Combined Framework for Real-Time Head Pose Estimation using Facial Landmark Detection and Salient Feature Tracking

Jilliam María Díaz Barros[1,2] *, Frederic Garcia[1], Bruno Mirbach[1], Kiran Varanasi[2] and Didier Stricker[2]

[1]*PTU Optical, IEE S.A., Contern, Luxembourg*

[2]*German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany*

{*Jilliam.Diaz, Frederic.Garcia, Bruno.Mirbach*}*@iee.lu*, {*Kiran.Varanasi, Didier.Stricker*}*@dfki.de*

Keywords: Head pose estimation, Real time, Fusion.

Abstract: This paper presents a novel approach to address the head pose estimation (HPE) problem in real world and demanding applications. We propose a new framework that combines the detection of facial landmarks with the tracking of salient features within the head region. That is, rigid facial landmarks are detected from a given face image, while at the same time, salient features are detected within the head region. The 3D coordinates of both set of features result from their intersection on a simple geometric head model (*e.g.*, cylinder or ellipsoid). We then formulate the HPE problem as a perspective-n-point problem that we separately solve by minimizing the reprojection error of each 3D features set and their corresponding facial or salient features in the next face image. The resulting head pose estimations are then combined using Kalman Filter, which allows us to take advantage of the high accuracy when using facial landmarks while enabling us to handle extreme head poses by using salient features. Results are comparable to those from the related literature, with the advantage of being robust under real world situations that might not be covered in the evaluated datasets.

1 INTRODUCTION

Monitoring the attention of human users by automatic tracking of their head pose is of fundamental importance in many areas: psychological analysis, assisting patients with disability, behavior understanding of children with autism, driver monitoring, *etc*. It is also relevant in assessing human performance in areas which warrant critical attention: factory monitoring, production pipelines, security, among others. For some of these applications, automatic head pose estimation (HPE) can be followed by gaze analysis, *e.g.*, for driver monitoring, as it can help to detect when the person is drowsy or distracted and assist the driver by alerting with an immediate warning. However, computer vision systems for HPE still face serious challenges in deployment for real world situations.

Firstly, any HPE system has to work for a wide range of human users, without any requirement for an additional calibration step. Secondly, it needs to use simple hardware that does not consume a lot of power. This currently rules out computationally intensive models that require GPUs. Thirdly, the system has to be extremely fast in order to assess the situation in real-time. Fourthly, it should be able to launch the estimation task without pose constraints, *e.g.*, that the person should be facing the camera. Finally, the system has to work for a wide range of head pose orientations, face occlusions, and external lighting conditions. Hence, a fast and efficient HPE system remains an active area of research.

In the recent past, several HPE approaches have been proposed using RGB to depth data, as well as thermal IR cameras, in monocular and stereo setups (Baker et al., 2004; Borghi et al., 2017; Cheng et al., 2007; Guo et al., 2006; Murphy-Chutorian and Trivedi, 2010; Zhu and Fujimura, 2004). Apart from estimating 3D head pose, some systems are also able to track human facial expressions in real-time. However, many of these approaches still suffer from limitations such as sensitivity to illumination changes, increasing costs, or difficulty to integrate into a real scenario. The latter is the case of depth cameras, that has also the downside of not being suitable for gaze estimation. Keeping that in mind, we focus our investigation on HPE from intensity (monochrome)

*This work is funded by the National Research Fund, Luxembourg. Project ID 9235599.

images, where gaze could also be recovered in a separate follow-up project.

We present a novel HPE approach which integrates a tracking-by-detection scheme using facial landmarks, with a salient-features tracking method. Facial landmarks are detected using a machine learning system, that effectively provides the corresponding 2D pixel locations of these points. However, these locations may be inaccurate or missing due to certain head poses or (self-)occlusions. To address these cases, we introduce a scheme that detects and tracks salient features on the acquired video frame sequences. These features can be person-specific facial birthmarks, wrinkles, hair or accessories that can provide alternative source of information for head pose estimation.

More in detail, we extract two sets of 2D features, one of 2D facial landmarks and one of salient features on the detected face. The two sets are projected onto a simple geometric 3D model through ray-tracing, and thus hypothesized in 3D space. Then, the respective 2D correspondences are computed on the next frame, running face alignment for the first set and optical flow for the second. Consequently, two head poses are estimated, by minimizing the image reprojection error with the 3D-2D correspondences, each with six degrees of freedom (DoF). Finally, we combine both poses using a dedicated Kalman filter.

The main contributions of our work are:

- A novel framework for HPE combining the relative strengths of facial landmark detection and salient feature tracking, through fusion with a dedicated Kalman filter.
- A novel approach for refined initialization of the head pose when it is different to frontal, using the 2D facial landmarks detected in the input image and a set of 3D facial landmarks, when the pose is frontal.

2 RELATED WORK

HPE is an extended researched topic in computer vision, spanning in the literature approaches based on diverse technologies. Video-based methods are comprised in the survey conducted in (Murphy-Chutorian and Trivedi, 2009). New approaches based on depth data have been proposed in (Fanelli et al., 2011; Fanelli et al., 2013; Meyer et al., 2015; Papazov et al., 2015; Borghi et al., 2017), as well as combined methods using both RGB and depth data (Baltrušaitis et al., 2012; Saeed and Al-Hamadi, 2015), or IR and depth data (Schwarz et al., 2017).

Following the categorization used in (Borghi et al., 2017), we classify the HPE approaches in three groups: model-based, appearance-based and 3D head model registration approaches. This classification should not be considered absolute, as there might be methods that fall in more than one category.

