3D registration based perception in augmented reality environment

Heen Chen*, Hanwu He, Jianqing Mo, Jinfang Li and Xian Yang

Abstract: Early augmented reality (AR) community has paid a lot of attention to identifying 2D fiducial markers in AR scene, only a little works have been devoted to 3D objects perception. With the massively increased usage of 3D representation, it is highly desirable that powerful processing tools and algorithms should be developed for 3D recognition purpose in the augmented reality domain. In this paper, we address the 3D depth understanding problem for AR assembly applications. We propose a 3D registration based perception method designed for desk assembly environments. The 3D perception method involves two tasks: the optical flow based structure from motion (SFM) and a coplanar 4 points set registration of the SFM point cloud and CAD models point cloud. From the transformation, the position and orientation of the real industrial object in the assembly environment can be determined and an AR operation can be applied. The novelty of our method resides in that it does not require any marker attached on the scene and obtain a more accurate registration result.

Subjects: Simulation & Modeling; Computer Graphics & Visualization; Computer Aided Design & Manufacturing

Keywords: augmented reality; registration; perception; pose estimation

1. Introduction
Augmented reality (AR) technology (Azuma, 1997) can be applied to mix real scene with virtual objects to create a mixed reality interface. This mixed reality is a powerful methodology for assembly evaluation and product development in the next manufacturing generation. In an AR interface, it would be possible to realize the concept of mixed prototyping, where a part of the design is available...
as physical prototypes and a part of the design exists only in the virtual form. With such an interface, it would be possible to combine some of the benefits of both physical and virtual prototyping. This allows time saving, convenient retrieving and sending of information vital to support the users’ assembly tasks. The users can thus focus attention on the operation at hand without having to physically change gesture to receive the next set of assembly instructions.

One of the challenges in building AR systems is to track objects in AR environment and align the virtual objects with specific target properly. Common AR operates with prior fiducial marker presented in the user’s environment. Tracking is then limited to the special area. Thus the assembly tasks will be constrained to a small region. When new regions are explored, due to marker visibility requirements, the system will fail to superimpose virtual model. Thus, its flexibility will be reduced greatly. Moreover, the presentation of an AR marker in the assembly environment will inevitably cause some kind of inconvenience.

In this paper, we address the registration problem for AR assembly applications. Though this problem has received a lot of attention in the 2D recognition of the AR community, only a little works have been devoted to 3D scene perception algorithms for industrial assembly environments. We propose a 3D registration-based perception method designed for desk assembly environments. The novelty of our method resides in that it does not require any marker in the scene and obtain a more accurate result.

The paper is organized as follows: Section 2 gives a rough survey of 2D based and 3D based tracking respectively. Section 3 deals with scene and virtual model representation. Then the registration of virtual model and scene point cloud is introduced, and we evaluate all our algorithm on synthetic and real data. Finally, we give our conclusions and discuss future works.

2. Related work

2.1. 2D based tracking
The ubiquity of high-quality mobile cameras with ever-increasing computational capacity, packaged along with other rich sensors, is providing many new opportunities for the robotics and augmented reality applications domain. When dealing with camera localization or pose computation, most of the approaches proposed in the relevant literature rely on a 2D marker recognition. In the related computer vision literature, geometric features considered for the estimation are often segments, straight lines, contours or points on the contours cylindrical objects, or a combination of these different features. The main advantage of these approaches is their robustness and accuracy. The main drawback is that they may be subject to marker occlusion and, worse, system fail if the marker is out of the viewing region.

In an attempt to track a camera in an unknown environment, Azuma (1997) did a lot of impressive improvement in this research for AR application. Simultaneous Localization and Mapping (SLAM) based tracking method (Davison, 2003) and Real-time key-frame based SFM (Klein & Murray, 2007) are two common techniques proposed to solve the tracking problem in unknown scenes, which have been successfully adopted by many practical systems. MonoSLAM (Davison, 2003) using a single camera to replace traditional sensor is reported to be the first system capable of performing visual odometry.

