A Markerless Augmented Reality Approach Based on Real-Time 3D Reconstruction using Kinect

Márcio C. F. Macedo*†, Antônio L. Apolinário Jr.*, Antonio C. S. Souza†
*Department of Computer Science
UFBA
Salvador, Brazil
† Department of Electro-Electronic Technology
LABRAGAMES - IFBA
Salvador, Brazil

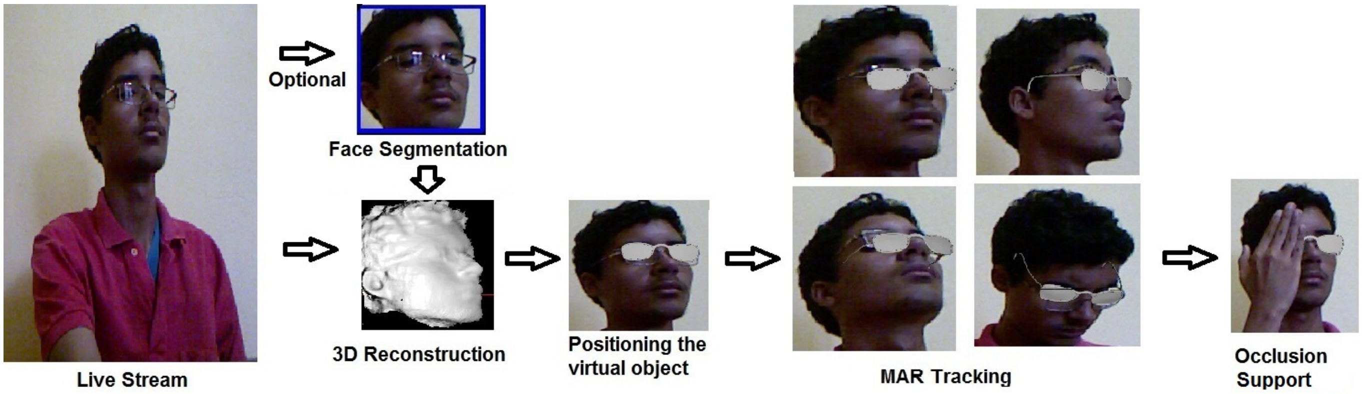

Fig. 1. A schematic view of the markerless augmented reality approach: A) RGB-D live stream. B) A face detector is used to locate the face in the whole image (this step is required only if the face is chosen as the real object that will be augmented). C) Reference 3D model is reconstructed with KinectFusion algorithm [1]. D) The 3D reconstruction is stopped and the user positions a virtual object into the scene. E) MAR tracking is done based on the reference 3D model. F) Our approach supports occlusion.

Abstract—**In this paper we present a markerless augmented reality (MAR) approach based on real-time 3D reconstruction using a low-cost depth camera, the Kinect. We intend to use this MAR approach for 3D medical visualization. In the actual state of developing, our approach is based on the KinectFusion algorithm with some minor adaptations, as long as our goal is not reconstruct scenes but use the reconstructed model as a reference to MAR.**

Keywords-**Augmented Reality; 3D Reconstruction; Kinect;**

I. Introduction

Augmented Reality (AR) is a technology in which a user's view of a real scene is augmented with additional virtual information. Accurate tracking, or camera pose estimation, is required for the proper registration of virtual objects. However, tracking is one of the main technical challenges of AR.

AR can be marker-based or markerless. Marker-based AR uses artificial markers to help the system to estimate the camera pose. These markers need to be printed and placed in the real world to be tracked by the application. Markerless AR (MAR) uses a part of the real scene as the marker. Tracking becomes more complex in MAR. However, because there are no ambient intrusive markers that are not really part of the scene, MAR is desirable in several AR application scenarios such as medicine. An example of such application is the AR system for 3D medical visualization proposed in [2]. In this case, instead of placing artificial markers on the patient's face, it is desirable to use the patient's face as the marker and track it through all frames. One way to achieve this goal is building a reference 3D model of the patient's head and aligning it to the current head captured by the sensor. With the recent advances in hardware, like modern GPUs and low-cost capture devices, this solution can be used with enough accuracy and in real-time.

We present a general MAR approach based on real-time 3D reconstruction using a low-cost depth camera, the Kinect. We intend to use this approach to build a MAR system for 3D medical visualization, in which the patient's head will be augmented with volumetric data of his cranium. Currently, our approach can be used for two different purposes: MAR with a head (fixed Kinect camera by the user turning his head in front of it) or with a scene (Kinect camera moving around the scene). In both cases, our approach consists of five basic steps shown in Fig. 1. First, in the MAR with a head, we apply the Viola-Jones face detector [3] to locate the face in the whole image. Afterward, a reference 3D model is built with a real-time 3D reconstruction algorithm. Next, the user positions the virtual object into the reconstructed model. Finally, the Kinect raw data is aligned to the reference 3D model, predicting the current camera pose. Also, we improve the robustness of the alignment step using a face tracking solution [4].