Model-based approaches. They utilize prior information regarding the geometry of the head, including detection of facial landmarks and the use of rigid or non-rigid face models. Feature-based HPE me-thods that rely on facial landmarks, *e.g.*, nose tip or eyes corners, need them to be detected in every frame and can be sensitive to extreme head poses, partial occlusions, facial expressions and low resolution images. (La Cascia et al., 2000) presented a method for 3D HPE using a cylindrical head model, based on registration of texture map images. (Kumano et al., 2009) presented an approach which used a face model given by a variable-intensity template with a particle filter, to estimate the pose and facial expression simultaneously. (Jang and Kanade, 2008; Jang and Kanade, 2010) proposed an user-specific head tracking framework using a cylinder head model (CHM). The estimated motion and a pose retrieved from a dataset of SIFT feature points were combined into a pipeline with Kalman Filter. (Choi and Kim, 2008) introduced a framework using templates, combining a particle filter with an ellipsoidal head model (EHM). The 3D motion parameters were initialized using active appearance model (AAM) and an online appearance model was used as the observation model. (Sung et al., 2008) presented a pipeline combining the AAM with a CHM. (An and Chung, 2008) used an EHM for HPE and face recognition. The pose estimation was formulated as a linear system assuming a rigid body motion under perspective projection. (Valenti et al., 2009; Valenti et al., 2012) introduced a pipeline for HPE with CHM and eye tracking. They were calculated and updated based upon a crossed feedback mechanism, which compensated the estimated values and allowed to re-initialize the head pose tracker. (Asteriadis et al., 2010) performed the estimation task using a facial-feature tracker with Distance Vector Fields (DVFs). (Kun et al., 2013) estimated 2D salient points and their location on the 3D space with an EHM. The face was detected in each frame, in order to set the area for feature extraction and correspondences estimation. POSIT and the Perspective-n-Point problem were employed to estimate the pose from the 3D-2D correspondences. (Prasad and Aravind, 2010) implemented a parametrized 3D face mask to model the head and extract feature points using SIFT. Similarly to the previous method, the pose was estimated with POSIT from the 3D-2D corre-

spondences. (Diaz Barros et al., 2017), random feature points were employed to estimate the pose. The features were tracked using optical flow and the respective 3D points were computed using a geometric model. The pose was recovered by minimizing the reprojection error of the 3D features and the 2D correspondences. (Vicente et al., 2015) proposed an approach oriented to driver's activities monitoring. Facial landmarks were detected and tracked using a facial alignment scheme with parameterized appearance models (PAMs). The head pose was estimated by minimizing the reprojection error, in this case from a 3D deformable head model and the tracked facial landmarks. (Yin and Yang, 2016) introduced a real-time HPE method optimized for on-board computers. The face was detected using pixel intensity binary test, while pose regression along with local binary feature were utilized for the alignment. The pose is retrieved by solving the 2D-3D correspondences using a rigid face model.

Appearance-based approaches. These methods use machine learning techniques to estimate the pose, based on visual features of the face appearance. The methods are robust to large head rotation, but generally the output comes from a classifier which uses discrete head poses for training and thus assigns the pose to a specific range, instead of continuous estimation. These methods usually perform better for low-resolution face images (Ahn et al., 2014; Drouard et al., 2015). (Wang et al., 2012) introduced a head tracking approach using invariant keypoints. A learning scheme was implemented, combining simulation techniques with normalization. In (Fanelli et al., 2011), the head pose as well as some facial features were estimated with random regression forests from depth data. A voting scheme was implemented, where patches from different parts of the depth face image were used to recover the pose. A large dataset with annotated data was necessary for training. (Ahn et al., 2014) proposed a deep-learning-based approach for HPE from RGB images. A particle filter was included to refine and increase stability on the estimated pose. (Liu et al., 2016) presented a pipeline for head pose estimation from RGB images using convolutional neural networks, where HPE was formulated as a regression problem. The network was trained using a large synthetic dataset obtained from rendered 3D head models. Both (Ahn et al., 2014) and (Liu et al., 2016) required a GPU to achieve real-time performance. (Borghi et al., 2017) introduced a real time deep learning approach for upper-body and head pose estimation from depth images. The proposed regression neural network, POSEidon, integrated depth with motion features and appearance. On the other hand, (Schwarz et al., 2017) presented a deep learning method for HPE under driving conditions, by fusing IR and depth data with cross-stitch units.

3D head model registration approaches. They register a 3D head model with the measured input data. (Ghiass et al., 2015) presented an approach for HPE from RGB-D data. The pose estimation was performed through a fitting process employing a 3D morphable model. (Papazov et al., 2015) introduced a method for HPE from depth data using triangular surface patch (TSP) descriptors. The input head data is divided in several patches and each one is matched with TSP descriptors from synthetic head models stored in a database. A voting scheme is then implemented from the patches to recover the head pose. (Jeni et al., 2017) presented an approach for 3D registration of a dense face mesh from 2D images through a cascade regression framework. This framework was trained using a database of high-resolution 3D face scans. The head pose and the 3D shape were iteratively refined, by registering a part-based deformable 3D model on 2D facial landmarks.

Other methods formulate an optimization problem to estimate the pose. For instance, (Morency et al., 2008) proposed a probabilistic scheme, GAVAM: Generalized Adaptive View-based Appearance Model, using an EHM. Pose was extracted by solving a linear system with normal flow constraint (NFC). (Baltrušaitis et al., 2012) proposed an extension of GAVAM, by combining a head pose tracker with a 3D constrained local model. The framework used both depth data and intensity information to track facial feature points, but it was also tested using only intensity images. (Saragih et al., 2011) presented a method for HPE by fitting a deformable model, using an optimization strategy through a non-parametric representation of the likelihood maps of landmarks locations. In (Drouard et al., 2015), a Gaussian mixture of locally-linear mapping model is used to map HOG features extracted on a face region to 3D head poses. The approach was suitable for HPE from low-resolution data.

We propose a model-based HPE approach using only intensity images, where facial landmarks are extracted in each frame and salient features on the area of the face are tracked through the entire video sequence. Thus, we extend the working range of facial-landmarks-based approaches to extreme head poses, being able to handle (self-)occlusions. Moreover, our method does not require the person to perform an initialization step, *e.g.*, face the camera frontally, an important constraint in several applications.

