Infrared Tracking System for Immersive Virtual Environments

Filipe Gaspar[1]
filipe.goncalves.gaspar@gmail.com

Miguel Sales Dias[1, 2]
miguel.dias@microsoft.com

Rafael Bastos [3]
rafael.bastos@vision-box.com

[1]ISCTE, Lisboa. [2]MLDC, Microsoft Languange Developement Centre, Lisboa, Porto Salvo

[3]Vision-Box, Alfragide, Amadora

Abstract

In this paper, we describe the theoretical foundations and engineering approach of an infrared-optical tracking system specially design for large scale immersive virtual environments (VE) or augmented reality (AR) settings. The system described is capable of tracking independent retro-reflective markers arranged in a 3D structure (artefact) in real time (25Hz), recovering all possible 6 Degrees of Freedom (DOF). These artefacts can be adjusted to the user's stereo glasses to track his/her pose while immersed in the VE or AR, or can be used as a 3D input device. The hardware configuration consists in 4 shutter-synchronized cameras attached with band-pass infrared filters and the artefacts are illuminated by infrared array-emitters. The system was specially designed to fit a room with sizes of 5.7m x 2.7m x 3.4 m, which match the dimensions of the CAVE-Hollowspace of Lousal where the system will be deployed. Pilot lab results have shown a latency of 40ms in tracking the pose of two artefacts with 4 infrared markers, achieving a frame-rate of 24.80 fps and showing a mean accuracy of 0.93mm/0.52º and a mean precision of 0.08mm/0.04º, respectively, in overall translation/rotation DOFs, fulfilling the system requirements initially defined.

Keywords

Immersive Virtual Reality, CAVE, Infrared Tracking, Blob Segmentation, Epipolar Geometry, 3D Reconstruction, Model Fitting, Pose Estimation.

1. INTRODUCTION

During the last decade, Virtual and Augmented Reality technologies have became widely used in several scientific and industrial fields. Large immersive virtual environments like the CAVE (Cave Automatic Virtual Environment) [Cruz-Neira92], can deliver to the users a unique immersive virtual experience with rich human-computer interaction modalities that other environments cannot achieve. Accordingly, through a collaboration of a vast group of Portuguese and Brazilian entities, namely, institutional (Ministry of Science, Câmara de Grândola), academic (ISCTE, IST, FCUL, PUC Rio) and industrial (SAPEC, Fundação Frederic Velge, Petrobrás, Microsoft), the first large scale immersive virtual environment installed in Portugal, "CaveHollowspace of Lousal" or "CaveH" in short [Dias07], started its operation in the end of 2007. The CaveH of Lousal (situated in the south of Portugal, near Grândola) is part of an initiative of the National Agency for the Scientific and Technological Culture, in the framework of the Live Science Centres network. CaveH aims at bringing to the educational, scientific and industrial sectors in Portugal, the benefits of advanced technologies, such as immersive virtual reality, digital mock-up and real-size interactive simulation. Its physical configuration assembles six projection planes in a U topology with 5.6 m wide, 2.7 m height and 3.4 m in each side (Figure 1), giving a field of view of more than 180º, with a resolution of 8.2 mega pixel in stereoscopy for an audience of up to 14 people (where one is being tracked). Apart from the entertainment industry, there is a large range of applications which use immersive virtual environments, particularly in critical industries where simulation and real-size digital mock-up observations are imperative, such as in aerospace, natural resources exploration (oil, mining), industrial product design (automotive, architecture, aeronautics), therapy (phobia, stress), etc. In most of these applications, the interaction with the user is crucial to fulfil the application purpose. Despite of CaveH being an immersive virtual environment, this awareness can be lost if the 3D view frustums (truncated pyramids usually associated to visualization volume of a virtual camera in rigorous perspective), are not correctly adjusted to the main user's head position and the CaveH topology, while he/she is in inside the physical space of the CaveH (Figure 2) during interactive sessions.

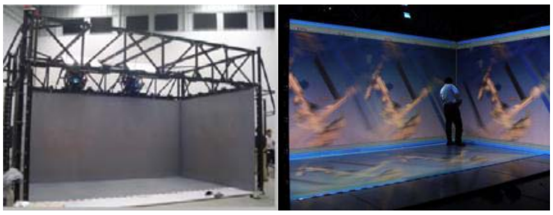

Figure 1: CaveHollowspace physical set-up.

For this reason, it is essential to integrate in an infrastructure like the CaveH, a precise tracking system that follows an artefact attached to user's head position, enabling the underlying distributed 3D graphics system to adjust the image on the CaveH displays in real-time (6 in total for the case in study, Figure 1), to reflect perspective variations (Figure 2). In this paper we describe a prototype of an infrared-optical tracking system capable of tracking in real-time a set of artefacts that the user can adjust to his/her stereo glasses, thus tracking his/her pose or use as 3D input device, while interacting in virtual environments. The artefacts are rigid body targets with 6DOF allowing the system to detect motion in a three-dimensional space in respect to a reference coordinate system, i.e., translation across X, Y and Z axis and rotation over X, Y and Z axis. This system can be used in virtual reality settings such as the CaveH, or in augmented environments, being an alternative in this last case to video-based tracking [Dias06].

