ACCURATE 3D FACE MODELING AND RECOGNITION FROM RGB-D STREAM IN THE PRESENCE OF LARGE POSE CHANGES

Donghyun Kim, Jongmoo Choi, Jatuporn Toy Leksut, Gérard Medioni

Institute for Robotics and Intelligent Systems,
University of Southern California
3737 Watt way PHE 101, Los Angeles, CA 90089, United States
{kim207, jongmooc, leksut, medioni}@usc.edu

ABSTRACT

We propose a 3D face modeling and recognition system using an RGB-D stream in the presence of large pose changes. In the previous work, all facial data points are registered with a reference to improve the accuracy of 3D face model from a low-resolution depth sequence. This registration often fails when applied to non-frontal faces. It causes inaccurate 3D face models and poor performance of matching. We address this problem by pre-aligning each input face ('frontalization') before the registration, which avoids registration failures. For each frame, our method estimates the 3D face pose, assesses the quality of data, segments the facial region, frontalizes it, and performs an accurate registration with the previous 3D model. The 3D-3D recognition system using accurate 3D models from our method outperforms other face recognition systems and shows 100% rank 1 recognition accuracy on a dataset with 30 subjects.

Index Terms— 3D Face Recognition, 3D Face Modeling

1. INTRODUCTION

We aim to provide a 3D face modeling and recognition system using a low-cost RGB-D sensor in the presence of large head pose changes. Face recognition performance using a reconstructed 3D face model from a full RGB-D video (3D-3D matching) can be better than using only a single raw or multiple depth frame [2]. However, the performance of 3D-3D matching recognition system depends on the quality of reconstructed 3D face models. Prior work has been focused on providing accurate 3D face models [1, 3, 4]. Recently, it has been demonstrated that laser scan quality face models can be reconstructed from a low-quality RGB-D sequence by aggregating multiple frames in [1]. Since the facial data are registered with respect to a reference data using a point cloud registration method in order to improve the accuracy of the 3D face model, the registration method plays the key role in the method [1].

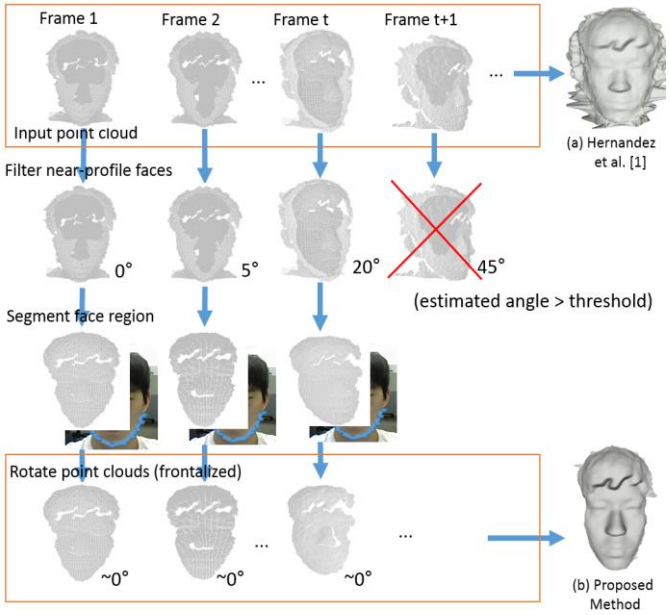


Fig. 1. Illustration of the proposed method

The iterative closest point (ICP) algorithm [5, 6] is one of the widely used registration methods [1]. ICP performs well under two conditions [6]. One is that two point clouds are close to each other (i.e. good initialization). The other condition is that there should be no outliers in the point clouds.

In natural environments, a user of the system can have large head motion (e.g., rotating one's head from -90 to 90 degrees), which may produce a bad 3D face model because ICP may converge to a local minimum due to the bad initialization [6]. Facial data with non-facial regions (e.g., a part of the neck) can also causes a sub-optimal registration result. We address these issues by rectifying input data and eliminating outliers in the source point cloud.