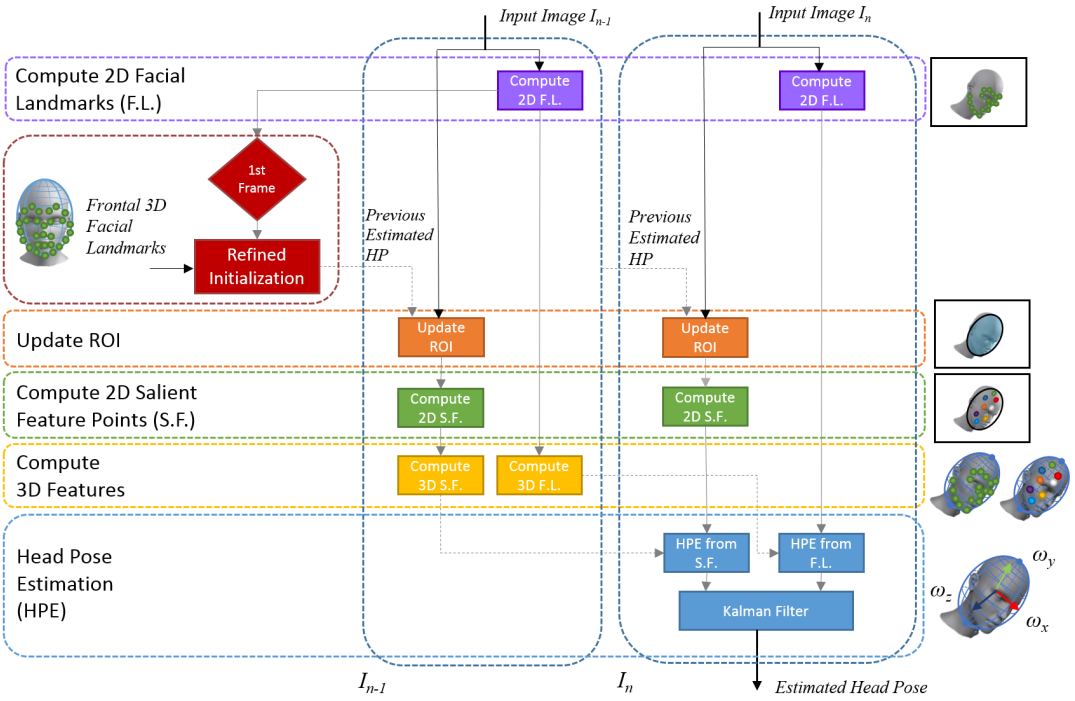

Figure 1: Pipeline of the proposed HPE approach.

3 PROPOSED HPE FRAMEWORK

HPE approaches based on facial landmarks seem to reach high HPE performances (Baltrušaitis et al., 2012; Jeni et al., 2017), as reported in Section 4. However, some of these approaches fail as soon as the face is partially or totally occluded. On the other side and despite the cumulated error along frames, the use of salient features (Diaz Barros et al., 2017) seems to be an appropriate solution in case the face detector fails. We thus propose a new framework for HPE (see Figure 1) that combines the use of both facial landmarks (purple box in Figure 1) and salient features (green box). By doing so, we take advantage of the HPE accuracy given by the use of facial landmarks while being able to handle extreme head poses, in which the face might be partially or totally occluded. Each set of feature points are back projected onto a simple 3D geometric shape, either an EHM or a CHM (yellow box in Figure 1). Moreover, we compute the 2D correspondences for each set of features in the next frame.

We formulate the HPE as a Perspective-n-Point problem, that we separately solve by minimizing the reprojection error between each set of features, facial landmarks and salient features, with their corresponding locations in the 3D space. The final HPE results from the combination of the resulting head poses using a dedicated Kalman Filter (blue box in Figure 1).

Figure 2: Facial landmarks aligned to a detected face.

3.1 Robust Facial Landmarks

Facial landmarks $\mathbf{p}_{FL}$ result from the alignment of facial features (Kazemi and Sullivan, 2014) on a detected face image. Only those facial landmarks that are less prone to be affected by facial expressions or self-occlusion are considered in the set of facial landmarks, *i.e.*, eye corners, nostrils, and nose tip. The reader is free to use any available face detector method, being the Viola and Jones face detector (Viola and Jones, 2001) one of the most commonly employed, *e.g.*, in (Baltrušaitis et al., 2012; Ghiass et al., 2015; Diaz Barros et al., 2017). With regards to the alignement of the facial landmarks, we propound to use regression trees as in (Kazemi and Sullivan,

2014), from a sparse subset of intensity values indexed to an initial estimate of the shape. By doing so, we are able to retrieve in real-time the set of facial landmarks displayed in Figure 2, as long as the face is detected. However, a problem might arise for extreme head rotations or (self-)occlusions, since the pose cannot be retrieved in case the facial landmarks are not available.

3.2 Robust Salient Features

Salient features $\mathbf{p}_{SF}$ are the result of a machine learning-based corner-detection algorithm on the region of interest (ROI), corresponding to the bounding box enclosing the human head. From our experience, FAST (Rosten et al., 2010) seems to be an appropriate approach since, aside from being very well suited for real-time applications, it provides accurate enough corners. Similarly to (Diaz Barros et al., 2017), we also weight each feature according to its location within the ROI. That is, salient features further away from the ROI border are treated as more reliable and with greater weight, enabling us to reject outliers and/or non-reliable features. This feature weighted scheme results from applying the distance transform onto the ROI, *i.e.*, each pixel within the ROI has a normalized weight related to the Euclidean distance to the closest boundary.

