HeadFusion: 360° Head Pose Tracking Combining 3D Morphable Model and 3D Reconstruction

Yu Yu, Kenneth Alberto Funes Mora, and Jean-Marc Odobez, *Member, IEEE*

Abstract—Head pose estimation is a fundamental task for face and social related research. Although 3D morphable model (3DMM) based methods relying on depth information usually achieve accurate results, they usually require frontal or mid-profile poses which preclude a large set of applications where such conditions can not be garanteed, like monitoring natural interactions from fixed sensors placed in the environment. A major reason is that 3DMM models usually only cover the face region. In this paper, we present a framework which combines the strengths of a 3DMM model fitted online with a prior-free reconstruction of a 3D full head model providing support for pose estimation from any viewpoint. In addition, we also proposes a symmetry regularizer for accurate 3DMM fitting under partial observations, and exploit visual tracking to address natural head dynamics with fast accelerations. Extensive experiments show that our method achieves state-of-the-art performance on the public BIWI dataset, as well as accurate and robust results on UbiPose, an annotated dataset of natural interactions that we make public and where adverse poses, occlusions or fast motions regularly occur.

Index Terms—Head pose, 3D head reconstruction, 3D morphable model

1 Introduction

Head pose plays an important role in face analysis. On one hand, it is strongly related to the positions of important facial features, and thus its estimation is often used as a pre-processing step for tasks like face alignment [1], expression analysis [2], identity recognition [3] or gaze estimation [4]. On the other hand, the head motion dynamics, which can be used to convey meaningful signals in daily interaction [5], is composed of a series of pose changes. Head pose estimation is thus also useful in fields like social interaction analysis [6] and human robot interaction.

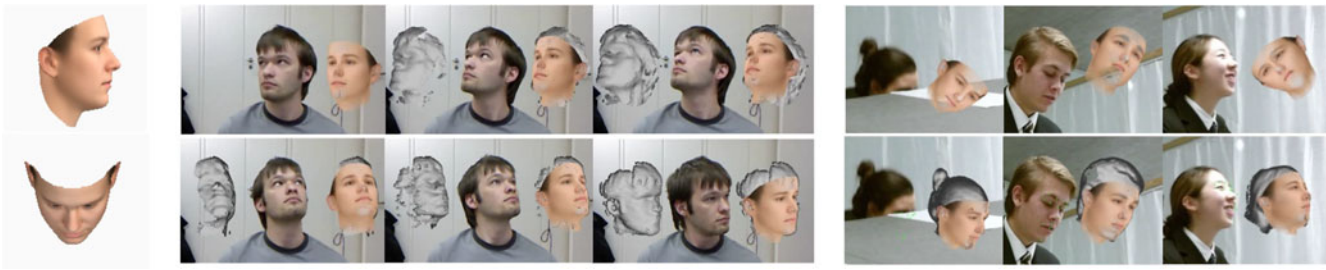
Although there have been important advances in recent years, traditional visual based head pose estimation suffers from difficulties such as human shape and appearance variability, extreme head poses, facial expressions, the non-rigid nature of the face, and illumination variations. The development of consumer 3D RGB-D sensors offers an alternative solution. Instead of only providing 2D observations in which fundamental information is lost after projection, the 3D sensor measures the depth information that is inherently required for 3D head pose estimation.

A number of depth-based approaches have been proposed for head pose estimation, such as feature matching [7], nonlinear regression [8] or model based methods [9], [10]. Continuous estimations are achieved by all these methods. However, the former two approaches do not report as precise results as the model based ones [10]. The model based methods rely on a predefined face model and retrieve the head pose parameter by minimizing the discrepancy between the observation and the head model. They often use 3D Morphable Models (3DMM) [11] to retrieve the subject's face model, since they provide a linear and low dimensional representation of the 3D facial shape variations across a population allowing online and well constrained model adaptation by finding the coefficients for the subject of interest. Furthermore, the 3DMM model also provides semantic information which is fundamental for other tasks such as gaze estimation. The 3DMM models are usually learned from the 3D scans of a group people. However, as illustrated in Fig. 1a, the limitation of most 3DMM models is that they only cover the frontal region of the face. The top, side and back parts of the head are missing since it is actually quite difficult to extract a linear statistical basis from the large variations of the hair (and even the ears) in these parts across the population.

Most applications so far consider applications where the subject's face is nearly frontal. However, there are many applications where such an assumption can not be garanteed, like when setting sensors in the environment and monitoring people's activities and interactions. Being able to track the head uninterruptedly in non-constrained natural scenarios (such as our UBImpressed sequences, shown in Fig. 6c), where unexpected cases such as fast motions, occlusions, and more profile or adverse poses are presented, is thus also very

• *Y. Yu, K. A. Funes Mora, and J.-M. Odobez are with the Perception and Activity Understanding Group, Idiap Research Institute, Martigny 1920, Switzerland and EPFL, Lausanne, Switzerland.*
E-mail: rainyucool@gmail.com, {kenneth.funes, odobez}@idiap.ch.

Manuscript received 1 Mar. 2017; revised 10 Dec. 2017; accepted 1 Feb. 2018. Date of publication 28 May 2018; date of current version 10 Oct. 2018. (Corresponding author: Yu Yu.)
Recommended for acceptance by S. Escalera, X. Baró, I. Guyon, H. J. Escalante, G. Tzimiropoulos, M. Valstar, M. Pantic, J. Cohn, and T. Kanade.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2018.2841403



(a) BFM model (b) online reconstruction (c) estimation results

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

important. However, a model only relying on the frontal face representation lacks the support to handle these cases, as shown in the top line of Fig. 1c.

In this paper, we thus propose a novel robust and accurate head pose estimation method which fuses the strenghs of two head representations:

- a 3DMM facial model automatically adapted online from a collection of samples, able to provide very accurate head pose estimations for near frontal head poses, but which has difficulties at tracking heads otherwise;
- an online reconstruction 3D head model based on a variant of KinectFusion [12], bringing the robustness of tracking of the head over a 360 degree range.

Furthermore, we propose to exploit a symmetric regularizer for the non-linear fitting of the 3DMM, preventing unwanted deformations that can degrade performance when mainly observing the face from a singe viewpoint away from the frontal pose. Combined with visual motion tracking cues based on KLT to enforce a temporal coherence and handle fast and natural head dynamics, we show that both accurate and robust head pose estimation can be achieved in natural and challenging scenarios as shown in Fig. 1c. In summary, the main contributions of our paper includes:

- A 3D head representation combining the semantic and the precision of a 3DMM fitting and tracking under restricted poses with the robustness of a full head representation reconstructed from depth data. This includes the estimation and the maintenance of a fine pose correspondence between the 3DMM and 3D reconstruction.
- A symmetry regularizer for robust online 3DMM adaptation;
- A framework exploiting both visual motion tracking and 3D model semantics for frame-to-frame pose initialization;
- UbiPose, a dataset composed of 22 videos from the UBImpressed dataset [13] featuring natural role played interactions, with more than 10k frames annotated with head pose ground-truth;
- Extensive experiments, with performance beyond the state-of-the-art on both the BIWI benchmark dataset and UbiPose.

The rest of the paper is organized as follows. After having presented the related work, Section 3 present our approach. Our experimental setting is described in Section 4, followed by our result analysis and discussion in Section 5 and a conclusion in Section 6.

2 Related Works

A large number of works have been proposed to address head pose estimation. They mainly differ on how the face is represented, the tracking approach, and the method used for pose estimation itself. In the following, we present a review of methods which are closely related to our work, either from the sensors used (e.g., depth sensor) or from a modeling perspective, and contrast them with our work.

2.1 General Methods

Due to the difficulty to model face appearance, early works relied on keyframes, i.e., face image samples with associated head poses. The GAVAM model of Morency et al. [14] 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, and adds new ones when needed.

Regression methods also avoid defining an explicit face model representation. Using depth data, Fanelli et al. [8] achieved this by extracting weak features from depth patches to train a random-forest regression model. However, their model did not achieve good generalization and suffers 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.

Facial features tracking is an alternative line of work. Head pose estimation then 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 the CLM framework 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.

Deep learning is gaining increased traction for tasks related to face analysis, such as detection [18], verification [3], and even gaze estimation from the full face [19]. Deep learning has also been used for the localization of facial

features [20], [21]. For instance, Sun et al. [20] proposed a cascaded model composed of three levels, each of them having a set of parallel CNNs for which subgroups predict the location of the same landmark(s) and their response is averaged to reduce the variance. This process is repeated at the 3 levels, successfully achieving a coarse-to-fine prediction of the landmarks. Nevertheless, the expectation of these methods is that facial landmarks are visible, thus restricting the method to head poses with visible face.

2.2 Model-Based Methods

Model based methods provide both semantic reasoning and may give better support against missing features. The 3D Morphable Model, as an extension of the 2D case ASM (Appearance Shape Model) [22] and AAM (Active Appearance Model) [23], is a parametric linear representation of 3D shape and appearance. The 3DMM linear basis can be learned from real data, modeling variations related to identity [24], [25] or even facial expressions [26].

As with AAMs, the 3DMM can be fit to image data [1], [27], [28], [29] or depth or shape data to adapt the model to the subject [30], [31]. One approach proposed is to conduct feature matching in the depth space, as proposed for instance by Papazov et al. [7] which relies on view-invariant descriptors encoding the face 3D shape for matching and pose inference.

Nevertheless, instead of feature matching, most approaches rely on registration for model fitting, where the aim is to minimize the discrepancy between the data and the parameterized model. For instance, Weise et al. [9] build a user specific 3D mesh face model offline using non-rigid registration, and then use the iterative closest points (ICP) algorithm for real-time head tracking. Funes and Odobez [31] extends [30] by applying a multi-instance fitting to build the offline model and also use ICP for tracking. However, as ICP is a local optimization technique, it requires a good initialization, and thus often needs to process data at a high frame rate. To solve this problem, Meyer et al. [10] combined ICP and Particle Swarm Optimization (PSO) together for joint tracking and online fitting, thus allowing to propose and evaluate multiple initializations. Higher pose estimation accuracy is achieved at the expense of a much higher computational cost.

Finally, further works have been proposed to address the 3D non-rigid facial expressions, mainly for transfer to animated avatars. Methods like [2], [32], [33] 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 deformation basis online. On the other hand, the authors in [32] 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. However, due to their focus, these papers lack a real evaluations on head pose estimation and on tracking robustness in non near-frontal pose conditions.