In the actual state of developing, our approach is inspired by the KinectFusion [1]: an algorithm that allows the dense mapping of extended scale environments in real-time using only Kinect raw data. However, our goal is not reconstruct scenes but use the reconstructed model as a reference to MAR. The preliminar results show that we have a MAR approach that supports occlusion.

The rest of the paper is arranged as follows. Section 2 provides a review on the related work of MAR. Section 3 presents the proposed approach. Section 4 discusses the experimental results. The paper concludes in Section 5, with a summary and discussion of future work.

II. Related work

The literature contains several works on MAR, which can be conveniently divided into two types: model based and Structure from Motion (SfM) based.

In model based techniques, the knowledge about the real world is obtained before the MAR tracking and is stored in a 3D model. This model is used for the camera pose estimation. An example of a model based approach is [5], where the tracking is done based on the objects' edges. In SfM based techniques, there is no previous knowledge about the scene, which is acquired during tracking. An example of a SfM based approach is MonoSLAM [6], where the camera movement is estimated based on the selection of good features in the scene.

A full review of MAR techniques is beyond the scope of this paper and we refer to [7] for a more detailed discussion.

With the recent increasing availability of depth sensors, a few notable works have shown the usefulness of the depth for solving the problem of MAR tracking. In 2011, Izadi et al. [1] described a system that enables real-time detailed 3D reconstruction of a scene using the depth stream from a Kinect. The system was called KinectFusion. Using GPU, it was the first system to reconstruct high-detail models at 30 fps [8] from a low-cost depth camera. Izadi et al. [1] presented some MAR applications, showing the level of the user interaction in their system.

In the same year, Weise et al. [9] presented a system that enables active control of facial expressions of a digital avatar in real-time. The system is called FaceShift [10]. It was the first system to enable high-quality reconstruction and control of facial expressions using blendshape representation in real-time. FaceShift represents a great advance in the field of MAR and non-rigid surface reconstruction.

Our approach is based on the KinectFusion algorithm with some minor adaptations and, different from [9], we do not use a deformable registration algorithm during the tracking.

III. MAR Approach Based on Real-Time 3D Reconstruction

The main goal of our approach is to track robustly and in real-time a real object through a Kinect live stream (Fig. 1-A). The first step of our approach is to segment the object from the scene applying a Z-axis threshold on the depth map provided by the Kinect sensor. Currently, the threshold is defined by the user. As long as our object of interest is a human face, we apply the Viola-Jones face detector [3] to locate and segment the face in the color image (Fig. 1-B). This face detector uses a representation called integral image to compute Haar-like features quickly over the image and a combination of simple classifiers built using the Adaboost learning algorithm [11] to detect the face regions. By calibrating the depth and color sensors of the Kinect, from the face segmentation we can achieve a more restricted area of the depth map to be used in the next step of the process.

Once with the real object identified and segmented, we need to track it through the Kinect live stream to build its reference 3D model (Fig. 1-C). Instead of using an image-based tracking, we use a real-time variant of a well-known depth-based algorithm, the Iterative Closest Point (ICP) [12], [13]. We choose a depth-based method because it does not suffer from changes in illumination or the presence of textureless regions in the scene. The registration can be done frame-to-frame or frame-to-model. Frame-to-frame consists in the registration of sequential depth frames and it is not appropriate to use with Kinect, as it provides noisy depth data. Frame-to-model consists in the registration of the current depth frame provided by the depth sensor and the previous depth view of a reconstructed model. To do a real-time frame-to-model registration with no previous knowledge about the scene and a 3D reconstruction of the object, we use the KinectFusion algorithm [1]. KinectFusion is a system that integrates raw depth data from a Kinect camera into a voxel grid to produce a high-quality 3D reconstruction of a scene. The system first applies a bilateral filter [14] to the depth map to reduce the noise preserving discontinuities of the raw data. The filtered depth map is then converted into a vertex and a normal map. Once with the current camera pose estimated by the ICP, the raw depth data can be integrated into the voxel grid. The grid stores at each voxel the distance to the closest surface and a weight that indicates uncertainty of the surface measurement. This distance is a truncated signed distance function (TSDF). Surface extraction is achieved by detecting zero-crossings through a raycaster. All these operations are made using the GPU. An overview of KinectFusion can be seen in Fig. 2.

