Robust and Accurate 3D Head Pose Estimation through 3DMM and Online Head Model Reconstruction

Yu Yu, Kenneth Alberto Funes Mora, Jean-Marc Odobez
Idiap Research Institute, CH-1920, Martigny, Switzerland
EPFL, CH-1015, Lausanne, Switzerland
{yyu, kfunes, odobez}@idiap.ch

***Abstract*—Accurate and robust 3D head pose estimation is important for face related analysis. Though high accuracy has been achieved by previous works based on 3D morphable model (3DMM), their performance drops with extreme head poses because such models usually only represent the frontal face region. In this paper, we present a robust head pose estimation framework by complementing a 3DMM model with an online 3D reconstruction of the full head providing more support when handling extreme head poses. The approach includes a robust online 3DMM fitting step based on multi-view observation samples as well as smooth and face-neutral synthetic samples generated from the reconstructed 3D head model. Experiments show that our framework achieves state-of-the-art pose estimation accuracy on the BIWI dataset, and has robust performance for extreme head poses when tested on natural interaction sequences.**

I. INTRODUCTION

Estimating head pose is an important and useful task often required before performing face alignment [1], facial expression analysis [2], head gesture recognition [3], or social interaction analysis [4], [5], [6]. Although there have been important advances in recent years, traditional visual based head pose estimation suffers from difficulties such as human shape and appearance variability, extreme head poses, facial expressions, the non-rigid nature of the the face, and illumination variations. The development of consumer 3D RGB-D sensors such as Kinect offers an alternative solution, since the depth measures they provide represents the fundamental information that is inherently required for pose estimation.

A number of depth-based approaches have been proposed for head pose estimation [7][8][9]. To the best of our knowledge, those relying on registering a 3D face mesh potentially learned online to depth observation sequences have achieved the best performance so far [10]. They often use 3D Morphable Models (3DMM) [11] as mesh representation, since they provide a linear and low dimensional representation of the 3D facial shape allowing online and well constrained model adaptation by finding the coefficients for the subject of interest.

However, as illustrated Fig. 1a), a limitation of most 3DMM face models is that they do not contain the top, side and back parts of the head because it is actually quite difficult to extract linear statistical basis from the variations of the hairs (and even the ears) in these parts. Although observing the frontal face is often the main interest in applications, being able to track the head even under more profile or extreme poses is required to avoid tracking interruptions or failures and manage such poses as encountered in non-constraint natural setting such as our UBImpressed sequences, Fig. 6b. A model only relying on the frontal face lacks the support to handle these cases, as shown in Fig. 1b where the face becomes almost invisible and the estimation is not robust.

978-1-5090-4023-0/17/$31.00 ©2017 IEEE

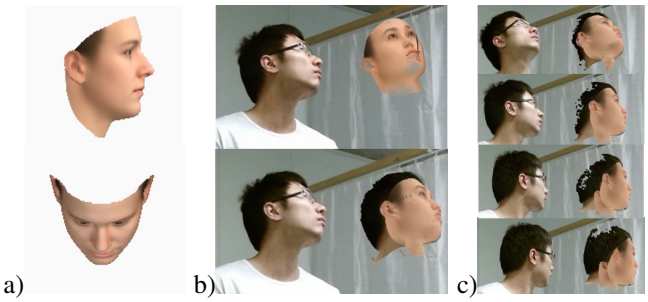


Fig. 1: Head model and pose estimation. (a) the 3DMM head representation only covers part of the head. (b) head pose estimation using only a 3DMM (top) and incorporating a reconstruction component (bottom). (c) online head reconstruction progressively incorporating observations.

In this paper, we propose a method for robust and accurate head pose estimation. The method relies on two main steps performed jointly: the online reconstruction of a full 3D head model, which is based on a variant of KinectFusion [12] and online fitting of a 3D morphable model by selecting multi-view observation samples and rendering smoothed synthetic samples. By combining the strengths of these two models in a single representation, we show that both accurate and robust performance can be obtained even under very extreme head poses as shown in Fig. 1. Our method achieves state-of-the-art results in the benchmark BIWI dataset [8]. In addition, the performance of our approach also exceeds the 3DMM face model in a more natural and challenging dataset where large head poses are relatively frequently.

II. RELATED WORKS

Many head pose tracking methods can be found in the literature, differing on how the face are modelled, and on the tracking approach using such representations.

Due to the difficulty to model the face appearance, some authors relied on keyframes, i.e. face image samples with associated head poses. The GAVAM model of Morency et al. [13] is a typical example. It uses differential tracking to compare to previous observations as well to the set of

keyframes, and constantly updates the current keyframe pose estimates, or adds new ones when needed.

To avoid defining an explicit face model representation, other methods have investigated pose-dependent statistical face representations [14] estimated through bayesian tracking, or regression methods. For instance, Fanelli et al. [8] achieved this by extracting weak features from depth data patches to train a random-forest regression model. However, their model did not generalize well and suffered from low accuracy. Also, regression methods in general lack semantics on facial features, which can be of importance for tasks such as eye-gaze estimation or facial expression recognition.

An alternative line of work focuses on facial features tracking, in which head pose estimation becomes a secondary problem solved through PnP techniques. Constrained local models (CLM) [15] represent the appearance of local features as linear subspaces. Their location is found from filter responses of patch experts, constrained by a shape model. Baltrusaitis et al. [16] extended CLM by including depth patches observations, performing better than CLMs or the GAVAM model. Later, Baltrusaitis et al. [17] proposed the Constrained Local Neural Fields (CLNF), used in the OpenFace software, a variant of CLM addressing feature detection under more challenging scenarios. However, feature based methods suffer from self occlusions, as they depend on features visibility.

Model based methods provide both semantic reasoning and may give better support against missing features. The 3D Morphable Model (3DMM), as an extension of the 2D case ASM (Appearance Shape Model) [18] and AAM (Active Appearance Model) [19], is a parametric linear representation of 3D shape and appearance. This 3DMM linear basis can be learned from real data, modelling variations related to identity [20][21] or even facial expressions [22].