2.3 3D Reconstruction Approach

Reconstruction methods aiming at building 3D models of objects have attracted more attentions since the emergence of consumer 3D sensors. KinectFusion [12] is a standard approach which creates representations of static and rigid object or scene using a moving camera. Roughly speaking, it works by estimating the camera pose in each frame through the registration of successive scene observations, and projects the multi-angle viewed observations into a unified model representation for averaging. This approach has recently been extended via DynamicFusion [34] to also handle non-rigid objects through the estimation of a dense non-rigid warp field. Both KinectFusion and DynamicFusion rely on a volumetric representation which can be large and time consuming to process when aiming for precise reconstruction. To alleviate this problem, Keller et al. [35] proposed a lighter point based reconstruction and fusion method, removing in the same way the static scene requirement through the robust detection of dynamic objects.

2.4 Our Approach

As we have seen, most 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 situations of interest (cf. Fig. 6).

On the other hand, online reconstruction methods can potentially handle a large variety of poses, but are usually much more time consuming and have limitations. In particular they lack face and head semantic information and are more sensitive to fast motions. In addition, as faces and heads are not rigid, one could wonder how well such methods can work when being applied to natural interaction data with talking or facial expressions, or if the head shape is actually sufficient to obtain a precise registration when the face is almost not visible.

To address the above issues, in this paper we propose a model combining the strengths of both approaches, through the online fitting of a 3DMM to the face region whereas the subject specific head representation is augmented on-the-fly through a variant of KinectFusion [12]. Thanks to additional carefull model fitting (Laplacian fitting with symmetric regularizer) and KLT tracking cues, the resulting method is capable of achieving high accuracy, to create a face model representation with associated semantics, and to maintain track under very challenging dynamic head pose sequences in real settings.

3 METHOD

3.1 Overview

The proposed framework is illustrated in Fig. 2. It consists of three main modules: head pose estimation, 3DMM fitting and head reconstruction. The head pose estimation module aligns, at every time step i, the current estimate of the head model $\mathbf{h}^i$ with the observed RGBD data $(I^i, \mathbf{o}^i)$ (in which I denotes the RGB image and $\mathbf{o}$ denotes the depth map) using the ICP algorithm. This module also exploits two other submodules for initialization: one relying on face detection and landmark localization to initialize tracking; a second one relying on visual KLT tracking for coarse pose temporal alignment from the previous frame, allowing to handle the fast head acceleration motions regularly observed in natural sequences.

The aim of the 3DMM fitting and head reconstruction modules is to learn and update the head model $\mathbf{h}^i$ of the

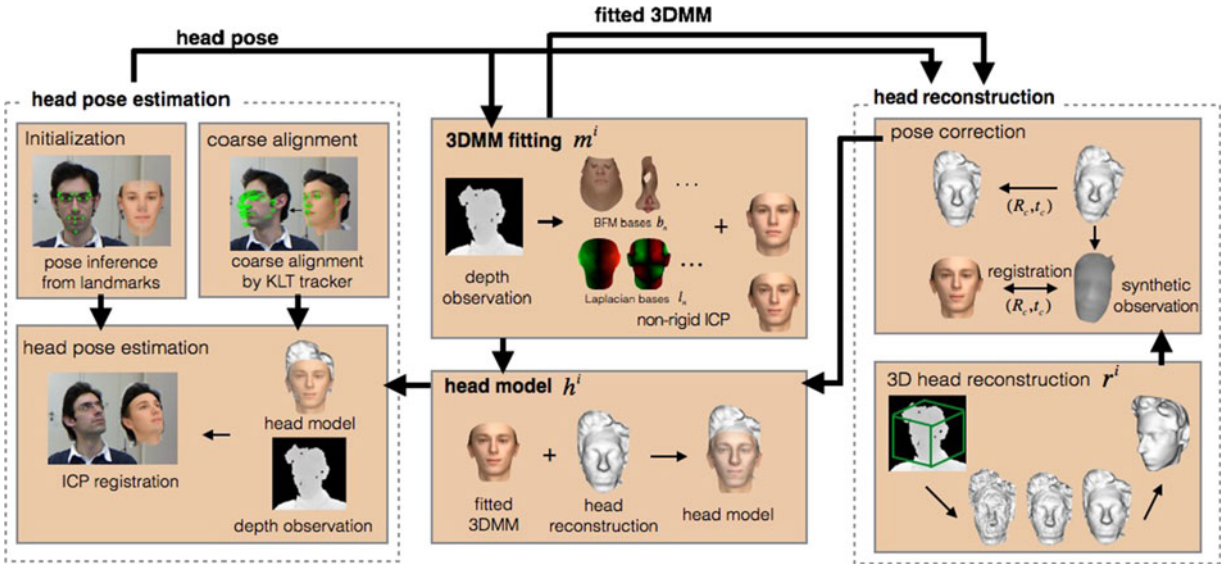


Fig. 2. Proposed framework. The head pose estimation module registers the current head model $\mathbf{h}$ to the observations. The 3DMM fitting module personalizes a 3DMM face model $\mathbf{m}$ to sample frames online. The reconstruction module aggregates pose rectified depth images into a full head representation $\mathbf{r}$. Vertex samples from the 3DMM face models $\mathbf{m}$ and from the reconstructed one $\mathbf{r}$ are used to define the head model $\mathbf{h}$.

given person using the sequence of past 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 3D mesh face representation, $\mathbf{m}^i$, built and adapted online from a multi-instance 3DMM fitting algorithm relying on automatically selected depth frames.

Note that, in the following sections, we will refer to a representation "$\mathbf{h}$" as a set of vertices $\mathbf{v_h} := \{\mathbf{v_h}[k]\}_{k=1}^{N_v^{\mathbf{h}}}$ and normals $\mathbf{n_h} := \{\mathbf{n_h}[k]\}_{k=1}^{N_v^{\mathbf{h}}}$, such that $\mathbf{h} := \{\mathbf{v_h}, \mathbf{n_h}\}$. We will use this notation to refer to the different face representations $\mathbf{h}$, $\mathbf{m}$ and $\mathbf{r}$, while using the "$[k]$" to index specific vertices or normals. In this view, the resulting head model, used for pose estimation, is thus given by the joint set of vertices and normals coming from the two representations, i.e., $\mathbf{h}^i = \{\mathbf{m}^i, \mathbf{r}^i\}$.

While in principle after several frames we could rely only on the reconstructed model $\mathbf{r}$ for head pose estimation, we keep the 3DMM-based face model $\mathbf{m}$ as part of the head representation 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 if we want to further combine the model with appearance information provided by facial landmark detectors. Second, besides personalization of the face model to specific individuals, the 3DMM-based face model can be extended to include further elements, e.g., deformations due to expressions, which could be useful for further facial analysis. Third, the existence of the 3DMM can prevent possible tracking failures caused by the sudden emergence of face regions which had not been seen so far and are thus not yet reconstructed, thus regularizing the resulting model.

Note that both the 3DMM-based face model and the head reconstructions are built online without any manual intervention. Details of pose estimation and head representation learning are provided in the following sections.

3.2 3D Head Pose Estimation

The objective of this module is to estimate the 3D head pose $\mathbf{p}^i = (\mathbf{R}^i, \mathbf{t}^i)$ at time i from the RGBD map $(I^i, \mathbf{o}^i)$. Here $\mathbf{p}^i$ represents a rigid transformation relating the head coordinate system (in which $\mathbf{h}$ is defined) to the world coordinate system, parametrized by a rotation matrix $\mathbf{R}^i \in R^{3\times3}$ (also characterized by three rotation angles yaw, pitch and roll), and a translation matrix $\mathbf{t}^i \in R^{3\times1}$.

The head pose estimation problem is formulated as finding the transform $\mathbf{p}^i$ of the head model $\mathbf{h}^i$ which minimizes the surface alignment error to the depth observations $\mathbf{o}^i$. However, this is intractable, as it requires to estimate jointly the pose and the point-wise semantic alignment between the surfaces. Thus, the cost is usually minimized using some form of ICP algorithm.

In the following, we first present in Section 3.2.1 the approach for ICP-based head pose estimation. Since being trapped in local minima is a typical weakness of ICP, in Section 3.2.2 we describe our method to initialize ICP close to the target pose either in the first frame (tracking initialization) or from frame-to-frame (during tracking).

3.2.1 Pose Estimation

Pose estimation is achieved using a variant of ICP, i.e., minimizing the registration error by iterating between the correspondence search and the rigid pose estimation steps.

More precisely, at each iteration, we first find the vertex correspondences of the head model, rigidly transformed by the current pose estimate, to the data $\mathbf{o}^i$ using the method in [36], which is a fast implementation of normal shooting. We will denote $c^i(k)$ the index of the vertex in $\mathbf{o}^i$ found to be the correspondence of the vertex k in $\mathbf{h}$. Then the pose estimate is improved by minimizing the point-to-plane ICP cost $E_1(\mathbf{R}^i, \mathbf{t}^i)$ given by:

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

where the time index i indicate that the set of vertices $\mathbf{v}_{\mathbf{h}}^{i}$ and normals $\mathbf{n}_{\mathbf{h}}^{i}$ refer to the head model $\mathbf{h}$ at time i.

The robust weights $\{w[k]\}_k$ 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 by more than $45°$, or if $\delta[k] > 4$ cm; ii) $w[k]$ is 1 for $\delta[k] < 1$ cm; iv) otherwise, $w[k] = \frac{r_1}{(\delta[k]-r_2)^2}$, where r_1 and r_2 are two parameters controlling the weight decay. We use the same weighting strategy for all ICP methods in this paper.

3.2.2 Pose Initialization

Initializing the ICP algorithm is needed in two distinct situations: to start the tracking of a newly detected head (or restart it upon a detected tracking failure); and during tracking, given the output result from the previous frame. Below we describe these two cases.

Tracking Initialization. In most applications, the tracking may have to start with any pose from the head. To do so, we initialize the system by inferring the head pose from facial landmark detections. The toolkit Dlib [37] is used to detect the face and facial landmarks from the RGB image I^i using the method of Kazemi and Sullivan [38]. These landmarks are then back-projected into the 3D space using the depth map $\mathbf{o}^i$ and used to form one-to-one point pairs with the corresponding 3D landmarks of $\mathbf{m}$, whose indices are known from the 3DMM semantics. Then the rigid rotation and translation of the head are inferred from these 3D point pairs using the method in [39].