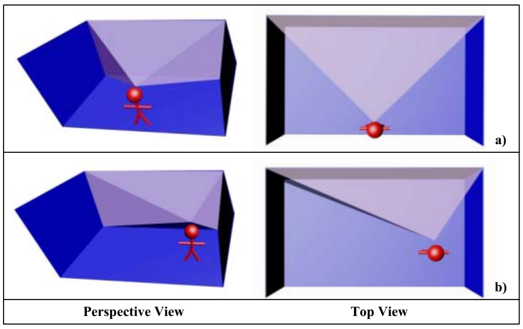

Figure 2: Dynamically adjustment of view frustums and viewpoints required to correctly display images on several projection planes, when the user moves from a) to b).

The paper starts by summarizing the related state-of-the-art work in infrared and vision-based tracking systems (Section 2) and then presents a system overview of the chosen hardware configuration as well as the developed software architecture (Section 3). Subsequently we describe the proposed infrared camera calibration algorithm (Section 4) and present the in-house developed algorithms that solve three common problems found in infrared tracking systems: **Feature Segmentation** (Section 5), **3D Reconstruction** (Section 6) and **3D Model Fitting** (Section7). In Section 8 we present and discuss our results and, in Section 9, we extract conclusions and plan for future directions.

2. RELATED WORK

Several commercial tracking systems can be found in the market. Vicon [Vicon09] or ART [ART09] propose systems with configurations that usually allows 4 to 16 cameras, with update rates from 60Hz to 120Hz. However, due the heavy cost of these commercial solutions, we turned to the scientific community and have analysed the literature regarding the development of accurate and precise algorithms to track independent rigid body markers and artefacts. We found that the multi or single camera systems using infrared or vision-based tracking technologies share the same problem formulation: computing the position and orientation of a rigid body target or artefact relatively to a reference coordinate system. To solve this problem a sufficient number of geometric correspondences between a real feature in the world reference space and its 2D projection in the camera's image space, is required. Several approaches to the tracking problem can be found in the literature. In PTrack [Santos06], a marker-based infrared single-camera system is presented. Similar to ARToolKit [Kato99], this contribution takes advantage of a non-symmetric square geometric arrangement label in object space (an infrared marker) and its correspondence in projected image space. PTrack uses an iterative approach to twist the square label projection in order to derive a correct orientation, and then performs iterative scaling to the label, to correct the position of the square infrared marker. Another single-camera tracking contribution is ARTIC [Dias05], where the marker design is based on color evaluation. Each marker has a different color which can be segmented using vision-based-techniques and a 3D artefact structure has been designed with five of such markers. The algorithm used to estimate the artefact pose is PosIT [DeMenthon92]. Analogous to Lowe [Lowe91], PosIT is an interactive algorithm but does not require an initial pose. The algorithm needs four non-coplanar points as features in image space and the knowledge of the corresponding object model points. The object pose is approximated through scaled orthographic projection. Another well known example of a multi-camera infrared tracking system is the ioTracker (nowadays is a commercial system, even though started as a research project) [Pintaric07]. This system uses rigid body targets that take advantage of the following constraint: every pair-wise feature Euclidean distance is different and the targets construction is done to maximize the minimum difference between these features' pair-wise distances. By having several cameras, the feature information in several images spaces is correlated via Epipolar Geometry (the geometry of stereoscopic projection) and the 3D reconstructed feature in object space is performed via Singular Value Decomposition [Golub93]. The pose estimation is a 3D-3D least square pose estimation problem [Arun87] and requires the object model points and at least 3 reconstructed points to estimate a 6DOF pose. Our work has been largely influenced by the ioTracker, especially in two specific algorithms: Multiple-View Correlation (see Section 6.1) and Model Fitting where we follow the complexity reduction approach proposed by Pintaric and Kaufmann (Section 7) with two important modifications to avoid superfluous computational time (for further details see Section 7.1.1).

3. SYSTEM OVERVIEW

In order to develop a tracking system our first step was to build a comparative assessment between different tracking technologies that can be used in immersive virtual environments like the CaveH, based on a clear definition of system requirements which are:

1. Real time update rate (at least 25 Hz, which is the maximum supported by affordable infrared cameras);
2. Without major motion constraints: this means that the tracking technology shall be less obtrusive as possible, allowing the user to interact with the CaveH environment through several artefacts and enabling the best freedom of movement;
3. Robust to interference of the environment, that is, the technology chosen should not have interferences with system materials and hardware (e.g. acoustic interference, electromagnetic interference);
4. Without error accumulation in the estimated poses, that is, the accuracy of the estimated artefact pose should be below 1 mm and 0.5º, respectively, for the overall translation (average over the 3DOF in translation) and overall rotation (average over the 3DOF in rotation);
5. No significant drift of the estimated pose, that is, the precision of the estimated artefact 3D pose (position and orientation) should be below 0.1mm and 0.1º, respectively, for the overall translation (average over the 3DOF in translation) and overall rotation (average over the 3DOF in rotation).