3.3 System Initialization

More recent HPE estimation approaches are based on adaptive 3D face models (Baltrušaitis et al., 2012; Meyer et al., 2015) that despite they might provide a better performance, they are more computationally and hardware demanding. We propose instead to use a simple geometric head model such as a cylindrical (CHM) or an ellipsoidal (EHM) head model, avoiding the use of dedicated hardware devices, *e.g.*, GPUs, while still having a very decent performance. Since the EHM has a better representation of the human head than the CHM, in the following we will only refer to the EHM as geometric head model.

In order to initialize the suggested EHM, we need first to detect the facial landmarks described in Section 3.1. Indeed, the EHM dimensions are linked to the width and height of the detected face.

The second step corresponds to the initialization of the 3D geometric model using the aligned facial landmarks. This model is scaled with regard to the detected face. The scaling and the initial location of the head in the 3D space are based on the assumption that the distance between the eyes is 60 mm. This value is an approximative distance obtained from anthropometric measurements, which average the mean interpupillary distance δ_{mm} to 64.7 mm for men and 62.3 mm for women (Dodgson, 2004; Gordon et al., 1989). From the detected facial landmarks, we extract the midpoint of the outline of each eye and measure the interpupillary distance in pixels δ_{px} between them. We then compute the distance Z_{eyes} from the optical center of the camera $\mathbf{C}$ to the baseline of the eyes, which is given by:

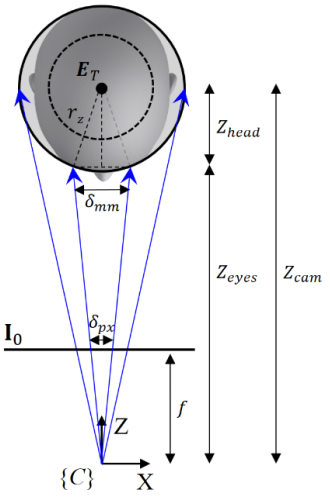

Figure 3: EHM Depth Estimation.

$$Z_{eyes} = f \cdot \frac{\delta_{mm}}{\delta_{px}}, \tag{1}$$

with f being the focal length of the camera (see Figure 3).

The distance from $\mathbf{C}$ to the vertical axis of the ellipsoid results from the sum of Z_{eyes} and the distance from the eyes' baseline to the axis of the ellipsoid Z_{head}, *i.e.*, $Z_{cam} = Z_{eyes} + Z_{head}$ (Figure 3). Z_{head} corresponds to:

$$Z_{head} = \sqrt{r_z{}^2 - (\delta_{mm}/2)^2}, \tag{2}$$

where r_z is the radius of the ellipsoid in the Z axis.

The 2D bounding box of the face, enclosed by points $\{\mathbf{p}_{TL}, \mathbf{p}_{TR}, \mathbf{p}_{BL}, \mathbf{p}_{BR}\}$, is computed from the outer facial landmarks. We use this box to calculate the width and height of the EHM. The radii r_x and r_z are set equal to half of the width of the detected face, and r_y to half of the height of the face, having as a result a prolate ellipsoid, or spheroid, as shown in Figure 4. That is,

$$r_x = r_z = \frac{1}{2}|\mathbf{p}_{TR} - \mathbf{p}_{TL}| \cdot \frac{\delta_{mm}}{\delta_{px}}, \tag{3}$$

where $\mathbf{p}_{TR}$ and $\mathbf{p}_{TL}$ correspond to the top right and top left corners of the bounding box. On the other hand,

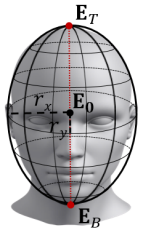

Figure 4: Ellipsoidal head model.

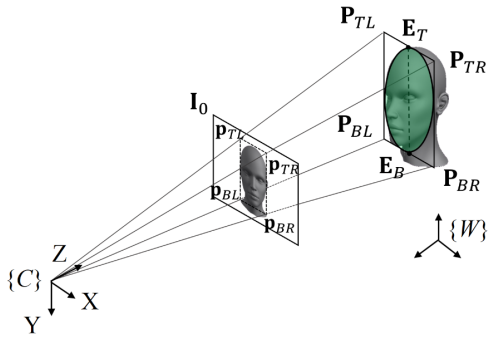

Figure 5: Initialization of EHM.

r_y is given by:

$$r_y = \frac{1}{2}|\mathbf{p}_{TR} - \mathbf{p}_{BR}| \cdot \frac{\delta_{mm}}{\delta_{px}} \tag{4}$$

with $\mathbf{p}_{BR}$ being the bottom right corner of the bounding box.

The initialization of the EHM is depicted in Figure 5.

As can be observed, the estimation of the head pose is given by its location ($\mathbf{t} = [t_x, t_y, t_z]$), and its orientation ($\omega = [\omega_x, \omega_y, \omega_z]$) with respect to the X, Y and Z perpendicular axes of a known coordinate system. The rotation angles, ω_x, ω_y, and ω_z are commonly termed as pitch, yaw, and roll angles.

In our framework, HPE is formulated as a Perspective-n-Point problem, using features computed in each frame. Therefore, the calibration of the camera is required in advance.

3.4 Initial Pose Estimation

Model-based and appearance-based HPE approaches in which the face must be detected require the user to be facing the camera during the initialization stage. However, in any real scenario this constraint results in a strong limitation, especially for assistive technologies that need to compute the head pose on the fly.

We instead propose a novel scheme to provide an approximate initialization of the head pose as soon as the head is detected and the facial landmarks are visible. The scheme utilizes the 2D facial landmarks retrieved in the first frame along with a prior set of the corresponding 3D facial landmarks on a geometric model when the face is frontal (red box in Figure 1). This set is initially normalized and later re-scaled according to the detected face in the 3D space.

