FaceCept3D: Real Time 3D Face Tracking and Analysis

Sergey Tulyakov, Radu-Laurenţiu Vieriu, Nicu Sebe
University of Trento, Italy
sergey.tulyakov@unitn.it, {vieriu, sebe}@disi.unitn.it

Abstract

We present an open source cross platform technology for 3D face tracking and analysis. It contains a full stack of components for complete face understanding: detection, head pose tracking, facial expression and action units recognition. Given a depth sensor, one can combine FaceCept3D modules to fulfill a specific application scenario. Key advantages of the technology include real time processing speed and ability to handle extreme head pose variations. Possible application areas of the technology range from human computer interaction to active aging platform, where precise and real-time analysis is required. The technology is available for scientific community.

1. Introduction

Over the past years, there has been an increasing interest in technologies aimed at supporting or enhancing people's lives (especially elderly class) in various environments, such as shopping malls, museums or at home [1, 2]. Understanding the affective state of these subjects offers important clues in decoding their state of mind, useful in monitoring tasks. In addition, many studies require estimates of the direction and level of attention for modeling different types of interactions. In such cases, the head pose estimation becomes a valuable proxy.

There is one important constraint all these scenarios share when looking for solving the above mentioned tasks: *non-invasiveness*, *i.e.* the solution must not hinder the naturalness of the subject's behavior. As a consequence, the vision sensors are typically placed out of the direct sight of the subject. FaceCept3D is motivated by challenges arising from these types of scenarios and is able to successfully address them in a unified, open source and cross-platform solution. Additionally, our system can be deployed in a much broader spectrum of applications (*e.g.* those cases for which the face is fully visible to the sensor), being able to maintain state-of-the-art performance, as shown in [8]. The technology is available on GitHub[1].

[1] https://github.com/sergeytulyakov/FaceCept3D

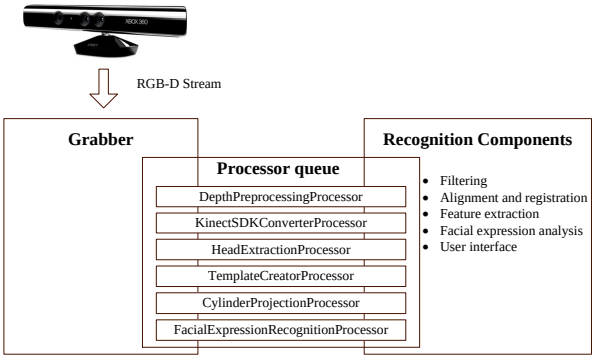

Figure 1. A pipeline for tracking the head pose and recognizing facial expressions. Processors are executed by the grabber one after another.

2. Modular architecture

FaceCept3D is a set of independent modules. All modules are split into three major parts:

- **Recognition modules** include filtering, registration, feature extraction, machine learning methods and other components.
- **Pipeline modules**, that encapsulate underlined platform and sensor-specific technical details.
- **User interfaces modules**, that enable viewing, annotating and displaying the results.

Figure 1 shows a typical pipeline for an automatic head pose tracking and facial expression recognition. A sensor dependent grabber module executes a queue of processors that perform necessary actions using the recognition components.

3. Head pose tracking

In order to track a face, FaceCept3D builds offline a person-specific 3D head template for a person in front of the sensor. When the template is ready a modified version of the Iterative Closest Point (ICP) [3] method is used to register it with a scene and obtain the head pose (more details in [7]).

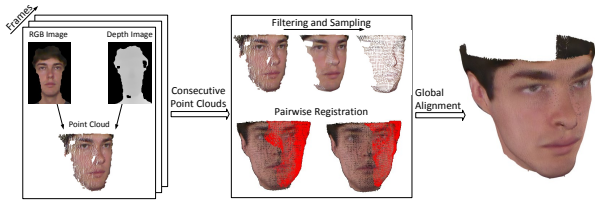

Figure 2. Person-specific template creation pipeline. Prior to creating the point cloud, we filter out noise by convolving the depth image with a Gaussian kernel. Voxel-grid algorithm on the smooth cloud is used to obtain a cloud with fewer points. Pairwise registration is performed on the consecutive clouds.

