

Polyhedral Object Detection and Pose Estimation for Augmented Reality Applications

Ali Shahrokni, Luca Vacchetti, Vincent Lepetit, Pascal Fua¹

Computer Graphics Laboratory, Swiss Federal Institute of Technology Lausanne, CH 1015
Lausanne, Switzerland

{ali.shahrokni, luca.vacchetti, vincent.lepetit, pascal.fua} @epfl.ch

Abstract

In augmented reality applications, tracking and registration of both cameras and objects is required because, to combine real and rendered scenes, we must project synthetic models at the right location in real images. Although much work has been done to track objects of interest, initialization of these trackers often remains manual. Our work aims at automating this step by integrating object recognition and tracking into an AR system. Our emphasis is on the initialization phase of the tracking. We address all the three major aspects of the problem of model-to-image registration: feature detection, correspondence and pose estimation. We have developed a novel approach based on facet detection that greatly reduces the number of possible feature correspondences making it possible to directly compute the transformation which best maps 3-D object to the image plane. We will argue that this approach offers a one-fold speed-up over existing methods. Results of our AR system which integrates initialization and tracking are shown. Our method takes about 5 seconds on our example images.

1. Introduction

The emergence of Augmented Reality techniques allows the creation of mixed environments [1] in which real and virtual elements coexist. This is valuable to implement effective interaction techniques for the following reasons: Firstly, studies have shown that putting users in a real environment, such as a familiar room, helps them to accomplish or to learn their real tasks. Secondly, adding virtual elements increases flexibility and allows the creation of new situations or scenarios at very little cost. Although much work has been done to track objects of interest, initialization of these trackers often remains manual. Our work aims at automating this step. In this work we propose an initialization method which is based on efficient model-based object recognition tuned for Augmented Reality applications for tracking polyhedral objects.

We address all the three major aspects of the age-old problem of *model-to-image registration*, namely, *feature detection*, *correspondence* and *pose estimation*. It is well known the pose of a target object in a calibrated image can be computed in real time, given known correspondence between object and image points [2,3].

However, due to the exponential nature of establishing correspondences between image and object points, our final goal is to use a method for feature extraction which generates minimum number of features in the input image and also obtain a criterion to narrow down the number of acceptable hypothesis configurations for pose estimation. Moreover our feature detection method can deal with objects that are partially occluded.

For the AR applications that involve registration of a polyhedral object, we exploit the knowledge about the geometry of the target object to determine the features. In this work we first try to detect visible facets of the polyhedral objects in the images. Furthermore we exploit information about geometrical configuration of facets and apply it to candidate facets to reduce the number of feasible hypotheses.

We reduce the number of plausible correspondences in two levels. This is equivalent in partitioning the set of image points into several subsets prior to establishment of correspondences and then confining the correspondences within each subset. This partitioning is inspired by enforcing two constraints. The first level is the constraint imposed by *viewpoint consistency*. The second constraint is the *geometry* of the object model. Our correspondence establishment scheme has one order of magnitude of computational advantage over existing methods.

This paper is organized as follows. Related work is reviewed in next section. Feature detection and hypothesis generation and verification is described in section 3 and analytic discussion of the complexity of our method is given in section 3.3. Experimental results are shown in section 4.

¹ This work was supported in part by the Swiss National Science Foundation.

2. Related Work

The ability to do pose estimation of polyhedral objects from 2D images is an age-old computer vision problem that remains a challenge, especially when it needs to be done in real-time [4,5]. However, the generic nature of the problem and the complexities associated with it has thus far prevented introduction of a general solution. The recognition problem can be stated as follows. Assume we are given a set of image features N (say, in \mathbf{R}^2) and a set of model features M (say in \mathbf{R}^3); identify the best transformation for which the mapped 3-D features satisfy a given error model with respect to image features [6]. By transformation we refer to a projection matrix relating 3-D world to the image plane.

Breuel [6] proposed a method which uses a fixed error model in conjunction with generalized Hough transform in multiresolution sense. Their method is restricted to 2-D affine transformation. Jurie [7] incorporates a probabilistic error model rather than a bounded error model to improve the efficiency of the pose search algorithm. Hough transform-based methods suffer from the problems which are associated with it such as false peaks and quantization error. Genetic algorithms have also been utilized to search the transformation space [8,9]. However these methods require extensive computations and it is often necessary to incorporate additional information about the targets to narrow down the search space.

A different approach to pose computation with unknown correspondences is to generate random correspondences and try to minimize the error in an iterative manner. By constraining the feature correspondences in different way it is possible to reduce the complexity of computations. Generally the order of complexity of these methods is $O(M^a N^b)$, $b > 1$, $a \geq 1$ [10] which is expensive for interactive applications.