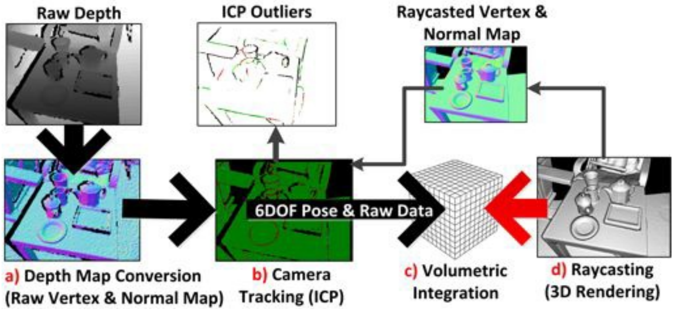
Raw Depth
ICP Outliers
Raycasted Vertex & Normal Map
6DOF Pose & Raw Data
a) Depth Map Conversion (Raw Vertex & Normal Map)
b) Camera Tracking (ICP)
c) Volumetric Integration
d) Raycasting (3D Rendering)

Fig. 2. Overview of KinectFusion's pipeline [1].

With the reference 3D model properly reconstructed in real-time, the user positions the virtual object into the scene (Fig. 1-D). Afterward, the MAR tracking can be started (Fig. 1-E). We apply the ICP algorithm to compute the current camera pose. The ICP uses the projective data association [15] to find correspondences between the current depth frame and the accumulated model. In this association, each point is transformed into camera coordinate space and perspective projected into image coordinates. The corresponding points are that on the same image coordinates. The ICP may fail (i.e. does not converge to a correct alignment) when there is not a small pose variation between sequential frames. We solve this problem by using the solution proposed in [4]. In their solution, a head pose estimation algorithm is used to give a new initial guess to the ICP algorithm to compute correctly the current transformation. The head pose estimation used is the algorithm proposed by Fanelli et al. [16]. They trained random forests [17] to estimate head pose from low-quality depth images. To train the trees, each depth map was annotated with labels indicating head center and Euler rotation angles. These labels were estimated automatically using ICP after a 3D facial reconstruction. After labeling and training, the head pose can be estimated letting every image region to vote it. The vote consists of a classification whether the image region contains a head and a retrieval of a Gaussian distribution computed during the training and stored at the leaf. This probabilistic approach achieves high accuracy and runs in real-time even running completely in CPU.

Our MAR approach can solve occlusion (Fig. 1-F) using a GLSL fragment shader [18] that process the depth buffer data of the virtual and real objects to check whether the virtual object is in front of the real object. Also, we use the color buffer data to color the fragment according to the depth test result.

IV. Preliminary Results and Discussion

In this section we analyze the approach's performance and describe the experimental setups we used.

For all tests we used an Intel(R) Core(TM) i7-3770K CPU @3.50GHz 8GB RAM. We used the open source C++ implementation of the KinectFusion [19] released by the PCL project [20].

We tested the prototype with real data captured from a Kinect sensor using a grid with volume size of $50cm$x$50cm$x$130cm$ considering that the object of interest is a human face.

Our approach runs in real-time and the live markerless tracking is done properly even with the noisy data provided by the Kinect. When the tracking algorithm fails, the solution proposed in [4] recovers the algorithm in most of the cases. Currently, one problem is that, as the tracking is mainly based on registration of the current and the previous depth frame, when there is occlusion, the real object is partially occluded and the tracking may fail. We empirically verified that the tracking fails when more than 70% of the real object is occluded.

We solve the problem of occlusion by using a fragment shader that check whether the virtual object is in front of the real object. However, one problem of the current solution is that the real object used to verify the occlusion is a 3D mesh built from the bilateral filtered depth map. The depths of the real object can have holes distributed along the model and around the edges of the object. In this case, the solution proposed in [1] cannot be used because, in their work, the MAR occlusion is done with a scene already reconstructed. In our case, the user can interact with parts of his body deforming in front of the camera (Fig. 3).

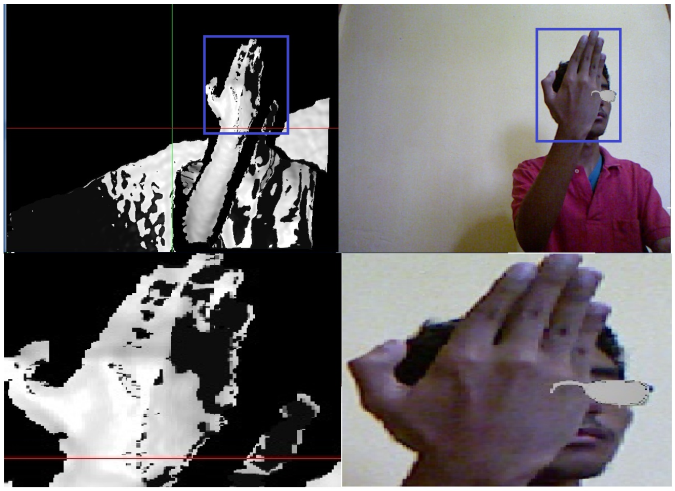

Fig. 3. Occlusion problem: The middle part of the glasses is not occluded because the corresponding part of the hand in the real object is not visible due to holes. The zoom in the bottom highlights the visualization of the area in which the occlusion is not done properly.

