\nu^\frac{3}{2}} \left(1 + \frac{\|e\|^2}{\nu\Sigma} \right)^{-\frac{\mu + 3}{2}}
\]

where \( \mu \) and \( \nu \) are the parameters of the prior gamma distribution of the precision variable \( \nu \), and \( \Gamma(\cdot) \) is the gamma function. Without loss of generality, we set \( \nu = 1 \) [19]. Notice that using the conjugate prior property, the posterior distribution of \( \nu \) is also a gamma distribution, namely the posterior gamma distribution:

\[
P(w|e; \Sigma, \mu, \nu) = \mathcal{N}(e; 0, w^{-1}\Sigma)\mathcal{G}(w, \mu, 1)
\]

with parameters:

\[
a = \mu + \frac{3}{2}, \quad b = 1 + \frac{1}{2}\|e\|^2
\]

The posterior mean of the precision is:

\[
\overline{\nu} = E[w|e] = \frac{a}{b}
\]

The expected complete-data negative log-likelihood (4), \( E_W[-\log P(e_{1:J}, w_{1:J}|e_{1:J}; \theta)] \) and in this case the parameter vector is a concatenation of the alignment and pdf parameters, namely \( \theta = \{ \rho, \mathbf{R}, \mathbf{t}, \Sigma, \mu \} \) since we set \( \nu = 1 \). This yields the minimization of:

\[
Q(\theta) = \sum_{j=1}^J (\overline{\nu}_j \|Z_j - \rho\mathbf{R}X_j - \mathbf{t}\|^2 + \log |\Sigma|)
\]

By taking the derivatives of (20) with respect to the parameters and equating to zero, we obtain the following analytical expressions for optimal values of the following parameters:

\[
t^* = Z - \rho\mathbf{R}X,
\]

\[
\rho^* = \sqrt{\frac{\sum_{j=1}^J w_j Z_j^\top \Sigma^{-1} Z_j}{\sum_{j=1}^J w_j (\mathbf{R}X_j)^\top \Sigma^{-1} (\mathbf{R}X_j)}}
\]

\[
\Sigma^* = \frac{1}{N} \sum_{j=1}^J w_j (Z_j - \rho\mathbf{R}X_j)(Z_j - \rho\mathbf{R}X_j)^\top
\]

with the following substitutions:

\[
\begin{align*}
\overline{\mathbf{X}} &= \frac{\sum_{j=1}^J w_j X_j}{\sum_{j=1}^J w_j}, & \mathbf{X}' &= \mathbf{X} - \overline{\mathbf{X}}, \\
\overline{\mathbf{Z}} &= \frac{\sum_{j=1}^J w_j Z_j}{\sum_{j=1}^J w_j}, & \mathbf{Z}' &= \mathbf{Z} - \overline{\mathbf{Z}}.
\end{align*}
\]

By substituting (21) in (20) and using the notations above, the rotation is estimated by minimization of:

\[
\mathbf{R}^* = \arg\min_{\mathbf{R}} \frac{1}{2} \sum_{j=1}^J (w_j \|Z_j - \rho\mathbf{R}X_j\|^2)
\]

The minimizer (26) yields a closed-form solution for an isotropic covariance, i.e. \( \Sigma = \sigma^2 \mathbf{I} \). In the general case, however, one has to make recourse to nonlinear minimization. It is practical to represent the rotation with a unit quaternion and to replace (26) with 6, namely:

\[
q^* = \arg\min_q \text{trace} (\Sigma^{-1}(\rho^2\mathbf{R}(q)\mathbf{A}(q)^\top - 2s\mathbf{R}(q)\mathbf{B})) \quad \text{s.t.} \quad q^\top q = 1,
\]

where \( \mathbf{A} = \sum_{j=1}^J w_j X_j'X_j'^\top \) and \( \mathbf{B} = \sum_{j=1}^J w_j X_j'Z_j'^\top \). The minimization just above is carried out using a sequential quadratic programming method [6]. The parameter \( \mu \) is updated by solving the following equation, where \( \Psi(\cdot) \) is the digamma function:

\[
\mu = \Psi^{-1}\left(\frac{1}{n} \sum_{j=1}^J \log b_j\right)
\]

This yields GStudent-EM, namely Algorithm 1.