In the following sub-sections we present an evaluation of several tracking technologies regarding their performance and reliability (Section 3.1), the hardware components (Section 3.2) chosen and the software architecture and workflow (Section 3.3).

3.1 Tracking Technologies Assessment

In Table 1, we present a summarized comparison between the main advantages and drawbacks of different available tracking technologies.

By analysing Table 1 we can conclude that only optical technology can cover all our specified requirements. However, this option brings three problems to be solved: (1) line of sight occlusion; (2) ambient light interference; and (3) infrared radiation in the environment, which we will tackle in Section 3.2. Considering the nature of our problem (tracking artefacts poses in a room-sized environment), an outside-in tracking approach is the best fit solution. Outside-in tracking employs an external active sensor (usually a video camera) that senses artificial passive sources or markers on the body of a person or on an artefact, and retrieves its pose in relation to a reference coordinate system.

3.2 Hardware Setup

Once the technology to be used has been defined, the hardware setup was chosen based on a minimization cost strategy, without compromising the system reliability and performance and the specified requirements.

3.2.1 Cameras

To minimize the line of sight occlusion problem, at least two cameras are required. We have opted to use a setup of four cameras, AVT Pike F-032 B, with IEEE 1394b interface [AVT09], with a maximum resolution of 640x480 at a capture rate of 209 fps. We have also assessed several lens models for the selected camera model in order to choose the optimal overture angle at best precision, since precision and operational volume are inversely proportional. The multiple cameras create a volume where artefacts can be seen (the diamond shape). In Figure 3 we see a simulation, with:

- (left) 3.5 mm Lens, field of view of 81.20º and resolution at the centre of 3.50 mm;
- (right) 4.5mm Lens, field of view of 67.40º and resolution at the centre of 2.70 mm.

The retained solution was the left one. Despite the reduced resolution, comparing to the right solution, we

Technologies	**Accuracy and Precision**	**Real time update rate**	**Robust to interference**	**Motion Constraints**	**Additional problems**
Acoustic	**Low, Low** (centimeters, degrees)	**No** (speed of sound variation)	**No** (acoustic interference)	**Yes** (line of sight occlusion	None.
Electromagnetic	**High, High** (sub-millimetre, sub-degree)	**Yes** (>100Hz)	**No** (interference with metal objects)	**No**	Small working volume
Inertial	**High, Low** (error accumulation)	**Yes** (>100Hz)	**No** (gravitic interference)	**No**	Limited by cabling
Mechanical	**High, High** (sub-millimetre, sub-degree)	**Yes** (100Hz)	**Yes**	**Yes** (mechanical arm paradigm	Small working volume
Optical	**High, High** (sub-millimetre, sub-degree)	**Yes** (>=25Hz)	**No** (ambient light and infrared radiation)	**Yes** (line of sight occlusion)	None.

Table 1: Tracking technologies requirements comparison.

have selected the configuration that allows a larger tracking volume.

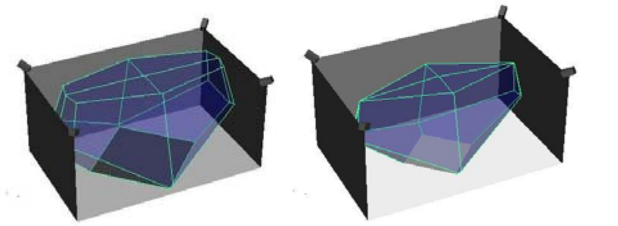

Figure 3: Simulation of CaveH "diamond shape volume" created by multiple cameras, where artefacts can be seen. The retained solution was the left one.

3.2.2 IR Filters and IR LEDs

Most CCD-based cameras are sensitive to the infrared spectrum. Usually these sensors come with an attached infrared filter, which can be replaced by an infrared band-pass filter – we have selected a Schneider Kreuznach IR filter [SK09] (cut-on wavelength 725 ±10 nm). This gives us the ability of obtaining finer measurements in the infrared spectrum. To enhance infrared detection by light reflection, we have chosen to mount an infrared light ring emitter in each camera (Figure 4). Therefore, since the CaveH in operation, is completely absente of environmental light (the CaveH room is in total darkness and usually with a room temperature of 21º C), we have a controlled lighting setting. With this set-up the only relevant source of infrared light is emitted by the LED arrays, which is subsequently detected by the applied infrared band-pass filters.

3.2.3 Shutter Controller

In order to synchronize the cameras frame grabbers, we need to use a shutter controller. We have opted by a National Instruments USB shutter controller NI-USB6501 [NI09], a digital and programmable device that trigger camera's capture on dynamic or static periods of time.