Klein and Murray (2007) presented an SFM-based tracker that builds the model of the environment on the fly but only works in small workspaces. Langlotz et al. (2011) combines vision tracking with the absolute orientation measured by inertial and magnetic sensors to track smartphone’s moving camera. Wagner, Reitmayr, Mulloni, Drummond, and Schmalstieg (2008) employ visual features to estimate the camera’s pose in relation to the model of the environment, but limit to planar object. Arth, Wagner, Klopschitz, Irschara, and Schmalstieg (2009) presented a method for localizing a mobile user’s 6DOF pose in a wide area using a sparse 3D point reconstruction and visibility constraints.
These methods typically use the frame-to-frame matching and bootstrapping-adding new features strategies for environment mapping. However, this allows the error to accumulate over longer sequence because of some of the interest points may not be visible all the time.

Lim, Sinha, Cohen, and Uyttendaele (2012) proposed an image based 6-Dof fast location recognition technique based on offline SFM point clouds, which is tightly related to our work. Similarly, our method selects an optimal set of key frames from the input video sequences shooting the initial space, as the camera moves we estimate the pose of the camera and add a new feature to the constructed map if necessary. In addition, our motion average matching can obtain reliable 6-Dof orientation of camera.

Other SLAM methods inspired by PTAM have been introduced. ORB-SLAM (Mur-Artal, Montiel, & Tardos, 2015) employs ORB features and PTAM skeleton, is much more robust and relatively simple to implement.

2.2. 3D base tracking

With the development of computer vision and the advent of new-generation depth sensors, the use of 3D data for objects representation is becoming increasingly popular. Both computer vision based method and sensors can acquire 3-D data easily and in real time. With the massively increased usage of 3D data for perception tasks, it is desirable that powerful processing tools and algorithms are available for the growing community of the AR domain.

Registering and recognizing rigid objects from 3D point clouds is a well-known problem in computer vision. In this article, we focus on 3D object perception and pose estimation. Specifically, our goal is to recognize rigid objects from a point cloud, estimate their position and orientation in the real world.

Two main approaches to solving the problem are primitive-based approaches and descriptor-based approaches. Primitive approaches recognize objects by relying on points, segments and planes distance measurement. It is difficult to handle partial shapes using these approaches since primitive features are sensitive to both absences of shape parts and occurrence of clutter.

Descriptor-based method have shown a great advantage in 3D object detection and object class categorization when dealing with occlusions and clutter. Johnson and Hebert (1999) proposed Spin Images descriptor, which represented points falling within a cylindrical region by means of a plane that “spins” around the normal as 2D histograms and recognition is done by matching spin images, grouping similar matches and verifying each output pose. Frome, Huber, Kolluri, Bülow, and Malik (2004) presented a 3D shape context based descriptor to computing 3D histograms of points within a sphere around a feature point. The descriptor extended the 2D shape context method to a 3D surface, and have a higher recognition rate on a set of 3D models of passenger vehicles. Rusu, Bladow, Marton, and Beetz (2008) proposed Point Feature Histograms (PFH) to encode a local geometry of a point and its k nearest neighbors. They also extended PFH to a simplified and fast version which reduces computational complexity but retains the majority discriminative power of the PFH. Many other local descriptors have been proposed, featured with being more discriminative, more repeatable, and more robust to noise and more invariant to local rigid transformations.

The descriptor registration methods can be applied to process point cloud obtained by the sensor because all the data are accurate. However, when dealing with raw noisy data generated from SFM, the method may fail. It is impossible to compute such local descriptors in presence of significant noise and outliers. Moreover, the data generated from SFM method can only recover part of the object, and it might also contain points that do not belong to the scene. In such scenarios, instead of using local descriptors, an effective alternative is to rely on the principle of the primitive method that does not need to compute local descriptors.
3. Scene and virtual model representation

3.1. Scene representation

Our representation of the user’s environment consists of a collection of $M$ key-points located in a world coordinate frame $W$ and a set of calibrated images, a set of 2D-3D matches that encode the views from which a particular 3D point was triangulated from during SFM.