Similar to AAMs, the 3DMM can be fit to image data [23][24][1][25] or depth or shape data to adapt the model to the subject [26][9]. Alternatively, for head pose estimation, Papazov et al. [7] presented a depth feature matching framework, in which view-invariant descriptors encoding the face 3D shape are used to infer head pose through matching.

Instead of feature matching, registration methods aim to minimize the discrepancy between the data and the parameterized model. In the work of Weise et al. [27], a user specific 3D mesh face model is built offline using non-rigid registration, whereas the iterative closest points (ICP) algorithm is used for real-time head tracking. Funes and Odobez [9][28] proposed to first fit a 3DMM to the user by extending the method of [26], and then using such model for tracking through ICP. However, ICP suffers from local minima, requiring sufficient frame rate during tracking.

To solve this problem, Meyer et al. [10] combined ICP and Particle Swarm Optimization (PSO) together to iteratively reinitialize the ICP with the result of PSO to achieve online fitting. Though high accuracy of head pose estimation is achieved, the computation cost is relatively high.

Finally, further works have been proposed to address the 3D non-rigid facial expressions, mainly for transfer to animated avatars. Methods like [2], [29], [30] model facial deformations through blendshapes which linearly extend a standard 3DMM. An advantage of these methods, as done by Bouaziz et al. [2] is that by decomposing the face model, it is possible to retrieve the components related to face identity even under facial deformations, as well as adapting the facial deformations basis online. On the other hand, the authors in [29] also achieved robust head tracking under occlusion. They identified outliers by measuring the difference between the current observation and the head model posed with the previous estimation. These papers however lack evaluations on head pose estimation.

Up to this point, all previous model based methods are strongly focused on the face region. Although this is justified, as the main interest is on this region, it is nevertheless insufficient to address the large range of head pose variations observed in many natural human interactions (cf. Fig. 6).

To address this problem, in this paper we propose an approach that fits a 3DMM online to the face region, whereas the subject specific head representation is augmented on-the-fly through a variant of KinectFusion [12]. The resulting method is capable of achieving high accuracy, to create a face model representation with associated semantics, and to maintain track under significantly challenging head poses.

III. METHOD

A. Overview

The proposed framework is illustrated in Fig. 2. It consists of three main modules: head pose estimation, 3DMM fitting and head reconstruction. The pose estimation module aligns at every time step i the current head model $\mathbf{h}^i$ with the observed depth map data $\mathbf{o}^i$ using a variant of the Iterated Closest Point (ICP) algorithm. The aim of the two other modules is to learn and update the head model $\mathbf{h}^i$ of the given person using the sequence of observations. This is achieved using two main representations: the first one, $\mathbf{r}^i$, is a 3D reconstruction of the head obtained through the temporal registration and integration over time of the incoming depth frames. Its main advantage is that it can represent the full head without any prior knowledge. The second one is a 3DMM face representation, $\mathbf{m}^i$, built and adapted online using a 3DMM fitting algorithm relying on automatically selected depth frames, complemented by synthesis data from the reconstruction model, whenever it becomes available. The resulting head model used for pose estimation if thus given by a set of vertices coming from the two representations, $\mathbf{h}^i = \{\mathbf{m}^i, \mathbf{r}^i\}$.

Although in principle after several frames, we could rely only on the reconstructed model, we keep the 3DMM face model $\mathbf{m}$ as it has several advantages. First, the semantic meaning of vertices from $\mathbf{m}$ is well known, which can be useful for face analysis or to combine the model with appearance information provided by facial landmark detectors. Secondly, besides personalization of the face model to specific individuals, the 3DMM face model can be extended to include further elements, e.g. deformations due to expressions.

Note that both face 3DMM and head reconstructions are built online without any manual intervention. Details of pose

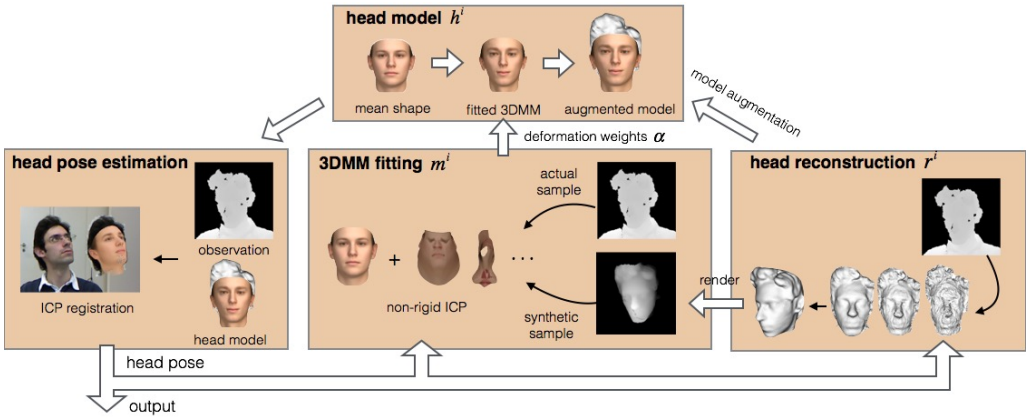


Fig. 2: Proposed framework. At time i, the head pose estimation module registers the current head model $\mathbf{h}^i$ to the observations. The 3DMM fitting module personalizes a 3DMM face model $\mathbf{m}$ to sample frames and their synthetic version generated from the head reconstruction. The reconstruction module aggregates pose rectified depth images into a full head representation $\mathbf{r}^i$. Vertex samples from the 3DMM face model $\mathbf{m}^i$ and from $\mathbf{r}^i$ are used to define the head model $\mathbf{h}^{i+1}$.

estimation and head representation learning are provided in the following sections.