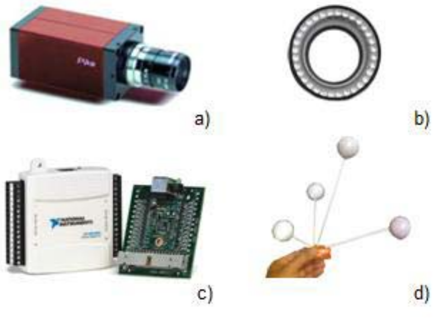

Figure 4: Hardware components of the Infrared Camera system. a) AVT Pike Camera; b) Infrared LED emitter; c) Shutter controller; d) Artefact.

3.2.4 Artefacts

Infrared targets or artefacts are a set of markers arranged in space with known pair-wise Euclidean distances in a pre-defined structure. Each target is a unique object with a detectable pose within a tracking volume. We have opted by passive retro-reflective markers instead of active markers because the last ones require an electric source, which would become more expensive, heavy and intrusive. Our lab-made markers are built from plastic spheres with 20 mm of radius and covered with retro-reflective self-adhesive paper. We have chosen spheres since it is a geometric form that allows an approximation to an isotropic reflection system. The final infrastructure of CaveH can be seen in Figure 4.

3.3 Software Architecture and Process Workflow

Our software architecture comprises a real-time and threaded oriented pipeline. The diagram in Figure 5 illustrates the full real-time infrared tracking pipeline process since multiple frames (one per camera), reach the application to generate the final output: the pose of each target in a know reference coordinate system.

However, two other important steps are done previously in an off-line stage:

- **Camera calibration** – For each camera, we determine its intrinsic parameters (focal length, the principal point, pixel scale factors in the image domain and the non-linear radial and tangential distortion parameters), and extrinsic parameters (position and orientation in relation to a reference coordinate system);
- **Artefact calibration** – For each artefact structure, we compute the Euclidean distances between each pair of markers. This procedure allows us to use artefacts with different and non pre-defined topologies (under some constraints, see Section 7.1) and overcomes the precision error in their construction. A "reference pose" is also computed for comparison with poses retrieved by application.

Each on-line pipeline cycle in Figure 5 starts when multiple frames are grabbed by the application (*Multiple Video Input*). For each frame an image segmentation algorithm identifies the 2D image coordinates of every blob feature, i.e. the 3D real infrared marker projected in the image, and recovers its centre and radius (*Feature Segmentation*). Through epipolar geometry [Hartley03], the recovered 2D features are correlated from the different views establishing feature correspondences (*Multiple View Correlation*). Via Singular Value Decomposition (SVD) Triangulation [Golub93], we are then able to reconstruct 3D markers through *n* multiple projected views of each marker, where *n* range is [2; number of cameras] (*3D Metric Reconstruction*). The resulting 3D point collection is examined in order to determine which points belong to each artefact (*Model Fitting*). Finally, for each discovered artefact its position and orientation, in relation to a reference coordinate system or in relation to "reference pose", are estimated (*Pose Estimation*).

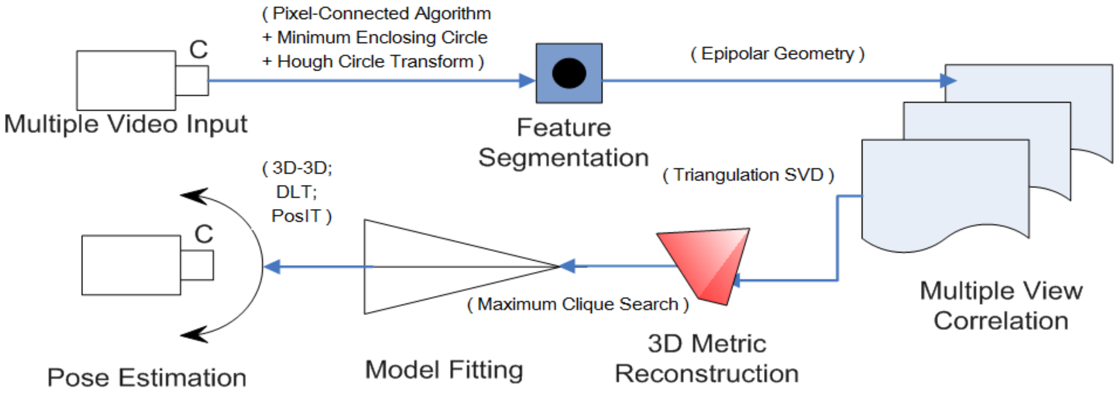

Figure 5: Infrared tracking process.