The calibrated images are used to build a database of feature descriptors for the 3D points, and a kd-tree index is constructed to support efficient approximate nearest neighbor queries during 2D-3D correspondence feature matching. As key-points are tracked through the input sequence, a subtrack optimization algorithm is used to determine their locations and incrementally build the complete cloud. Each key-frame has an associated camera-centered coordinate frame, denoted $F_i$ for the $i$th key-frame. The transformation between this coordinate frame and the world is denoted as $^wT_i$.

Each key-frame also stores a three level pyramid of grayscale 8bpp images; level zero stores the full 640 × 480 pixel camera snapshot, and this is sub-sampled down to level three at 80 × 60 pixels.

We extract key-points using the FAST detector and compute the matching correspondences (Rosten & Drummond, 2006). We further optimize the point retrieval by grouping cameras into overlapping clusters and use them for local bundle adjustment and pose recognition.

The $j$th point in the map ($P$) has coordinates $^wP_j = (x_j, y_j, z_j, 1)^T$ in coordinate frame $W$. The interest points which make up each patch feature are not stored individually, we associate them with a source key-frame—typically all the key-frame in which this point was extracted and observed. Thus each landmark point stores a reference to all source key-frame, its pyramid level within these key-frames, and pixel location within this level.

3.2. Training data generation

To train the system, the CAD models representing our objects are transformed to partial point clouds, simulating the output that will be given by a laser scanner. For this purpose, we construct a bounding sphere with a radius large enough to enclose the object. The sphere is generated using an icosahedron as starting shape and subdividing each triangular face with four equilateral triangles. The resulting triangles are used to place the virtual camera at their barycenter. The virtual camera will simulate the real laser scanner which will render the object seen from that viewpoint into a depth buffer. A partial point cloud serves as training data can be extracted from the buffer by using the virtual scanner of point cloud library (Point Cloud Library, 2015; Rusu & Cousins, 2011).

4. Registration of virtual model and scene point cloud

Attempt to detect and identify the interest region of the point cloud, the aim of this stage is to align a more accurate 3D model to the sparse point cloud obtained by SFM. Through the registration results, the system can deduce what kind of the object the point cloud cluster describes and also the object’s orientation and position in the physical space. Figure 1 shows the workflow of the proposed registration based perception method. The algorithm takes as input two point clouds, one from the real scene which is obtained from SFM, the other transformed from the virtual 3D model. After an adaptively resampling to regular point density, the CAD point cloud serves as the training data which will be matching with the SFM point cloud using nearest point distance minimization algorithm.
Matching results is a set of corresponding coplanar 4-point, from which the rigid-body transformation for alignment is estimated, and optionally refined with ICP (Besl & McKay, 1992).

Compared to the well-known problem of registering couples of point cloud sets obtained from a laser scanner, this case has serious complications (Corsini et al., 2013). First, the point clouds produced by the SFM methods are a sampling of the real assembly environment with clutter background. Therefore, the sparse point will be presented with outliers and noise. And the second difficulty is that the density of the point cloud varies unevenly over all the range data involving in the registration stage, thus we cannot use it with any certainty to find the transformation matrix. Finally, the two kinds of point cloud may share any fraction of the surface, thus, it is impossible to get a correct alignment using the oriented bounding boxes of the two point clouds.

