A Tracking Algorithm for Rigid Point-Based Marker Models

Bernd Schwald
Computer Graphics Center (ZGDV), Department Visual Computing
Fraunhoferstraße 5
D-64283 Darmstadt, Germany
bernd.schwald@zgdv.de

ABSTRACT

Tracking of objects and persons by using a stereo camera system setup and markers is applied since several years in Virtual and Augmented Reality or other applications. In especial infrared tracking systems are getting increasingly popular, because of their precision and robustness. Often, markers for such systems are spherical and therefore not distinguishable, making the consideration of geometric constellations of several markers necessary to identify objects and to determine their transformations. This paper presents an algorithm for tracking rigid constellations of markers, which can be adapted to the needs of corresponding applications.

Keywords

Tracking, Marker, Stereo Camera System, Geometric Constellation, Infrared

1. INTRODUCTION

Using punctual objects as markers for optical tracking is very typical for infrared stereo camera systems. Those markers are passive retroreflective spheres or active infrared diodes with a big angle of radiation. Such markers have the advantage, that they can be chosen quite small, but still their position can be reconstructed precisely in a large interaction volume.

To determine translation and rotation of objects, it is necessary to consider rigid constellations of markers, here called models, respectively Multi Point Models (MPMs). One important task is to achieve a high reliability of the detection of MPMs by regarding properties like distances between points and translation and rotation of the MPM in the previous frame. Nevertheless a limitation to some criteria for the detection of MPMs is necessary, in order to prevent extensive computations causing too much latency between capturing images and returning the calculated transformations.

Conference proceedings ISBN 80-903100-7-9
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

In the following, the existence of a set of $m \in \mathbb{N}$ 3D points $\mathcal{X}_W = \{X_W^1, \ldots, X_W^m\}$, reconstructed in one frame from synchronously captured camera images, is supposed. The 3D points are the result of a segmentation on the camera images and a triangulation thanks to a calibrated system, as described e.g. in [Har97], which are then processed by the tracking algorithm. The 3D points are the world coordinates of markers and therefore more or less accurate, depending on the used system. Furthermore, disturbing points not representing a really existing marker may be elements of this set.

2. MULTI POINT MODEL

A Multi Point Model (MPM) consists of parameters

- $\mathcal{X}_M$ the set $\{X_M^1, \ldots, X_M^n\}$ of 3D points, $n \geq 3$, and
- id a unique identifier for the model.

These parameters are defined once, when the model is initialised. From the set $\mathcal{X}_M$ of 3D model points the set $\mathcal{D}_M = \{d_M^{jk} \in \mathbb{R} : 1 \leq j \leq n-1, j < k \leq n\}$ of $n_d = \frac{n(n-1)}{2}$ distances between model points X_M^i can be calculated once and stored, whereas $d_M^{jk} = |X_M^j - X_M^k|$. Furthermore the model consists of parameters

- $\mathcal{K}$ the set of correspondences between reconstructed points X_W^j from the previous frame and model points X_M^i,
- R the model rotation from the previous frame and
- T the model translation from the previous frame.

Normally, these parameters change during tracking of the model in each frame.

3. DETECTION OF MODELS

For detecting a MPM, the model points X_M^i, representing the real markers, have to be known. In simple cases their coordinates can be measured by hand or by just placing the model once in the interaction volume of the tracking system. In more complex cases a special procedure, allowing movements of the model during the learning procedure [Sch04], can be applied.

The fact, that 3D points are reconstructed not perfectly, is important when comparing distances and point coordinates. To take this reconstruction error into account, a distance tolerance $\delta > 0$ is defined, expressing that a distance $d_{ij} = |X_i - X_j|$ between two points X_i, X_j is regarded as equal to a given distance $d \in \mathbb{R}$, if $|d_{ij} - d| < \delta$. For the world and the model coordinate system have the same scaling, the distance tolerance δ is not defined depending on a certain coordinate system, but is the same for all coordinate systems.