B. 3D Head Pose Estimation

The goal is to estimate the head pose $\mathbf{p}^i = (\mathbf{R}^i, \mathbf{t}^i)$ at time i from the depth map $\mathbf{o}^i$, where $\mathbf{R}^i \in R^{3\times3}$ is a rotation matrix characterized by three rotation angles (yaw, pitch, roll) and $\mathbf{t}^i \in R^3$ is the translation. Classically, it is formulated as finding the alignment between head model $\mathbf{h}^i$ and the depth observations which minimizes a rigid registration cost. As this is intractable, the cost is minimized iteratively. At each iteration, given the current estimate of the pose, the indices $c^i(k)$ of the vertices in $\mathbf{o}^i$ corresponding to the the vertices k of our model are found using the method in [31], which is a fast implementation of normal shooting. Then the pose is refined by minimization of the point-to-plane ICP cost $E_1(\mathbf{R}^i, \mathbf{t}^i)$ given by [9]:

$$\sum_k w[k] \left((\mathbf{R}^i \mathbf{n}^i_{\mathbf{h}}[k])^{\mathrm{T}} (\mathbf{R}^i \mathbf{v}^i_{\mathbf{h}}[k] + \mathbf{t}^i - \mathbf{v}^i_{\mathbf{o}}[c^i(k)]) \right)^2 \quad (1)$$

where the set of vertices and their normal vectors to the 3D surface $\{(\mathbf{v}^i_{\mathbf{h}}[k], \mathbf{n}^i_{\mathbf{h}}[k]), k = 1 \dots N^{\mathbf{h}}_v\}$ represents our head model $\mathbf{h}^i$ at time i.

The robust weights $w[k]$ aim to discard bad correspondences. Assuming $\delta[k]$ is the euclidean distance between a transformed vertex and its correspondence, $w[k]$ is computed at each ICP iteration as follows: i) $w[k]$ is set to zero for correspondences whose normals differ for over $45°$; ii) $w[k]$ is zero if $\delta[k] > 4cm$; iii) $w[k]$ is 1 for $\delta[k] < 1cm$; iv) otherwise, $w[k]$ is inversely proportional to $\delta[k]$. We use the same weighting strategy for all ICP methods in this paper.

Initialization. For each frame i, the pose is initialized with the pose estimated from the previous frame. Upon failure, or at the very beginning, the Haar detector is applied to each new RGB image until a detection is found. The pose is then initialized with the identity matrix for the rotation, and, for the translation, with the 3D location of the detected face mapped in the 3D space using the depth map.

C. 3D Morphable Model (3DMM) Fitting

The 3D Morphable Model relies on a set of deformation bases $\mathbf{b}_l$ to model the face variations across different face identities. More precisely, a 3D frontal face $\mathbf{m}$ can be represented as a linear combination of the mean shape μ and the deformation bases $\mathbf{b}_l$ according to:

$$\mathbf{v}_{\mathbf{m}}(\alpha) = \mathbf{v}_\mu + \sum_{l=1}^{N_{\mathbf{b}}} \alpha_l \lambda_l \mathbf{v}_{\mathbf{b}_l} \quad (2)$$

where λ_l is the eigenvalue associated to the deformation base $\mathbf{b}_l$. We use the Basel Face Model (BFM) [20] as 3DMM.

Online Model Fitting. Humans have different face shapes. Since pose estimation is defined as a registration task aligning the head model to the observations, the 3DMM should be deformed to be as close to the observation as possible. To achieve this, we rely on a non-rigid multiple instance fitting method minimizing the discrepancy between our 3DMM model $\mathbf{m}(\boldsymbol{\alpha})$ and a set of frames $\mathcal{J}^i$ collected until time i. As with pose estimation, this discrepancy is minimized iteratively by minimizing at each step the non-rigid ICP cost (with $(\mathbf{R}, \mathbf{t}) = \{(\mathbf{R}^j, \mathbf{t}^j), j \in \mathcal{J}^i\}$):

$$E(\boldsymbol{\alpha}, \mathbf{R}, \mathbf{t}) = \sum_{j\in\mathcal{J}^i} \left(\sum_k w^j[k] \left((\mathbf{R}^j \mathbf{n}_{\mathbf{m}}(\alpha)[k])^{\mathrm{T}} (\mathbf{R}^j \mathbf{v}_{\mathbf{m}}(\alpha)[k] + \mathbf{t}^j - \mathbf{v}^j_{\mathbf{o}}[c^j(k)]) \right)^2 \right) + \gamma \sum_{l=1}^{N_{\mathbf{b}}} \alpha_l^2 \quad (3)$$

representing the sum of the rigid alignment errors with each frame of the sample set $\mathcal{J}^i$, and a regularization over the coefficients α. The parameter γ is the stiffness and controls how much $\mathbf{m}$ can be deformed [32]. The cost function is optimized with Gauss-Newton method [26].

Sample set online selection. A simple scheme is used to built $\mathcal{J}$ online. In essence, the goal is to collect observation samples whose estimated poses are close to 9 predefined poses [9] (see Fig. 3), to guarantees that the observation samples cover the whole 3D face. Whenever a new frame arrives, its pose is

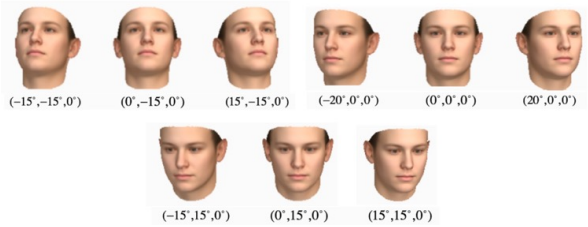


Fig. 3: Set of predefined poses (yaw,pitch,roll) used to collect data samples for online 3DMM fitting.

(a) Visual observation (b) Depth sample (c) Synthesized sample

Fig. 4: Illustration of actual observation and synthetic observations. Due to temporal integration, bad observations (dropping hair covering the face, missing depth measurement) are reduced or eliminated in the synthesized samples.

estimated using the current head/face model and the closest of the predefined poses is identified. If no frame had been yet added to this latter pose, the current frame is added to form $\mathcal{J}^i$, and the model fitting optimizing Eq. 3 is conducted with all samples in $\mathcal{J}^i$. Note that as the number of samples increases, the value of γ decreases to allow more flexibility for the fitting.