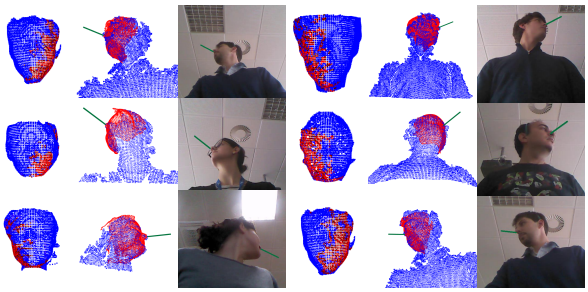

Figure 3. For every subject three images are given. The left one represents a template with the most important points marked in red. The image in the middle shows the template fitted to the point cloud. The right image shows the view from the walker. Note that for some subjects the face is almost completely hidden.

The process of person-specific template creation is outlined in Figure 2 and takes around 3 seconds on a embedded Intel processor.

Our modified version of the ICP algorithm uses history-based points weighting as described in [7] to guide the optimization procedure of ICP to a promising descend direction and reach local minima faster. Table 1 shows that our version of ICP converges almost 4 times faster. Several examples of recognized head poses are given in the Figure 3. Note the difficult viewing and head orientation correctly handled by the system.

Table 1. Comparison between history-based weighted ICP and generic ICP in computational time

	# Iterations	Fps
Generic ICP	14.64	10.05
History-based weighted ICP	**3.16**	**38.87**

3.1. Head pose invariant face representation

FaceCept3D head pose tracker returns head pose orientation in real-time. Since subjects are not constrained in head movements, many parts of the face could be self-occluded. Therefore a head pose invariant representation is required. We build such representation by constructing a cylinder around the face and projecting the face onto the cylinder. Figure 4 shows this cylindrical sampling pipeline.

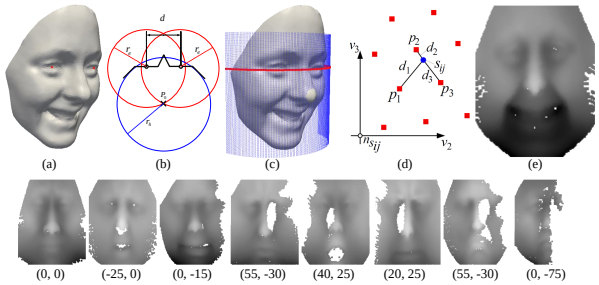

Figure 4. Top row (left to right): an example of face scan with two eyes detected. The cylindrical head model (CHM) parameters. Our CHM with 150×120 sampling points imposed on the face scan. Sampling point values computed based on the 3 nearest neighbors. An example of pose-invariant face representation. Bottom row: examples of sampled faces under varying head poses and facial expressions. The head rotation $(tilt, yaw)$ is given in the brackets.

Several examples of head pose invariant face representation are given in Figure 4 bottom row. Note how the head pose problem is transformed into a missing information problem. Nearest neighbor interpolation is the most computationally expensive step in this pipeline. In order to run it in real-time FaceCept3D has an efficient way to compute it.

4. Facial Expression and Action Unit Recognition

Once computed, the head pose invariant face representation is subject to a dense sampling procedure with overlapping patches of fixed size (see Figure 5). For each patch position, we train a separate classifier, followed by a late fusion stage for the final estimate. In the case of action unit (AU) recognition, we employ a 1-vs-all strategy for every patch. The dense sampling approach comes along with two important benefits: (i) it offers an elegant way to cope with missing information, as the *empty* patches are simply discarded at decision making stage and (ii) it is naturally suited for modeling patch importance, as different patch votes can be weighted differently (especially in the case of AU recognition).

From each face image encoding depth information (*i.e* each pixel value reflects the distance between the object and the sensor), we first compute channel representations [4], then we split the channels into overlapping patches, from which generalized Haar features are extracted. Random Forests are then used to perform patch level predictions, which in turn are aggregated for the final estimate [8].