B. Performance Analysis

The proposed GStudent-EM is compared with: a Gaussian distribution with isotropic covariance (Horn), i.e. [24], a Gaussian distribution with anisotropic (full) covariance (Gen-Horn), and a Gaussian-uniform distribution (GUM-EM). Please refer to Figure 4 and Figure 5. The RMSE errors in scale follow the same patterns.
C. Deformable Shape Model

In this appendix we summarize the deformable shape model that we use. The model is based on a training set \( S = \{S_1, \ldots, S_m, \ldots, S_M \} \) of \( M \) 3D face scans. Each 3D face scan is described with a triangulated mesh composed of \( N \) 3D points, or vertices, \( V_{1:N} = \{V_1, \ldots, V_N\} \), where the coordinates of each vertex are \( V_n = (V_{n1}, V_{n2}, V_{n3})^\top \). Hence, a shape in \( S \) is described by the vector \( S' \):

\[
S = (V_{11}, V_{12}, V_{13}, \ldots, V_{N1}, V_{N2}, V_{N3})^\top \in S \subset \mathbb{R}^{3N}.
\]

It is assumed that all the shapes are represented in the same coordinate frame, that they share the same number of vertices and that there is a one-to-one correspondence between their vertices, namely \( \{V_{nm} \leftrightarrow V_{n'm'}\}_{n=1}^N \), for any two shapes in the training set, \( S_m \) and \( S_{m'} \) and for each vertex index \( n \). Notice, however, that the problem of finding a rigid alignment between \( M \) shapes based on point-to-point correspondences is not trivial because of the presence of non-rigid shape deformations and of large shape variabilities. In what follows we will treat each shape/face as a point set. The covariance matrix associated with these \( M \) registered shapes, \( \{S_m\}_{m=1}^M \), is:

\[
\Omega = \frac{1}{M} \sum_{m=1}^M (S_m - \bar{S})(S_m - \bar{S})^\top,
\]

where \( \bar{S} = (V_{11}^\top, \ldots, V_{N1}^\top, \ldots, V_{N3}^\top)^\top \in \mathbb{R}^{3N} \) is the mean shape with, \( \bar{V}_n = 1/M \sum_{m=1}^M V_{nm} \). The covariance \( \Omega \in \mathbb{R}^{3N \times 3N} \) is a symmetric semi-definite positive matrix, hence its eigendecomposition writes \( \Omega = \mathbf{U} \Lambda \mathbf{U}^\top \), where
Figure 6: A training set of $M$ shapes (black dots), characterized by their mean (orange dot) and covariance, is projected onto the space spanned by the principal eigenvectors of the shape covariance matrix, $U_1$ to $U_K$. The number $K$ of principal eigenvectors is selected such that the sum of the associated eigenvalues represents 95% of the total variance. A shape $S_m$ projects onto a shape embedding $s_m$. Conversely, shape $S_m$ (blue dot), reconstructed from $s_m$, approximates shape $S_m$.

$\mathbf{Algorithm 1: GStudent-EM:}$ Robust estimation of the rigid transformation between two 3D point sets.

**Data:** Centered point coordinates $X_{1:N}$ and $Z_{1:N}$

**Initialization of $\theta^{\text{old}}$:** Use a closed-form solution for the rotation to evaluate $\mathbf{R}^{\text{old}}$ and $\rho^{\text{old}}$; evaluate $\Sigma^{\text{old}}$ using (23). Provide $\mu^{\text{old}}$.

**while** $\|\theta^{\text{new}} - \theta^{\text{old}}\| > \epsilon$ **do**

**E-step:** evaluate $a^{\text{new}}$ and $b^{\text{new}}_{1:N}$ using (18) with $\theta^{\text{old}}$, then evaluate $\mu^{\text{new}}_{1:N}$ using (19);

update the centered coordinates $Y_{1:N}$;

**M-scale-step:** estimate $\rho^{\text{new}}$ using (22);

**M-rotation-step:** estimate $\mathbf{R}^{\text{new}}$ via minimization of (26);

**M-covariance-step:** evaluate $\Sigma^{\text{new}}$ using (23);

$\theta^{\text{old}} \leftarrow \theta^{\text{new}}$;

**end**

**Optimal translation:** Evaluate the translation vector using (21);

**Result:** Optimal scale $\rho^*$, rotation $\mathbf{R}^*$, translation $t^*$, covariance $\Sigma^*$, and weights $\mu_{1:N}$.