4.1. Alignment with 4 points congruent sets (4PCS)

Our proposal is inspired by a recent work of Aiger, Mitra, and Cohen-Or (2008) and Corsini et al. (2013). They employed a RANSAC approach to align pairs of surfaces (hereafter P and Q) in arbitrary initial poses. Their idea is to select 4 points in P, 3 points at random and the 4th point to ensure coplanarity and then, for each of these quadruples, to look for all the quadruples in Q that are approximately congruent. If the quadruples of Q can be registered with the quadruple extracted from P, we note it as a candidate. For all the candidate quadruples in Q, we randomly select one and applied a respective transformation to the whole point cloud. For a transformed Q, the number of points which is within a threshold distance of their closest point in P is calculated. The quadruples sampling and transformation are repeated until an alignment with enough alignment candidates is found. The correspondence of the two point clouds is considered established when the candidate transformation achieved the highest number of such points.

The 4PCS method follows the principle that in a planar quadrangle of 4 points where the intersection of the diagonals (the red point illustrated in Figure 2) can be determined. The resulting intersection ratios \( r_1 \) and \( r_2 \) of the two diagonals w.r.t. that point are preserved under affine transformation, and the corresponding points set from Q also satisfy the following relations:

\[
\begin{align*}
    r_1 &= \frac{||a - e||}{||a - b||} \\
    r_2 &= \frac{||c - e||}{||c - d||}
\end{align*}
\]

The 4-points congruent sets method has proven to be fairly robust with noisy data, mostly because it exploits the affine properties of 4 coplanar points which are resistant to the noise and outliers. However, the efficiency of the 4PCS algorithm relies on a prior knowledge of the density of the overlap of the two objects, and on the assumption that there is no obvious density difference between the objects.

We propose a density independent version of the 4PCS algorithm by introducing a modification, as shown in algorithm. The first step is to overcome the density difference in sampling between the scanned model and the point cloud obtained by SFM. This is because the sampling of the latter does not depend on object geometry alone, as it does with the sampling with laser scanners, but on the reconstruction stage as well. We employ a coarse-to-fine strategy to down sample the Q point cloud. By expressing the point cloud P as a set of planar regions and resampling them uniformly, we try to obtain a representation that is as dependent as possible on the actual shapes and not on the sampling provided by the SFM algorithm. \( q \) is a parameter which will decide the sampling density.

![Figure 2. Principle of 4PCS, with point a, b, c, d ∈ P and a’, b’, c’, d’ ∈ Q, both are the coplanar 4 points set.](image-url)
Scheme of the proposed algorithm

\[
\text{DensityIndependent4PCS}(P, Q) \\
\{ \\
\text{for}(k = 0; k < 2; k++) \\
\quad Q_r = \text{Downsampling}(Q)\text{; with grid leaf } (\rho/2^k); \\
\quad P_r = \text{Resampling}(P, \text{Density}(Q)); \\
\quad n_{\text{Best}} = 0; \\
\quad \text{for } i = 0 \text{ to } L \text{ do} \\
\quad \quad B = \text{Coplanar base }(P); \\
\quad \quad U = \text{FindCongruent}(Q_r, B, \delta); \\
\quad \quad \text{For all } U_i \text{ in } U \text{ do} \\
\quad \quad \quad T = \text{FindTransformation}(U_i, B); \\
\quad \quad \quad \text{if} \quad \text{AverageDistanceRejectionTest}(T) \\
\quad \quad \quad \quad n_{\text{Closest}} = \text{CountClosest}(T); \\
\quad \quad \quad \text{if}(n_{\text{Closest}} > n_{\text{Best}}) \\
\quad \quad \quad \quad \quad n_{\text{Best}} = n_{\text{Closest}}; \\
\quad \quad \quad \quad \quad T_{\text{Best}} = T; \\
\quad \quad \quad \quad \text{RefineWithICP}(T_{\text{Best}}); \\
\quad \quad \quad \text{End if} \\
\quad \quad \text{End if} \\
\quad \text{End for} \\
\quad \text{End if} \\
\text{End for} \\
\text{Return}(T_{\text{Best}}) 
\}
\]

5. Experimental results

We performed the evaluation of our monocular exploration system using the data-set of desk assembly environment sequence. Images were captured using a low-cost Logitech Quick-CaPro 9,000 camera with resolution 640 × 480, and using a lens with 80-degree horizontal field of view. The computation was performed on a desktop computer with an Intel Xeon(R) CPU E31225 3.10 Ghz processor and a NVIDIA 8,800 GT GPU which is used for dense optical flow computation, Win 7 system, 4G memory. Point cloud visualization is using MeshLab and VTK software.

Figure 3 illustrates the 3D objects perception results in a sparse point cloud. The point cloud of a cover, a hammer, a wrench and a pair of pliers are detected and identified in the scene point cloud, as shown in Figure 3(a–d). The perception result of each object will be represented by a transformation matrix \((R, t)\), where \(R\) is a 3 × 3 rotation matrix, and \(t\) a 3 × 1 translation vector. We label the region of interest with colors. The assembly tools are identified in the physical space also. From the experiment, we can find that the cover’s recognition is more accurate than that of the other objects. Because we take registration method to understand the point cloud scene, if the point cloud of a CAD model can match some part of the sparse point cloud, we consider the region as a perception object. The final result will be affected by the point cloud’s accuracy and also the CAD model’s accuracy. The CAD model of the pliers takes a little different shape from the real one, thus the registration performance is very poor, see the red circle in Figure 3. In our experiment, the cover’s average registration rate can reach 85%, while the pliers drop to 40%.
Figure 4 illustrates an augmented assembly using the implicit constraint between the components of the gear pump. A point cloud set of the AR scene is reconstructed, then the cover of the gear pump is recognized and located, depicted in Figure 4(a). Using the assembly constraint between the cover and the wheel, we can insert a virtual wheel model into the real assembly environment which is presented by point cloud, see Figure 4(b). In a practical application, the point cloud can be set invisible, and the implicit constraint will guide a proper installation of the virtual wheel on the frame of the video sequences which is useful for assembly design and evaluation in the early design stage, as illustrated in Figure 4(c). We also describe the augmented reality application based on 3D object perception. In Figure 4(d), we superimpose text “cover” “pump body” on respective objects, as shown in the red circle. While in a clutter manufacturing environment, this could bring significant benefits for an operator to identify industrial products accurately using the AR interface.

We make a comparison between our 3D perception based AR application and another method in Figure 5. Figure 5(a) is an original key frame selected from video sequences. Figure 5(b) is our experimental result and Figure 5(c) is a Qualcomm Vuforia AR application based on image marker (Qualcomm, 2016). We use the SDK developed by Qualcomm to design an AR application. The original image is a marker on which a virtual cover model is overlaying. From the experiment, we can see that our 3D virtual object seamlessly mixed with a 2D image, while the image marker AR performs no better than we expect.
However, there are various ways in which registration may fail. Most of these are due to the system’s dependency on the extraction of salient interest feature from images. Low-quality image will preclude most feature’s extraction, and this will obtain incorrect scene construction result. Without an accurate scene representation, the registration is impossible to carry out.

6. Conclusion and discussion
In this paper, we presented a 3D registration based perception augmented reality method. We address the 3D depth understanding problem for AR assembly applications. Our method involves two tasks: first employs SFM to simultaneous estimate the camera pose and obtain a sparse map of the scene, then a coplanar 4 points set registration algorithm is used to align the SFM point cloud and CAD models point cloud. From the transformation, the position and orientation of the real industrial object in the assembly environment can be determined and an AR operation can be applied. Results show that the method is capable of providing recognizing and localizing quality adequate for seamless registration. We believe the level of registration accuracy we achieve advances the state-of-the-art 2D image-based method.

In the future, the SFM result could be improved by a much more accurate method using a high-resolution camera or a laser scanner. With a growing number of salient interest features being extracted from the input images, the accuracy of depth measurement can be improved significantly. In some cases, this should obtain a correct point cloud representation of the AR scene and thus improve the perception performance of the subsequent stages.
Acknowledgments
We thank the anonymous reviewers for their insightful comments and suggestions.