Several techniques have been proposed in order to make the feature matching more reliable and also to reduce the number of possible matches which is desirable for real-time applications. A wide variety of features have been purposed in the literature which are suitable for different applications. Han et. al.[11] use angle pairs defined by line segments with respect to a basis line segment as a feature set. They measure the Hausdorff distance between two feature sets to determine the similarity.

3. Pose estimation for augmented reality

In this work we integrate object recognition and tracking into an AR system called the Augmented Reality Virtual Human Director (VHD++) [11,12]. The initialization procedure consists of three steps. Suitable

features are detected by exploiting a priori knowledge about the targets. Here we use corners of facets as features. Hypotheses are constructed as feasible configurations of the features. We then use the set of hypotheses to identify the transformation which best describes the object pose. These steps are described in more detail in the following section.

3.1 Feature detection

Our goal is to select minimum number of vertices in the input image and also obtain a criterion to narrow down the number of acceptable hypothesis configurations to be investigated. This can be done by detecting facets in the image which represent the visible facets of a polyhedral object. Here we focus on polyhedral objects with parallelogram faces, however our method can be generalized to apply to other geometric shapes as well. This will allow robust vertex detection and greatly shrinks the number of acceptable points as features. Direct detection of parallelograms (e.g. 6D Hough transform) is not an easy task. Therefore, we try to investigate the connection between line segments to detect parallelograms.

In order to detect straight-line segments to build the basis for parallelogram detection, traditional methods such as Hough transform and probabilistic kernels have been used. These approaches lack sufficient accuracy in determining the endpoints of segments reliably. Furthermore, methods based on Hough transform generate multiple lines for each single segment. Adequate processing in the form of Hough space filtering and post processing is required to refine the results. Based on the above discussion we have used an approach to detect the segments which is based on image contours.

Image contours, extracted from the edge image are approximated by polygons. This allows us to determine linear parts of the contours easily by keeping polygonal edges which are longer than a preset threshold. Approximation of edge contours by polygons allows robust hypothesis generation and parallelogram detection. The disadvantage of the contours is that discontinuities in the edge segments results in multiple segments along a long edge. This problem is dealt with in the hypothesis generation step.

Later the spatial proximity of endpoints is used to cluster and label them. The labels are further exploited to merge the segments which have the same labels on both sides (i.e. they represent the same segment).

3.2 Hypothesis generation

Our object model consists of a set of labeled vertices of the polyhedral object. Vertices on the model are labeled based on the number of facets attached to them when seen in the image. This requires having different object models for cases when different facets of the object are visible or hidden in the scene. For ordinary polyhedral objects with planar symmetry this number is small (e.g. for a rectangular parallelepiped we require 4 object models for cases when three faces or two faces are seen- we do not consider the case that we see only one face).

The segments and their labels are used to detect parallelograms by checking their connection and parallelism. Analysis of the angles between the segments provides information about the way the segments are connected to each other. Furthermore, it can be exploited to merge the cut segments generated by contour detection at discontinuities in the edge image. We can also recover the fourth edge of a parallelogram in case it is not detected in the previous stage using the knowledge about the other edges as vectors. This gives robustness to our algorithm against partial occlusion. Figure 2 illustrates an example of occlusion which can be handled by our algorithm. Results of parallelogram detection for image of Figure 1.a is shown in Figure 1.c. Parallelograms are in turn used to construct our hypothesis set.

Vertices of the detected facets in the image can be labeled in the same manner as object vertices. For example for a cubic object, we can have vertices connected to 3 faces, 2 faces and 1 face as seen from a camera. The labels are used to determining which object point can be matched to which image points. This is equivalent in partitioning the set of image points into several subsets prior to establishment of correspondences (*viewpoint consistency*) and then restricting the correspondences within each subset (*geometry consistency*).

3.3 Computational complexity evaluation

Assuming an object model with f facets and M vertices, the first layer of partitioning divides the set of N image points $S = \{p_i | i=1, \dots, N\}$ into $N' \leq N/M$ subsets $S_k = \{p_i | i=1, \dots, M\}$, $k = 1, \dots, N'$, with M points which lie on a set of f connected facets in the image corresponding to the facets on the 3-D model. This is the constraint imposed by *viewpoint consistency* (i.e. the locations of all projected model features in an image must be consistent with projection from a single viewpoint.). Each subset S_k is further partitioned into r

subsets S_{kl} , $l=1, \dots, r$ where r is the number of labels of 3-D model features. Each S_{kl} has M_l elements so that $\sum_{l=1}^r M_l = M$. The ratio of improvement in the correspondence problem obtained by the second level of partitioning is obtained by the following expression.

$$\left(\begin{matrix} M \\ M_1, M_2, \dots, M_r \end{matrix} \right) = \frac{M!}{M_1! M_2! \dots M_r!} \quad (1)$$

The greater the number of labels, the greater the improvement achieved. For example for a cubic object with 7 vertices and 3 labels as defined before, this ratio is $7! / (1! 3! 3!) = 140$ and for an object with 11 vertices as shown in Fig. 2 this ratio becomes $11! / (3! 3! 5!) \cong 10^4$.

The set of hypotheses is constructed using the hierarchy of subsets. Each first level subset S_k is used independently to establish correspondence between object and image points. Within the points in S_k image points in each partition S_{kl} are matched against object points in the same partition S_{kl} (the object model is selected to have the same number of facets as the hypothesis). Finally the number of feasible correspondences between image points in S_k and M model points will become $M_1! M_2! \dots M_r!$ as compared to $M!$.

We repeat this procedure for all N' subsets. That yield total number of correspondences as $M_1! M_2! \dots M_r! N' \leq M_1! M_2! \dots M_r! N / M$. Therefore computational complexity of our method is $O(CN)$, with $C = M_1! M_2! \dots M_r! / M$. Since we usually deal fewer model points compared to image points, i.e. $M \ll N$, this approach has one order of magnitude of computational advantage over existing methods with complexity order of $O(M^a N^b)$, $b > 1$, $a \geq 1$.

3.4 Pose Estimation and Verification

Next, for each hypothesis configuration, pose of the object is estimated in terms of rotation and translation wrt 3-D model coordinate system using POSIT algorithm [2]. As a first stage of verification the rotation matrix obtained by POSIT algorithm is verified to have orthogonal columns. Next, an error function is evaluated and is minimized over the set of all possible configurations, S . The error function used is given in equation 2.

$$E_s = \sum_i \text{dist}(P[R \ T]M_i - m_i)^2 \xrightarrow{s} \min \quad (2)$$

Where P is the matrix of internal parameters of the camera (which is calibrated off-line), T and R are the translation and rotation matrix obtained by POSIT algorithm and M_i and m_i are 3-D object and 2-D image features respectively. Finally the verified results of pose estimation are passed to the tracker and the tracking phase is triggered.

4. Experimental Results

We made several experiments to evaluate the performance of our algorithm. We used models of simple polyhedral objects to test the recognition phase. Figure 1.d shows the position of the vertices of the detected target (scaled to fit on the page) among other polyhedral objects in the image of Figure 1.a. In one scenario a box is thrown on a table and the system correctly detects the object and computes its pose. This is shown in Figure 3, where a wire frame model is fitted on the detected target. After successful pose estimation the tracker is triggered. Currently, the initialization phase works reasonably fast (around 5 seconds).

5. Conclusion

We have proposed a method for polyhedral object recognition for augmented reality applications which focuses on constrained and partitioned hypothesis generation, in contrast to conventional approaches that tend to make approximations in the search method and space. We prune the set of acceptable hypothesis about polyhedral targets by detecting facets in the image using segment detection and robust labeling. Furthermore we exploit the assumption of *viewpoint consistency* together with the knowledge about *object geometry* to greatly constrain the correspondences. Our correspondence establishment scheme has one order of magnitude of computational advantage over other methods with complexity order of $O(M^a N^b)$, $b > 1$, $a \geq 1$, where M is number of model points and N is the number of image points and under the assumption that $M \ll N$. Attention has been given in robust detection of facets in the images. Occlusion is handled in two different manners. Segment labeling allows recovery of those facet edges which were not detected in segment detection stage or not seen in the image. The error minimization scheme also allows recognition of partially occluded objects. This is done by selecting the transformation which generates the closest approximation of available data about the targets with respect to the error bound. Our method is useful for applications that deal with recognition of well-defined polyhedral objects and require near real-time performance. It can be used to implement fully automated interaction systems in which

tracking and registration part can be automatically initialized at the start of the application or in cases where the tracking fails. We are currently working on extending our method to apply to more complex objects.