4. CAMERA CALIBRATION

Camera calibration is a critical step in many computer vision applications. Considerable effort has been made on the development of effective and accurate procedures to address this problem since the reliability of computer vision systems, such as optical tracking, are highly dependent of the camera calibration phase. Several approaches can be found in the literature [Wang04], [Zhao07], to evaluate the internal camera geometric and optical characteristics (intrinsic parameters) and to find the three-dimensional position and orientation of the camera reference frame in relation to a reference coordinate system (extrinsic parameters). We have decided to solve the two problems separately, essentially because intrinsic parameters have to be determined only once (ideally) and combined solutions to solve intrinsic and extrinsic parameters tend to be more imprecise. Our camera calibration offline stage development was entirely based on the technique developed by Zhang [Zhang99] available in the OpenCV library [OpenCV09].

4.1 Intrinsic Calibration

Using a classical pinhole camera model [Heikkila97] it is possible to find a set of parameters which define a 3D transformation matrix that maps pixel coordinates of an image point into the corresponding coordinates in the camera reference frame. The method used, allows finding the focal length, the principal point, the pixel scale factors and the non-linear radial and tangential distortion using a chessboard pattern (see Figure 6). The chessboard has geometric known properties– its squares width and height, number of horizontal and vertical squares – which allows extracting the object corners and determining intrinsic parameters through projective transformation.

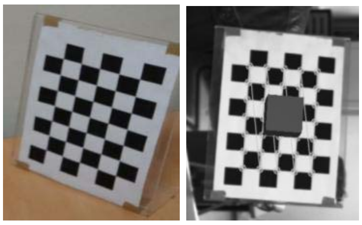

Figure 6: Camera Intrinsic Calibration. Left) 6x8 chessboard calibration patern. Right) Calibration output: chessboard corners extration.

4.2 Extrinsic Calibration

In multi-camera systems, to perform a reliable 3D points reconstruction from a scene, all cameras' extrinsic parameters have to be calibrated under the same world coordinate system. The method used requires at least four 3D-2D correspondences between the world reference coordinate system and each camera image plane, in order to compute the camera extrinsic parameters through a well documented computer vision algorithm: Direct Linear Transform [Abdel-AzizKarara71]. The DLT performs the homography that transforms a 3D object plane into 2D image coordinates. In order to establish 3D-2D correspondences we have used a co-planar square arrangement of five markers as presented in Figure 7, with known positions in world space. In image space, through our Feature Segmentation algorithm (see Section 4), we can identify the markers centre and radius via a projective transformation.

By observing Figure 7, we see that our markers' distribution takes advantage of two important constraints: the "orientation point" is the one with smaller Euclidean distance to the average of all points; the remaining points are indentified by their proximity to the image corners; the point of index 1 has the smaller Euclidean distance to the "orientation point". With the presented constraints we easily order the points from the "orientation point" to the fifth point following an anti-clockwise order. The result achieved with extrinsic calibration is within a reprojection error of 1.68 pixels/mm.

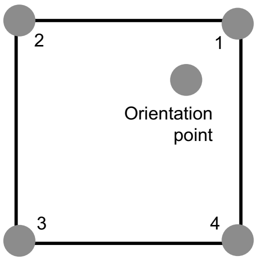

Figure 7: Extrinsic Calibration Marker.

5. FEATURE SEGMENTATION AND IDENTIFICATION

Feature segmentation and feature identification refers to the process of analysing a digital image and change its representation into something more meaningful. Our features are blobs, i.e., 2D projections of 3D objects (infrared markers). This section describes how to approximate blobs to circles recovering the feature's radius and centre.

5.1 Feature Segmentation through non-Destructive Threshold

The first image processing operation is a non-destructive image thresholding. Since our work environment is lighting controlled and most of information is filtered by infrared filters, we have used a threshold operation to eliminate noise and discard possible false artefacts in the image. The output of the camera is a grayscale image where the retro-reflective markers have extremely high value of luminance (> 60% maximum value). It is quite easy to segment our features establishing a static threshold value (discovered experimentally), where the pixels with luminance value below the threshold are changed to zero (Figure 8). This shows the importance of hardware configuration. We have used a non-destructive operation because in a later algorithm, Hough Transform [Kimme75], the gradient information will be extremely useful to accelerate circles searching.

5.2 Feature Identification

A classic approach to indentify the retro-reflective markers in image space is the adoption of Hough Circle Transform (HCT) [Duda72], [Kimmy75]. HTC is a well-known technique to find geometric primitives in an image (in our case, circles). The principle of this technique is to transform the image space into a tri-dimensional accumulator, where each entry of accumulator is a potential circle parameterized by its centre (x-coordinate and y-coordinate) and radius. The local-maxima entries on the accumulator will be the retrieved circles. However, our features can assume a wide range of radii in image, especially if the image space has partially-occluded features. The Hough accumulation is highly demanding of memory which turns this technique too much heavy, computationally speaking, to be applied in a real-time application with the robustness and the precision needed. Therefore, instead of applying HCT to the whole image (of 640 x 480 of resolution), our feature identification algorithm consists in a combination of one technique (Pixel-Connected Algorithm) and one decision metric (Occlusion Metric), to select the best algorithm to be applied to the image (Minimum Enclosing Circle Fitting or Hough Circle Transform), depending on whether occluded features exist or not. The feature segmentation workflow can be seen in Figure 8.