3DMM Fitting with Synthetic Observations. The above fitting approach relies only on a few observation samples so as to minimize the computational cost. A risk is that the fitting samples might be noisy and include temporary occlusions, large depth noise, or facial expressions (see Fig. 4(b)) which might not be adapted to the fitting which requires face only observations, and neutral faces due to the use of the BFM basis functions. When such samples are added to $\mathcal{J}$, they may adversely affect the 3DMM fitting and result in distorted face.

To mitigate the possible effect of such samples, we investigated whether the reconstruction would be useful, as it usually results in a more *robust and complete* representation of a *neutral* face as it relies on much more frames thanks to temporal integration. More specifically, each time we decide to add a fitting sample j to $\mathcal{J}$, we not only add the corresponding depth map $\mathbf{o}^j$, but render as well a synthetic depth map $\mathbf{s}^j$ from the current reconstructed head model (cf Next Section), and whose vertices are given by:

$$\mathbf{v}_\mathbf{s}^j = \left([\pi(\mathbf{R}^j\mathbf{v}_\mathbf{r} + \mathbf{t}^j)]_x, [\pi(\mathbf{R}^j\mathbf{v}_\mathbf{r} + \mathbf{t}^j)]_y, [\pi(\mathbf{R}^j\mathbf{v}_\mathbf{r} + \mathbf{t}^j)]_z\right)$$

that is, the vertices from the reconstruction models are rotated to the j^{th} pose and projected as a depth image (the notation $[\cdot]_{x,y,z}$ consist of using these depth map points to build the vertices).

Then, the fitting is straightforwardly adapted from Eq. 3 to optimize the alignment to both the real and the synthetic depth maps for all j (and sharing the rigid transform parameters between the real and synthetic maps). It is important to note the following points. First, we prefer to generate depth images rather than, for instance, fitting the 3DMM directly to the reconstruction surface, since the correspondence search along the normal vectors can be quite time consuming when working with 3D meshes. Secondly we prefer to keep the actual observations rather than relying just on the synthetic ones, since the latter ones can be smooth (depends on the voxel sampling, cf next section) and may not benefit from more crisper/detailed observations, for instance if the person would come closer to the sensor.

Another important motivation to use the synthetic samples is to keep the pose correspondence or alignment between the reconstructed data and the 3DMM. Indeed, due to the inaccurate representation of the 3DMM, at the beginning of the tracking, the reconstruction may happen with a small but fix bias with the 3DMM and thus with the actual head pose. By fitting or registering the 3DMM to the synthetic observations, a semantic correspondence between the 3DMM and reconstructed model can be maintained, which is important to our future work, for instance, eye gaze analysis.

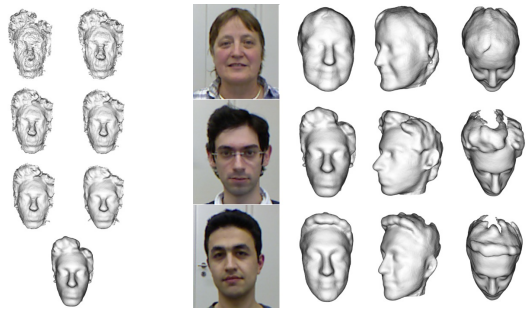
D. Head Reconstruction for Robust Head Pose Estimation

To handle head tracking from any pose, our goal is to augment the 3DMM with a head reconstruction built from the observed data. To achieve this, we rely on an adaptation of KinectFusion [12]. It is a classical method for 3D object reconstruction, targeting scenarios where a cameras moves in the 3D space or around a rigid 3D object. Our case is slightly different, as the sensor is static, and the head is moving; in addition, the face is not rigid, so one could wonder how well it can work in that case. Although DynamicFusion [33] has been proposed recently for non-rigid objects, it is more time consuming and has some limitations like lacking face and head semantic information, or being more sensitive to fast motions.

The principle is to use a 3D volume of the head represented by a set of vertices $\mathbf{v}_g$ regularly sampled in a $(depth = 28) \times (height = 28) \times (width = 19)$ volume (size in cm; we used 128 samples per dimension), and to accumulate from observations a function TSDF$[g]$ indicating whether the point is inside (negative value) or outside (positive value) the head.

The methods comprises 4 main steps. The first one consists in estimating the head pose We rely on the robust method described in Section III-B. Interestingly, this benefits from the availability of the 3DMM to obtain an accurate head pose, esp. at the beginning when only few frames have been observed. The second and third steps are volumetric mapping, which consists of rotating the vertex samples in the camera pose according to $\mathbf{v}_g^i = \mathbf{R}^i\mathbf{v}_g + \mathbf{t}^i$, and per-frame TSDF (truncated signed distance function) [34] computing for surface representation, defined by:

$$\text{tsdf}^i[g] = \max(-1, \min(1, \frac{[\mathbf{v}_g^i]_z - [\pi(\mathbf{v}_g^i)]_z}{\tau})) \qquad (4)$$



(a) online process (b) head reconstruction results

Fig. 5: 3D head reconstruction from the BIWI dataset.

in which $\pi(\mathbf{v}_g^i)$ denotes the 3D point associated to the pixel in the observed depth map $\mathbf{o}^i$ to which the mapped vertex g projects to (so, the point on the observed 3D surface), and $[\cdot]_z$ denotes the depth of a 3D point. In other words, tsdf records for vertex k the signed distance between its actual location $\mathbf{v}_g^i$ and the observed surface point. The parameter τ represents the thickness around the observed surface for which such distance is computed, and actually used (see equation 5 below).