References

1. R.T. Azuma. A Survey of Augmented Reality. Presence, Teleoperators and Virtual Environments, 6(4):355–385, August 1997.
2. D.F. DeMenthon and L.S. Davis. Model-Based Object Pose in 25 Lines of Code. International Journal of Computer Vision, 15, pp. 123-141, June 1995.
3. C.-P. Lu, G.D. Hager, & E. Mjølness, “Fast and Globally Convergent Pose Estimation from Video Images,” IEEE Trans. PAMI, vol. 22, pp. 610–622, 2000.
4. David G. Lowe. Three-dimensional object recognition from single two-dimensional images. Artificial Intelligence, 31, 3 (March 1987), pp. 355-395.
5. P. Suetens, P. Fua, and A. Hanson. Computational Strategies for Object Recognition. ACM computing surveys, 4(1):5–61, March 1992.
6. Breuel, T.M. Fast recognition using adaptive subdivisions of transformation space. Proceedings of Computer Vision and Pattern Recognition, CVPR '92., pp. 445–451, 1992.
7. Jurie, F. Model-based object tracking in cluttered scenes with occlusions. Intelligent Robots and Systems, IROS '97, pp. 886–892 vol.2, 1997.
8. Kayanuma M., Hagiwara M. A new method to detect object and estimate the position and the orientation from an image using a 3-D model having feature points. Systems, Man, and Cybernetics, IEEE SMC '99 Conference Proceedings, pp. 931–936 vol.4, 1999.
9. Kawaguchi T., Baba T. 3-D object recognition using a genetic algorithm. Circuits and Systems, ISCAS '96, pp. 321–324 vol.3, 1996.
10. Phil David, Daniel DeMenthon, and Ramani Duraiswami. SoftPOSIT: Simultaneous Pose and Correspondence Determination. ECCV, 2002, Accepted for publication.
11. Han I, Il Dong Yu, Sang Uk Lee. Model-based object recognition using the Hausdorff distance with explicit pairing. Image Processing, ICIP 99. pp. 83–87 vol.4, 1999.
12. G. Sannier, S. Balcişoy, N. Magnenat-Thalmann, D. Thalmann. VHD:A System for Directing Real-Time Virtual Actors. The Visual Computer, Springer, Vol.15, No 7/8, 1999, pp.320-329.
13. R. Torre, S. Balcişoy, P. Fua, D. Thalmann. Interaction Between Real and Virtual Humans: Playing Checkers. Proc. Eurographics Workshop On Virtual Environments 2000.

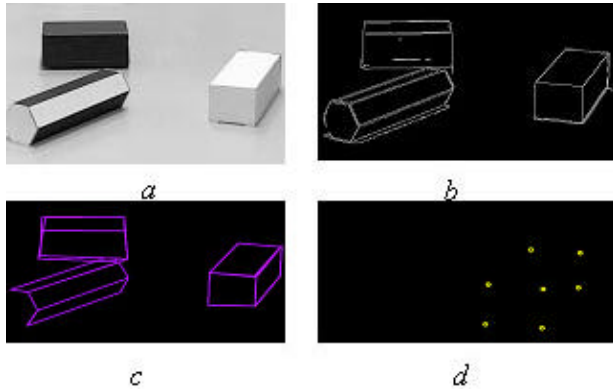


Figure 1. Recognition phase. a) Input image with the target shown in the right corner, b) edge image and c) detected parallelograms, used to estimate the pose of the target as shown in d.

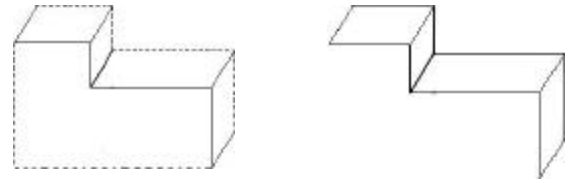


Figure 2. Handling occlusion. An example of a case where it is possible to recover the parallelograms even if all the dashed lines are missing in the image. Detecting the solid lines gives enough information to detect the parallelograms shown on the right and then performing pose estimation.

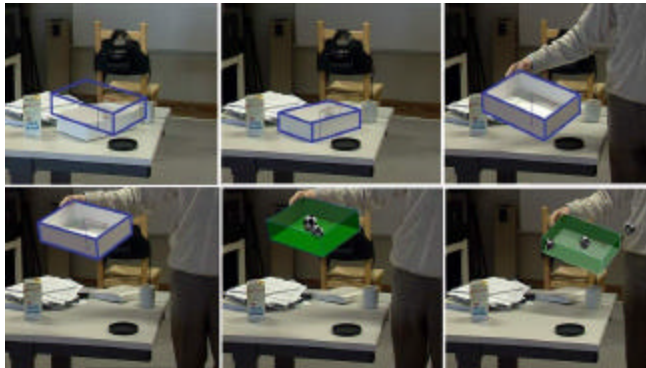


Figure 3. Integration of recognition results with the tracker. A box is detected on a cluttered background. The computed pose is used to initialize the tracker. The pose of the object in each frame provides the dynamics to control the virtual objects (balls).