Figure 6 shows the recognition rate distribution over the yaw/tilt space on BU-3DFE dataset [9]. The angle ranges are divided into blocks of equal size $15^\circ \times 15^\circ$ and

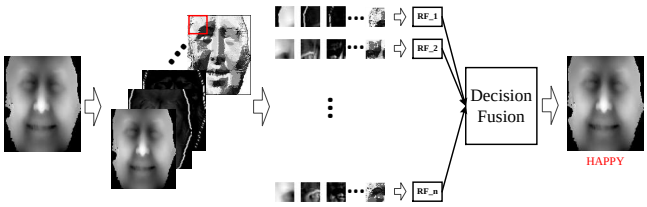

Figure 5. From head pose invariant face representation to the expression label (left to right): initial 2D face representation, channel computation, dense sampling with overlapping patches, random forest classification, decision fusion and labeled sample.

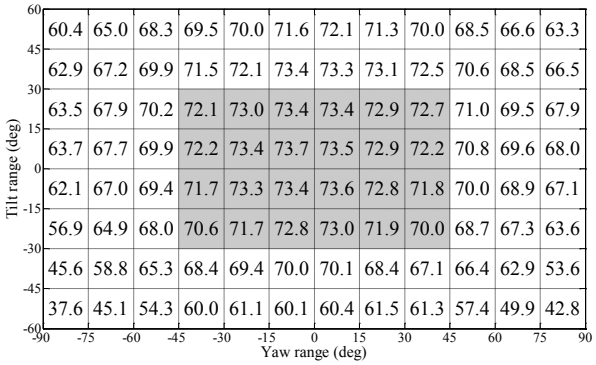

Figure 6. Recognition rate distribution over the yaw/tilt space. The gray area shows the reduced head-pose range reported in [6].

performance is computed on samples belonging to each block. The gray area corresponds to a reduced set of angles, commonly used in previous work (*e.g.* [6]). While maintaining state-of-the-art performance on the reduced set, FaceCept3D is able to extend its operating point to severe head rotation angles with only a reasonable loss in recognition accuracy.

Table 2. Action Unit recognition results obtained on BP4D

AU Index	F1 Norm	Acc Norm
1	0.46	0.60
2	0.12	0.50
4	0.36	0.56
6	0.80	0.79
7	0.73	0.70
10	0.79	0.77
12	0.82	0.81
14	0.68	0.66
15	0.33	0.56
17	0.58	0.63
23	0.43	0.60
Avg	**0.56**	**0.65**

Finally, in Table 2, we show preliminary results on AU recognition on BP4D dataset [10], following a leave-one-subject-out protocol. As a performance measure, we report the normalized F1 score with a skew factor [5], computed as $F1Norm = \frac{2sPR}{2sR+P}$, where R and P are the Recall and Precision, respectively, and s is the ratio between the number of negative samples and the number of positive ones included in the test set. In a similar manner we compute the skew-normalized accuracy, as $AccNorm = \frac{TP+TN/s}{TP+TN/s+FP/s+FN}$.

5. Conclusions

In this paper we introduce FaceCept3D, an open source cross platform system for 3D face analysis. FaceCept3D is able to accurately infer head pose, perform face frontalization and estimate facial expressions in real-time. Our system is designed to cope with a wide range of head pose variations, typically seen in applications for which non-invasiveness is a particularly important requirement.

References

[1] `http://cordis.europa.eu/project/rcn/101220_en.html`.

[2] `http://cordis.europa.eu/project/rcn/194087_en.html`.

[3] P. Besl and N. D. McKay. A method for registration of 3-D shapes. *PAMI*, 14(2):239–256, 1992.

[4] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, 2009.

[5] L. Jeni, J. F. Cohn, F. De La Torre, et al. Facing imbalanced data–recommendations for the use of performance metrics. In *ACII*, pages 245–251, 2013.

[6] O. Rudovic, M. Pantic, and I. Patras. Coupled gaussian processes for pose-invariant facial expression recognition. *PAMI*, 35(6):1357–1369, 2013.

[7] S. Tulyakov, R. L. Vieriu, S. Semeniuta, and N. Sebe. Robust Real-Time Extreme Head Pose Estimation. In *ICPR*, 2014.

[8] R.-L. Vieriu, S. Tulyakov, S. Semeniuta, E. Sangineto, and N. Sebe. Facial expression recognition under a wide range of head poses. In *FG*, 2015.

[9] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato. A 3D facial expression database for facial behavior research. In *FG*, 2006.

[10] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, and P. Liu. A high-resolution spontaneous 3d dynamic facial expression database. In *FG*, pages 1–6, 2013.