5.2.1 Pixel-Connected Algorithm

Instead of analysing the whole image, the purpose of this "in-house" developed algorithm is to isolate image regions of interest) (ROIs) for further examination. The statement is: by analysing pixel by pixel, a blob can be seen as a set of adjacent pixels with luminance value greater than zero. Pixel-Connected is a recursive algorithm that finds closed sets of pixels. Each pixel visited is marked to stop the recursive cycle. The result is a vector of bounding boxes where each one is a ROI of the image and represents at least one feature (or more, when features are occluded).

5.2.2 Minimum Enclosing Circle Fitting

An alternative technique to HCT is the Minimum Enclosing Circle Fitting (MECF) algorithm [Fitzgibbon95], [Zhang96]. Given a set of pixel coordinates, the algorithm retrieves the minimum circle (centre and radius) that includes all pixels. Additionally, MECF finds feature parameters faster that HCT, although it is not applicable in the case of occluded features.

5.2.3 Occlusion Metric

Having two different techniques able to find feature's radius and centre, we have decided to add to our feature identification workflow an occlusion metric in order to determine whether or not a ROI of the image has occlusions. If it shows occlusions, we use HCT as our feature identification algorithm, otherwise, MECF is selected. An image ROI is labelled as "occluded" in the following situations:

1. ROI of the image has less than 65% of nonzero pixels;
2. Circle retrieved by circle fitting algorithm exceeds the ROI limits in more than 2 pixels.

To validate the second rule presented above, we need to run always the circle fitting algorithm. The processing time saved (10.06 ms, see Section 8.2) supports this decision. Both rules were established empirically.

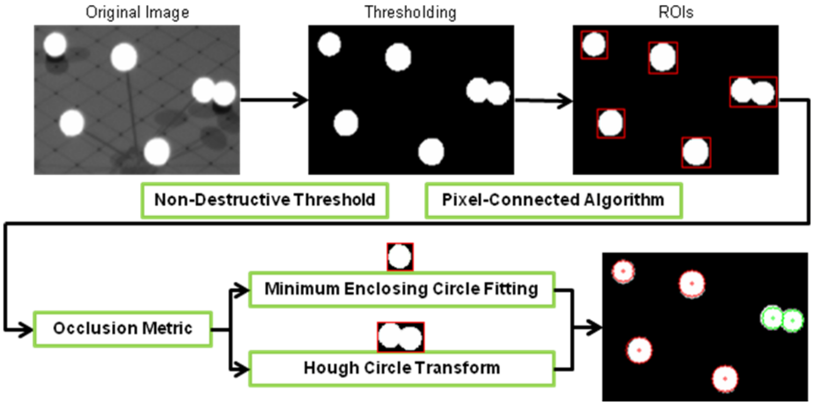
Original Image
Thresholding
ROIs
Non-Destructive Threshold
Pixel-Connected Algorithm
Minimum Enclosing Circle Fitting
Occlusion Metric
Hough Circle Transform

Figure 8: Feature Segmentation Workflow.

6. 3D RECONSTRUCTION VIA TRIANGULATION

Once indentified the feature's position in different views, the subsequently problem is how use these information to reconstruct the markers position in 3D-scene space. In this section we cover the geometry of several perspective views. First, we clarify how to establish correspondences across imaged features in different views by using epipolar geometry [Hartley03], [Trucco98]: we call it Multiple View Correlation. Therefore, having groups of correspondent features, we present our 3D metric reconstruction to recover the marker position in 3D space via SVD [Golub93], the 3D Metric Reconstruction [Dias07].

6.1 Multiple View Correlation

A useful and widely used mathematical technique to solve feature's correlation from two or more views is epipolar geometry [Hartley03]. The epipolar geometry theory defines that a 3D point, X, imaged on the first camera view of a stereo pair, as x_1 restricts the position of the correspondent point x_2 in the second camera view to a line (epipolar line), if we know the extrinsic parameters of the stereoscopic camera system. This allows us to discard features as possible correspondences whose distance to the epipolar line are above a certain threshold. Our multiple-view correlation is based in this constraint and works as follows. We define a set of stereo pairs between the reference camera (the camera with the most features in current frame) and the remaining ones. For every stereo pair and for each feature in the first view (reference camera image), we compute the distance between the features centres in the second view and the epipolar line that corresponds to the feature in the first view. All features with a distance above our threshold are discarded. The corresponding feature is the one with the smallest distance to the epipolar line. The result is a set of feature correspondences between the reference camera and the remaining ones which can be merged giving a Multiple-View Correlation.