Given a pair of depth and RGB frames from an RGB-D sequence, we extract 2D facial landmarks from the RGB

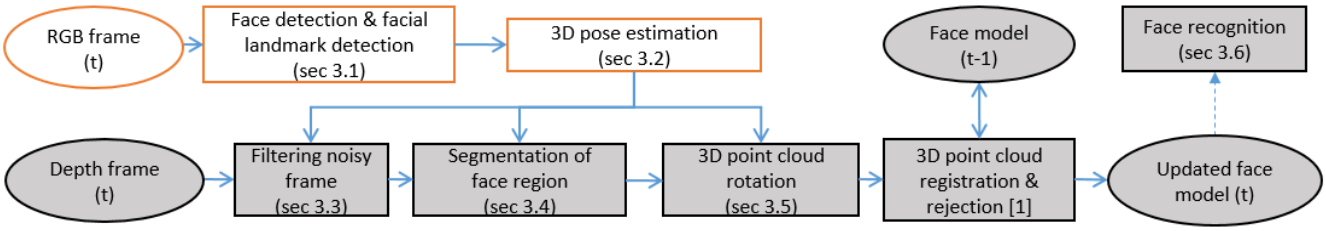


Fig. 2. Overview of the proposed system

image, estimate the 3D head pose, assess the quality of data using the pose. Using the detected facial landmarks, we segment the facial region using the boundary, which helps the registration by removing outliers such as the neck, shoulder, hair, and clothes. Then, we frontalize (i.e. repositioning to an upright, frontal face) the segmented point cloud from a rotated face in order to avoid a local minimum of the registration algorithm. The reconstructed 3D face model is used for the following 3D recognition module. Fig. 1 demonstrates our approach.

Our contributions are twofold:

• We frontalize the facial point cloud and remove outliers before the registration using the estimated 3D head pose computed from extracted 2D facial landmarks. Our method provides more accurate modeling results compared to the previous state of the art method [1].

• The experimental results using our proposed 3D modeling method on a small database show 100% rank 1 recognition accuracy in the presence of large head motion.

In the following sections, we review related work of a 3D face recognition using low-cost depth sensors, explain our methods, and present experimental results.

2. RELATED WORK

3D face modeling and recognition have been active research topics for many years [7, 8, 9, 10, 11]. In recent years, low-cost depth sensors have become readily available, such as PrimeSense camera [12] and Kinect [13]. We focus on 3D face modeling and recognition with low-cost depth sensors in real-time. There are two categories of prior proposed methods of 3D face modeling.

Model-based methods. It starts with a generic model and deforms the generic model as input frames come in [3, 4, 14]. Zollhöfer [3] proposed a method to build a 3D face model by fitting a morphable face to input data using non rigid registration with a Kinect. However the reconstructed models can be biased toward the generic face.

Data-driven methods. These approaches do not use a generic model but infer a 3D face model by integrating sequences of depth frames, such as the method proposed by Hernandez et al. [1]. Hernandez et al.'s method [1] used a cylindrical representation, segmented the face regions of every depth frame, registered and integrated the frame to the reconstructed 3D face model using the ICP algorithm. However, any errors in the ICP algorithm can lead to a poorly integrated 3D face model. Newcombe et al. [15] proposed a method that extends the KinectFusion to reconstruct dynamic moving objects and scenes in real-time, which is limited to a face with slow motion.

Face recognition. Min et al. [16] presented a 'Video-to-Image' method where each frame in the video is identified and all results are combined to determine one's identity because a single raw input from a low-cost sensor is quite noisy. A 3D-3D recognition using reconstructed 3D models has been presented in [2]. Face recognition systems under pose changes from an RGB-D stream are proposed in [17, 18]. Li et al. [17] used ICP and facial symmetry to canonicalize non-frontal faces.

We leverage the 3D-3D matching framework as in [2], but our method provides an accurate 3D face model in the presence of large pose changes while the prior work [1] requires only near frontal faces.

3. METHOD

We take depth and RGB frames from a fixed PrimeSense camera [12] as inputs to the system, reconstruct a 3D face model, and match it with gallery data in real-time. We detect a face and 2D facial landmarks from each RGB frame using dlib library [19]. We convert the corresponding depth frame to a 3D point cloud, then estimate the 3D head pose using the 2D landmarks and a generic 3D face model. After estimating the 3D head pose, we reject the frame if the computed rotations are greater than threshold values. Otherwise, we frontalize the segmented point cloud to have an upright, frontal face. Our method allows all point clouds to have the same frontalized 3D pose. Therefore, the ICP-based algorithm converges well and shows accurate registration results. The final updated 3D face model is used for the 3D face recognition. The overview of this proposed system is shown in Fig. 2.

3.1. Face detection and 2D facial landmark detection

We detect a rectangular face bounding box and 2D facial landmarks from each RGB frame in real time using the dlib library which is an implementation of the method in [20]. Dlib uses the iBUG 300-W dataset in the i-bug [21] to train a model to detect 68 facial landmark points.