This guarantees that the reconstructed shape belongs to the family of training shapes with 99% confidence.

**D. Robust Estimation of the Alignment Between a Deformable Shape and a 3D Point Set**

In this appendix we extend the robust alignment method of Appendix A to deformable shapes. The expression (20) can be re-written as:

$$Q(\theta) = \sum_{j=1}^{J} (\pi_j \|Y_j - Q(V_j + W_j s)\|_2^2 + \log |\Gamma|)$$

$$+ \kappa s^T \tilde{\Lambda}^{-1} s,$$

(35)

with the property $\tilde{U}^T \tilde{U} = I_K$. Matrix $\tilde{U}^T$ projects the centered shapes from $\mathbb{R}^{3N}$ onto $\mathbb{R}^K$:

$$s_m = \tilde{U}^T (S_m - \mathcal{S}),$$

(30)

where $s_m \in \mathbb{R}^K$ is a low-dimensional embedding of $S_m$. Conversely, it is possible to reconstruct an approximation of $S_m$, denoted $\hat{S}_m$, from its low-dimensional embedding:

$$\hat{S}_m = \mathcal{S} + \tilde{U} s_m,$$

(31)

which is a concatenation of:

$$\hat{V}_n = \tilde{V}_n + W_n s_m,$$

(32)

where the $N$ block-matrices $W_n \in \mathbb{R}^{3 \times K}$ are such that:

$$\tilde{U}^T = (W_1^T \ldots W_n^T \ldots W_N^T)$$

(33)

The concept of statistical shape model is summarized in Fig. 6. A training set of $M$ shapes is embedded into a low-dimensional space spanned by the $K$ principal eigenvectors of the covariance matrix (29) associated with the training set and is characterized by the $K$ principal (largest) eigenvalues, where an eigenvalue $\lambda_k$ corresponds to the variance along direction $U_k$. Importantly, a valid shape $\hat{S}$ is a shape reconstructed from an embedding $s$ that lies inside an ellipsoid whose half axes are equal to $\sqrt{\lambda_k}$, or:

$$\hat{S} = \mathcal{S} + \tilde{U} s, \text{ s.t. } s^T \tilde{\Lambda}^{-1} s \leq 1.$$  

(34)
where the statistical shape model is rotated, translated and deformed in order to be aligned with a set of landmarks, and where the last term constrains the embedding $s$ to lie inside a 99% confidence ellipsoid, i.e. (34).

By taking the derivatives of (35) with respect to the parameters, i.e. $\theta = \{Q, d, \Gamma, s, \mu\}$, and equating to zero, we obtain the following analytical expressions for the optimal values of the following parameters:

$$d^* = Y - Q\hat{V},$$

$$\Gamma^* = \frac{1}{N} \sum_{j=1}^{J} \pi_j (Y'_j - Q\hat{V}'_j)(Y'_j - Q\hat{V}'_j)^\top,$$

where similar notations as in Appendix A and Appendix C were used, i.e (24), (25) and (31). An identical nonlinear optimization problem for the rotation is obtained as well:

$$Q^* = \arg\min_Q \sum_{j=1}^{J} \pi_j \|Y'_j - Q\hat{V}'_j\|_1^2.$$  

Additionally, there is a closed-form solution for the optimal embedding:

$$s^* = \left(\sum_{j=1}^{J} \pi_j A_j \Gamma^{-1} A_j + \kappa \mathbf{I}\right)^{-1} \left(\sum_{j=1}^{J} \pi_j A_j \Gamma^{-1} b_j\right),$$

where the following notations were used: $A_j = QW_j$ and $b_j = Y_j - Q\hat{V}_j - d$. The associated EM algorithm is similar to Algorithm 1; the most notable difference is that the M-step alternates between the estimation of the rigid parameters and the shape parameters, namely Algorithm 2.

### Algorithm 2: Robust fitting of a deformable shape model to a 3D point set.

**Data:** Centered landmark coordinates $Y'_{1:j}$, mean shape $\bar{V}_{1:j}$ and shape reconstruction matrices $W_{1:j}$

**Initialization of $\theta^{old}$:** Set $s^{old} = 0$, use the closed-form solution [24] to evaluate $Q^{old}$; evaluate $\Gamma^{old}$ using (37). Provide $\mu^{old}$. Compute the centered vertex coordinates $\hat{V}'_{1:j}$.

**while** $\|\theta^{new} - \theta^{old}\| > \epsilon$ **do**

**E-step:** evaluate $q^{new}$ and $b_{1:N}^{new}$ using (18) with $\theta^{old}$, then evaluate $\pi_{1:N}^{new}$ using (19);

**Update the centered coordinates:**

**M-rotation-step:** Estimate $Q^{new}$ via minimization of (38);

**M-covariance-step:** Evaluate $\Gamma^{new}$ using (37);

**M-shape-step:** Evaluate $s^{new}$ using (39);

**M-mu-step:** Evaluate $\mu^{new}$ using (28);

$\theta^{old} \leftarrow \theta^{new}$;

**end**

**Optimal translation:** Evaluate the translation vector using (36);

**Result:** Optimal rotation $Q^*$, translation $d^*$, covariance $\Gamma^*$, and weights $\pi_{1:N}$.

Because of self occlusions, Eqs. (40) and (41) map several mesh vertices, $A^{1:K}$, at the same pixel location $(b_1, b_2)'$, but with different depth values $B_{3}^{1:K}$. Notice that only the vertex with the smallest depth value is visible in the input image. Consequently, vertices that are not visible in the input image don’t have any photometric information associated with them and hence, they give rise to blank areas in the frontalized image. The final face frontalization step consists of synthesizing a frontal image:

$$I_f(a_1, a_2) = \begin{cases} I_p(b_1, b_2), & \text{if } B_3 = \min_k \{B_{3}^{B_{3}}\}_{k=1}^{K} \\emptyset, & \text{otherwise,} \end{cases}$$

where $\emptyset$ denotes that there is no photographic information available with that pixel.

### F. Frontalization Examples from the LRW Dataset

Figure 7 shows a few frontalization results obtained with examples from the LRW dataset that is used for the lipreading benchmark. Since there are no ground-truth frontal faces available with this dataset, the results are qualitative. Notice that for frontally viewed faces with a neutral expression, the frontalization methods perform equally
well. However, one may notice that several existing methods don’t preserve facial expressions, e.g. mouth shape. The qualitative results observed with these examples correlate well with the ZNCC scores summarized in Tables 1 and 2 and with the lip reading performance summarized in Table 3.

F.1. Fine-tuning the Lip-reading Model of [33]

In this appendix we provide implementation details on fine-tuning the lip reading model of [33] for a 100 isolated word recognition (100-IWR) task and using the LRW dataset. 100 word categories were randomly selected from the dataset which contains 500 word categories. For each word, we use 150 videos for fine-tuning, 20 videos for validation and 20 videos for testing, from a total of 1000, 50 and 50 videos per word category, respectively. All the selected videos were previously frontalized using the proposed method. The TCN-based model of [33] is trained for a 500-IWR task. Therefore we replaced the 500 cell linear transformation and softmax layers with corresponding 100 cell layers. The Adam optimizer was used with a cosine scheduler. The weight decay was set to $1 \times 10^{-4}$ and the batch size was set to 32. The fine-tuning undergoes three stages, each stage being trained over 30 epochs as follows. In the first stage, the weights are frozen except for the last multi-scale TCN layer. The learning rate is set to $1 \times 10^{-5}$. As for the second stage, the fine-tuned model from the first stage is reloaded, with at this time the multi-scale TCN and the last layer of the ResNet being unfrozen. The learning rate is set to $1 \times 10^{-6}$. Similarly, the last stage uses the fine-tuned model from the second stage. The multi-scale TCN and the last two layers of the ResNet are unfrozen. The learning rate is set to $1 \times 10^{-6}$. After the three-stage fine-tuning, only the weights providing the highest validation accuracy are saved.

References


Figure 7: Frontalization examples from the LRW dataset. We remind that the methods of [22] and [4] combine 2D-to-3D geometric mapping with facial symmetry, that the method of [55] trains a GAN to predict a frontal face from a profile face, and that the method of [61] combines 2D-to-3D geometric mapping with 2D-to-2D nonlinear mapping based on GAN.
face alignment. In IEEE International Conference on Automatic Face & Gesture Recognition, pages 399–403. IEEE, 2018. 5


[27] Rui Huang, Shu Zhang, Tianyu Li, and Ran He. Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis. In IEEE International Conference on Computer Vision, pages 2439–2448, 2017. 2