Finally, in the fourth step, the tsdf values across frames are aggregated using a simple averaging strategy:

$$\mathrm{w}_{\mathrm{ts}}^i[g] = \begin{cases} 1 & \text{if } -1 < \mathrm{tsdf}^i[g] < 1 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

$$\mathrm{TSDF}^i[g] = \frac{\mathrm{w}_{\mathrm{TS}}^{i-1}[g]\mathrm{TSDF}^{i-1}[g] + \mathrm{w}_{\mathrm{ts}}^i[g]\mathrm{tsdf}^i[g]}{\mathrm{w}_{\mathrm{TS}}^{i-1}[g] + \mathrm{w}_{\mathrm{ts}}^i[g]} \quad (6)$$

$$\mathrm{w}_{\mathrm{TS}}^i[g] = \mathrm{w}_{\mathrm{TS}}^{i-1}[g] + \mathrm{w}_{\mathrm{ts}}^i[g] \quad (7)$$

Importantly, note that the fusion is only conducted on the voxels whose tsdf values are within the range $[-1, 1]$, i.e. the pixels near the observed surface. This is to avoid self-occlusion effects for concave parts, e.g. when seen from 45°, the visible nose surface hides other face surfaces which therefore are not 'inside' the head.

Reconstruction model. At each time step, a reconstruction model $\mathbf{r}^i$ is built from $\mathrm{w}_{\mathrm{TS}}^i$. In the original version of Kinect-Fusion, a ray-casting model was registered to the actual observation to ensure fast processing. Given the low resolution of our 3D volume (cf. Section IV), we used marching cubes [35] instead to derive a full 3D model, which is efficient enough. More concretely, the marching cubes method is applied to the set of voxels for which $\mathrm{w}_{\mathrm{TS}}^i$ is larger than 25 (i.e. they were at least 25 times within the observed surface region) to find the zero crossing surfaces and extract the vertices and their normals. Some reconstruction results at the end of BIWI sequences are shown in Fig. 5, and demonstrate that quite accurate models can be recovered.

E. Head model

As described in Section III-B, what we need for pose estimation is a set of vertices and normals, i.e. $\{(\mathbf{v}_{\mathbf{h}}^i[k], \mathbf{n}_{\mathbf{h}}^i[k]), k = 1 \dots N_v^{\mathbf{h}}\}$. To combine the 3DMM $\mathbf{m}$ and the reconstruction model, we simply randomly sample a fixed ratio of vertices from each of the model. That is, if $N_v^{\mathbf{m}}$ represents the number of vertices in $\mathbf{m}$, we sample $N_v^{\mathbf{r}} = \eta \times N_v^{\mathbf{m}}$ from $\mathbf{r}$, and hence we have $N_v^{\mathbf{h}} = N_v^{\mathbf{r}} + N_v^{\mathbf{m}}$.

Fig. 6: Dataset samples. a) BIWI. b) UBImpressed.

IV. Experimental protocol

In this section, we present the design of our experiments, including the datasets, the performance measures, the considered models and parameter settings.

A. Dataset

In our experiments, two datasets are used.

The BIWI Dataset is a public dataset collected by Fanelli [8]. It consists of 24 videos (15K frames in total) recorded with a Kinect 1 sensor, and where seated people keep moving their heads. Some samples are shown in Fig. 6a. The dataset provides the ground truth of head pose $(\mathbf{R}, \mathbf{t})$ for every frame.

The UBImpressed Dataset. This dataset has been captured to study the performance of students from the hospitality industry at their workplace [6]. The role play happens at a reception desk, where a student has to handle a difficult client. Students and clients are recorded using a Kinect 2 sensor (one per person). In this free and natural setting, large head poses and sudden head motions are frequently presented as people are observed from a relatively large distance, and people are seen from the side (see Fig. 6 for samples).

We used 10 video clips (9K frame in total) from the round 80 interactions. As head pose is not available, to identify tracking failure and evaluate accuracy, we annotated 6 landmarks on every frame, whenever they are visible: left and right corner of the left eye (l-l and r-l), left and right corner of the right eye (l-r and r-r), nose tip (n-t) and nasal root(n-r). These landmarks are rigid and seldom affected by facial expressions.

B. Performance Measures

Head pose estimation performance can be evaluated by two aspects, *accuracy and robustness*. Below we describe how we measure these two aspects on the two datasets.

BIWI Dataset. We report accuracy by the average error of the estimated rotation angles. Robustness reflects in general whether the error can be kept in an accepted range even when extreme head pose occurs. Therefore, we can evaluate it by the cumulative distribution function of errors (error CDF) reflecting the proportion of frames whose errors are below a given value. We further use this curve to report as in [10] the accuracy $\mathbf{ACC}_{10}$ as the percentage of frames with errors below 10 degrees. Finally, we also measure the robustness through the average pose error for different range of poses.

We take the maximum of the three ground-truth rotation angles (yaw/pitch/roll) as pose indicator for each frame and quantize the indicators in bins of size 10°.

UBImpressed Dataset. As pose is not available, we rely on the annotated landmarks. More specifically, we transform the 3DMM face model with the estimated pose and project the facial landmarks of the model to the image plane and compute as error the distances between these projections and the ground-truth. We then characterize the pose tracking performance with several measures. As our goal is to evaluate the tracking robustness, we first report the lost frame ratio **LFRatio** defined as the percentage of frames for which a given tracker does not report results (i.e. it has identified by itself a failure case)[1]. Then we report the average localization errors on the frame with reported results, and compute as well the CDF function of those errors.

C. Systems

We compared several models: the mean shape of the 3DMM and the online fitted 3DMM, which only uses the 3DMM model as face/head representation; the pure reconstruction model FHM and our proposed model 3DMM+FHM. Note that the pure FHM relies on the 3DMM in the first 25 frames to construct an initial model, and that we sample the same number of points that we use in the 3DMM. The 3DMM+FHM model selects different proportions of points from the reconstructed $\mathbf{r}$ and 3DMM model $\mathbf{m}$. We tested the ratios $\eta = 0.5, 1.0, 1.5$, where $\eta = \frac{N_v^{\mathbf{r}}}{N_v^{\mathbf{m}}}$ (see Sec. III-E). In addition, we compare our results to those of [10], which obtained the best performance on BIWI, and to the OpenFace system [36] which relies on both image and depth data and has been primarily optimized for landmark localization.