6.2 3D Metric Reconstruction

By setting a well-known camera topology during the camera calibration step and by establishing optimal feature's correlation through multiple view correlation, we are able to perform the 3D metric reconstruction of the artefact. This process can be obtained in two different methods: by triangulation or via Singular Value Decomposition [Golub93]. When we have correspondences between only two views, we can use triangulation instead of SVD to obtain the 3D point location, i.e., the 3D point can be computed directly through epipolar geometry as the intersection of rays fired from the camera positions that hit the corresponding image points and that intersect the 3D point. This analytic method is clearly faster than using SVD. However, the line intersection problem can lead us to numerical instability and subsequently to numerical indeterminacies which affect the system stability. Having several views for a given feature correspondence, several possible solutions for the reconstruction derive from epipolar geometry and we are left with a set of linear equations that can be solved to compute a metric reconstruction for each artefact feature via SVD (presented also in [Dias07]). The SVD usually denotes that a matrix A, can be decomposed as $A = V\Lambda U$. Using each camera's intrinsic (K) and extrinsic parameters (M), we stack into matrix A the existing information for each view (2D point location – x, y). Solving the A matrix by SVD and retaining the last row of the V matrix, the reconstruction point coordinates (x, y, z) are the singular values in Λ. After testing both solutions in the two views scenario, we have decided to preserve the system reliability using the SVD technique despite the computational cost.

7. MODEL FITTING

The purpose of model fitting is to determine which set of reconstructed 3D points belong to each artefact, labelling its spatial model and estimating the artefact relative pose (see Figure 10). Fitting a model to a cloud of points is typically a NP-hard problem. However, Pintaric and Kaufmann have proposed an approach to reduce the complexity of this problem [Pintaric07] by combing Candidate Evaluation via Artefact Correlation and Maximum Clique Search. Our model fitting technique is largely influenced by Pintaric and Kaufmann. In the following sections the technique is presented, focusing in our proposed improvements.

7.1 Candidate Evaluation via Artefact Correlation

The candidate evaluation goal is to decide which points are "good candidates" to belong to a specific artefact. The metric used to compare pairs of 3D points is the Euclidean distance between them, where we can take advantage of three important constraints: artefacts are rigid-body objects with constant Euclidean distances between pair-wise markers; since we control the artefacts construction, its design can be done to maximize the minimum difference between Euclidean distances across all targets; knowing the minimum Euclidean distance detectable by our system we maintain the previous constraint above this value. However, this metric does not allow us to design the artefacts arbitrarily.

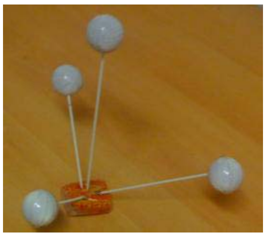

Figure 9: Rigid body artefact.

7.1.1 Our implementation of Pintaric and Kaufmann's Algorithm

The candidate evaluation technique works as follows. For each artefact, a calibration process is needed in order to construct a Correlation Score Look-up Table (CSLT) with all artefact Euclidean distances between all M pair-wise markers. This topology information is assumed to follow a Gaussian distribution whose deviation is chosen according to the system accuracy. In runtime, for each system pipeline cycle, a Euclidean Distance Matrix

(EDM) of size NxN is constructed, containing Euclidean distances between all pairs of the N points reconstructed. For each artefact, a Correlation Score Matrix (CSM), of size NxN, is computed. In the authors' original problem formulation [Pintaric07], each entry of CSM corresponds to a correlation between the EDM correspondent entry and the artefact's Euclidean distances stacked in the CSLT. In other words, each entry of CSM represents the probability that a pair of makers belongs to the artefact concerned. From the CSM, a vector containing the accumulated correlation scores is computed through row or column-wise summation. Since each CSM row or column represents a reconstructed point, the accumulated vector pass through a threshold determining which points are candidates to belong to the artefact. Our implementation differs from [Pintaric07] in two main aspects. First, despite each Correlation Score Matrix entry represent the probability that a pair of makers belongs to a certain artefact, there is no information about which artefact' markers are responsible for this high probability. Our CSM implementation keeps record of markers indexes with high correlation score. This information avoids the introduction of an additional processing step to establish point's correspondences between the output of model fitting and the calibrated artefact (required for pose estimation). Second, our threshold implementation is a simple integer. Any point with less than two high correlation entries on CSM is discarded, i.e. any point that can't belong to a set of at least three points (we need a minimum of three points to retrieve 6DOF pose estimation). The final output of Candidate Evaluation via Artefact Correlation, is a set of points, each one with a reference to points which are connected.

7.2 Maximum Clique Search

The output of the previous step can be actually seen as a graph, where each candidate marker is a vertex and has a connection with a set of other markers, creating with each one an edge. Given a graph, which can be denoted by $G = (V, E)$, where V is a set of vertices and E is a set of edges, a clique is a set of vertices where any pair of two is connected by an edge. In the Maximum Clique Search problem (MCS), the goal is to find the largest amount of cliques, i.e., find the largest sub-graph of G, denoted by $C = (V_c, E_c)$, where any pair of vertices V_c are connected by an edge E_c. To address this problem we have developed an algorithm based on [Konk07]. The markers with high probability to belong to a certain artefact are the input of maximum clique search algorithm. The vertex-clique returned is the set of points more probable to be the corresponding artefact.

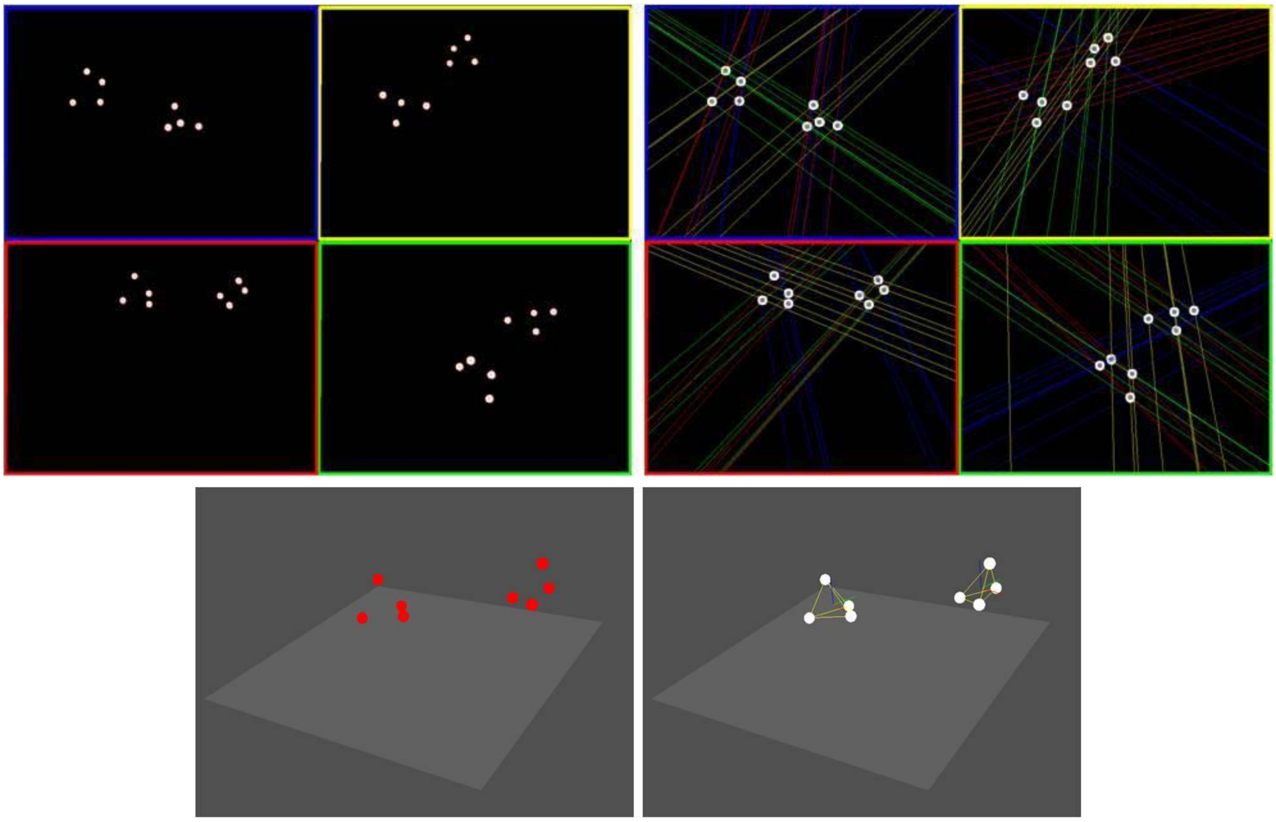

Figure 10: Software main stages. Top Left) Feature Segmentation output in each view (each one represented by a colour); Top Right) Epipolar lines from all views projected in each view; Bottom Left) 3D Reconstruction of markers; Bottom Right) Artefacts' pose retrieval after Model Fitting.

7.2.1 Analysis Algorithm Input and Output

Developing metrics to deal with the model fitting algorithm input and output is desired in order to decrease the required computational time.

A simple input metric can be followed. For an artefact with N markers and M vertices in the input, only one of the subsequent scenarios can happen:

- **Unique maximum-clique:** N ≤ M ≥ 2 and each Mth vertex have exactly M-1 edges. In this case there is no need to run the MSC algorithm;
- **No maximum-clique:** M < 3, we need at least 3 points to estimate the 6DOF pose;
- **Find maximum-clique:** we need to run the MCS to determine the maximum vertex-clique.

Also important is the output metric. We account for three possible outcomes:

- **Single solution:** The graph has exactly one vertex-clique of size N ≥ 3. No further processing is needed;
- **Multiple solutions:** The graph has multiple vertex-cliques of size N ≥ 3. We compute the pose estimation for each solution and choose the one with minimal squared sum of differences from the last retrieved pose;
- **No solutions**: Any vertex-cliques of size N ