# A Hybrid Tracking Solution to Enhance Natural Interaction in Marker-based Augmented Reality Applications

Rafael Radkowski, James Oliver
Virtual Reality Applications Center
Iowa State University
Ames, USA
rafael | oliver@iastate.edu

***Abstract*—In this paper a method for enhanced natural interaction in Augmented Reality (AR) applications is presented. AR applications are interactive applications, designed to facilitate the handling of virtual objects, represented by a physical proxy object. Ideally, interaction should be natural, in that the user should not notice its technical realization. Interaction capability relies on tracking technology, which enables spatial registration of virtual objects in 3D. Markers are a common solution for this. However, the marker must stay in line of sight of a video camera. In highly interactive applications, the user's hands regularly cover the markers. Thus, the virtual object disappears. This paper describes a hybrid tracking solution, which incorporates marker-based tracking and optical flow-based tracking. The optical flow-based tracking supports the marker-based tracking: it seeks corners of the marker to keep track of them. If no markers are visible, the optical flow tracking extrapolates the position of the object to track. Thus, the virtual object remains visible. A prototype implementation and example application show the feasibility of the solution.**

***Keywords- augmented reality; hybrid tracking; interaction***

## I. INTRODUCTION

Augmented Reality (AR) technology is a type of human-computer interaction that superimposes the natural visual perception of a human user with computer-generated information (i.e., 3D models, annotation, and text) [1]. AR presents this information in a context-sensitive way that is appropriate for a specific task, and typically, relative to the users physical location. Special viewing devices are necessary to use AR. A common viewing device is the so-called head mounted display (HMD), a device similar to eyeglasses that use small displays instead of lenses.

AR applications are designed to facilitate natural interaction. In general, an interface is described as natural when its technical realization is effectively veiled from the user [2]; i.e., the user should not consider how the interaction is working.

Spatial tracking is a key factor of AR and the interaction within an AR application. The term tracking denotes the continuous determination of the position and the orientation of a physical item (i.e., the user, a video camera, etc.). Marker-based tracking is a commonly used tracking technique in the field of AR. It is based on fiducial markers, which incorporate a defined geometry for position and orientation estimation and a unique pattern for identification. A video camera captures an image of a maker and a computer vision algorithm locates and identifies the marker in the image and estimates the spatial relationship between the marker and camera. The pose of that marker is thus known and a 3D model can be rendered and composited with the video image with correct alignment to the user. Usually markers need to be attached on the body of a physical object to track it. Marker-based tracking is popular, because applications are easy to deploy, the technique is convenient to use, and the algorithms are mostly free. The ARToolkit, for instance, is a common tracking system that works this way [3].

However, marker-based tracking technology can hinder the naturalness of the AR user interface. Marker tracking requires a free line of sight between camera and marker [4]. Today solutions suffer performance degradation due to the hands of the user that partially covers the markers. For example, in a highly interactive application, e.g., an assembly training application, the user's hands regularly cover the marker. An assembly training application displays assembly instructions as 3D models and text annotations spatially registered to physical objects. The user needs to grasp these objects in order to assemble them. While doing so the user's hands cover a large area of the scene often occluding the marker from the video camera's field of view. The consequence is, the virtual information disappears. This problem occurs in many highly interactive applications.

To address this problem, this paper describes a hybrid marker-based tracking system. The tracking system uses multiple video cameras. The marker information of all cameras is merged to one so-called marker object model. Thus, the probability of losing the marker is reduced. In addition, the marker and physical object movement is analyzed using optical flow. When no marker is in the line of sight of a video camera, its position and orientation propagation can be estimated using a Kalman Filter-like approach. Thus, all markers can be hidden for a short time.

The paper is structured as follows. In the next section related work is introduced. Section 3 describes the hybrid tracking system, which is the basis for the enhanced natural feature tracking. Section 4 demonstrates the advantages of this approach for enhanced natural interaction. An assembly application is used as an example. The paper closes with a summary and an outlook for future research.

 ISBN: 978-1-61208-250-9 

## II. Related Work

The related work addresses two areas: fiducial marker tracking systems and hybrid tracking systems. The first subsection presents existing maker based solutions. The second subsection introduces approaches that increase tracking stability and improve the interaction capabilities. It concludes with a summary that explains the gap.

### A. *Fiducal marker tracking*

Kato et al. [4] presented the ARToolKit marker tracking, which is the basis for this work. The ARToolkit uses template markers that consist of a black border with a unique image embedded. The ARToolkit uses the image to identify the marker and the black border to calculate the position and orientation of the marker with respect to a video camera. It is easy to use and an application is easy to deploy. However, the tracking quality depends on the lighting conditions, and tracking is impossible when a marker is partially covered.

Fiala [5] introduced ARTag, a fiducial marker tracking system that uses and id-enhanced marker. An id marker uses a binary pattern that represents a digital code. The ARTag is more robust than the ARToolkit: a marker can be partially covered without loss of tracking.

Wagner et al. [6] introduced a grid dot marker. They draw a grid of dots on a common map to combine a fiducial marker system (grid of dots) with a feature-based marker system. Thus, the marker can also be partially covered.

Wagner et al. [7] also developed the ARToolkit Plus. It is an enhancement of Kato's ARToolkit that is specialized for mobile devices.

Naimark et al. [8] developed data matrix marker. That is a fiducial marker tracking system with a 2D barcode surrounded by a black frame. The barcode incorporates a set of black and white areas. The pattern of black and white corresponds to a marker ID.

Recently, Uchiyama et al. [9] presented random dot markers. A random dot marker is a set of distributed points on a limited area. This random set of points acts as a template that facilitates the identification of the marker.

In summary, fiducial marker tracking systems work well and are widely used in academic research. However, they have different limitations. For instance, the robustness of the tracking system depends on the image quality and the light conditions in the surrounding environment. If the tracking system retrieves a video image that contains noise, it can cause jitter of the calculated pose. In addition, the tracking algorithm utilizes computer vision, whose feasibility depends on the lighting conditions. If the image processing does not comply with the lighting conditions, it can also causes jitter.

### B. *Hybrid Tracking*

Hybrid tracking systems are developed to address the limitations of a single tracking technology. Usually they merge the tracking data of two or more tracking systems to facilitate robust and jitter-free tracking of a physical object.

Seo et al. [10] developed a hybrid tracking system that solves jitter and occlusion problems of common fiducial markers. They use a fiducial tracking system and enhance the tracking with a corner tracking system. They use Kanade-Lucas-Tomasi Feature Tracker to detect and track additional key points of a marker. This data is merged with the pose calculated by the marker tracking system to facilitate a robust tracking.

Marimon et al. [11] developed a hybrid tracking system that incorporates a marker tracking system and a particle filter-based tracking system. The particle filter searches for feature points of the marker and estimates a 3D pose. The feature points are described as a 2D back-projection of the 3D corners.

The tracking systems presented in [10] and [11] work similar to the approach taken in this research. However, both systems require a corner model of the marker, which facilitates a continuous tracking of the marker. In addition, the systems have been tested with one fiducial marker only. Thus, it is unclear, if they work with multiple markers. Finally, they have not been tested in a realistic application scenario like virtual assembly training.

Piekarski et al. [12] combined a marker tracking system with a GPS (global positioning system) tracking system to facilitate indoor and outdoor tracking with a single system. They attached large-sized markers on the walls and the ceiling of a room and tracked them with multiple cameras. The estimated pose is merged with the GPS position. However, their tracking system facilitates indoor tracking and is not designed to track objects with which a user interacts.

Kalkusch et al. [13] incorporates marker tracking and inertial tracking to facilitate robust indoor tracking. Their system requires markers on the wall of a building to track the position of the user. The inertial tracking acts as a backup system for the marker tracking. The system is also not designed to track objects.

Yang et al. [14] proposed a camera tracking approach that combines inertial sensor tracking and marker-based tracking. They use the data retrieved from the inertial sensor to reduce jitter. Yang et al. [15] combines a marker-based tracking system and a feature-based tracking system to realize a robust tracking for their AR book. They use small-sized marker (2cm x 1.3cm) to identify the pages of the book and combine it with a random tree, a machine learning computer vision method to describe and detect feature points. The pose retrieved from the marker tracking is used as the initial position for the feature tracking. Thus, enhancing accuracy.

Fischer et al. [16] presents a hybrid tracking approach that incorporates tracking data from an infrared tracking system and a computer vision-based tracking system. The infrared tracking system tracks rigid objects. A computer vision algorithm tracks distinct points on the subject to track. The data of both systems is merged to reduce noise.

The research summarized above is only part of all relevant works. Further approaches can be found in [17], [18], and [19]. In general, most of the approaches are similar: they incorporate two or more tracking technologies to address the limitations of a single technology.

However, the hybrid tracking systems do not generally consider the interaction of the user. If an object to be tracked must also be grasped by the user, markers are occasionally covered. Thus, the marker tracking is limited. Robust tracking in this scenario remains a challenge.

In addition, the tracking systems presented generally use two equivalent tracking systems and merge data from both. In contrast, in the approach taken in this research, only a backup system is necessary that becomes active when the main tracking system is not able to track the fiducial marker. The advantage of this approach is robustness and computational efficiency.

## III. HYBRID TRACKING

This section presents the hybrid tracking. First, an overview of the tracking setup and the tracking process is presented. Secondly, the hybrid tracking approach is described in detail, including the mathematical foundations and logic during tracking.

### A. *Overview*

Figure 1 shows an overview of the hardware setup for the hybrid tracking solution. This implementation employs a monitor-based AR application setup in which the resulting superimposed video stream is shown on a monitor, with screen oriented face-to-face with the user. The system is designed for a single user. However, since it is a monitor-based AR application more persons can see the output video image.

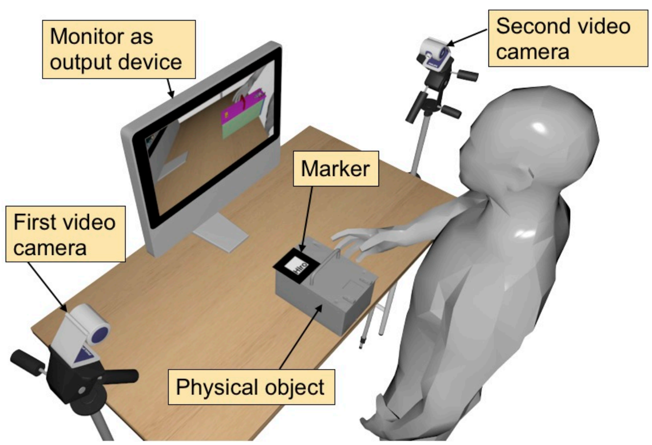


Figure 1. The hardware setup for the hybrid tracking solution.

The user works with a physical object that is located on the table-top work area. It is the object to be tracked. Therefore, one or more fiducial markers are attached on the body of this object. Two cameras are used to retrieve video images for tracking. The cameras must be arranged at different positions around the working area. After calibration, they must remain at their position and cannot be moved. The working area is roughly 1m in square and depends on the resolution of the camera and the size of the marker. This setup is a common setup for industrial applications like virtual assembly training or maintenance training. It is inexpensive, works with common computer hardware, and is easy to deploy.

The approach taken for hybrid tracking in this research is to use the marker tracking as the primary source for tracking and to back it up with optical flow tracking. It works similar to a Kalman Filter with measurement and prediction components.

Figure 2 shows an overview of the hybrid tracking method. The starting point are two images that are retrieved from the video cameras. The left side (green boxes) of the figure depicts the common marker tracking that is extended by a position estimation function. Using a Kalman Filter analogy, this part of the system acts as the measurement. The right side (blue boxes) of the image shows the optical flow tracking. It estimates the movement of an object. Thus, it can be considered analogous to the prediction part of a Kalman Filter. However, it is an extrapolation only. The result is a matrix in homogenous coordinates that describes the pose of the tracked object and is used as transformation data for virtual objects.

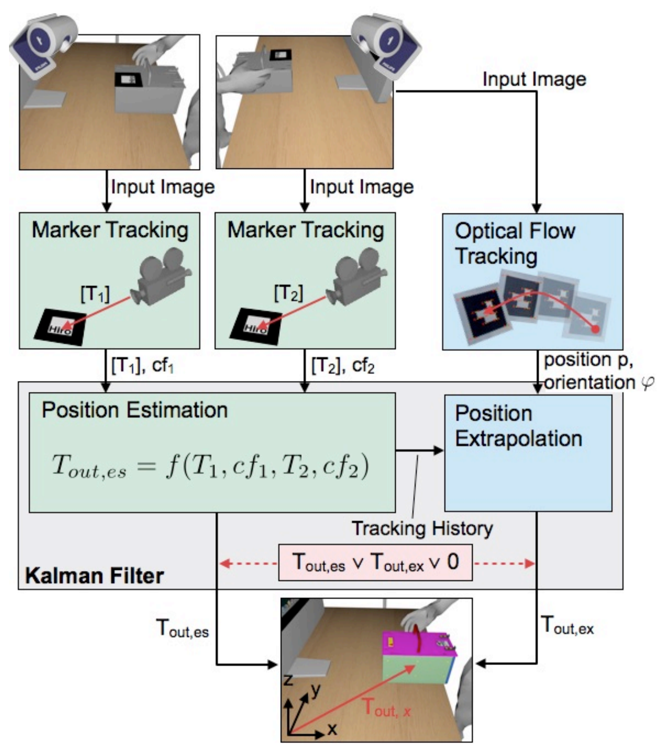


Figure 2. Overview of the hybrid tracking method.

The marker tracking module tracks the fiducial markers that are attached on the body of a physical object. In contrast to the common approaches, multiple cameras are used to track markers. Each module is related to one camera. A module identifies and tracks the markers separately, without information about the other tracking module. It provides a matrix in homogenous coordinates that keeps the position and orientation information of the marker with respect to a global coordinate system, regardless of whether a single marker or a multi marker (a set of multiple single markers that act similar to a single marker) is used. In addition, the module provides a confidence value $c$ for each single marker. This confidence value describes how certain the module is that the provided information belongs to a marker [3].

The position estimation module gets the information of both marker tracking modules and estimates a position and orientation of the marker (and the tracked object) with respect to a global coordinate system. Figure 2 shows two marker tracking modules, but theoretically, the position estimation facilitates the estimation on the basis of an arbitrary number of tracking modules. The estimated position is denoted as $T_{out,es}$.

The optical flowtracking module tracks corners of the markers. The movement of each corner is tracked frame by frame. The tracking data is merged, and the propagation of the position $p$ and orientation $\varphi$ is calculated.

The position extrapolation module uses the data of the optical flow tracking to extrapolate the locomotion of the physical object position and orientation. In addition, it gets access to a pose history of the markers. The outgoing data is denoted as $T_{out,ex}$,

Following a Kalman Filter-like approach, the marker tracking data $T_{out,es}$ is used when at least one marker tracking module is able to track a marker. The extrapolated position and orientation data $T_{out,ex}$ is used when no marker tracking module provides tracking information. Finally, the data provides the transformation information $T_{out,x}$ (with $x$ denoting either $ex$ or $es$) for spatial registration of a 3D model.

In the following, the marker tracking module and its pose estimation is explained. Then the optical flow tracking and the pose extrapolation is described. In the description the tracking is described for one marker only in order to simplify the presentation. The method can track more than one marker or multi marker.

## B. Marker Tracking

The objective of the marker tracking modules is to estimate the position and orientation of a single marker or multi marker with respect to a global coordinate system. Figure 3 shows the relationships between marker coordinates, camera coordinates, and the global coordinate system.

The desired position and orientation of the marker is denoted by $T_i$., where $i$ is the number of the camera. It is the global position, calculated by one tracking module and camera $i$. It is calculated by:

$$\mathrm{T}_i = T_{i,w->c} + T_{i,c->m} \tag{1}$$

$T_{i,c->m}$ is the relation between the camera and the marker in the camera coordinate system of each camera. The ARToolkit is used for marker tracking [3], which provides the relation $T_{i,c->m}$.

The position and orientation of each camera $T_{i,w->c}$ is calculated during an initial calibration procedure. Therefore, a calibration marker is used. All cameras must see this marker at the same time. Thus, the marker coordinate system can be assumed as the global coordinate system. All cameras must remain at their position after its global position has been specified.

The result of the marker tracking is a matrix with the global pose $T_i$ of each tracking module $i$, in homogenous coordinates. In addition, a confidence value $cf_i$ is provided. The ARToolkit provides it for each marker.

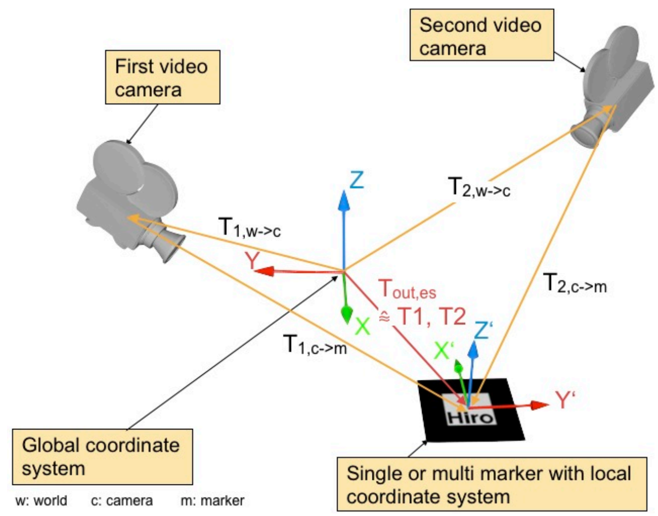


Figure 3. Relationships between the cameras, the marker coordinate system, and the global coordinate system.

## C. Pose Estimation

The objective of the pose estimation is to calculate the global position of the marker $T_{out,es}$ with respect to the world coordinate system (Figure 3). Therefore, a linear combination merges all pose data into one resulting value, considering the confidence value of each marker:

$$T_{out,es} = \sum_{i=0}^{n} T_i \frac{c_i}{c_n} \tag{2}$$

with $T_i$, the value of the single marker and $c_i$, its confidence value, and $n$, the amount of used marker tracking modules. The value $c_n$ is the sum of all values $c_i$:

$$c_n = \sum_{i=0}^{n} c_i \tag{3}$$

Thus, the ratio $c_i/c_n$ rates the quality of the marker tracking.

When a multi marker pattern is used, all single markers provide their own confidence value. In this case the value $c_i$ is the arithmetic average value of all single markers' confidence values.

In addition to the linear combination, two further improvements can be used: a priority of the cameras and a threshold value. The priority of the cameras allows it to specify one camera as a primary tracking camera. This is helpful when it is known in advance that one camera has an uncovered view to the markers most of the time. The priority for a camera, in this case for camera $j$, can be modeled using the confidence value:

$$c_j \geq c_{threshold} \implies c_i = 0.0 | \forall c_i \ni c_j \tag{4}$$

and

$$c_j < c_{threshold} \implies c_i := max\{c_0 \ldots c_m\} \quad (5)$$

with $c_{threshold}$, the minimum accepted confidence value, and $c_j$, the confidence value obtained from camera $j$. This acts as a switch. As long $c_j$ exceeds $c_{threshold}$, the marker data are used. Otherwise, the next best marker data are used for tracking.

The result of this module is a value that describes the pose of the marker $T_{out,es}$ with respect to a global coordinate system.

*D. Optical flow tracking*

The objective of optical flow tracking is to determine the movement of the physical marker. A commonly used optical flow tracker, the Lukas Kanade-Tracker [20], is adapted for this application. Figure 4 shows the six-step procedure of the optical flow tracking approach. In the first step, a region of interest (ROI) is calculated. The ROI describes an area in the image in which the optical flow tracking can probably find a marker. The ROI is located at the last known position of the marker. Its size correlates to the marker size scaled by the factor 1.3 to ensure that all significant points will be inside the ROI.

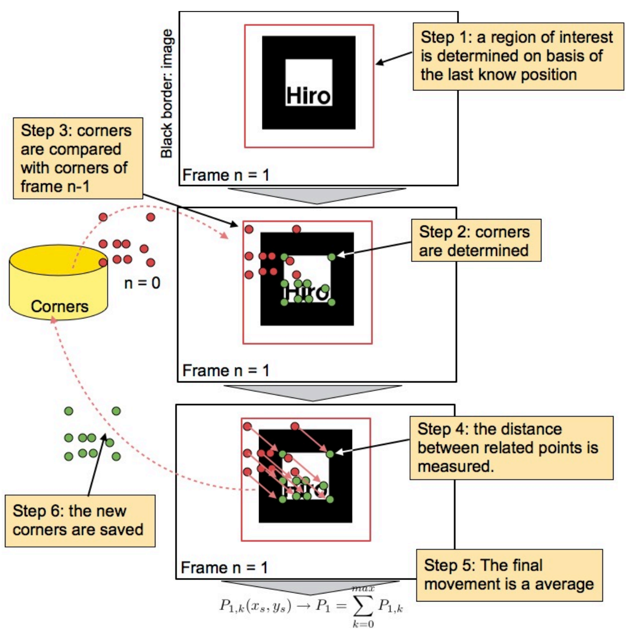


Figure 4. Concept of the optical flow tracking

In the second step, the optical flow tracker searches for corners inside the ROI frame. The Lukas Kanade-Tracker utilizes Shi Tomasi-Corners [21]. Each identified corner $P$ is represented as a two-dimensional point $p_k\,(x_s,\, y_s)$ in screen coordinates. The corners have to be strong corners according to Harris and Stephens [22]. All corners that are used for tracking have to meet the criteria:

$$P_1 := \{p_k, ROI(p_k) \cup e > e_{threshold} | p_k \in P\} \quad (6)$$

with ROI, the region of interest function, $e$, the eigenvalue of the corner and $e_{threshold}$, the minimum allowed eigenvalue.

In the third step, the set $P_1$ is compared with a predefined set of corners. The predefined set of corners $P_0$ describes the geometry of the pattern. It is defined during an initialization step. For this comparison the Lucas Kanade-optical flow method is used [21]. It provides all points of the set $P_1$ that are also part of the set $P_0$:

$$P_{final} := P_1 \cup P_0 \quad (6)$$

Next the movement of each corresponding point is calculated and summed (Step 5) to one movement vector by:

$$\Delta p = \sum_{k=0}^{max} p_{1,k} - p_{0,k} \quad (7)$$

with $p_{1,k}$ the corner of the set $P_0$ and $p_{1,k}$, the corners of set $P_1$, and $k$, the corner index. The result of this step is $\Delta p$, the movement of the marker between two frames in frame coordinates. In addition, the value $\Delta\varphi$, is calculated, which describes the orientation change of the vector $\Delta p$.

Finally in step six, the new corner points $P_{final}$ are stored. These steps are continuously repeated.

Figure 5 shows a test application for the optical flow tracking. The main image shows the output image (taken from a video) and contains one marker. A blue box is rendered on top of this marker. The black box on the bottom of the image is a miniaturized window of the main image. The white area shows the region of interest. The window on the lower right corner is a debug window for optical flow tracking. The red dots are the corner points and the red lines depict the movement of the pattern between two frames.

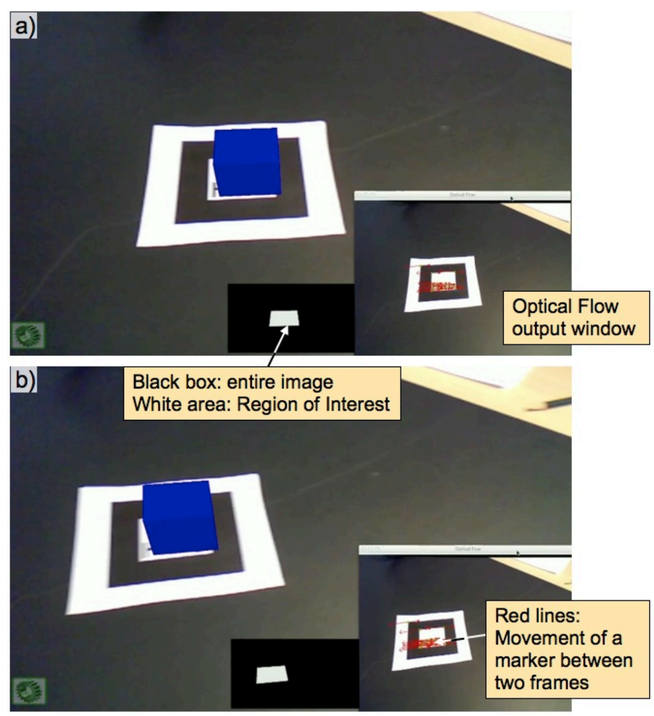


Figure 5. Optical Flow tracking test application

### *E. Pose Extrapolation*

The objective of pose extrapolation is to extrapolate the movement of the recognized pattern and to describe it in a special coordinate system. Therefore, a Kalman Filter-like method is used. In general, a Kalman Filter is a mathematical method that uses measurements of a system observed over time and a prediction model to estimate the current state of a system [23]. Usually, the prediction model is based on a physical model, which describes the behavior of the physical system under observation. The solution developed here works similar to a Kalman Filter in principle. However, the Kalman Filter uses a model to predict the next value. Our approach uses a second measurement and an estimation to predict a next value.

Figure 6 shows an overview of the pose extrapolation process. Input data for this process is the output of the optical flow measurement and the previous, as well as, the current marker tracking data.

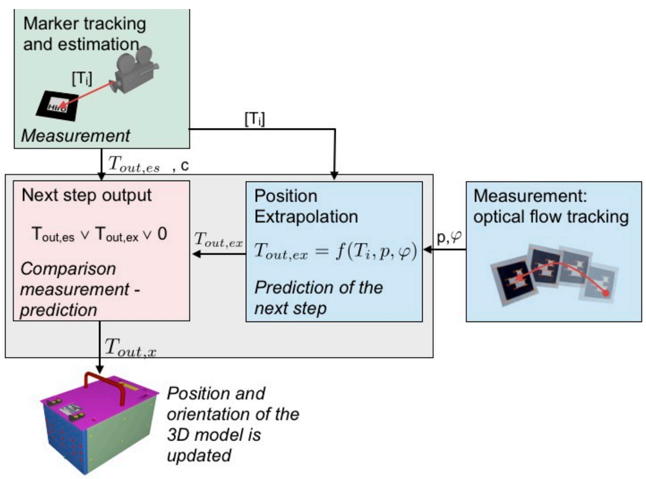


Figure 6. Overview of the position extrapolation

The observed measurement, in the sense of a Kalman Filter, is the marker-based object tracking and estimation process. The prediction process is a position extrapolation, which is based on optical flow tracking. The position $T_{out,ex}$ is extrapolated by a linear approximation of the movement of the physical object:

$$T_{out,ex} = T_{out,es,t-1} + (K \times \Delta p) + (K \times \Delta\varphi) \tag{8}$$

with $T_{out,ex,t-1}$, the previous tracking data, $\Delta p$, the movement of the pattern, and $\Delta\varphi$, the orientation of this vector. $K$ is a camera projection matrix, which projects the 2D values of the optical flow from pixel coordinates to 3D coordinates. A common camera projection is used according to the camera calibration according to Tsai [24].

To determine the output for the next step, the marker tracking data and the position extrapolation data are compared. Two error values are used that act as a probability value for both measurements. The probability value for the marker tracking is the confidence value $c$, which the ARToolkit provides. The probability value $c_p$ for the position extrapolation is a time and distance dependent value that is based on the time of the last update of $T_{out,es}$. and the moving distance $\Delta p$:

$$c_p = \begin{cases} 1 & : \quad t = t_0 \\ -\Delta p_s \cdot t + \frac{1}{\Delta p_s} & : \quad t > t_0 \end{cases} \tag{9}$$

with $\Delta p$, the movement of the pattern between two frames, $t$, the current time, and $t_0$, the time of the last update of the values $T_{out,es}$. Because the position extrapolation is based on the pattern position and orientation, it is assumed that its quality and accuracy decrease with time. Thus, the function is time dependent. In addition, it depends on the movement distance. Short movements are assumed because they can be estimated with a higher accuracy than long movements. This assumption is considered by the vector $\Delta p_s$, which is a scaled version of $\Delta p$.

The final position $T_{out}$ is determined by:

$$T_{out} = \begin{cases} T_{out,es} & : \quad c \geq c_p \\ T_{out,ex} & : \quad c_p > c \wedge c_p \geq c_{p,threshold} \\ 0 & : \quad c_p > c \wedge c_p < c_{p,threshold} \end{cases} \tag{10}$$

with $c_{p,threshold}$, a threshold value (specified empirically) that limits the minimum acceptable value $c_p$. If no value is sufficient, the output is set to 0, and the 3D model disappears. This is necessary because a user may intend to remove a marker and the object from the working surface. Thus, the 3D model must disappear.

## IV. INTERACTION ADVANTAGES

In this section the advantages for interaction gained by the hybrid tracking solution are explained. Two tests were conducted to demonstrate these advantages. The section starts with an introduction of the utilized hardware and software. Next, the first test is presented that shows the advantages. The second test results in a set of quantitative measurements, which underpin the results of the first test.

The application example is an AR assembly support application, which presents an assembly sequence as a set of 3D models superimposed on the parts to be assembled. The user handles the parts to be tracked and regularly covers the markers.

### *A. Application setup*

The application runs on a Windows 7 PC with Intel Xeon 3.6 GHz Processor and 6GB RAM. The graphics card uses a NVIDIA Quadro 5500 processor. Two Creative LiveCam Chat HD video cameras are implemented for tracking with a resolution of 1024 x 768 pixels at 30 fps. The cameras were arranged on the left and right position of the working area. The average distance between working area and both video cameras was 80cm.

Figure 7 shows an image of the application main window. WorldViz Vizard 4.02 (http://www.worldviz.com) was used as the development platform with the ARToolworks ARToolkit multi marker tracking for marker tracking (http://www.artoolworks.com/).

Vizard is a platform that facilitates the development of virtual reality applications. It uses Python as its programming language. ARToolworks is a plug-in for

Vizard. It provides functions for AR applications like image capturing, tracking, and spatially correct rendering. The optical flow tracking has been implemented with OpenCV 2.3 (http://opencv.willowgarage.com/wiki/), an open source computer vision library. It is written in C++ and provides functions to realize optical flow tracking. The optical flow tracking algorithm described above was developed as plug-in for Vizard in order to integrate this functionality with the AR application.

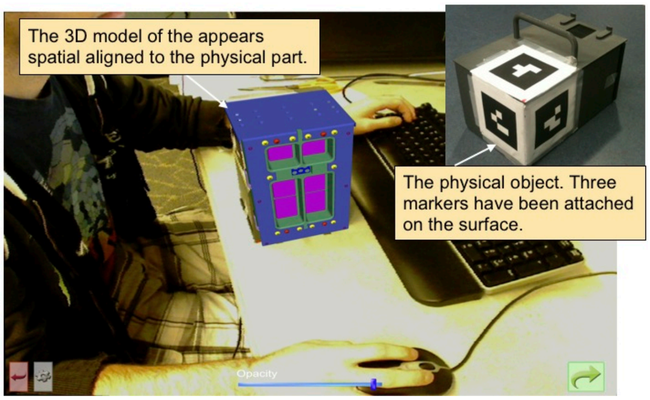


Figure 7. Main view of the AR application

## B. Interaction test

The objective of this test was to assess the performance of the hybrid tracking in comparison to the common ARToolkit tracking. In the following, the term single-camera tracking refers to common ARToolkit tracking, i.e., the tracking of all markers using one camera only.

The question addressed by this test is: does the hybrid tracking facilitate manual interaction with an object to track? Hybrid tracking is assumed to be more robust than single-camera tracking: the physical object should be tracked continuously even when the markers are out of view of one camera or partially covered.

The physical object shown in Figure 7 (upper right corner) was used as a test object. Three ARToolkit markers were attached on the body of this object. Each marker has a size of 50mm in square. The AR application superimposes a video image with a 3D model of the assembly of that object.

Figure 8 shows images of the video sequences of the test. The images show the virtual object superimposed by a 3D model. The user rotates the objects in order to inspect it. A 90-second video sequence was used for this test. The video ensures equal input data for hybrid tracking and single-camera tracking.

During the test a user interacts with this object. The user grasps it with two hands, turns it, and inspects it from different orientations. Thus, the object changes its orientation to the camera continuously. While doing so the three markers are partially covered by the hands of the user.

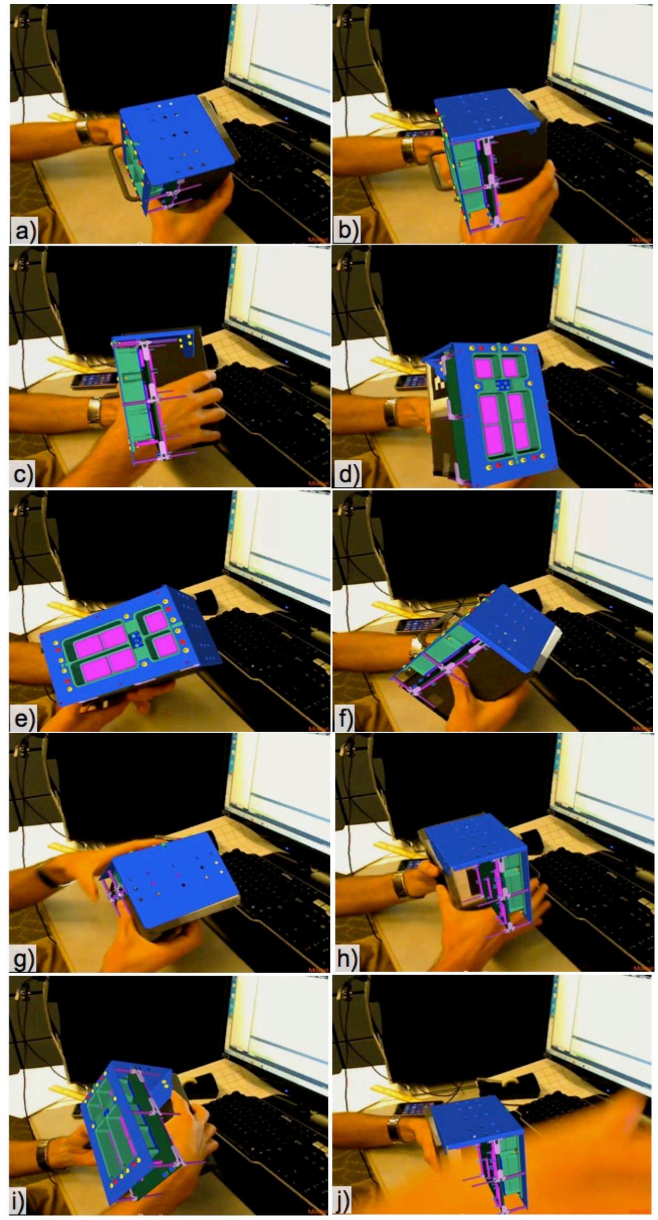


Figure 8. Image sequence retrieved from the test video

In Figure 8h) marker visibility to one video camera is partially covered by one hand of the user, however the 3D model was always visible and in its expected position.

To measure the robustness of the hybrid tracking and the single-camera tracking the confidence value $c_n$ (Eq. 3) was recorded. Figure 9 shows a diagram with the results. The abscissa displays the number of frames, the ordinate the confidence value $c_n$. The upper graph depicts the results of the hybrid tracking, the lower graph the results of the single-camera tracking. The horizontal sectioning of the charts shows the threshold values, thus, it shows the mode of operations according to Eq. 10.

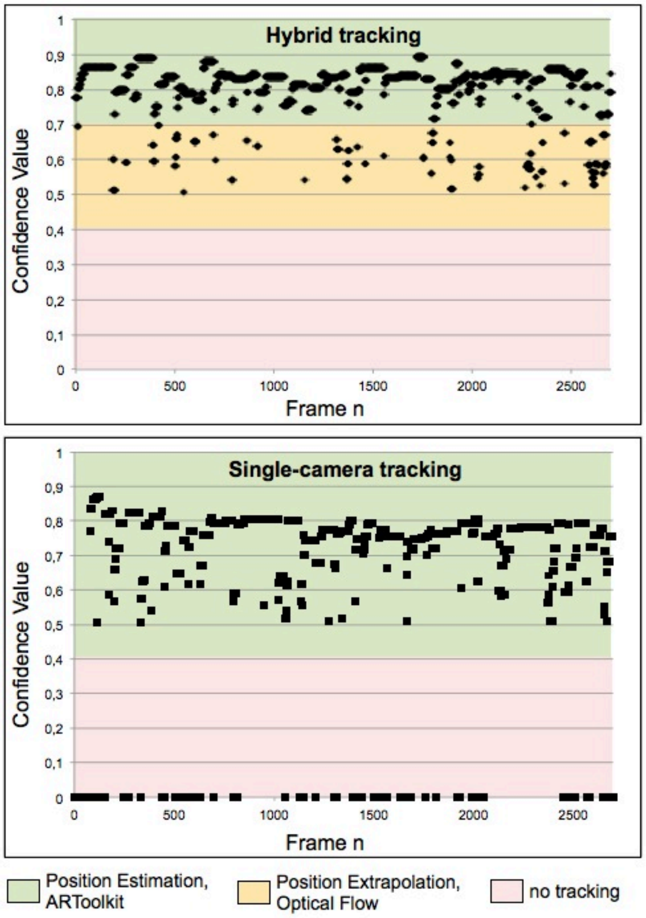


Figure 9. Confidence value of the a hybrid tracking test and a single-camera test.

For the hybrid tracking: the AR application switches between a position estimation mode (upper area), the position extrapolation mode (middle area), and an off mode when the confidence value declines beneath the given threshold. For the tests, a threshold of $c_p = 0.7$ and $c_{p,threshold} = 0.4$ were used. Below a value of 0.4 the AR application hides the 3D model. For the single-camera tracking: the threshold also was $\boldsymbol{c_p} = 0.4$.

The results show, the hybrid tracking can continuously track the physical object and show the 3D model. The confidence value moves within a range of 0.5 and 0.9. Thus, the AR application switches between the position estimation mode and the position extrapolation mode multiple times, but it never declines below the threshold value $c_{p,threshold}$. Thus, the 3D model never disappears during this test.

In comparison, the single-camera tracking is not able to track the physical object continuously. The graph shows that the confidence value occasionally declines below the threshold value $c_p$. In this case the AR application hides the 3D model.

### C. Optical Flow Test

The objective of this test was to assess the performance and the advantages of the optical flow tracking. The optical flow tracking should be able to track the marker even if the marker is partially covered and the ARToolkit is not able to detect the 3D model. For this test a single ARToolkit marker was used that showed a semi-transparent 3D cube (Figure 10). The edges of the cube are aligned to the edges of the 3D model. Thus, an incorrectly calculated position and orientation of the cube can be observed.

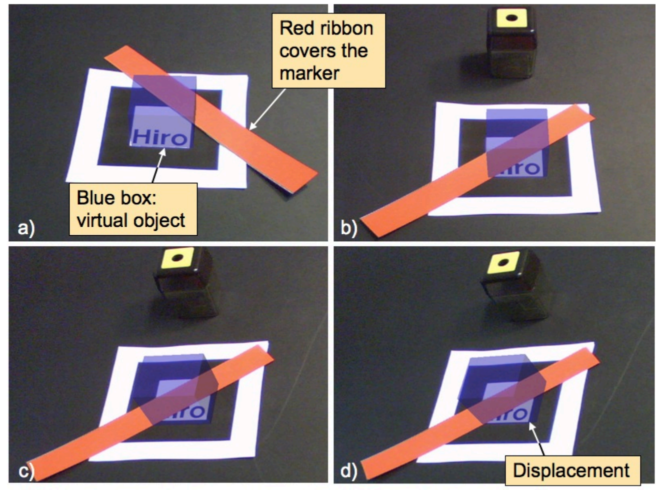


Figure 10. The optical flow tracking backs-up the ARToolkit marker tracking. Thus, partially covered markers could be tracked.

Initially, the position and orientation of the marker was calculated by the ARToolkit tracking. It was disabled after an initial transformation matrix was calculated. During the test a user covers the ARToolkit marker partially using a red ribbon. When doing this, the 3D cube must remain at its position (Figure 10). In addition, the camera was moved in one test (Figure 10 b-d). The images indicate that, in general, the tracking is able to estimate the position and orientation of the marker and to keep the 3D model at the expected position.

### D. Discussion

The results of both tests are a strong indication of the interaction advantages gained by the hybrid tracking solution: it facilitates more natural interaction. The objective of an AR application is to allow natural interaction meaning, the user does not need to consider the technical solution of an interaction technology. In marker-based AR applications, interaction relies on the quality of tracking. Thus, for natural interaction, the user should not be asked to consider markers during interacting. The common marker-based solution does not allow that. The results presented in Figure 9, Single-camera tracking, indicates that the 3D model disappears when the user handles the physical object; the confidence value, which is the parameter responsible for switching between extrapolation and estimation, declines below the threshold value $c_{p,threshold}$. In general, it is also well known that the user must keep the marker in the line of sight of the video camera in order to keep the virtual information visible. In contrast, the hybrid tracking makes the interaction more natural. The 3D model does not disappear regardless of the interaction of the user (Figure 9, Hybrid tracking); the confidence value never drops below the critical threshold value. Thus, the user does not need to consider the optical markers while working with a physical component.

The results of the second test show that it is possible to estimate the correct position for at least one second. Thus, the optical flow tracking is able to backup the ARToolkit when pattern recognition is not possible.

However, some limitations must be pointed out. First, the tests do not cover all possible situations and settings. Only one video with one parameter set has been tested. While this demonstrates the feasibility of the hybrid tracking, the reliability has not been demonstrated. However, experience with the ARToolkit alone in the demonstrated use case, indicates that hybrid tracking is a suitable solution for virtual assembly training.

Secondly, the optical flow tracking is a backup and works for a few frames only. It cannot be used as a second fully functional tracking system that replaces the ARToolkit. In addition, the quality of the tracking results decrease with time. This results in incorrect alignment and position of the 3D model. The entire approach works only when the movement of the marker between each frame is small. The optical flow tracking provides a linear approximation of the movement in a spatial coordinate system. The simplicity of the solution is an advantage at this point.

Thirdly, only three markers of one size were used to track a physical object. Thus, the results of hybrid tracking are limited to this setup. More markers can cause more disturbances because the software has to merge more data.

## V. RESUMEE AND OUTLOOK

In this paper a hybrid tracking solution is presented that facilitates natural interaction in marker-based AR applications. Usually, interaction in marker-based AR applications relies on the handling of physical markers. Therefore, an unobstructed view between a video camera and the marker is mandatory. Tracking, and thus interaction, is impossible, if a user, his/her hands, or anything else covers the maker. In virtual training applications this regularly happens, because a user interacts with the physical objects that are subject to tracking. The hybrid tracking system developed uses two video cameras as input devices and combines the determined tracking data to one position and orientation estimation. In addition, an optical flow tracking approach is used to estimate the movement of a physical object when an ARToolkit marker is partially covered.

Thus, this paper makes two contributions: First, is a tracking solution that facilitates tracking of objects with which the user interacts. Hybrid tracking facilitates natural interaction with marker-based AR applicaitons.

Secondly, the optical flow tracking and a position estimation technique presented can act as a tracking backup for several seconds. The approach is simple from a technical point of view, compared with other tracking solutions, and tests indicate that it is feasible and necessary for virtual training applications.

Future work will address displacement problems and subassembly tracking. As mentioned above, the optical flow tracking and the switch between the position estimation and position extrapolation mode causes a displacement of the 3D model. Since the optical flow tracking becomes inaccurate with the time, this cannot be avoided. The goal is to replace the switch by a soft fade. Thus, the visual effect of the displacement will be reduced and it will be less significant for the user.