D. Parameter setting

For 3DMM fitting, we use $N_{\mathbf{b}} = 50$ deformation bases from the BFM model. For head reconstruction, the size of the 3D volume is $128 \times 128 \times 128$.

V. Results

A. BIWI dataset

The overall estimation accuracy of BIWI is listed in Tab. I. The error CDF is provided in Fig. 8a), while Fig. 8b) provides the average error for different pose ranges. From these results, the following conclusions can be drawn.

First, from Tab. I, our head model 3DMM+FHM achieves the best results and all our results exceed the performance of [10] which reported so far the best accuracy on BIWI. Since [10] relies on the combination of ICP and Particle Swarm Optimization (PSO), this shows that when using an augmented 3DMM model with head reconstruction, ICP alone can achieve equal or even better accuracy. In particular, the estimation of the pitch angle is much improved when comparing with [10].

[1]Note that in the BIWI case, when a failure is identified by the tracker, we set the estimation as being frontal so that all frames are into account for evaluation and a fair comparison with other works [10] can be made.

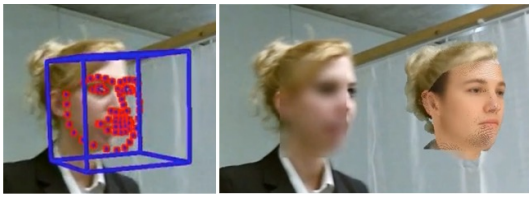


Fig. 7: OpenFace common failure case. Although the error distance with respect to the visible landmarks is small, the head pose is badly estimated. We show our result on the result. Note that OpenFace is using depth information as well.

Secondly, while the 3DMM alone performs in par with the PSO model, it performs worse than the FHM model and the proposed model for all values of η, showing that the use of head reconstruction improves the pose estimation accuracy. In particular, we can notice from Fig. 8b) that the errors from large pose are much reduced, which, when looking at result video, is due to a reduction in tracking failures. We also notice that on BIWI not much differences can be observed when varying η.

Compared to OpenFace, we note that the head pose error is much larger with OpenFace, which is understandable since it does not attempt at building a 3D face model. This shows the limitations of such approach for head pose estimation.

Finally, the performance of the Mean shape model is lower than the 3DMM model, illustrating that personalized 3DMM fitting is important for pose estimation.

B. UBImpressed dataset

Results are shown in Table II. They report the amount of frames for which no output is provided (Lost frames ratio **LFRatio**), the mean landmark location error, as well as the percentage of frames (FailRatio) for which this error is above 20 pixels and which can be considered as being in a tracking failure situation. The error CDF curve is shown in Fig. 8. The main comments are as follows.

First, as on BIWI, our proposed 3DMM+FHM models performs much better than the 3DMM alone. In particular, the model with $\eta = 1$ (same number of vertices from the 3DMM and the reconstruction) provides a good compromise between the face and head modeling component. Compared to 3DMM, it fails around 3 times less (FailRatio), and has a localization error decreased by 40%.

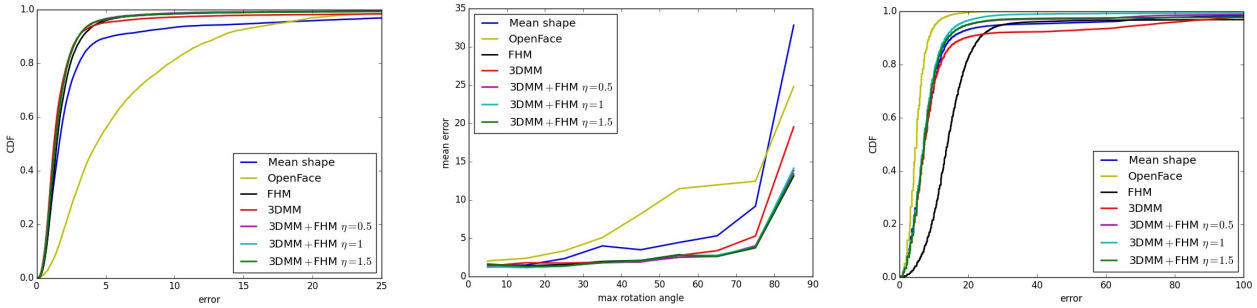
Secondly, according to the mean error the OpenFace system seems to perform better than our approach. However, this mainly shows that our indirect estimation of head pose estimation accuracy using landmark has limitations. Indeed, we can first notice that **LFRatio** is almost 13% for OpenFace, which is 3.6 times that of the (3DMM+FHM, $\eta = 1$) model. Thus OpenFace is reporting results on less frames than the other methods, frames which usually are problematic. Secondly, as already seen in BIWI, its pose estimation accuracy is limited, while its landmark localization accuracy is high as it was specifically trained for that. This contrast is illustrated in Fig. 7, and such situations are relatively frequent in the results.

TABLE I: BIWI: average error and accuracy.

System/Head model	yaw	pitch	roll	mean	$\mathbf{ACC}_{10}$
OpenFace	7.8	8.0	4.6	6.8±6.8	52.3%
PSO [10]	2.1	2.1	2.4	2.2	94.6%
Mean shape	5.0	2.5	3.9	3.8±8.6	89.5%
3DMM	2.9	1.8	2.6	2.4±5.7	95.3%
FHM	2.5	1.7	2.3	2.2±4.6	95.7%
3DMM+FHM $\eta = 0.5$	2.5	1.5	2.3	2.1±4.5	96.2%
3DMM+FHM $\eta = 1.0$	2.4	1.7	2.2	2.1±4.1	96.4%
3DMM+FHM $\eta = 1.5$	2.5	1.5	2.2	2.1±5.2	96.6%