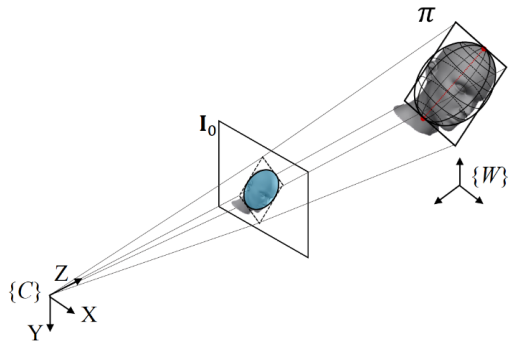

Figure 6: Projection of EHM on the image plane.

Similarly to our approach for pose estimation, we estimate the head pose from the 3D-2D correspondences, by formulating the system as a Perspective-n-Point problem and minimizing the reprojection error in the image plane.

3.5 ROI Update

As depicted in Figure 1, one of the cornerstone of our HPE framework is the use of salient features, which extend the working range of our approach to extreme head poses. These features are extracted only in a region of interest (ROI) in the input image, that corresponds to the head of the user.

The ROI is calculated from the projection of the 3D geometric model onto the 2D image plane, as shown in Figure 6, and it is updated in each frame as soon as a new head pose is recovered (orange box in Figure 1). To do so, we first find the plane parallel both to the axis of the ellipsoid and to the X-axis of the image plane, and which cuts the EHM in two equal parts. Secondly, we extract the surface that results from the intersection of the computed plane and the ellipsoid. Finally, we project the surface onto the image plane, assuming a perspective camera model. For the EHM, the projection corresponds to an elliptical ROI that covers the area of the face.

3.6 Feature Detection and Tracking

The backbone of our HPE approach is given by the detection of facial landmarks and the tracking of salient features on the ROI of the face. Therefore,

we divide the 2D feature estimation step in two main tasks, detailed below.

In the case of facial landmarks, we proceed similarly to the initialization stage, where each facial landmark $\mathbf{p}_{FL}$ is aligned on the detected face using (Kazemi and Sullivan, 2014), as described in 3.1. With this method, we can find the 2D facial landmarks correspondences in each pair of frames, as long as the face is detected.

We compensate the limitation of facial-landmarks-based approaches, by integrating a set of salient features which can track the pose of the head, even when the facial landmarks are not available. The extraction of these features are detailed in 3.2. In order to track the salient features and find the correspondences, the iterative Lucas-Kanade feature tracker with pyramids (Bouguet, 2001) is used.

3.7 Sparse 3D Reconstruction

In the following, we describe the process to compute the 3D feature points $\mathbf{P}_i$ from each set of 2D features: facial landmarks $\mathbf{p}_{FL}$ and salient feature points $\mathbf{p}_{SF}$, by means of ray tracing. To do so, we compute the intersection point on the ellipsoidal model from the ray that start at the optical center of the camera $\mathbf{C}$ and pass through each 2D feature $\mathbf{p}$ in the image plane. This line is given by $\mathbf{P} = \mathbf{C} + k\mathbf{V}$, where $\mathbf{V}$ is a vector parallel to the ray. k is the scalar parameter obtained from the quadratic equation of the ellipsoid, given by:

$$|\mathbf{W}|^2 k^2 + 2(\mathbf{W} \bullet \mathbf{X})k + |\mathbf{X}|^2 - 1 = 0 \qquad (5)$$

with $\mathbf{W} = \mathbf{M}\mathbf{R}^T\mathbf{V}$ and $\mathbf{X} = \mathbf{M}\mathbf{R}^T(\mathbf{C} - \mathbf{E_0})$. $\mathbf{M}$ is the diagonal matrix of the inverses of the ellipsoid radii $\{\frac{1}{r_x}, \frac{1}{r_y}, \frac{1}{r_z}\}$, $\mathbf{R}$ is the rotation matrix and $\mathbf{E_0}$ is the center of the ellipsoid.

As a result, we obtain two set of 3D features, one of 3D facial landmarks $\mathbf{P}_{FL}$ and the other of 3D salient features $\mathbf{P}_{SF}$.

3.8 HPE and Measurement Fusion

Similarly to the initial HPE process described in 3.4, we estimate the pose after formulating a Perspective-n-Point problem from the 3D-2D correspondences and solve by minimizing the reprojection error on the image plane, using the global Levenberg-Marquardt algorithm. To define the 3D-2D correspondences, we use the set of 3D features $\mathbf{P}_i$ that were calculated from the previous frame and the 2D features obtained at the current frame. This procedure is applied using both set of features, salient features $\{\mathbf{P}_{SF}, \mathbf{p}_{SF}\}$ and facial landmarks $\{\mathbf{P}_{FL}, \mathbf{p}_{FL}\}$. Consequently, we obtain two measured head poses: ω_{SF} and $\mathbf{t}_{SF}$ from the salient features and ω_{FL} and $\mathbf{t}_{FL}$ from the facial landmarks.

We introduce a method for combining both measurements using a dedicated Linear Kalman Filter. Although Kalman Filter is a very common approach in sensor fusion, it is the first time, to the best of our knowledge, that it is used for fusing measurements given by facial landmarks and by salient features. As long as we have two different head pose measurements from the two sets of feature points, it is possible to apply this framework. In general, Kalman Filter is defined on the state-space representation of the system, where the dynamical system is given by the following state equation:

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{w}_{k-1}. \qquad (6)$$

$\mathbf{x}_k$ represents the state of the system at time k, $\mathbf{u}_{k-1}$ the deterministic inputs to the system at time $k-1$ and $\mathbf{w}_{k-1}$ is the random noise affecting the system at time $k-1$, commonly termed as process noise.

The measurement equation is defined as:

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k, \qquad (7)$$

where $\mathbf{v}_k$ corresponds to the measurement noise, *i.e.*, the random noise in the observation $\mathbf{z}_k$.

$\mathbf{w}_k$ and $\mathbf{v}_k$ are assumed to be independent, white and having Gaussian zero mean, with covariance matrices given by Eq. 8 and 9.

$$E[\mathbf{w}_n \mathbf{w}_k{}^T] = \begin{cases} Q_k, & \text{for } n = k \\ 0, & \text{otherwise} \end{cases} \qquad (8)$$

$$E[\mathbf{v}_n \mathbf{v}_k{}^T] = \begin{cases} R_k, & \text{for } n = k \\ 0, & \text{otherwise} \end{cases} \qquad (9)$$

In our framework, the state vector $\mathbf{x}$ contains the position $\mathbf{t}$ and orientation ω of the head, with their respective first and second derivatives, *i.e.*, the velocity and acceleration. The measurement (or observation) vector $\mathbf{z}_k$ is composed of both estimated poses. The observation model is given by:

$$\mathbf{H} = \left[\mathbf{H}_{SF}^T, \mathbf{H}_{FL}^T\right]^T \qquad (10)$$

being $\mathbf{H}_{SF}$ and $\mathbf{H}_{FL}$ the respective observation models from the salient features and the facial landmarks.

The covariance matrix of the observation noise is then defined by:

$$\mathbf{R} = \begin{bmatrix} \Sigma_{SF} & 0 \\ 0 & \Sigma_{FL} \end{bmatrix} \qquad (11)$$

It is important to note that facial landmarks are not always detected in every frame. In order to address this problem, we propose to include a dynamic

Kalman Filter framework that corrects the state vector from the available measurements. In case the face is not detected, the Kalman Filter is updated using only the pose estimated from the salient features, that is:

$$\mathbf{H} = \mathbf{H}_{SF} \tag{12}$$

and the observation noise covariance as:

$$\mathbf{R} = \sum_{SF}. \tag{13}$$

Otherwise, the fusion scheme defined by Eq. 10 and 11 is employed. From this framework, we are able to update the filter and estimate the new pose having either one or two measurements.

4 EVALUATION AND RESULTS

In this section, we describe the methodology implemented to evaluate our HPE approach and the analysis of the results in comparison to other methods of the state of the art.

4.1 Evaluation and Experiments

We evaluated our framework using the Boston University dataset with uniform illumination proposed by (La Cascia et al., 2000). This dataset comprises a set of 45 videos recorded with a RGB camera, from 5 different persons on an office setup. The ground truth was acquired using a Flock of Birds magnetic sensor, with nominal accuracy of 1.8 mm in translation and 0.5 degrees in rotation. With this dataset, we also investigated the effect of combining facial landmarks with salient features using two other pipelines than the fusion-based framework described in Subsection 3.8. These two pipelines are described below.

Integrating feature points in common pipeline: For this first method, we integrated the set of 2D salient features $\mathbf{p}_{SF}$ along with the set of 2D facial landmarks $\mathbf{p}_{FL}$, to create one combined set of 2D features points $\mathbf{p}_{SF+FL}$. The set of 3D feature points $\mathbf{P}_{SF+FL}$, composed as well of 3D salient features $\mathbf{P}_{SF}$ and 3D facial landmarks $\mathbf{P}_{FL}$, were computed as described in Subsection 3.7, by using ray tracing. The head pose was also retrieved by finding the 3D-2D correspondences and minimizing the reprojection error from the combined set of features, $\{\mathbf{P}_{SF+FL}, \mathbf{p}_{SF+FL}\}$. Similarly to the fusion approach, the face ROI was updated as detailed in 3.5. This method is listed in the results as *Integration SF+FL*.

Integrating feature points and adding Kalman Filter: For the second pipeline (*Integration with KF*), we proceeded similarly to the previous method, *Integration SF+FL*, but we included an extra step to incorporate a Kalman Filter to update the state vector using the pose obtained by minimizing the reprojection error in the image plane.

Table 2: Averaged runtime for 100 launches.

Process	Time in ms
Face detection & Init. EHM	31.61
Initial HPE	17.87
Runtime for first frame	49.48
ROI Update	0.49
Feature detection and tracking	23.84
3D Reconstruction	0.27
Head pose estimation	18.6
Runtime of other frames	43.2

4.2 Results

Three different metrics, the root mean square errors (RMSE), standard deviation (STD) and mean absolute error (MAE), were used to compare our results with alternatives methods of the state of the art. Not only we compare the approach proposed in Section 3, but also the two other pipelines described in Subsection 4.1. The results are reported in Table 1.

As can be noted from Table 1, our current results are comparable to the state of the art approaches. Although the low MAE from (Baltrušaitis et al., 2012), this method is not suitable for real-time applications. The same applies to methods proposed in (Jang and Kanade, 2010; Prasad and Aravind, 2010). (Jeni et al., 2017) on the other hand, can perform on real time, but a large dataset of high-resolution 3D face scans are required for training (the method used around 300,000).

4.3 Runtime Analysis

For many applications, it is of great importance the approach performs in real time. We measured the average time to execute each step of the proposed pipeline using a CPU with an Intel Core(TM) i5-4210U processor. The results are presented in Table 2.

From Table 2 we can observe the average runtime of the proposed framework. At the first frame, it requires around 50 ms to detect the face and provide an initial estimation of the head pose using only facial landmarks. For the rest of the frames, the pipeline is also suitable for real time applications, as it can run at 23fps. This is an important advantage with respect to methods of the state of the art that do not perform in real time, as (Ahn et al., 2014; Prasad and Aravind, 2010; Wang et al., 2012) (see last column of Table 1).

Table 1: Results. Comparison of the RMSE, STD and MAE errors with other methods of the SoA using the BU dataset.

Method	RMSE ± STD			MAE			Time
	Roll	Pitch	Yaw	Roll	Pitch	Yaw	(FPS)
(Sung et al., 2008)	**3.1**	5.6	5.4	-	-	-	-
(Morency et al., 2008)	-	-	-	2.91	3.67	4.97	6
(An and Chung, 2008)	-	-	-				-
Ellipsoid	-	-	-	2.83	3.95	3.94	-
Cylinder	-	-	-	3.22	7.22	5.33	-
Plane	-	-	-	2.99	7.32	18.08	-
(Choi and Kim, 2008)	-	-	-				
Ellipsoid	-	-	-	2.82	3.92	4.04	14
Cylinder	-	-	-	2.45	4.43	5.19	14
(Jang and Kanade, 2008)	-	-	-	2.1	3.7	4.6	-
(Kumano et al., 2009)	-	-	-	2.9	4.2	7.1	-
(Jang and Kanade, 2010)	-	-	-	**2.07**	3.44	4.22	-
(Prasad and Aravind, 2010)	-	-	-	3.6	**2.5**	3.8	-
(Asteriadis et al., 2010)	3.56	4.89	5.72	-	-	-	-
(Saragih et al., 2011)	-	-	-	2.55	4.46	5.23	8
(Valenti et al., 2012)							
Fixed template with eye cues	**3.00± 2.82**	5.26± 4.67	6.10± 5.79	-	-	-	-
Fixed template w/o eye cues	3.85±3.43	6.00±5.21	8.07±7.37	-	-	-	-
Updated template with eye cues	3.93±3.57	5.57±4.56	6.45±5.72	-	-	-	-
Updated template w/o eye cues	4.15±3.72	5.97±4.87	6.40±5.49	-	-	-	-
(Wang et al., 2012)	-	-	-	**1.86**	2.69	**3.75**	15
(Baltrušaitis et al., 2012)	-	-	-	**2.08**	3.81	**3.00**	-
(Vicente et al., 2015)	-	-	-	3.2	6.2	4.3	25
(Diaz Barros et al., 2017)							
Ellipsoid	3.36±2.98	4.46±3.84	**5.09±4.56**	2.56	3.39	3.99	56
Cylinder	3.66±3.35	5.73±4.54	6.16±5.42	2.80	4.58	4.87	42
(Jeni et al., 2017)	-	-	-	2.41	**2.66**	3.93	50
Our approach							
Integration SF+FL	3.44±3.11	4.51±3.97	5.84±4.98	2.62	3.59	4.73	
Integration with KF	3.37±3.00	**4.31±3.60**	5.30±4.70	2.56	3.29	4.13	
Fusion with dynamic KF	3.36±2.99	**4.32±3.62**	**5.25±4.70**	2.54	3.27	4.07	23

5 CONCLUSIONS

We introduced a new framework to estimate the 6 DoF head pose from 2D images, by combining the head poses estimated from facial landmarks detected in each frame and tracked salient features, through a dedicated Kalman Filter. It performs in real time, with no need for parallel computing.

We extend the application of feature-based HPE methods that rely on facial landmarks detected in every frame, to cases where the face is not detected, but the head is still visible. In this way, our approach can handle extreme pose variations and (self-) occlusion. On the other hand, we avoid the drifting which might results from cumulative error on salient-feature-based methods, by considering the head pose estimated from facial landmarks.

We also propose a method for initializing the head pose when it is different to frontal. To do so, we use the facial landmarks detected in the first frame and their corresponding 3D points on a synthetic head model that is facing the camera. We then compute the initial pose by minimizing the reprojection error from the 3D-2D correspondences.

An ellipsoidal model provides a good representation of the human head. However, it might add approximation errors when estimating the pose. For that reason, we are interested in exploring in the future the performance of the pipeline with a refined head model, that adjusts better to the shape of the head.

We have also discussed about improving the framework by replacing the linear Kalman Filter by an Extended Kalman Filter (EKF), as it provides a more accurate modelling of the 3D motion of the head, and thus a better estimation of the rotation angles.

REFERENCES

Ahn, B., Park, J., and Kweon, I. S. (2014). Real-time head orientation from a monocular camera using deep neural network. In *Asian Conference on Computer Vision (ACCV)*, pages 82–96. Springer.

An, K. H. and Chung, M. J. (2008). 3d head tracking and pose-robust 2d texture map-based face recognition using a simple ellipsoid model. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 307–312. IEEE.

Asteriadis, S., Karpouzis, K., and Kollias, S. (2010). Head pose estimation with one camera, in uncalibrated environments. In *Workshop on Eye Gaze in Intelligent Human Machine Interaction*, pages 55–62. ACM.

Baker, S., , Matthews, I., Xiao, J., Gross, R., Kanade, T., and Ishikawa, T. (2004). Real-time non-rigid driver head tracking for driver mental state estimation. In *11th World Congress on Intelligent Transportation Systems*.

Baltrušaitis, T., Robinson, P., and Morency, L.-P. (2012). 3d constrained local model for rigid and non-rigid facial tracking. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Borghi, G., Venturelli, M., Vezzani, R., and Cucchiara, R. (2017). Poseidon: Face-from-depth for driver pose estimation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Bouguet, J. Y. (2001). Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, 5:1–10.

Cheng, S. Y., Park, S., and Trivedi, M. M. (2007). Multi-spectral and multi-perspective video arrays for driver body tracking and activity analysis. *Computer Vision and Image Understanding*, 106(2):245–257.

Choi, S. and Kim, D. (2008). Robust head tracking using 3d ellipsoidal head model in particle filter. *Pattern Recognition*, 41(9):2901–2915.

Diaz Barros, J. M., Garcia, F., Mirbach, B., and Stricker, D. (2017). Real-time monocular 6-dof head pose estimation from salient 2d points. In *International Conference on Image Processing (ICIP)*. IEEE.

Dodgson, N. A. (2004). Variation and extrema of human interpupillary distance. In *Stereoscopic Displays and Virtual Reality Systems XI*, volume 5291, pages 36–46. SPIE.

Drouard, V., Ba, S., Evangelidis, G., Deleforge, A., and Horaud, R. (2015). Head pose estimation via probabilistic high-dimensional regression. In *International Conference on Image Processing (ICIP)*, pages 4624–4628. IEEE.