V. Conclusions and Future Work

We presented the preliminary results of a MAR approach based on real-time 3D reconstruction using a low-cost depth camera, the Kinect. We used the KinectFusion algorithm to reconstruct the reference 3D model and used the Viola-Jones face detector and a face tracking solution. when the real object was a human head. Our approach can deal with occlusion, and the MAR tracking can be done properly in real-time. However, the presence of holes distributed along the real model is a problem to the occlusion test and, when there is occlusion, the real object is partially occluded and the tracking may fail.

Encouraged by the work of Meister et al. [8], for future work we plan to analyze the accuracy of our approach to check if it can be used for medical applications. Further improvements can be achieved by implementing a deformable registration algorithm to track the face, as proposed by [9]. Or implementing a better rendering algorithm for mixed reality, as proposed by [21]. The next steps of this work are improve its use in the 3D medical visualization, taking advantage of the KinectFusion's grid representation to implement a method for automatic position of medical volumetric data into the scene.

Acknowledgments

We are grateful to the PCL project for providing the open-source implementation of the KinectFusion. This research is financially supported by FAPESB and CAPES.

References

[1] 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 *Proceedings of the 24th annual ACM symposium on User interface software and technology*, ser. UIST '11. New York, NY, USA: ACM, 2011, pp. 559–568.

[2] J.-D. Lee, C.-H. Huang, T.-C. Huang, H.-Y. Hsieh, and S.-T. Lee, "Medical augment reality using a markerless registration framework," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 5286–5294, Apr. 2012.

[3] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, no. 2, pp. 137–154, May 2004.

[4] M. Macedo, A. Apolinario, and A. C. Souza, "A robust real-time face tracking using head pose estimation for a markerless ar system," in *Symposium on Virtual and Augmented Reality*, Cuiaba, MT, Brazil, May 2013.

[5] A. Comport, E. Marchand, M. Pressigout, and F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 12, no. 4, pp. 615–628, 2006.

[6] A. Davison, I. Reid, N. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 1052–1067, 2007.

[7] V. Teichrieb, E. Apolinário, M. Bueno, J. Kelner, and I. Santos, "A survey of online monocular markerless augmented reality," *International Journal of Modeling and Simulation for the Petroleum Industry*, vol. 1, no. 1, Aug. 2007.

[8] S. Meister, S. Izadi, P. Kohli, M. Hämmerle, C. Rother, and D. Kondermann, "When can we use kinectfusion for ground truth acquisition?" in *Workshop on color-depth fusion in robotics, IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[9] T. Weise, S. Bouaziz, H. Li, and M. Pauly, "Realtime performance-based facial animation," in *ACM SIGGRAPH 2011 papers*, ser. SIGGRAPH '11. New York, NY, USA: ACM, 2011, pp. 77:1–77:10.

[10] (2013, Jan.) Faceshift. [Online]. Available: http://www.faceshift.com/

[11] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Proceedings of the Second European Conference on Computational Learning Theory*, ser. EuroCOLT '95. London, UK, UK: Springer-Verlag, 1995, pp. 23–37.

[12] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and Vision Computing*, vol. 10, no. 3, pp. 145 – 155, 1992.

[13] P. Besl and H. McKay, "A method for registration of 3-d shapes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 14, no. 2, pp. 239 –256, feb 1992.

[14] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Computer Vision, 1998. Sixth International Conference on*, jan 1998, pp. 839 –846.

[15] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*, 2001, pp. 145 –152.

[16] G. Fanelli, T. Weise, J. Gall, and L. V. Gool, "Real time head pose estimation from consumer depth cameras," in *33rd Annual Symposium of the German Association for Pattern Recognition (DAGM'11)*, September 2011.

[17] L. Breiman, "Random forests," in *Machine Learning*, 2001, pp. 5–32.

[18] R. J. Rost, *OpenGL(R) Shading Language (2nd Edition)*. Addison-Wesley Professional, 2005.

[19] (2013, Mar.) Kinfu. [Online]. Available: http://svn.pointclouds.org/pcl/trunk/gpu/kinfu/

[20] R. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, may 2011, pp. 1 –4.

[21] M. Knecht, C. Traxler, O. Mattausch, and M. Wimmer, "Reciprocal shading for mixed reality," *Computers and Graphics*, vol. 36, no. 7, pp. 846 – 856, 2012.