3.2. Head pose estimation based on 2D facial landmarks

We estimate the head pose by computing the camera matrix using 3D-to-2D correspondent points:

$$u_p \cong K[R\ \mathrm{t}]\ X_w \qquad (1)$$

where u_p and X_w represent 2D and 3D points in the homogeneous coordinate system. The intrinsic camera parameters K are obtained from the sensor, while the extrinsic camera parameters $[R\ \mathrm{t}]$ which encode the camera's rotations and translations need to be computed. Based on available 68 2D landmarks, we selected 25 rigid landmark positions (e.g. the nose, inner, and outer coner of eye) and annotated their 3D positions on our 3D generic face model. The 25 3D landmark points are fixed, while the 25 2D landmark points vary from frame to frame. Given u_p, X_w, and K, we use the EPnP [22] algorithm to solve for $[R\ \mathrm{t}]$. Note that while we use the generic face model to estimate the rotations, we do not alter the point cloud based on the shape of the generic model as described in the sec 3.5.

3.3. Filtering frames

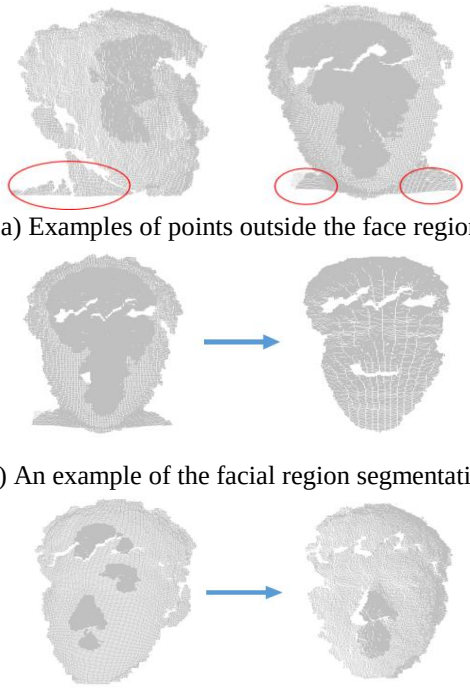
We propose to filter depth frames if the rotation (yaw angle) of the head is greater than a threshold (45 degrees) since these point clouds were empirically determined to make the face model worse. We also filter frames where we are unable to get person's facial landmarks (e.g. face occluded by hands). In other words, we collect and aggregate only good facial data in terms of 3D head rotation to build an accurate 3D face model. Our implementation depends on whether the dlib detector can detect all 68 facial landmark points.

3.4. Segmentation of face region

As the dlib face detector captures a face in the form of a rectangular bounding box, point clouds of faces could contain regions outside of the face, possible outliers, such as shoulder and neck, shown in Fig. 3(a). Those points cause ICP to converge to a local minimum. To improve this, we shape a curve along the jawline based on the 2D facial landmark points and remove all 3D points lower than the curve, as shown in Fig. 3(b).

3.5. ICP registration using frontalized point clouds

We frontalize the 3D source point cloud using the estimated pose by rotating the source point cloud around its center to make a canonical face (i.e., upright frontal face). In addition, we set the middle of two eye brows as the rotation center using the 2D facial landmarks. Setting an appropriate rotation center is important because the point of the nose can be more inaccurate when the head is largely rotated. In this way, we get point clouds that have the same head pose and center, shown in Fig. 3(c). This method of filtering and normalizing allows us to produce an accurate 3D face model from a sequence of raw and noisy depth frame data even in the presence of large head motion. We aggregate all frontalized point clouds to build an accurate 3D face model.