Funding
This work was supported by the National Natural Science Foundation of China [grant number 51275094, 61300106].

Author details
Heen Chen1
E-mail: chheen@gmail.com
Hanwu He1
E-mail: hwhe@gdut.edu.cn
Jianqiong Mo1
E-mail: momolan@gdut.edu.cn
Jinfang Li1
E-mail: 2894599@qq.com
Xian Yang1
E-mail: 181642641@qq.com

1 School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou 510006, China.

Citation information
Cite this article as: 3D registration based perception in augmented reality environment, Heen Chen, Hanwu He, Jianqiong Mo, Jinfang Li & Xian Yang, Cogent Engineering (2016), 3: 1263037.

References
Alger, D., Mitra, N. J., & Cohen-Or, D. (2008). 4-points congruent sets for robust pairwise surface registration. ACM Transactions on Graphics (Proceeding of SIGGRAPH 2008), 27(3), 1–10.

Arth, C., Wagner, D., Klopfisch, M., Irshara, A., & Schmalstieg, D. (2009). Wide area localization on mobile phones. 8th IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Washington, DC.

Azuma, R. T. (1997). A survey of augmented reality. Presence-Teleoperators and Virtual Environments, 6, 355–385. http://dx.doi.org/10.1162/pres.1997.6.4.355

Besi, P. J., & McLay, N. D. (1993). A method for registration of 3D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14, 239–254.

Corsini, M., Dellepiane, M., Ganovelli, F., Gherardi, R., Fusilloy, A., & Scopigno, R. (2013). Fully automatic registration of image sets on approximate geometry. International Journal of Computer Vision, 102, 91–111. http://dx.doi.org/10.1007/s11263-012-0552-5

Davison, A. J. (2003). Real-time simultaneous localisation and mapping with a single camera. 9th IEEE International Conference on Computer Vision, Washington, DC.

Frome, A., Huber, D., Kolluri, R., Bulsion, T., & Malik, J. (2004). Recognizing objects in range data using regional point descriptors. Computer Vision-ECCV 2004 (pp. 224–237). New York, NY: Springer Berlin Heidelberg.

Johnson, A. E., & Hebert, M. (1999). Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 21, 433–449. http://dx.doi.org/10.1109/34.765655

Klein, G., & Murray, D. (2007). Parallel tracking and mapping for small AR workspaces. IEEE and ACM International Symposium on Mixed and Augmented Reality, Washington, DC.

Langlotz, T., Degendorfer, C., Mulloni, A., Schall, G., Reisnayr, G., & Schmalstieg, D. (2011). Robust detection and tracking of annotations for outdoor augmented reality browsing. Computers & Graphics, 35, 831–840. http://dx.doi.org/10.1016/j.cag.2011.04.004

Lim, H., Sinho, S. N., Cohen, M. F., & Uyttendaele, M. (2012). Real-time image-based 6-dof localization in large-scale environments. IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC.

Mur-Artal, R., Montiel, J., & Tardos, J. D. (2015). ORB-SLAM: A versatile and accurate monocular slam system. IEEE Transactions on Robotics, 31, 1147–1163. http://dx.doi.org/10.1109/TRO.2015.2463671

Qualcomm. (2016). Vuforia augmented reality SDK. Retrieved from http://www.qualcomm.com/Vuforia

Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. 9th European Conference on Computer Vision. New York, NY: Springer Berlin Heidelberg.

Rusu, R. B., Blodow, N., Marton, Z. C., & Beetz, M. (2008). Aligning point cloud views using persistent feature histograms. IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice.

Rusu, R. B., & Cousins, S. (2011). 3d is here: Point cloud library (PCL). IEEE International Conference on Robotics and Automation, Shanghai.

Wagner, D., Reisnayr, G., Mulloni, A., Drummond, T., & Schmalstieg, D. (2008). Pose tracking from natural features on mobile phones. International Symposium on Mixed and Augmented Reality, Washington, DC.