Assembly training also necessitates tracking of subassemblies. At this time only one physical part is tracked. To track subassemblies, ARToolkit markers may also be attached to each subassembly. To do this, several markers may be hidden. The hybrid tracking system must be enhanced to not consider this as an error.

## VI. REFERENCES


[1] Azuma, R., 1997, “A Survey of Augmented Reality,” in Presence: Teleoperators and Virtual Environments Vol. 6 (4) (August 1997), pp. 355-385

[2] Blake J., 2011, “The natural user interface revolution,” Natural User Interfaces in .NET, pp. 1–33.

[3] Kato, H. and Billinghurst, M., 1999, “Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System,” in 2nd International Workshop on Augmented Reality, October 20-21, 1999, San Francisco, USA, p. 10

[4] Welch G. and Foxlin E., 2002, “Motion tracking: No silver bullet, but a respectable arsenal,” IEEE Computer Graphics and Applications, Nov./Dec. 2002, Vol. 22 (6), pp. 24-38

[5] Fiala, M., 2005, „ARTag, a fiducial marker system using digital techniques,“ in: Conference on Computer Vision and Pattern Recognition, 2005 (CVPR 2005), 20-25 June 2005, San Diego, CA, USA, IEEE Computer Society, Vol. 2, pp. 590 - 596

[6] Wagner, D., Langlotz, T., and Schmalstieg, D., 2008, “Robust and unobtrusive marker tracking on mobile phones.” In Proc. of the Int. Symposium on Mixed and Augmented Reality (ISMAR 2008), 15-18th September 2008, Cambridge, UK, pp. 121–12.

[7] Wagner, D. and Schmalstieg , D., 2007, "ARToolKitPlus for Pose Tracking on Mobile Devices", In Proceedings of 12th Computer Vision Winter Workshop (CVWW'07), 6-8 February 2007, St. Lambrecht, Austria

[8] Naimark, L. and Foxlin, E., 2002, “Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker,” In: ACM Int. Symposium on Mixed and Augmented Reality (ISMAR 2002), 30 September - 1 October 2002, Darmstadt, Germany. IEEE Computer Society 200, pp. 27–36

[9] Uchiyama, H. and Saito, H., 2011, “Random dot markers,” IEEE Virtual Reality (VR ‘11), 19-23 Mar. 2011, Singapore, pp. 35-38

[10] Seo, J., Shim, J., Choi, J.H., Park, C., and Han, T., 2011, “Enhancing Marker-based AR Technology,” in: Shumaker, R. (Ed.): Virtual and Mixed Reality, Part I, Proceedings of Human Computer Interaction International 2011 (HCII’11), 9-14 July 2011, Orlando, FL, USA, pp. 97-104

[11] Marimon, D., Maret, Y., Abdeljaoued, Y., and Ebrahimi, T., 2007, "Particle filter-based camera tracker fusing marker- and feature point-based cues," in: Proc. of the Conf. on Visual Communications and Image Processing, IS&T/SPIE Electronic Imaging

[12] Piekarski, W., Avery, B., Thomas, B., and Malbezin, P., 2003, "Hybrid Indoor and Outdoor Tracking for Mobile 3D Mixed Reality," in: International Symposium on Mixed and Augmented Reality (ISMAR2003), Oct. 7-10, 2003, Tokyo, Japan

[13] Kalkusch, M., Lidy, T., Knapp, M., Reitmayr, G., Kaufmann, H., and Schmalstieg, D., 2002, “Structured Visual Markers for Indoor Pathfinding.” In: 1st Int'l Augmented Reality Toolkit Workshop, 29. Sep. 2002, Darmstadt, Germany

[14] Yang, P., Wu, W., and Moniri, M., 2007, "A Hybrid Marker-based Camera Tracking Approach in Augmented Reality" In: Proceedings of the 2007 IEEE International Conference on Networking, Sensing and Control, 15-17 April 2007, London, UK, pp. 622-627

[15] Yang, H., Cho, K., Soh, J., Jung, J., and Lee, J., 2008, “Hybrid Visual Tracking for Augmented Books,” in: Stevens, S.M. and Saldamarco, S. (Eds.): Proceedings of the 7th International Conference Entertainment Computing, ICEC 2008, Pittsburgh, PA, USA, September 25-27, 2008, LNCS 5309, pp. 161–166

[16] Fischer, J., Eichler, M., Bartz, D., and Straßer, W., 2006, “Model-based Hybrid Tracking for Medical Augmented Reality” In: Proceedings of Eurographics Symposium on Virtual Environments (EGVE), Lisbon, Portugal, pp 71-80

[17] Pensyl, W.R., Tran, C.T., Qui, Pei Fang, Hsin, Shang Ping Lee and Jernigan, D. K. , 2009, "Large Area Robust Hybrid Tracking with Life-size Avatar in Mixed Reality Environment – for Cultural and Historical Installation", The International Journal of Virtual Reality, 2009, Vol. 8, Issue 2, pp. 13-18

[18] Newman, J., Wagner, M., Bauer, M., MacWilliams, A., Pintaric, T., Beyer, D., Pustka, D., Strasser, F., Schmalstieg, D., and Klinker, G., 2004, "Ubiquitous Tracking for Augmented Reality", in: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality, ISMAR '04, Nov. 2 - 5, 2004, Arlington, VA, USA, pp. 192 - 201

[19] Su, H., Kang, B., and Tang, X., 2008, "Research and Implementation of Hybrid Tracking Techniques in Augmented Museum Tour System," In: Pan, Z. et al. (Eds.): Proceedings of the Third International Conference on Technologies for E-Learning and Digital Entertainment, Edutainment 2008, June 25-27, Nanjing, P. R. China, pp. 636–643

[20] Szeliski, R., 2010, "Computer Vision". Springer Verlag, Berlin

[21] Bradski, G. and Kaehler, A., 2008, “Learning OpenCV,” First Edition, O’Reilly Media Inc., Sebastopol, CA

[22] Harris, C. and Stephens, M., 1998, “A Combined Corner and Edge Detector,” in Fourth Alvey Vision Conference, Manchester, UK, pp. 147-151

[23] Welch, G and Bishop, W., 2006, “An Introduction to Kalman Filter,” Technical Report TR 95-041, Department of Computer Science University of North Carolina at Chapel Hill, Chapel Hill, NC, USA

[24] Tsai, R., 1987, “A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses,” in: IEEE Journal of Robotics and Automation, Vol. No. 4, August 1987, pp. 323-344