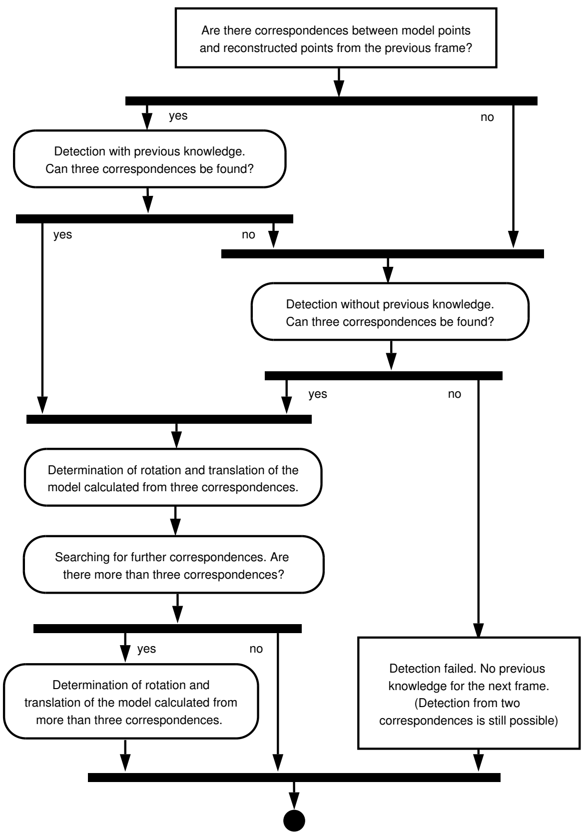

Figure 1. Tracking algorithm for MPMs based on both detection with and without previous knowledge from last frame.

Starting point for the detection of a MPM in the present frame is the set $X_W = \{X_W^1, \ldots, X_W^m\}$ of reconstructed points in world coordinates in one frame. Two different methods are applied: one method is based on previous knowledge from the last frame, the other does not use previous knowledge. In both cases the first step consists in trying to identify the MPM by finding three correspondences between model points X_M^i and reconstructed points X_W^j. Therefore the distances between the reconstructed points are calculated and compared with the distances stored in the model.

The detection method without previous knowledge is mainly based on comparing distances between world points with distances between model points. In order to make this more efficient, knowledge form the previous frame is added in the second method. Nevertheless the first method is important, because previous knowledge is not always available. Both detection methods, are combined in one tracking algorithm, see Figure 1.

4. DISCUSSION AND RESULTS

The presented tracking algorithm is implemented as part of an infrared tracking system, allowing to track several models. To evaluate the processing time, needed by the tracking algorithm, up to four models with two times three, four and nine markers were tracked. The tracking software was running on an Athlon XP 1800+ with 1 GB RAM. The models were moved within and out of the interaction volume, such that the number of actually tracked models varied between 0 and 4 at a given time. Table 1 shows the results from ≈ 700 measurements. The processing time Δ, includes image processing, 3D reconstruction from two camera images and the presented tracking algorithm.

#models	#tracked frames	average Δ in ms	average #markers	approx. #dist.
0	85	4.8	1.4	0
1	267	5.9	9.9	45
2	148	6.2	14.3	91
3	117	6.6	17.0	136
4	104	7.0	18.9	171

Table 1. Between 0 and 4 models were tracked in ≈ 700 measurements. Δ is the processing time for the corresponding number of tracked models.

Obviously, the processing time increases with the number of models and the number of reconstructed points, but ≈ 7.0 ms is an acceptable value for a setup with four models consisting of 19 markers. In especial, the processing time is not increasing the same way as the number of distances between reconstructed points does.

5. REFERENCES

[Har97] R. Hartley and P. Sturm, Triangulation, Computer Vision and Image Understanding 68 (1997), no. 2, 146–157.

[Sch04] B. Schwald and P. Figueiredo, Learning of Rigid Point-Based Marker Models for Tracking with Stereo Camera Systems, 1. Workshop der GI VR/AR (Chemnitz), 2004, pp. 23–34.