(a) Examples of points outside the face region

(b) An example of the facial region segmentation

(c) An example of a frontalized point cloud

Fig. 3. Exmaples of segmentation and frontalization

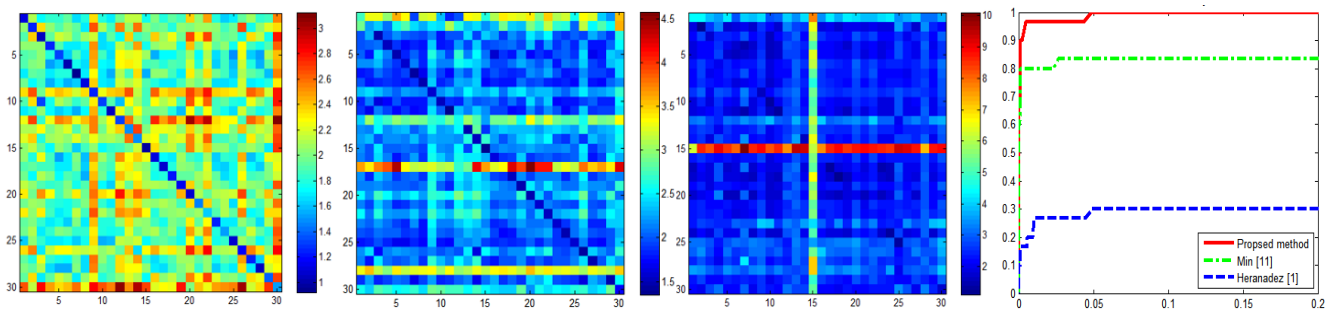
3.6. 3D face recognition

We use the 3D face recognition (3D-3D matching) method described in [2]. We take the reconstructed 3D face model as probe data. We measure the Euclidean distances between probe facial data which is registered to canonical faces using ICP and gallery facial data. The identity of the probe facial image is determined by finding the gallery facial image which has the lowest distance.

4. EXPERIMENTAL RESULTS

We have compared our 3D recognition method against two state of the art methods: a 3D-3D matching using a reconstructed 3D face from an RGB-D video [1, 2] (baseline 1) and a video-to-3D matching using a set of input depth frames [16] (baseline 2). We also qualitatively compared our reconstructed 3D models with the method of [1].

Data. Since most of the existing databases contain limited head pose motion, we have collected a small database containing 60 RGB-D video segments from 30 subjects with large pose changes. The database consists of a



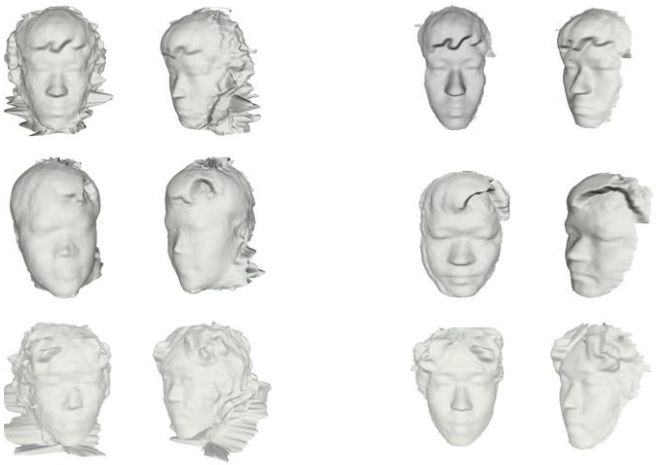
(a) Proposed method (b) Min [11] (baseline 2) (c) Hernandez [1] (baseline 1) (d) ROC

Fig. 4. Similarity matrices and ROC of proposed, baseline 1, and baseline 2 method.

probe set and a gallery set. Each video in the gallery set contains only near frontal faces while each video in the probe set includes large head motion from -90 to 90 degrees. We used a fixed PrimeSense camera and set its resolution to 640×480. The distances between the subjects and the camera were about 50 cm and the recording time of each video was about 15 to 25 seconds.

4.2. Qualitative analysis of face modeling

We generated 3D face models using both our proposed method and the baseline 1 from the probe videos containing large head motion. We qualitatively compared two 3D mesh models per subject. We have observed that the quality of results from our method surpasses the baseline 1, as shown in Fig. 5. The eyes, noses, lips and surfaces of the our results are more distinct and smoother than those of the baseline 1.



(a) Hernandez [1] (baseline 1) (b) Proposed method

Fig. 5. Quantitative evaluation results

4.3. 3D Face recognition

Method. We compared the recognition performances of our method with the two baseline methods. In the 3D-3D matching method (baseline 1), we first generated 30 3D face models from the 30 gallery videos and enrolled them the gallery database. Given each probe video, we build a 3D face, compute the distances against all 30 gallery models [2], and generate a similarity matrix (Fig. 4(c)). In the video-3D matching method (baseline 2), we randomly select 10 frames from each probe video and compute the distances between each frame and the specific 3D model in the gallery database [16]. The similarity score between the probe and the gallery is the minimum distance (Fig. 4(b)). The gallery database is identical to the gallery in the baseline 1. In our method, we use our proposed 3D face modeling algorithm and the identical recognition method with the baseline 1.

Results. Our proposed method outperforms the two baseline recognition methods, as shown in Fig. 4(d). In this experiment, the baseline 2 outperforms the baseline 1 because the quality of 3D models has been dramatically degraded by the large head motion (Fig. 5) and the video-based matching method is robust to outliers in the input sequence. The similarity matrices show the quality of matching results. All diagonal elements, representing the matching distance between the same subjects, in our method (Fig. 4(a)) show the discriminant power of our method compared to others (Fig. 4(b), 4(c)). The 15th subject is wearing a hat which makes the result worse in the baseline 1 (lines in Fig. 4(c)). In this small dataset, our method showed 100% rank 1 recognition rate in the presence of large head motion. However, the performance of this method could be overestimated due to the small dataset.

5. CONCLUSION

Our proposed method provides very accurate 3D face models in the presence of large head motion in natural interaction environments. The 3D face models from our method shows better distinct facial features than the results from the baseline 1. Our experimental results on a small dataset containing large head motion showed that our proposed method outperforms the baseline methods. Our proposed method showed 100% rank 1 recognition rate on the small database with large head motion.

Our future work includes accurate 3D face modeling for non-cooperative persons considering pose and facial expression changes.

6. REFERENCES

[1] Matthias Hernandez, Jongmoo Choi, and Gerard Medioni, Near Laser-Scan Quality 3-D Face Reconstruction from a Low-Quality Depth Stream, Image and Vision Computing, 2015.

[2] Jongmoo Choi, Ayush Sharma, and G´erard Medioni, Comparing Strategies for 3D Face Recognition from a 3D Sensor, RO-MAN, 2013 IEEE, 2013.

[3] M. Zollhöfer, M. Martinek, G. Greiner, M. Stamminger, J. Süßmuth, Automatic reconstruction of personalized avatars from 3D face scans, Comput. Animat. Virtual Worlds (Proceedings of CASA 2011) 22 (3–4), 2011.

[4] L. Tang, T. Huang, Automatic construction of 3D human face models based on 2D images, IEEE International Conference on Image Processing (ICIP), pp. 467–470, 1996.

[5] Paul J. Besl, Neil D. McKay, A Method for Registration of 3-D Shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 2, pp. 239-256, 1992.

[6] U. Castellani and A. Bartoli, 3D shape registration, 3D Imaging, Analysis, and Applications, 2012.

[7] Kim, Jongsun, et al. "Effective representation using ICA for face recognition robust to local distortion and partial occlusion." Pattern Analysis and Machine Intelligence, IEEE Transactions on 27.12 (2005): 1977-1981.

[8] Medioni, Gérard, et al. "Identifying noncooperative subjects at a distance using face images and inferred three-dimensional face models." Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on 39.1 (2009): 12-24.

[9] Choi, Jongmoo, et al. "3D face reconstruction using a single or multiple views." Pattern Recognition (ICPR), 2010 20th International Conference on. IEEE, 2010.

[10] Lin, Yuping, Gérard Medioni, and Jongmoo Choi. "Accurate 3D face reconstruction from weakly calibrated wide baseline images with profile contours." Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010.

[11] Wael AbdAlmageed, et. al, Face Recognition Using Deep Multi-Pose Representations, WACV 2016: IEEE Winter Conference on Applications of Computer Vision, 2016.

[12] PrimeSense Camera, Available: https://en.wikipedia.org/wiki/PrimeSense [Accessed Jan. 01, 2016].

[13] Microsoft Kinect, Available: https://msdn.microsoft.com/en-us/library/hh438998.aspx [Accessed Jan. 01, 2016].

[14] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(9):1063–1074, 2003.

[15] Richard A Newcombe, Dieter Fox, and Steven M Seitz, Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 343–352, 2015.

[16] Rui Min, Jongmoo Choi, Medioni, G., Dugelay, J., Real-time 3D face identification from a depth camera, in Pattern Recognition (ICPR), 2012 21st International Conference on, vol., no., pp.1739-1742, 11-15 Nov. 2012.

[17] B. Y. L. Li, A. S. Mian, W. Liu, and A. Krishna, Using Kinect for face recognition under varying poses, expressions, illumination and disguise, in Proc. IEEE Workshop Appl. Comput. Vis., pp. 186–192, 2013.

[18] C. Ciaccio, L. Wen, and G. Guo, Face recognition robust to head pose changes based on the RGB-D sensor, in Proc. IEEE 6th Int. Conf. Biometrics, Theory, Appl., pp. 1–6, 2013.

[19] Dlib, C++ open source library for face landmark detection. Available: http://dlib.net/ [Accessed Jan. 01, 2016].

[20] V. Kazemi and S. Josephine, One millisecond face alignment with an ensemble of regression trees, in IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874 2014.

[21] I-bug, Intelligent Behaviour Understanding Group (iBUG), Department of Computing, Imperial College London. Available: http://ibug.doc.ic.ac.uk/ [Accessed Jan. 01, 2016].

[22] V. Lepetit, F. Moreno-Noguer, and P. Fua, EPnP: An accurate O(n) solution to the pnp problem, Int. J. Comput. Vision, 81:155-166, February 2009.