Fanelli, G., Dantone, M., Gall, J., Fossati, A., and Van Gool, L. (2013). Random forests for real time 3d face analysis. *International Journal of Computer Vision*, 101(3):437–458.

Fanelli, G., Gall, J., and Van Gool, L. (2011). Real time head pose estimation with random regression forests. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 617–624. IEEE.

Ghiass, R. S., Arandjelović, O., and Laurendeau, D. (2015). Highly accurate and fully automatic head pose estimation from a low quality consumer-level rgb-d sensor. In *2nd Workshop on Computational Models of Social Interactions: Human-Computer-Media Communication*, pages 25–34. ACM.

Gordon, C. C., Bradtmiller, B., Clauser, C. E., Churchill, T., McConville, J. T., Tebbetts, I., and Walker, R. A. (1989). Anthropometric survey of u.s. army personnel: Methods and summary statistics. In *Technical report 89-044. Natick MA: U.S. Army Natick Research, Development and Engineering Center.*

Guo, Z., Liu, H., Wang, Q., and Yang, J. (2006). A fast algorithm face detection and head pose estimation for driver assistant system. In *8th International Conference on Signal Processing*, volume 3. IEEE.

Jang, J. S. and Kanade, T. (2008). Robust 3d head tracking by online feature registration. In *8th International Conference on Automatic Face & Gesture Recognition (FG'08)*. IEEE.

Jang, J. S. and Kanade, T. (2010). Robust 3d head tracking by view-based feature point registration. Technical report, People Image Analysis (PIA) Consortium, Carnegie Mellon University.

Jeni, L. A., Cohn, J. F., and Kanade, T. (2017). Dense 3d face alignment from 2d video for real-time use. *Image and Vision Computing*, 58:13–24.

Kazemi, V. and Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1867–1874. IEEE.

Kumano, S., Otsuka, K., Yamato, J., Maeda, E., and Sato, Y. (2009). Pose-invariant facial expression recognition using variable-intensity templates. *International Journal of Computer Vision*, 83(2):178–194.

Kun, J., Bok-Suk, S., and Reinhard, K. (2013). Novel back-projection method for monocular head pose estimation. *International Journal of Fuzzy Logic and Intelligent Systems*, 13(1):50–58.

La Cascia, M., Sclaroff, S., and Athitsos, V. (2000). Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models. *Transactions on Pattern Analysis and Machine Intelligence*, 22(4):322–336.

Liu, X., Liang, W., Wang, Y., Li, S., and Pei, M. (2016). 3d head pose estimation with convolutional neural network trained on synthetic images. In *International Conference on Image Processing (ICIP)*, pages 1289–1293. IEEE.

Meyer, G. P., Gupta, S., Frosio, I., Reddy, D., and Kautz, J. (2015). Robust model-based 3d head pose estimation. In *International Conference on Computer Vision (ICCV)*, pages 3649–3657. IEEE.

Morency, L., Whitehill, J., and Movellan, J. (2008). Generalized adaptive view-based appearance model: Integrated framework for monocular head pose estimation. In *8th International Conference on Automatic Face & Gesture Recognition (FG'08)*, pages 1–8. IEEE.

Murphy-Chutorian, E. and Trivedi, M. M. (2009). Head pose estimation in computer vision: A survey. *Transactions on Pattern Analysis and Machine Intelligence*, 31(4):607–626.

Murphy-Chutorian, E. and Trivedi, M. M. (2010). Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness. *Transactions on Intelligent Transportation Systems*, 11(2):300–311.

Papazov, C., Marks, T. K., and Jones, M. (2015). Real-time 3d head pose and facial landmark estimation from depth images using triangular surface patch features. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Prasad, B. H. and Aravind, R. (2010). A robust head pose estimation system for uncalibrated monocular videos. In *7th Indian Conference on Computer Vision, Graphics and Image Processing*, pages 162–169. ACM.

Rosten, E., Porter, R., and Drummond, T. (2010). Faster and better: A machine learning approach to corner detection. *Transactions on Pattern Analysis and Machine Intelligence*, 32:105–119.

Saeed, A. and Al-Hamadi, A. (2015). Boosted human head pose estimation using kinect camera. In *International Conference on Image Processing (ICIP)*, pages 1752–1756. IEEE.

Saragih, J. M., Lucey, S., and Cohn, J. F. (2011). Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91(2):200–215.

Schwarz, A., Haurilet, M., Martinez, M., and Stiefelhagen, R. (2017). Driveahead - a large-scale driver head pose dataset. In *International Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE.

Sung, J., Kanade, T., and Kim, D. (2008). Pose robust face tracking by combining active appearance models and cylinder head models. *International Journal of Computer Vision*, 80(2):260–274.

Valenti, R., Sebe, N., and Gevers, T. (2012). Combining head pose and eye location information for gaze estimation. *Transactions on Image Processing*, 21(2):802–815.

Valenti, R., Yucel, Z., and Gevers, T. (2009). Robustifying eye center localization by head pose cues. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 612–618. IEEE.

Vicente, F., Huang, Z., Xiong, X., De la Torre, F., Zhang, W., and Levi, D. (2015). Driver gaze tracking and eyes off the road detection system. *Transactions on Intelligent Transportation Systems*, 16(4):2014–2027.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–511. IEEE.

Wang, H., Davoine, F., Lepetit, V., Chaillou, C., and Pan, C. (2012). 3-d head tracking via invariant keypoint learning. *Transactions on Circuits and Systems for Video Technology*, 22(8):1113–1126.

Yin, C. and Yang, X. (2016). Real-time head pose estimation for driver assistance system using low-cost on-board computer. In *15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry*, volume 1, pages 43–46. ACM.

Zhu, Y. and Fujimura, K. (2004). Head pose estimation for driver monitoring. In *Intelligent Vehicles Symposium*, pages 501–506. IEEE.