Temporal Coarse Alignment. A common strategy in tracking is to use the pose estimated in frame $i-1$ as prediction and then initialization for frame i, or to use a more complex state-based dynamic model. These strategies have nevertheless difficulties in case of sudden acceleration (lagging behind) or deceleration (over shooting). A better strategy, as demonstrated in other tracking framework (e.g., GAVAM [14]) is to exploit visual motion for prediction. More precisely, a coarse alignment between frame $i-1$ and i is conducted based on facial feature tracking. Concretely, 3D facial features $\{\mathbf{f}^{i-1}[l]\}_l$ (where l denotes the feature index) of the head model set with the estimated pose of frame $i-1$ are projected into the 2D image plane I^{i-1}. Their corresponding positions in the next frame are estimated using a robust variant of the KLT tracker and projected back into the 3D space, resulting in the 3D feature locations $\{\tilde{\mathbf{f}}^{i}[l]\}_l$ predictions. The relative pose transformation $(\tilde{\mathbf{R}}^{i-1,i}, \tilde{\mathbf{t}}^{i-1,i})$ between frame $i-1$ and frame i is then estimated from the set of paired features $\{(\mathbf{f}^{i-1}[l], \tilde{\mathbf{f}}^{i}[l])\}_l$ by minimizing the following cost function:

$$\sum_{l} w[l]\Big(\big(\tilde{\mathbf{R}}^{i-1,i}\cdot\mathbf{n}_{\mathbf{f}}^{i}[l]\big)^{\mathrm{T}}\Big(\tilde{\mathbf{R}}^{i-1,i}\mathbf{f}^{i-1}[l]+\tilde{\mathbf{t}}^{i-1,i}-\tilde{\mathbf{f}}^{i}[l]\Big)\Big)^2 + \gamma||\tilde{\mathbf{R}}^{i-1,i}-\mathbf{R}_I||^2, \tag{2}$$

where $\mathbf{R}_I$ denotes the identity matrix and $\mathbf{n}_{\mathbf{f}}^{i}[l]$ is the normal vector at the lth feature on the head model. The weight $w[l]$ is derived from the tracking confidence of the lth feature given by the robust KLT tracker. A regularizer for the rotation matrix is incorporated to favor the identity rotation matrix estimation and comparatively encourage large translations in the solution, if required, since large and fast head motions are often due to head translation. Finally, given the relative pose transformation $(\tilde{\mathbf{R}}^{i-1,i}, \tilde{\mathbf{t}}^{i-1,i})$, a coarse estimation of the head pose at frame i is given by:

$$\mathbf{R}^{i} = \tilde{\mathbf{R}}^{i-1,i}\cdot\mathbf{R}^{i-1}, \mathbf{t}^{i} = \tilde{\mathbf{R}}^{i-1,i}\cdot\mathbf{t}^{i-1}+\tilde{\mathbf{t}}^{i-1,i}. \tag{3}$$

In our implementation, we chose 70 facial landmarks and 80 random points on the head model for coarse alignment. We expect this to achieve a good balance between covering a wider face area (through random points) and points which normally result in high confidence motion estimates (facial landmarks).

3.2.3 Tracking Failure Identification

In some situations the ICP optimization may diverge. If detected, we denote it as a tracking failure and apply the Dlib library face detector to incoming frames until a face is detected and the tracking is reinitialized. In this paper, a tracking failure is identified using the weights $w[k]$. Concretely, we first select the visible points k_o from the aligned model. Then their weights are summed up as $\sum_{k_o} w[k_o]$ which indicates whether the registered model achieves an overall good correspondence with the observation. More precisely, if

$$\sum_{k_o} w[k_o] < 0.01\cdot\sum_{k_o} 1, \tag{4}$$

is verified, then we assume that the registered model does not align with the observations, and a tracking failure is detected.

3.3 3D Morphable Model (3DMM) Fitting

In this section we explain our approach to retrieve $\mathbf{m}$ from a 3DMM. We will describe 3DMMs in detail, the fitting algorithm, the regularization, and the online sample selection strategy.

3.3.1 3D Morphable Model (3DMM)

A 3DMM is a statistical linear representation of facial shape (and/or appearance) variations [27]. Concretely, it is a linear combination of a mean shape μ with a deformation basis $\mathbf{b}_n$ weighted by a set of coefficients α. Vertex-wise, this can be represented as follows:

$$\mathbf{v}_{\mathbf{m}}(\alpha) = \mathbf{v}_{\mu} + \sum_{n=1}^{N_{\mathbf{b}}} \lambda_{\mathbf{b}_n}\alpha_n\mathbf{v}_{\mathbf{b}_n}, \tag{5}$$

where we omit the index notation "[k]" to avoid clutter and $\lambda_{\mathbf{b}_n}$ is the eigenvalue associated to the deformation vector $\mathbf{b}_n$. Here, $\mathbf{b}_n$ models facial shape variations across different face identities, while α allows to encode a person-specific facial shape $\mathbf{m}$.

In this work we used the Basel Face Model (BFM) [24] as 3DMM. The deformation basis were learned from the 3D face scans of only 200 people, providing mainly global face variations. To obtain a finer modeling of a specific person's face, we rely on the work of [2] which used the eigenvectors of the Laplacian matrix of the 3DMM graph as additional

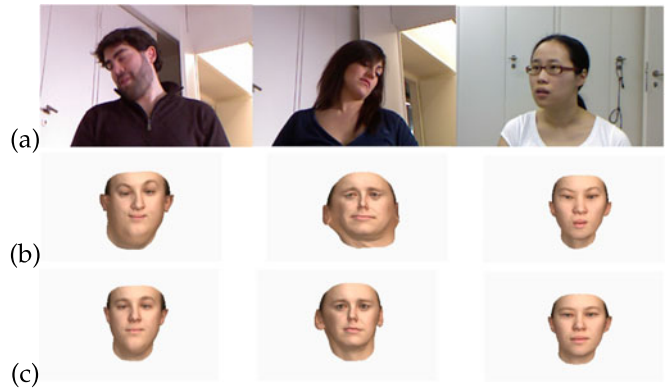


Fig. 3. Use of symmetry regularizer.(a) Fitting samples; (b) Fitting without symmetry regularizer; (c) Fitting with symmetry regularizer.

deformation bases. These Laplacian eigenvectors $\mathbf{l}_n$ corresponds to the smallest K eigenvalues, as shown in the 3DMM fitting module of Fig. 2 where the red and green region denotes positive deformation values and negative deformation values respectively, and the brightness of the region is proportional to the absolute deformation values. By adding the Laplacian eigenvectors, our final 3D face model is given, per each vertex k, by:

$$\mathbf{v}_{\mathbf{m}}(\alpha,\beta) = \mathbf{v}_{\mu} + \sum_{n=1}^{N_{\mathbf{b}}} \lambda_{\mathbf{b}_n}\alpha_n\mathbf{v}_{\mathbf{b}_n} + \sum_{n=1}^{N_{\mathbf{l}}} \lambda_{\mathbf{l}_n}(\beta_n^x\mathbf{v}_{\mathbf{l}_n}, \beta_n^y\mathbf{v}_{\mathbf{l}_n}, \beta_n^z\mathbf{v}_{\mathbf{l}_n}). \tag{6}$$

Note that unlike $\mathbf{b}_n$, the Laplacian eigenvectors are the same across the directions x, y, z.

3.3.2 Online Model Fitting

Since pose estimation is defined as a registration task aligning the head model to the observations, the head model itself should gradually be deformed to be as close as possible to the observations, and therefore adapt to the tracked person. To achieve this, we rely on a non-rigid multiple instance fitting method [31] 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 the non-rigid ICP cost (with $(\mathbf{R},\mathbf{t}) = \{(\mathbf{R}^j,\mathbf{t}^j), j \in \mathcal{J}^i\}$):

$$\begin{aligned} &E(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{R},\mathbf{t}) \\ &= \sum_{j\in\mathcal{J}^i} E_j(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{R}^j,\mathbf{t}^j) + \gamma_1\sum_{n=1}^{N_{\mathbf{b}}}\alpha_n^2 + \gamma_2\sum_{a\in\{x,y,z\}}\sum_{n=1}^{N_{\mathbf{l}}}\left(\beta_n^a\right)^2, \end{aligned} \tag{7}$$

where the cost E_j for each sample j is given by:

$$\begin{aligned} &E_j(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{R}^j,\mathbf{t}^j) \\ &= \sum_k w^j[k]\Big(\left(\mathbf{R}^j\mathbf{n}_{\mathbf{m}}(\alpha,\beta)[k]\right)^{\mathrm{T}}\left(\mathbf{R}^j\mathbf{v}_{\mathbf{m}}(\alpha,\beta)[k] + \mathbf{t}^j - \mathbf{v}_{\mathbf{o}}^j[c^j(k)]\right)\Big)^2, \end{aligned} \tag{8}$$

meaning Eq. (7) represents the sum of the rigid alignment errors with each frame of the sample set $\mathcal{J}^i$, and a regularization over the coefficients (α, β). $\gamma = (\gamma_1,\gamma_2)$ are stiffness parameters controlling how much $\mathbf{m}$ can deviate from the mean shape. The solution of Eq. (7) is found using the Gauss-Newton method by gradually reducing the stiffness [30].

3.3.3 Symmetry Regularizer

The deformation basis, especially the Laplacian basis $\mathbf{l}_n$, do not generate a symmetric deformation fields over the face. When the 3DMM fitting is based on samples where some parts of the face are not observed, the fitting may be conducted locally on the visible parts and the resulting deformation fields may diverge on the not visible ones. We show some fitting samples in Fig. 3b which are based on the profile face samples in Fig. 3a. As can be seen, the resulting meshes are asymmetric and distort the original face shape, especially in the second case which is also affected by the long hair near the face. To address this issue, we designed a symmetry regularizer to penalize the deformation coefficients which provide asymmetric structure on the iteratively fitted face. This extends Eq. (7) as follows:

$$\begin{aligned} &E(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{R},\mathbf{t}) \\ &= \sum_{j\in\mathcal{J}^i} E_j(\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{R}^j,\mathbf{t}^j) + \gamma_1\sum_{n=1}^{N_{\mathbf{b}}}\alpha_n^2 + \gamma_2\sum_{a\in\{x,y,z\}}\sum_{n=1}^{N_{\mathbf{l}}}\left(\beta_n^a\right)^2 \\ &\quad + \gamma_3\sum_k\left(\mathbf{v}_{\mathbf{m}}^x(\alpha,\beta)[k] + \mathbf{v}_{\mathbf{m}}^x(\alpha,\beta)[s(k)]\right)^2 \\ &\quad + \gamma_3\sum_k\left(\mathbf{v}_{\mathbf{m}}^y(\alpha,\beta)[k] - \mathbf{v}_{\mathbf{m}}^y(\alpha,\beta)[s(k)]\right)^2 \\ &\quad + \gamma_3\sum_k\left(\mathbf{v}_{\mathbf{m}}^z(\alpha,\beta)[k] - \mathbf{v}_{\mathbf{m}}^z(\alpha,\beta)[s(k)]\right)^2, \end{aligned} \tag{9}$$

where $s(k)$ denotes the symmetry index of vertex k. This mapping is derived from the original BFM model. For a symmetric BFM model, two symmetric points k and $s(k)$ should have the opposite x-axis values while the same y-axis and z-axis values. So the main idea of this regularizer is to maintain the symmetry of the face during progressive fitting, especially in absence of observations for some parts of the face. The fitting results using the symmetry regularizer are depicted in Fig. 3c, where better results are achieved than in Fig. 3b. Note that by preventing the fitting over the asymmetric long hair, the fitting in the second case is also improved.

3.3.4 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 [31] (see Fig. 4), and to guarantee that the observation samples cover the whole 3D face. Whenever a new frame arrives, its pose is estimated using the current head/face model and the closest of the predefined poses is identified. If the estimation is in the neighborhood of the closet predefined pose and no frame had been yet added to that predefined pose, the current frame is added to form $\mathcal{J}^i$, and the model fitting optimizing Eq. (7) is conducted with all samples in $\mathcal{J}^i$. Note that as the number of samples increases, the value of γ further decreases to allow more flexibility for the fitting.

3.4 Head Reconstruction Modeling

To handle head tracking from any pose, our goal is to dynamically augment the 3DMM-based mesh $\mathbf{m}$ with a

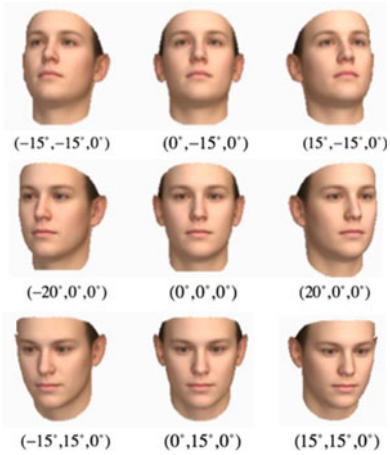


Fig. 4. Set of predefined poses (yaw,pitch,roll) used to collect data samples for online 3DMM fitting.

head reconstruction built from the observed data. To achieve this, we rely on an adaptation of KinectFusion [12]. KinectFusion originally targets 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.

The principle is to represent the head through a 3D dense volume, composed of regularly samples voxels $\mathbf{v}_g$, and to accumulate observations using a truncated signed distance function $\mathrm{TSDF}[g]$, indicating which of the points g are inside (negative value) or outside (positive value) the head surface. We here use a 3D volume of $28(depth) \times 28(height) \times 19(width)$ cm sampled with 128 steps per dimension.

The method comprises 4 main steps. The first one consists in estimating the head pose $(\mathbf{R}^i, \mathbf{t}^i)$. We rely on the robust method described in Section 3.2. Interestingly, this benefits from the availability of the 3DMM, esp. at the beginning when only few frames have been observed. The second step is the volumetric mapping, which consists in rotating the vertex samples in the camera pose according to $\mathbf{v}_g^i = \mathbf{R}^i \mathbf{v}_g + \mathbf{t}^i$. The third step consists of computing the per-frame TSDF [40] associated to the observed surface, denoted as tsdf^i and defined by:

$$\mathrm{tsdf}^i[g] = \frac{[\mathbf{v}_g^i]_z - [\pi(\mathbf{v}_g^i)]_z}{\tau}, \tag{10}$$

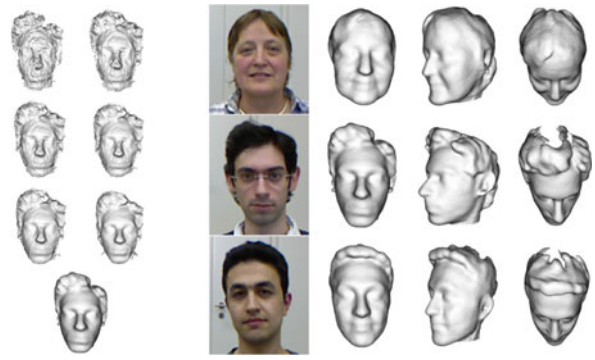
in which $\pi(\mathbf{v}_g^i)$ denotes the projection along the ray of the vertex $\mathbf{v}_g^i$ onto the observed 3D surface, and $[\cdot]_z$ denotes the depth of a 3D point. In other words, tsdf records for vertex $\mathbf{v}_g$ the signed distance between its actual location $\mathbf{v}_g^i$ and the observed surface point. The parameter τ represents the thickness around the observed surface for which such distance is computed, and actually used (see Eq. (11) below).

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

$$\mathrm{w}_{\mathrm{ts}}^i[g] = \begin{cases} 1 & \text{if } -1 < \mathrm{tsdf}^i[g] < +\infty \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$\mathrm{TSDF}^i[g] = \frac{\mathrm{w}_{\mathrm{TS}}^{i-1}[g]\mathrm{TSDF}^{i-1}[g] + \mathrm{w}_{\mathrm{ts}}^i[g]\mathrm{tsdf}^i[g]}{\mathrm{w}_{\mathrm{TS}}^{i-1}[g] + \mathrm{w}_{\mathrm{ts}}^i[g]} \tag{12}$$

$$\mathrm{w}_{\mathrm{TS}}^i[g] = \mathrm{w}_{\mathrm{TS}}^{i-1}[g] + \mathrm{w}_{\mathrm{ts}}^i[g]. \tag{13}$$



(a) online process (b) head reconstruction results

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

Importantly, note that the fusion is only conducted on voxels whose tsdf values are within the range $[-1, +\infty]$ (pixels in front of the surface or close behind the observed surface). This is to avoid self-occlusion effects for concave parts, e.g., when seen from 45 degree, the visible nose surface hides other face surfaces which might not necessarily lie 'inside' the head.

Reconstruction Model. At each time step, a reconstruction model $\mathbf{r}^i$ is built as a 3D mesh from $\mathrm{w}_{\mathrm{TS}}^i$. More concretely, the marching cubes method [41] is applied to the set of voxels for which $\mathrm{w}_{\mathrm{TS}}^i$ is larger than 25 (i.e., voxels having been observed at least 25 times within the observed surface region) to find the zero crossing surfaces and extract the vertices and their normals. Examples of reconstruction results at the end of the sequence from the BIWI dataset are shown in Fig. 5, and demonstrate that accurate models can be recovered.

3.5 Head Model

As described in Section 3.2, 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-fitted mesh $\mathbf{m}$ and the reconstruction model, we simply 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 randomly sample $N_v^{\mathbf{r}} = \eta \times N_v^{\mathbf{m}}$ from $\mathbf{r}$, and hence we have $N_v^{\mathbf{h}} = N_v^{\mathbf{r}} + N_v^{\mathbf{m}}$.

3.6 Pose Bias Correction

The 3D head reconstruction is a process which fuses the depth observations into a grid. To register the observations of different poses into a unified model, the grid is transformed with the estimated pose at every frame. Therefore, the estimated head pose is essential to the quality of reconstruction and needs to be consistent across frames.

However, in the initial frames, the pose estimation relies on the 3DMM-based representation $\mathbf{m}$, which is progressively fitted to the person's face. If this fit is good (which is usually the case when starting the sequence from a near frontal pose), the estimated pose will be very close to the real one, and the reconstruction will then implicitly be built and aligned with $\mathbf{m}$. However, if this fit is not yet fine, and/or if the initial estimated poses are biased, the reconstruction will be performed in a pose coordinate system slightly different than that of $\mathbf{m}$, and this difference may remain over time. In other words, the two head representations ($\mathbf{m}$ and reconstruction) are not fully aligned, the same facial feature may appear on two different positions, and this can confuse the registration.

Fig. 6. Dataset samples. a) BIWI. b) UbiPose.

A pose correction module aligning the reconstruction with $\mathbf{m}$ is necessary. It mainly requires to estimate the pose bias between the two representations and then use this bias to align in the same pose space the vertices sampled from them when building the common head representation.

To estimate the bias we simply rely on ICP registration. However, to achieve a fast correspondence search [36], we do not attempt at performing ICP directly between $\mathbf{m}$ and the 3D mesh of the reconstruction $\mathbf{r}$. Instead, we simply render a synthetic depth image $\mathbf{s}$ by projecting the vertices $\mathbf{v_r}$ from the reconstruction into the depth image plane (i.e., along the ray of the depth camera). Thanks to the depth averaging during reconstruction, most temporal occlusions and large depth noise are removed and the resulting images are usually of high quality. Then ICP is performed to register the 3DMM model to this map and obtain the resulting pose correspondence bias $(\mathbf{R}^c, \mathbf{t}^c)$.

Finally, to account for this bias, two modifications need to be done. First, when building the head model (Section 3.5), the vertices $(\mathbf{v}_\mathbf{r}^i[k], \mathbf{n}_\mathbf{r}^i[k])$ sampled from the reconstruction need to be mapped back into the semantic pose space of the 3DMM, according to

$$\begin{aligned} \mathbf{v}_\mathbf{r}^i[k] &\leftarrow \mathbf{R}^c \mathbf{v}_\mathbf{r}^i[k] + \mathbf{t}^c \\ \mathbf{n}_\mathbf{r}^i[k] &\leftarrow \mathbf{R}^c \mathbf{n}_\mathbf{r}^i[k]. \end{aligned} \tag{14}$$

Second, in order to keep the consistency with the previous tsdf in the head reconstruction (Section 3.4), the volumetric mapping needs to be the composition of the estimated pose $(\mathbf{R}^i, \mathbf{t}^i)$ and the inverse correction $(\mathbf{R}^c, \mathbf{t}^c)$.

Importantly, note that since the reconstruction and 3DMM-fitted model evolve slowly after the initial frames, the pose correction $(\mathbf{R}^c, \mathbf{t}^c)$ is only computed every 100 frame in our implementation (but it is used at all frames).

4 Experimental Protocol

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

4.1 Dataset

In our experiments, two datasets are used.

4.1.1 The BIWI Dataset

It is a public dataset collected by Fanelli et al. [42]. It consists of 24 videos (15K frames in total) recorded with a Kinect 1 sensor, and where seated people keep moving their heads in an artificial fashion. Some samples are shown in Fig. 6a.

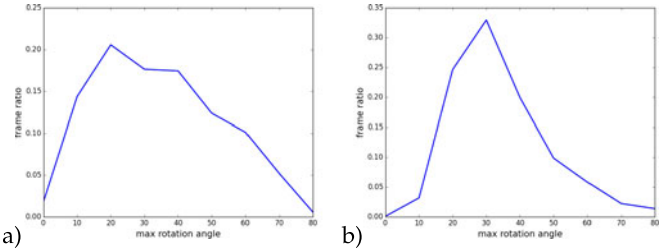


Fig. 7. Proportion of frames with a given pose GT (or IGT for UbiPose) for (a) the BIWI dataset (b) UbiPose dataset.

4.1.2 The UbiPose Dataset

This dataset relies on videos from the UBImpressed dataset, which has been captured to study the performance of students from the hospitality industry at their workplace [13]. 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 frequent as people are observed from a relatively large distance, and people are mainly seen from the side (see Fig. 6b for samples).

Out of the 160 interactions recorded in the UBImpressed dataset, we selected 22 videos (with 22 different persons) as evaluation data to build the UbiPose dataset. In 10 of these videos, 30-50 second clips were cut from the original videos and all frames were annotated (see Section 4.2.2). The other 12 videos were fully annotated at one frame per second. This allowed to gather a large diversity of situations. In total, this amounts to 14.4K frames. The UbiPose dataset with annotations and evaluation code can be found at www.idiap.ch/dataset/ubipose.

4.2 Ground Truth

4.2.1 BIWI Data

This dataset provides the ground truth of head pose $(\mathbf{R}, \mathbf{t})$ for every single frame, which was estimated using a supervised 3D face fitting and registration process. Fig. 7a indicates the distribution of the number of frames over pose ranges for this dataset

4.2.2 UbiPose Data: Inferred Pose Ground Truth (IGT)

To avoid interfering with the role play, no wearable sensors, e.g., motion capture, were used to obtain a head pose ground truth. So we inferred the ground truth indirectly from facial landmarks. Concretely, we first annotated 6 landmarks on the extracted RGB frames whenever they were 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). Generally speaking, these landmarks are rigid and seldom affected by facial expressions.

These 2D landmark annotations were projected into the 3D space using the depth image, and further paired with the corresponding landmarks in the 3DMM. Note that to foster precise head pose ground truth, the 3DMM had been previously fitted to the person's face using auxiliary data from a recording session made another day (with an interview scenario). The ground truth was then inferred from the available point pairs [39]. We denote the inferred ground truth as IGT.

TABLE 1
Average Error of IGT

	yaw	pitch	roll	mean ± std	ACC_{10}
IGT	3.26	3.79	2.48	3.18 ± 1.61	100.0%

Since for near profile poses the IGT might become noisy (see Section 4.4 for an evaluation of the annotation accuracy), we resorted to visual inspection to validate the IGT. More precisely, we examined the IGT frame by frame by projecting the IGT posed 3DMM model on the 2D image and compared it with the actual pose of the person in the image. If they were perceived as not matching, a new annotation was attempted. If the difference remained unacceptable after revision, the frame was definitely abandoned. After inspection, we obtained a dataset of 10.5K frames in total for experiment. The distribution of frames with respect to the head pose is illustrated in Fig. 7b. As can be seen, due to the scenario, most frames fall within a $[20°, 40°]$ interval. Compared with the BIWI dataset, we observe that there are much less frontal faces, whereas the percentage of frames above 80 degree is higher.

4.3 Performance Measures

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

4.3.1 Accuracy

Since we have head pose ground truth, we first report accuracy using on one hand the average error of the estimated rotation angles. On the other hand, since we have annotated up to six landmarks per frame on the UbiPose dataset, we also measure the accuracy of landmark localization. Note that the landmark location estimates are obtained by projecting the semantic 3DMM-fitted model set with the estimated pose, and the accuracy of landmarks is measured as the 2D distance between the estimated and annotated landmark locations.

4.3.2 Robustness

Robustness can be defined by several aspects. One of them is to evaluate whether the error can be kept in an accepted range even when extreme head pose occurs. This can be measured with the cumulative distribution function of errors (error CDF) showing the proportion of frames whose errors are below a given value. We further use this curve to report as in [10] accuracy measures ACC_{10} as the percentage of frames with L2 norm of angular errors below 10 degrees.

We can also analyse the robustness by measuring the accuracy across different head poses. We take the maximum of the three ground truth rotation angles (yaw/pitch/roll) as pose indicator per frame and quantize them in bins of size 10 degree. Then the average error is computed within each bin.

The robustness also usually means continuous and successful tracking. Therefore, we define the lost frame ratio (*LR*) to indicate the percentage of frames in which the tracker is in a failure state, because a tracking failure has occurred and that the tracker could not successfully reinitialize itself. In addition, to better analyze the robustness to occlusion, we annotated video segments where at least half of the face is occluded and computed the ratio *O-LR* of tracking failure which have occurred in such segments. Note that *O-LR* is an event based measurement which complements the frame-based measure *LR*.

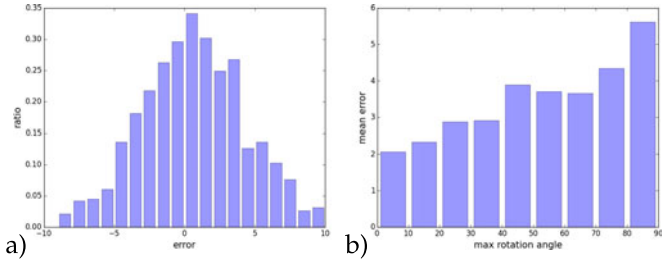


Fig. 8. IGT quality evaluation based on BIWI data. a) distribution of the yaw, pitch and roll errors between IGT and GT. b) distribution of IGT absolute pose errors.

Finally, we also measure the impact of facial expressions on pose estimation. As a proxy for this, with the help of microphone array data (recorded as part of the UbiPose database), we extracted frames where the participants are speaking since facial deformations are expected to be important in these frames due to mouth motion, but also because people are usually more expressive when speaking. The results on these frames are reported in the column "*S-mean*" (mean on speaking frames) of the result tables.

4.3.3 Significance Test

To evaluate whether the results of two methods are statistially significant (esp. with respect to the proposed model), and given the large number of samples available, we rely on a paired z-test and report the significance of the test for different p-values.

4.4 IGT Evaluation

Compared with the supervised dense ICP registration of BIWI, the inferred ground-truth (IGT) is only based on limited facial landmarks. Even if we used a visual inspection to validate them, we have no clear idea about its accuracy. To evaluate this, we conducted a small scale experiment in which we annotated a subset of the BIWI dataset with landmarks, and following the same procedure than with the UbiPose data, inferred the IGT. In practice, we used a subset of 450 random frames, and 381 remained after visual validation.

Table 1 and Fig. 8 provide the results of the IGT evaluation. As can be observed, the mean error is around 3 degree, with a small standard deviation. Fig. 8a shows that errors follow a zero mean Gaussian distribution, indicating that no bias is observed. In addition, as was to be expected (less available landmarks, higher inference sensitivity to location accuracy), Fig. 8b shows the limited increase of error in function of the observed pose, but even for large pose, the average error remains below 6 degree. In general, around 85 percent of the errors are below 5 degree, and all errors are below 10 degree.

Altogether, although not perfect, we believe that the IGT can be used as GT for UbiPose. While the reported error may not reflect the actual accuracy of methods, given the large number of samples (above 10000), which are dominantly independent and uncorrelated and with unbiased approximation, we expect the evaluation to provide a fair indication of which method performs best. This is particularly true for

performance measures like ACC_{10} which are indicative of robustness, and somehow already account for some uncertainty in the ground truth. In any case, although of a different nature, the raw annotated landmarks will also be used for evaluation.

4.5 Systems and Parameter Settings

We compared several models as listed below:

- *Mean shape:* the tracking is conducted with ICP using the mean shape of the Basel Face Model (BFM).
- *3DMM:* An online model fitting is conducted using the BFM and Laplacian basis function and the symmetric constraint. This differs from most previous works which only fit the BFM model [10], [31], [43]. Also, following [2], the 3DMM and Mean shape models rely on a sample subset of the vertices of the full BFM model, with a denser sampling on rigid face regions (forehead, eye regions) [2], [31].
- *FWH-ID and FWH-EXP:* FaceWarehouse [26] is a 3D facial expression database providing aligned 3D head models of 47 expressions from 150 participants. We derived the deformation bases of both identity and expressions from these 3D scans and built two models, namely FWH-ID and FWH-EXP. For the FWH-ID model, only the identity deformation bases are used and the online model fitting is conducted as with the BFM model. This allows to evaluate the impact of the used 3DMM mesh model (BFM versus FWH) on the tracking results. In the FWH-EXP model, the expression bases are added to the identity bases of the FWH-ID model, allowing to test the performance of using a richer (and in principle more relevant) model to fit the data.
- *FHM:* the tracking is only based on the reconstructed head model, except in the first 25 frames where the 3DMM is still used to build an initial 3D model.
- *HeadFusion:* this is the proposed model. It includes both the online model fitting with Laplacian basis function and symmetric constraint, the KLT tracking intialization, head reconstruction with pose correction. The default value for η (proportions $\eta = \frac{N_v^{\mathbf{r}}}{N_v^{\mathbf{m}}}$ of points coming from the reconstructed $\mathbf{r}$ and 3DMM models, see Section 3.5) is set to 1.5.
- *State-of-the-art:* we compared our work with three methods. The 3DMM fitting based approach [10] using Particle Swarm Optimization (PSO) for tracking, which achieves the best results on BIWI; the OpenFace system [44] which relies on both image and depth data and has been primarily optimized for landmark localization; and our previous work [43], which combines 3DMM and reconstruction but without several key elements like KLT, pose correction, symmetric fitting constraint.

Parameter Settings. All model parameters were kept the same for all experiments (except for reporting explicit changes, eg. the impact of η) and the two datasets. Whenever relevant, we used $N_{\mathbf{b}} = 50$ deformation bases from the BFM model and the same number for the Laplacian model. For all models involving head reconstruction, the size of the 3D volume is $128 \times 128 \times 128$.

5 Results

To analyse our results, we first present qualitative results in Section 5.1. We then detail numerically our results in Section 5.2, comparing the different head representation approaches, including against state-of-the-art methods. Finally, in Section 5.3, we evaluate the benefits of the different components of our method.

5.1 Qualitative Results

Fig. 9 illustrates the obtained results. As can be seen, robust and accurate tracking can be achieved, in both typical and more adverse conditions, like leaning, looking towards the back while calling on the phone, or putting the hand in front of the mouth. The impact and requirement for a full head representation is clearly visible, and our method allows to handle it, even when the head (eg the right three pictures in Fig. 9c) is only partially visible and could easily lead to uncertainty in pose estimation and tracking failure. In addition to the head representation, KLT tracking proved to be particularly useful, e.g., in handling people looking for objects in the registration desks, where non-frontal faces with fast motion and pose changes could be observed (Fig. 9c).

5.2 Quantitative Analysis

The tracking and head pose results of all methods are listed in Table 2 (BIWI) and Table 3 (UbiPose), whereas Table 4 display the results for the landmark localization task on UbiPose data. For further analysis of the models' properties, error CDF, error distribution on poses and *LR* distribution are also provided in Fig. 10. In the sequel, we will first analyze the performance of the different head pose modeling methods before comparing to the state-of-the-art.

5.2.1 Overall Result

We first compare the FaceWarehouse models with the BFM model. We first note that the FWH-ID model performs worse than the (BFM) 3DMM model. This might be explained by the fact that the BFM model was built from high resolution scans, compared to lower quality data for the FWH model (for which vertex subsampling was not necessary because of the lower resolution). Furthermore, we find that the performances (both for pose and landmark estimation) of the FWH-EXP expression model are worse than those of its ID only counterpart FWH-ID. This is not so surprising, as in presence of noise or non frontal head pose, the additional fitting capacity may lead to expression basis fitting pose or identity information rather than only the facial deformation, resulting in a distorted face model whose fitting reduces the accuracy of head pose estimation. In practice, as noted in [2], to handle facial deformation, it is more efficient (and better) to first fit the head pose, and then estimate the facial deformation. In contrast, the expression independent BFM model achieves better performance in head pose estimation.

We then compare the BFM Mean Shape, 3DMM, FHM and our HeadFusion models. As can be seen, our proposed model HeadFusion has the best accuracy and robustness for most performance measures: it has the lowest head pose error on both BIWI and UbiPose, the lowest landmark localization error on UbiPose, and the best robustness indicators (least error variance, lowest *S-mean*, *LR* and *O-LR*), indicating that it has a more stable tracking and suffers from much less tracking

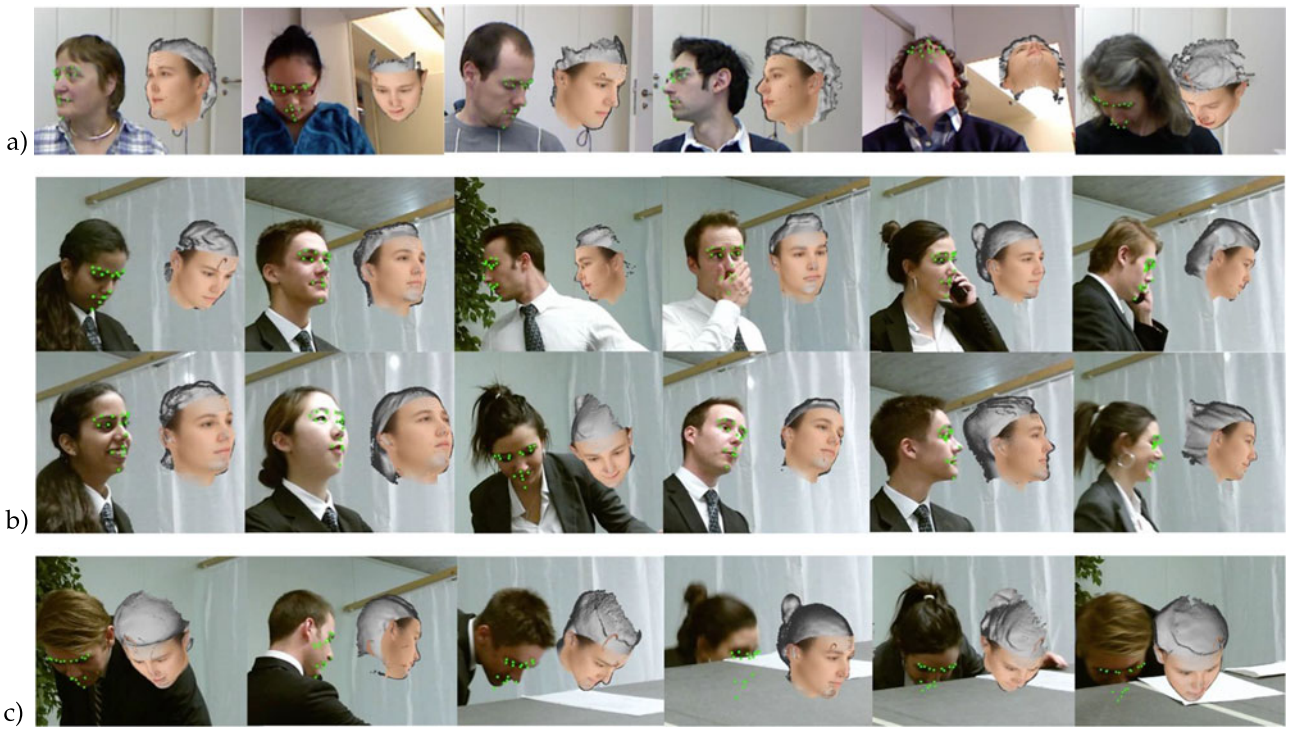


Fig. 9. 3D head reconstruction and tracking samples. a) BIWI dataset. b) Typical frames of UbiPose dataset. c) Extreme head pose cases and occlusion cases. Note that for better visualization, displayed image were cropped from original images.

TABLE 2
BIWI: Average Head Pose Error and Accuracy[1]

Approach	yaw	pitch	roll	mean (std)	ACC_{10}
FWH-ID	2.47	3.01	2.15	2.54 (3.5)†	94.7%
FWH-EXP	2.46	3.87	2.15	2.83 (3.6)†	92.4%
Mean shape	5.20	2.72	4.23	4.05 (9.2)†	88.7%
3DMM	2.96	1.58	2.65	2.40 (4.8)†	94.7%
FHM	3.30	1.82	2.45	2.52 (3.2)†	94.5%
HeadFusion	2.54	1.45	2.10	2.03 (3.0)	96.4%
OpenFace [44]	7.77	7.99	4.61	6.79 (6.8)†	52.3%
Yu et al. [43]	2.49	1.53	2.18	2.07 (5.2)	96.6%
PSO [10]	2.1	2.1	2.4	2.2	94.6%

$^{\dagger}p < 0.01$.

failures. In particular, the tracking failure in occlusion cases of our method is much less than the approaches without reconstruction. This is understandable since our model can rely on more points from the full head for model registration. The robustness is also reflected in Fig. 10d where for large poses, *LR* is much lower for our approach.

Notice that the accuracy gap between the proposed models and the others is larger on UbiPose than on BIWI, probably because the former dataset involves more natural behaviors and comprises much more diverse and adverse situations. Note as well that for UbiPose dataset average errors and curves in Fig. 10 are reported on frames without failures, thus results from our approach are computed from more frames. This explains why the ACC_{10} on UbiPose is better for the 3DMM than for our approach, since the 3DMM errors are gathered from less frames and in particular exclude those which often correspond to difficult situations and higher pose errors in general.

Looking at the *S-mean* results (speech frames with facial expressions) in Table 3 and in Fig. 10h, we can notice that almost all methods keep a stable performance compared with the overall results (mean), including under difficult poses. For the BFM based methods, this can be attributed to two main factors: first, our robust weighting strategy of ICP (see section 3.2.1) which can filter out bad correspondences caused by expression deformations; second, the selection of mesh samples in face regions less affected by facial deformations. On their side, by averaging faces over time, reconstruction models (FHM or HeadFusion) result in a neutral model which combined with the previous factors, avoids the addition of specific facial expression biases.

All in all, these results demonstrate that our method has the potential for continuous and uninterruptive tracking which is necessary for tracking in natural interaction setting. This is due to the good exploitation of the joint benefit of the 3DMM model and of the FHM approach.

Indeed, on one hand, compared to FHM, the 3DMM achieves higher accuracy, as demonstrated by a much higher ACC_{10} of 70.9 percent compared to 56.0 percent on UbiPose, but this is at the cost of less robustness: a much higher variance and difficulty to detect tracking failure (thus reporting larger errors, as can be noticed from the fact that the CDF of the 3DMM does not reach 100 percent in Fig. 10e), in particular for large head pose (see Fig. 10b for instance).

On the other hand, the FHM model is less accurate (see the worse CDF curves at small angles on BIWI and more importantly on UbiPose, Fig. 10e), but is more robust as shown by the much smaller pose error standard deviation on UbiPose, or the lower tracking failure *LR* and *O-LR* compared to the 3DMM. However, it is clear from the results

1. † indicates that the result is significantly lower than our method with $p < 0.01$. The test with PSO [10] is not possible.

TABLE 3
UbiPose: Average Head Pose Error and Accuracy

Approach	yaw	pitch	roll	mean (std)	*S-mean* (std)	ACC_{10}	*LR*	*O-LR*
FWH-ID	8.45	4.87	4.94	6.09 (9.7)†	6.22 (10.7)†	70.3%	4.1%	30.8%
FWH-EXP	11.55	6.65	7.16	8.45 (15.1)†	8.64 (15.5)†	64.7%	5.7%	46.2%
Mean shape	6.77	5.03	5.12	5.64 (7.5)†	5.75 (8.0)†	64.2%	3.5%	38.5%
3DMM	5.63	5.05	4.57	5.08 (6.9)†	5.10 (6.9)†	70.9%	4.1%	38.5%
FHM	5.33	4.96	4.61	4.97 (2.8)†	5.06 (3.3)†	56.0%	3.6%	15.4%
HeadFusion	4.63	4.37	3.83	4.28 (2.7)	4.25 (3.0)	70.0%	0.6%	15.4%
OpenFace [44]	9.49	4.45	4.89	6.27 (4.0)†	6.94 (4.7)†	44.3%	8.7%	100.0%
Yu et al. [43]	5.09	5.14	3.90	4.71 (4.5)†	4.60 (4.8)†	67.2%	3.3%	23.1%

$^\dagger p < 0.01$.

that the FHM model alone is not sufficient to achieve good tracking, and that it is the combination of the 3DMM and FHM which performs best.

Finally, regarding the Mean Shape model, one can notice that its results on BIWI and UbiPose (Tables 2 and 3) are lower than other models including the 3DMM model. In fact, the error of the Mean shape model is relatively large for almost every pose bin according to Fig. 10b and f[2], which reflects the importance of online model adaptation in model registration.

5.2.2 Comparison with State-of-the-Art Methods

Three state-of-the-art methods were used, as described in Section 4.5: PSO [10], the CMU OpenFace [44], and our previous work [43].

BIWI Dataset. Our HeadFusion model obtains the best results. It exceeds the performance of PSO [10] which relies on the combination of ICP and Particle Swarm Optimization (PSO), which shows that when combining a 3DMM with head reconstruction, ICP alone can achieve equal or even better accuracy. In particular, the estimation of the pitch angle is much improved compared to [10]. OpenFace provides by far the worst error, 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, on BIWI, we do not notice much improvement from our method compared to our previous work [43].

UbiPose Dataset. Table 3 demonstrates that our method performs much better than OpenFace for pose estimation, both in terms of accuracy and importantly robustness (much better ACC_{10}, *LR* and *O-LR* values). This claim is further supported by Fig. 10f, which shows that the error of OpenFace becomes much larger beyond 45 degree. Note that the *O-LR* value is 100 percent, which shows that the OpenFace has difficulty in handling cases where at least half of the face is occluded. However, its performance for landmark localization is usually better, as shown in Table 4, as it was specifically trained for that.[3] This is not contradictory: since localization accuracy is computed only for visible landmarks, the localization errors can still remain small even if the pose estimate is bad, esp. for adverse situations where only a limited set of landmarks is visible. This contrast is illustrated in Fig. 11 and such situations are relatively frequent for OpenFace.

Compared to our previous work, the difference of the mean error is not that large (but is still statistically significant). However, the robustness is much higher with our new method, as shown by the higher *LR* and *O-LR* values of [43], which, without the coarse temporal alignment module, can not handle most fast motions of our data.

5.3 Model Components Analysis

In this section, we study the contribution of the different modeling components to the success of the method. We present and contrast the results of 7 experiments in Table 5 (BIWI) and Table 6 (UbiPose) by changing system parameters or removing some components.

Our approach samples points from the 3DMM and the 3D reconstruction to build the head model, with a ratio η. The default value in HeadFusion is $\eta = 1.5$, meaning that more points are sampled from the reconstruction. As can be seen, when using smaller values, the model is slightly less accurate, and less robust (higher error and standard deviation), in particular for UbiPose dataset.

Head Pose Correction. Results show the requirement for this correction module. Without it, the error becomes larger and more variable, especially for the UbiPose dataset. This can be explained by the fact that sequences start with a semi-profile face, which usually result in a small but non negligeable initial bias between the 3DMM model and the head reconstruction. Interestingly, the removal of the component does not seem to result in much more additional tracking failures.

Temporal Alignment. When removing the coarse temporal alignment relying on the KLT tracker, we can notice that the

TABLE 4
UbiPose: Landmark Position Errors

Approach	l-l	r-l	l-r	r-r	n-r	n-t	mean
FWH-ID	9.6	9.1	10.2	9.5	13.3	13.9	11.3 (19.7)†
FWH-EXP	11.5	11.7	14.0	15.1	16.9	18.2	14.7 (27.8)†
Mean shape	7.7	10.8	10.7	11.1	9.3	10.1	9.7 (15.8)†
3DMM	7.5	10.9	10.1	11.5	8.9	10.5	9.6 (16.5)†
HeadFusion	5.4	7.9	7.1	9.3	6.0	6.5	6.7 (6.0)
OpenFace [44]	5.1	5.8	5.6	6.6	6.0	6.0	5.8 (4.1)
Yu et al. [43]	6.0	9.3	9.2	11.4	7.4	8.6	8.1 (12.1)†

$^\dagger p < 0.01$.

2. Note that in Fig. 10f, the error of the Mean shape in the first bin (<5 degree) is abnormally high. This is due to the fact that there are only 8 frames in that bin (see Fig. 6b), and for that method, the tracking results are bad for 6 contiguous frames due to the impact of an erroneous tracking right before these frames.

3. Remember however that due to the difference in tracking failures (*LR*) the average error of OpenFace is computed on 8.1 percent less frames than our method, frames in which the pose is usually large.

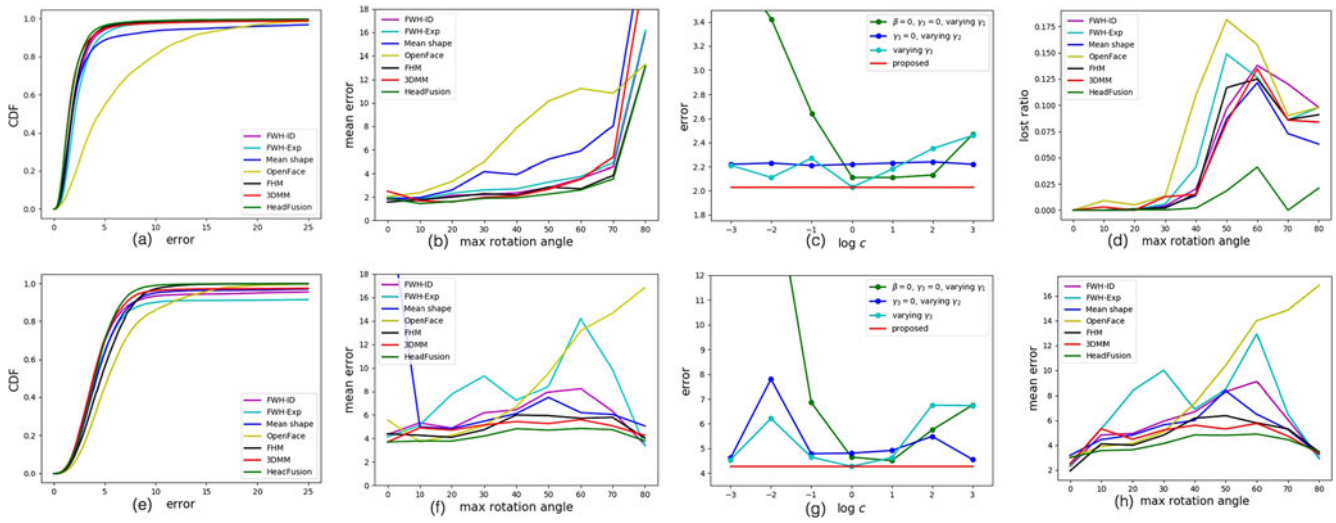


Fig. 10. Robustness curves for BIWI (a, b, c) and UbiPose data (d, e, f, g, h). (a, e) Pose error CDF. (b, f) Mean error per pose. (c, g) Impact of regularization coefficients. (d) Tracking lost ratio per pose (UbiPose). (h) Mean error on speaking frames, per pose (UbiPose).

performance does not decline too much in accuracy measurement. Rather, as shown by the results in Table 6, there is a higher number of tracking failures due to dynamical head motions that the tracker can not handle anymore.

Head Model Fitting. The end of Section 5.2.1 highlighted the complementarity and mutual benefit of using the 3DMM and FHM head models. Here we further study the impact of the 3DMM modeling on results by removing some deformation bases in Eq. (9) ($\beta = 0$), or by varying the weights of the regularizing terms (γ_1, γ_2, γ_3) by a factor c ($c = 0.001, 0.01, 0.1, 10, 100, 1000$) with respect to their default value. Results are reported in Tables 5 and 6, and Fig. 10c and Fig. 10g. We first remove both the symmetry regularizer and the Laplacian bases (green curve) and vary the weight γ_1. We observe that enforcing more ID shape bases regularization usually lead to better results. However, when γ_1 becomes too large, results quickly degrades in both datasets, and in practice, the fitted head models remain very close to the mean shape model. This is corroborated by results in Tables 5 and 6 ('without Fitting'), which show that simply using only the 3DMM mean shape actually achieves worse results than methods with online model adaptation. When adding Laplacian bases and adjusting the weight γ_2 ($\gamma_3 = 0$), we note that the performances are relatively stable for different values of γ_2. This is understandable, since the Laplacian bases mainly compensate the original deformation bases for a finer 3DMM modeling. However, the performances with Laplacian bases are inferior to the model with a suitable value of γ_1. Indeed, with fitting samples seen from semi-profile, a poor 3DMM fitting can be obtained (as already illustrated in Fig. 3) with asymmetric variations coming from Laplacian bases, for which a symmetry constraint is a must. Finally, we observe an improvement of performance (0.18 degrees on BIWI, 0.53 degrees on UbiPose) when using a suitable symmetric regularization. We also note from Fig. 10c and Fig. 10g that emphasizing too much on symmetry regularizer can make the 3DMM fitting too constrained and lead to worse performances. Altogether, results in Fig. 10c and Fig. 10g show that our model with selected weights achieves the best compromise between robustness, accuracy, and quality of face fitting.

Finally, when removing the regularization and the KLT tracking, the negative effects are cumulated, resulting in a performance decrease in both accuracy and robustness.

5.4 Computational Cost

We implement our system in Python/C++ based on CPU. Generally speaking, the coarse temporal alignment based on KLT tracker takes ~60ms and the following ICP based alignment costs ~9ms. The 3DMM fitting executed in a separate thread usually costs ~5s. 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.

TABLE 5
BIWI Contrastive Experiments

Approach	yaw	pitch	roll	mean (std)	ACC_{10}
HeadFusion	2.54	1.45	2.10	2.03 (3.0)	96.4%
$\eta = 0.5$	2.48	1.41	2.19	2.03 (3.1)	96.4%
$\eta = 1.0$	2.56	1.46	2.15	2.06 (3.3)	96.5%
Without Correction	3.24	1.66	2.35	2.42 (5.8)	95.7%
Without KLT	2.98	1.78	2.34	2.37 (6.0)	96.0%
Without Sym	2.67	1.68	2.31	2.22 (4.4)	96.0%
Without Fitting	2.93	1.97	2.36	2.42 (3.6)	94.6%
Without KLT, Lap, Sym	2.75	1.74	2.37	2.29 (4.7)	95.8%

TABLE 6
UbiPose Contrastive Experiments

Approach	yaw	pitch	roll	mean (std)	ACC_{10}	LR	$O\text{-}LR$
HeadFusion	4.63	4.37	3.83	4.28 (2.7)	70.0%	0.6%	15.4%
$\eta = 0.5$	5.51	4.62	4.56	4.90 (4.9)	69.5%	0.6%	7.7%
$\eta = 1.0$	4.88	5.02	4.12	4.68 (5.2)	69.5%	0.6%	15.4%
Without Correction	7.57	4.96	4.51	5.68 (10.5)	70.4%	0.7%	23.1%
Without KLT	4.72	4.37	3.88	4.33 (3.2)	70.0%	3.3%	23.1%
Without Sym	4.75	5.38	4.31	4.81 (5.1)	65.7%	0.7%	23.1%
Without Fitting	9.48	5.21	5.74	6.81 (9.8)	61.2%	0.6%	15.4%
Without KLT, Lap, Sym	9.39	5.99	5.49	6.96 (8.8)	58.3%	3.2%	23.1%

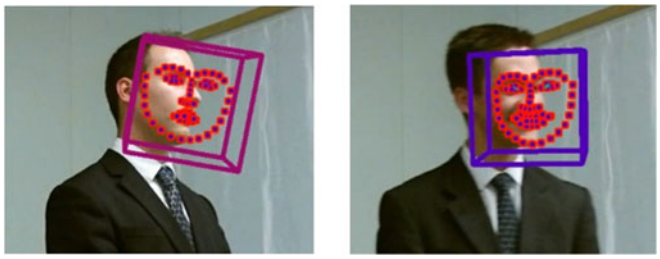


Fig. 11. CMU OpenFace common failures. Although the error distance with respect to the visible landmarks is small, the head pose is very badly estimated. Note that OpenFace is using depth information as well.

6 Conclusions and Future Works

We presented an accurate and robust 3D head pose estimation method effective even for challenging natural settings. The main idea is to build a full head model providing more support when dealing with arbitrary tracking situations. To achieve this, we simultaneously conduct a 3DMM online fitting and and online 3D head reconstruction using a KinectFusion methodology. In addition, we also proposed a coarse temporal alignment module to handle fast head motions and a symmetry regularizer for finer model adaptation. Results demonstrate that our method achieves state-of-the-art performance and is also accurate and very robust when dealing with challenging natural interaction sequences where adverse situations are frequent.

Recovering the semantic segmentation of the head model (eg which region is face or hair) is an interesting perspective to the work. Indeed, the semantic information could help the landmark localization estimation to be more accurate; and second, as our method is still challenged by long hairs moving around, the knowledge of the semantic information could help in obtaining even more robust results. Our work can also be expanded to other tasks, by serving as a preprocessing step for head gesture recognition, eye gaze tracking, or facial expression estimation and analysis as shown by our experiments on this topic.

Acknowledgments

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 Union's 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 146–155.

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

[3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1701–1708.

[4] K. A. Funes Mora and J.-M. Odobez, "Gaze estimation in the 3D space using RGB-D sensors. Towards head-pose and user invariance," *Int. J. Comput. Vis.*, vol. 118, pp. 194–216, Nov. 2016.

[5] U. Hadar, T. J. Steiner, and F. C. Rose, "Head movement during listening turns in conversation," *J. Nonverbal Behavior*, vol. 9, pp. 214–228, 1985.

[6] 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 *Proc. 18th ACM Int. Conf. Multimodal Interaction*, Nov. 2016, pp. 21–28.

[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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4722–4730.

[8] G. Fanelli, J. Gall, and L. V. Gool, "Real time head pose estimation with random regression forests," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 617–624.

[9] T. Weise, S. Bouaziz, H. Li, and M. Pauly, "Realtime performance-based facial animation," in *Proc. ACM SIGGRAPH*, 2011, Art. no. 11.

[10] G. P. Meyer, S. Gupta, I. Frosio, D. Reddy, and J. Kautz, "Robust model-based 3D head pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3649–3657.

[11] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," in *Proc. 26th Annu. Conf. Comput. Graph. Interactive Techn.*, 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 *Proc. ACM Symp. User Interface Softw. Technol.*, 2011, pp. 559–568.

[13] 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 *Proc. 18th ACM Int. Conf. Multimodal Interaction*, 2016, pp. 84–91.

[14] L.-P. Morency, J. Whitehill, and J. Movellan, "Generalized adaptive view-based appearance model: Integrated framework for monocular head pose estimation," in *Proc. 8th IEEE Int. Conf. Autom. Face Gesture Recognit.*, 2008, pp. 1–8.

[15] D. Cristinacce and T. F. Cootes, "Automatic feature localisation with constrained local models," *Pattern Recognit.*, vol. 41, no. 10, pp. 3054–3067, 2007.

[16] T. Baltrusaitis, P. Robinson, and L.-P. Morency, "3D constrained local model for rigid and non-rigid facial tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2610–2617.

[17] T. Baltrusaitis, P. Robinson, and L.-P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, 2013, pp. 354–361.

[18] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5325–5334.

[19] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "It's written all over your face: Full-face appearance-based gaze estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 2299–2308.

[20] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3476–3483.

[21] E. Zhou, H. Fan, Z. Cao, Y. Jiang, and Q. Yin, "Extensive facial landmark localization with coarse-to-fine convolutional network cascade," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, 2013, pp. 386–391.

[22] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models: Their training and application," *Comput. Vis. Image Underst.*, vol. 61, no. 1, pp. 38–59, Jan. 1995.

[23] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 681–685, Jun. 2001.

[24] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3D face model for pose and illumination invariant face recognition," in *Proc. 6th IEEE Int. Conf. Advanced Video and Signal based Surveillance (AVSS) for Sec., Safety and Monitoring in Smart Environments*, 2009.

[25] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway, "A 3D morphable model learnt from 10,000 faces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5543–5552.

[26] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3D facial expression database for visual computing," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 3, pp. 413–425, Mar. 2014.

[27] T. Vetter and V. Blanz, "Estimating coloured 3D face models from single images: An example based approach," in *Proc. Eur. Conf. Comput. Vis.*, 1998, pp. 499–513.

[28] B. S. Göktürk, J. Y. Bouguet, and R. Grzeszczuk, "A data-driven model for monocular face tracking," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, 2001, pp. 701–708.
[29] A. Jourabloo and X. Liu, "Large-pose face alignment via CNN-based dense 3D model fitting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4188–4196.
[30] B. Amberg, R. Knothe, and T. Vetter, "Expression invariant 3D face recognition with a morphable model," in *Proc. 8th IEEE Int. Conf. Autom. Face Gesture Recognit.*, 2008, pp. 1–6.
[31] K. A. Funes Mora and J.-M. Odobez, "Gaze estimation from multimodal kinect data," in *Proc. IEEE Comput. Society Conf. Comput. Vis. Pattern Recognit. Workshop*, 2012, pp. 25–30.
[32] P.-L. Hsieh, C. Ma, J. Yu, and H. Li, "Unconstrained realtime facial performance capture," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1675–1683.
[33] D. Thomas and R.-I. Taniguchi, "Augmented blendshapes for real-time simultaneous 3D head modeling and facial motion capture," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3299–3308.
[34] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 343–352.
[35] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb, "Real-time 3D reconstruction in dynamic scenes using point-based fusion," in *Proc. Int. Conf. 3D Vis.*, 2013, pp. 1–8.
[36] S.-Y. Park and M. Subbarao, "An accurate and fast point-to-plane registration technique," *Pattern Recognit. Lett.*, vol. 24, no. 16, pp. 2967–2976, Dec. 2003.
[37] D. E. King, "Dlib-ml: A machine learning toolkit," *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, 2009.
[38] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1867–1874.
[39] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
[40] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. 23rd Annu. Conf. Comput. Graph. Interactive Techn.*, 1996, pp. 303–312.
[41] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," in *Proc. 14th Annu. Conf. Comput. Graph. Interactive Techn.*, 1987, pp. 163–169.
[42] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, "Random forests for real time 3D face analysis," *Int. J. Comput. Vis.*, vol. 101, no. 3, pp. 437–458, Feb. 2013.
[43] Y. Yu, K. A. Funes Mora, and J.-M. Odobez, "Robust and accurate 3D head pose estimation through 3DMM and online head model reconstruction," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit.*, 2017, pp. 711–718.
[44] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "Openface: An open source facial behavior analysis toolkit," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2016, pp. 1–10.

Yu Yu received the BSc and MSc degrees in pattern recognition and intelligent system from Xian Jiaotong University, China, in 2011 and 2014 respectively. He is now working toward the PhD degree at EPFL and is research assistant with the Idiap Research Institute under the supervision of Dr. Jean-Marc Odobez. His current research mainly focuses on non-verbal cue extraction.

Kenneth Alberto Funes Mora is the CEO and Co-founder of Eyeware Tech SA. He obtained his PhD in Electrical Engineering at the École polytechnique fédérale de Lausanne (EPFL) in 2015, while being a research assistant at Idiap Research Institute in Switzerland. He obtained a multiple Erasmus Mundus MSc degree in computer vision and robotics in 2010 (Université de Bourgogne, Universitat de Girona and Heriot-Watt University). During his PhD, and later as postdoctoral researcher, he developed techniques for 3D eye gaze estimation using consumer RGB-D sensors. He is the author or coauthor of 11 papers in international journals and conferences, and 2 filed patents, and has received awards in both academic and entrepreneurial settings. He co-founded Eyeware Tech SA in 2016, a company aiming to bring unconstrained attention sensing technology to multiple applications.

Jean-Marc Odobez received his engineering degree from the Ecole Nationale Supérieure des Télécommunications de Bretagne (ENSTBr), France, in 1990, and the PhD degree at INRIA, Rennes University, France, in 1994. He was associate professor in computer science with the Université du Maine, Le Mans, France, from 1996 to 2001. He is now a senior researcher at both the Idiap Research Institute and EPFL, Switzerland, where he directs the Perception and Activity Understanding group. His main areas of research are computer vision and machine learning techniques applied to multimedia content analysis, tracking, and human activity and behavior recognition. He is the author or coauthor of more than 150 papers in international journals and conferences. He is or was the principal investigator of 13 European and Swiss projects. He holds two patents on video motion analysis and gaze estimation. He is the cofounder of the Swiss Klewel SA company and of the Eyeware Tech SA company. He is a member of the IEEE, and associate editor of the IEEE Transaction on Circuits and Systems for Video Technology and Machine Vision and Application journals.