Real-Time 3D Object Recognition for Automatic Tracker Initialization

Gábor Blaskó and Pascal Fua
Computer Graphics Lab (LIG)
Swiss Federal Institute of Technology
1015 Lausanne, Switzerland
Gabor.Blasko, Pascal.Fua@epfl.ch

Abstract

We propose a vision based real-time object recognition system, that provides object identification and 3D position data for the automatic initialization of a 3D tracking system. A-priori information is generated using the models of objects which may be present in the image. During recognition this data is accessed, using the position of corner features in the video image for indexing. A voting scheme is used to recognize objects and their positions in a single run. Upon recognition, a 3D object tracker can be automatically provided with initialization data.

1. Introduction

One of the main goals of Augmented Reality (AR) systems is to enhance the user's view of the real world with computer data, in the form of text, 2D images, or 3D graphics. It is desirable to have a system that recognizes objects in the user's environment, because the user should not have to constantly tell the system about the environment and trackable objects in it.

There have only been a few non-real-time systems that perform true 3D object recognition [3, 7]. Recently, appearance based systems have been proposed, that use multiple 2D image processing methods (e.g., color histograms, eigenspace matching, receptive field histograms [1], and probabilistic methods [2]). These systems treat 3D objects as a multitude of 2D images (appearances) with associated viewpoint data. They succeed in identifying complex objects, but they do not provide 3D pose data that is needed for AR tasks, such as augmenting an image of real objects with 3D graphics. Furthermore, they are inadequate for real-time interactive AR applications due to their time consuming algorithms for a-priori training and recognition.

Currently the major drawback for vision-based 3D tracking systems is that they have to be manually initialized by providing information about what objects are visible in the image, and in which part of the image they can be found. In the context of AR, automating tracker initialization was recently addressed in [6], where a textured planar object (paper poster) was found, tracked, and annotated with 3D data. Instead of finding a single colorful and textured planar object with color histograms, we address the problem of finding and recognizing textureless polyhedral 3D objects in a single image taken with a perspective camera.

The proposed system not only identifies the object in the image, but also provides 3D data about the placement of the object in the scene. With data provided by this recognition system, initialization of an object tracking system, such as [4], can be automated without requiring user interaction to begin 3D object tracking.

2. Object recognition

Almost all vision based recognition systems divide the recognition task into two parts: first, a time-consuming a-priori data processing phase, then a recognition phase, which accesses an optimally structured precomputed database to enable results to be produced as fast as possible. As discussed in Section 3, we have modified the *feature-pose* (FP) *map* memory-based technique described in [8] to enable real-time recognition and to decrease a-priori processing time.

The precalculated FP map is a table that associates recognizable image features to 3D objects and their possible poses as shown in Figure 1 for just one image feature.

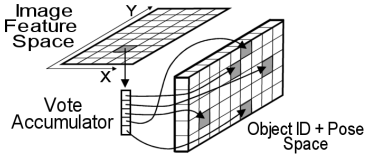

Figure 1. Feature-pose map

We precalculate and store a quantized version of the pose space for all objects based on model data. In a 2D image, a certain recognizable feature (e.g., corner) can be produced

by different objects in different poses. The FP map stores this 2D feature – 3D feature association. Upon recognition, we process the image and find the 2D features. We use these features as indices to retrieve from the database the possible objects and poses that could have produced them. We collect this data in a vote accumulator. The most frequent element in the accumulator identifies the object and its 3D position.

3. Implementation and Discussion

We modified the original FP map approach as follows, to increase its speed. We use a single FP map in memory for all objects, instead of one for each object. The original method uses post-processing to identify the object after the retrieval of hypothesis poses for each object. Instead we combine an Object ID index with Pose data in the FP map. Because of this, the peak in the accumulator provides both identification data as well as pose data in a single run without post-processing. Due to the lack of texture in our objects, we choose the corner as our 2D feature and the SUSAN [5] corner detector, because it is highly robust to image noise and extremely fast. By choosing this feature, we can decrease the a-priori processing time significantly, since the projection of the 3D model vertices into 2D image points can be calculated rapidly by using the graphics hardware. Other 2D features, such as edges could also be used to make the recognition more robust, however the extraction of these features is more time-consuming. We decided not to quantize the image feature space in the precalculated FP map to areas of multiple pixels, because there can be problems of oversampling or undersampling the feature space [8], whereby too many false positives can result without producing a high enough peak in the accumulator.

Our prototype system (450 Mhz Pentium III, Win NT) identifies low polygon objects that sit on a predefined plane. For a simple 8 vertex box object, the *precalculation* of the FP map takes less than a second, as opposed to several minutes or hours for other more complex precalculation methods. With such fast FP map generation, it is possible to regenerate the FP map database on-line when the camera's viewpoint relative to the plane changes or new objects have to be added to the compiled FP map on-line.

In the *recognition* phase, when the compiled FP map for all possible objects is in memory, the SUSAN feature extraction and the previously explained voting recognition cycle takes approximately 0.03-0.05 seconds to produce a result when processing half PAL video resolution (384x288 pixel) images. This performance is more than sufficient to recognize objects and provide their 3D positions on the plane surface in real-time. Figure 2 shows the placement of three different objects on a table, and the result of the recognition system, which selects and overlays the appropriate model based on the Object ID index and the retrieved position. The images were captured in real-time approximately 2 seconds apart.

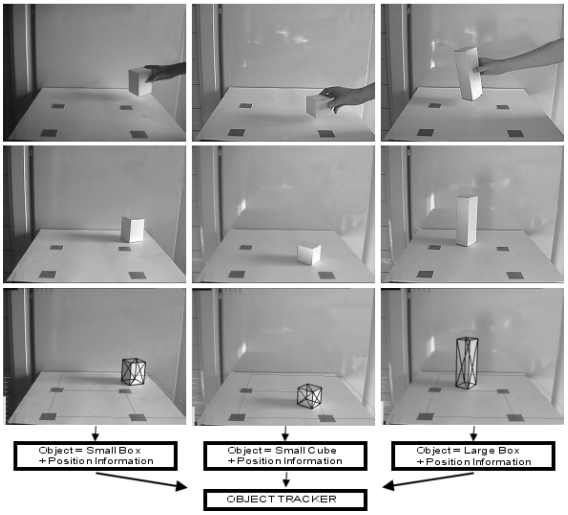

Figure 2. Recognition Results

4. Conclusions

We have presented an object identification system that is capable of running in real-time and constantly recognizing new trackeable objects in a video image. The system is able to differentiate between multiple objects, and at the same time, provides 3D pose data, which can be used to automate the initialization of our 3D object tracker [4].

References

[1] B. Schiele, N. Oliver, T. Jebara and A. Pentland. An interactive computer vision system. *ICVS'99 Intl. Conf. on Comp. Vision Systems*, 1999.

[2] A. R. Pope and D. G. Lowe. Probabilistic models of appearance for 3-d object recognition. *International Journal of Computer Vision*, 4(2):149–167, 2000.

[3] S. Procter and J. Illingworth. Foresight: Fast object recognition using geometric hashing with edge-triple features. *Proc. IEEE Int. Conf. Image Processing*, pages 889–892, 1997.

[4] R. Torre, S. Balcisoy, P. Fua, and D. Thalmann. Interacion between real and virtual humans: Playing checkers. *Proc. Eurographics Workshop on Virtual Environments*, 2000.

[5] S. Smith and J. Brady. SUSAN - a new approach to low level image processing. *Int. Journal of Computer Vision*, 23(1):45–78, May 1997.

[6] T. Okuma, T. Kurata and K. Sakaue. 3-d annotation of images captured from a wearer's camera based on object recognition. *Proceedings of ISMR 2001*, pages 184–185, 2001.

[7] H. van Dijk. *Object Recognition with Stereo Vision and Geometric Hashing*. PhD thesis, Univerity of Twente, 1999.

[8] M. Westling and L. Davis. Object recognition by fast hypothesis generation and reasoning about object interactions. *Proc. of ICPR96*, pages 148–153, 1996.