TABLE II: UBImpressed: Landmark position errors

System/Head model	l-l	r-l	l-r	r-r	n-r	n-t	mean	FailRatio	**LFRatio**
OpenFace	4.6	5.4	5.2	6.2	5.6	5.7	5.4±4.0	0.1%	13.0%
Mean shape	10.5	11.7	12.7	12.1	11.8	12.9	11.8±22.2	6.9%	4.5%
3DMM	13.5	13.4	7.5	9.3	14.7	16.4	14.0±25.5	10.5%	4.2%
FHM	20.0	20.2	16.7	17.5	21.1	23.7	20.9±40.0	17.7%	2.9%
3DMM+FHM $\eta = 0.5$	8.9	9.7	7.9	10.0	10.2	11.5	10.0±21.6	5.2%	4.3%
3DMM+FHM $\eta = 1.0$	7.0	9.0	7.9	10.1	8.5	9.8	8.6±15.0	3.4%	3.6%
3DMM+FHM $\eta = 1.5$	10.7	12.0	7.9	10.1	11.9	13.2	11.7±26.2	5.2%	3.7%



(a) CDF of error for BIWI (b) Error distribution across head poses for BIWI (c) CDF of error for UBImpressed

Fig. 8: Robustness measurement and comparison

Note that surprisingly, the Mean shape model performs better than the 3DMM model. This is because the system takes some false samples for 3DMM fitting when processing the challenging UBImpressed videos and the distorted fitting results can reduce the tracking accuracy. In addition, the pure FHM model produces the lowest accuracy among all models. Indeed, different from the BIWI dataset, the UBImpressed videos do not start with a frontal face. This means that the reconstructed head model is incomplete after the first 25 frames (e.g. one side of the face is missing) so that when the missing part becomes suddenly visible, the tracking can not rely on the 3DMM model to achieve its registration. This result also explains the necessity to combine 3DMM and FHM.

We illustrate some head pose estimation results in Fig. 9. We note that our augmented model can also handle the situation of occlusion, shown in the last column of Fig. 9.

C. Use of synthetic data for 3DMM fitting.

We observed in practice that using the synthetic samples $\mathbf{s}^j$ in addition to the actual depth maps $\mathbf{o}^j$ for fitting the 3DMM model produced improved head reconstruction. To evaluate the potential impact of synthetic observations on accuracy, we did the following. We used as set of frames $\mathcal{J}$ for fitting those automatically selected by the FHM algorithm. We then trained the 3DMM *offline* from these frames, using either only the depth maps, or using in addition the synthetic frames, and then ran the tracker on the videos with the resulting models. Results are shown in Table III and IV. No differences can be noticed, suggesting that the fitting with synthetic samples has no impact on pose estimation, or that the selected frames did not contain difficult situations where the synthetic data could have been more useful. However, we keep this module since it is essential to our future work in computing pose correspondence.

D. Time cost.

We implement our system in Python/C++ based on CPU. Generally, the ICP based alignment takes ∼9ms and the 3DMM fitting costs ∼5s and is executed in a separate thread. The reconstruction module which also includes the 3D meshing takes ∼0.25s per frame. This module is applied at every frame within the first 300 frames and every 5 frames afterwards. The whole system can be much faster by implementing some modules (especially reconstruction) on GPU.

VI. Conclusions and Future Works

We presented a method for robust head pose estimation. The main idea is to augment a 3DMM face model with a set of 3D points extracted from a reconstruction of the person's face. To achieve this, we first do online 3D head reconstruction using a KinectFusion methodology, and the result is combined with the 3DMM face model. The experiment results show that our framework can achieve not only accurate but also robust performance even when extreme head poses are presented.

An important future work includes the exploitation of visual information, esp. to handle people moving away from the sensor. This can be achieved for instance by exploiting landmark detection, which, as shown by the OpenFace results, can provide good results if we are able to select frames where they are reliable. Also, as ICP can be trapped in local optima, a rough pose initializer is needed for ICP registration at the beginning or to deal with situations where large pose changes occur, e.g. fast motions.

Our work can also be expanded to other tasks. In addition to serving as a preprocessing step for facial expression analysis or eye gaze tracking, the 3DMM fitting included in our framework can also be extended to estimate facial expressions by directly incorporating expression blendshapes.

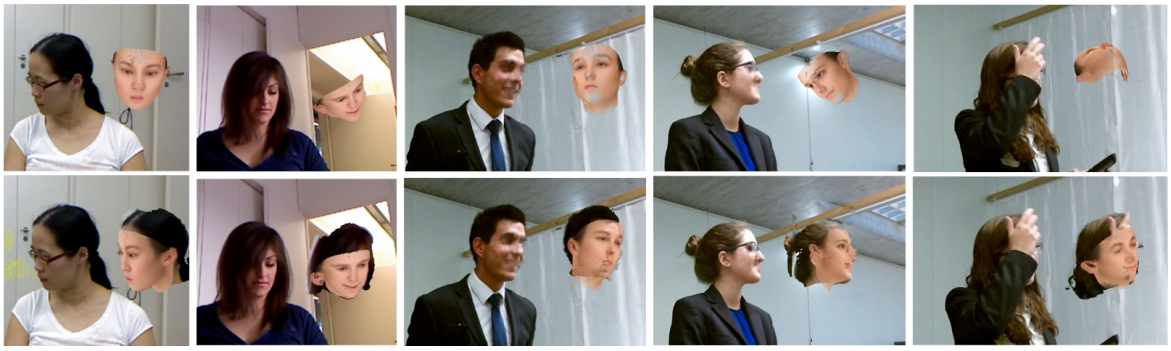


Fig. 9: Visualized results of head pose estimation with the 3DMM (top) and augmented model (bottom)

TABLE III: Error of rotation angles in BIWI

Head model	yaw	pitch	roll	mean
3DMM fitted to $\{\mathbf{o}^j, \mathbf{s}^j\}$	2.6	1.8	3.6	2.3
3DMM fitted to $\{\mathbf{o}^j\}$	2.8	1.8	2.6	2.4

TABLE IV: Error of landmark positions in UBImpressed

Head model	l-l	r-l	l-r	r-r	n-r	n-t	mean
3DMM fitted to $\{\mathbf{o}^j, \mathbf{s}^j\}$	10.4	11.2	9.2	10.4	11.5	13.0	11.4
3DMM fitted to $\{\mathbf{o}^j\}$	9.9	10.5	9.0	9.6	11.2	12.6	10.9

Acknowledgement. This work was partly funded by the UBIMPRESSED project of the Sinergia interdisciplinary program of the Swiss National Science Foundation (SNSF), and by the the European Unions Horizon 2020 research and innovation programme under grant agreement no. 688147 (MuMMER, mummer-project.eu).

REFERENCES

[1] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, "Face alignment across large poses: A 3d solution," in *CVPR*, June 2016.

[2] S. Bouaziz, Y. Wang, and M. Pauly, "Online modeling for realtime facial animation," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 40:1–40:10, 2013.

[3] Y. Chen, Y. Yu, and J.-M. Odobez, "Head nod detection from a full 3d model," in *Proceedings of ICCV Workshops*, Dec. 2015, pp. 528–536.

[4] S. Ba and J. Odobez, "A study on visual focus of attention recognition from head pose in a meeting room," in *Proc. Workshop on Machine Learning for Multimodal Interaction (MLMI)*, Washington DC, 2006.

[5] C. Oertel, J. Lopes, Y. Yu, K. Funes, J. Gustafson, A. Black, and J.-M. Odobez, "Towards building an attentive artificial listener: On the perception of attentiveness in audio-visual feedback tokens," in *ICMI 2016*.

[6] S. Muralidhar, L. S. Nguyen, D. Frauendorfer, J.-M. Odobez, M. Schmid Mast, and D. Gatica-Perez, "Training on the job: Behavioral analysis of job interviews in hospitality," in *ICMI*, Tokyo, Japan, 2016.

[7] C. Papazov, T. K. Marks, and M. Jones, "Real-time 3d head pose and facial landmark estimation from depth images using triangular surface patch features," in *CVPR*, June 2015.

[8] G. Fanelli, J. Gall, and L. V. Gool, "Real time head pose estimation with random regression forests," in *CVPR*, June 2011, pp. 617–624.

[9] K. A. Funes Mora and J.-M. Odobez, "Gaze estimation from multimodal kinect data," in *CVPR Workshop on Gesture Recognition*, Jun. 2012.

[10] G. P. Meyer, S. Gupta, I. Frosio, D. Reddy, and J. Kautz, "Robust model-based 3d head pose estimation," in *ICCV*, December 2015.

[11] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *SIGGRAPH*, New York, NY, USA, 1999, pp. 187–194.

[12] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon, "Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera," in *24th UIST*, 2011, pp. 559–568.

[13] L.-P. Morency, J. Whitehill, and J. Movellan, "Generalized Adaptive View-based Appearance Model: Integrated Framework for Monocular Head Pose Estimation," in *FG*, 2008.

[14] S. Ba and J.-M. Odobez, "A rao-blackwellized mixed state particle filter for head pose tracking," in *ACM-ICMI Workshop on Multi-modal Multi-party Meeting Processing (MMMP)*, 2005.

[15] D.Cristinacce and T.F.Cootes, "Automatic Feature Localisation with Constrained Local Models," *Pattern Recognition*, 2007.

[16] T. Baltrusaitis, P. Robinson, and L.-P. Morency, "3D Constrained Local Model for Rigid and Non-Rigid Facial Tracking," in *CVPR*, 2012.

[17] T. Baltruaitis, P. Robinson, and L.-P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *ICCV workshop*, Sydney, Australia, 2013, pp. 354–361.

[18] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models: Their training and application," *CVIU*, vol. 61, no. 1, Jan. 1995.

[19] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *PAMI.*, vol. 23, no. 6, pp. 681–685, Jun. 2001.

[20] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," in *AVSS for Security, Safety and Monitoring in Smart Environments*, 2009.

[21] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway, "A 3d morphable model learnt from 10,000 faces," in *CVPR*, June 2016.

[22] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3d facial expression database for visual computing," *IEEE Trans on Visualization and Computer Graphics*, vol. 20, no. 3, pp. 413–425, 2014.

[23] T. Vetter and V. Blanz, "Estimating Coloured 3D Face Models from Single Images: An Example Based Approach," in *ECCV*, 1998.

[24] B. S. Göktürk, J. Y. Bouguet, and R. Grzeszczuk, "A Data-Driven Model for Monocular Face Tracking." in *ICCV*, vol. 2, 2001, pp. 701–708.

[25] A. Jourabloo and X. Liu, "Large-pose face alignment via cnn-based dense 3d model fitting," in *CVPR*, June 2016.

[26] B. Amberg, R. Knothe, and T. Vetter, "Expression invariant 3D face recognition with a Morphable Model," in *FG*, Sep. 2008, pp. 1–6.

[27] T. Weise, S. Bouaziz, H. Li, and M. Pauly, "Realtime performance-based facial animation," *SIGGRAPH 2011*, vol. 30, no. 4, p. 1, Jul. 2011.

[28] K. A. Funes Mora and J.-M. Odobez, "Geometric generative gaze estimation (g3e) for remote rgb-d cameras," in *CVPR*, 2014.

[29] P.-L. Hsieh, C. Ma, J. Yu, and H. Li, "Unconstrained realtime facial performance capture," in *CVPR*, 2015.

[30] D. Thomas and R. Taniguchi, "Augmented blendshapes for real-time simultaneous 3d head modeling and facial motion capture," in *CVPR2016*.

[31] S.-Y. Park and M. Subbarao, "An accurate and fast point-to-plane registration technique," *Pattern Recogn. Lett.*, vol. 24, no. 16, 2003.

[32] J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt, "Real-time expression transfer for facial reenactment," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 6, 2015.

[33] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," in *CVPR*, 2015.

[34] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *SIGGRAPH*, 1996.

[35] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *SIGGRAPH*, 1987.

[36] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "Openface: an open source facial behavior analysis toolkit," in *WACV*, 2016.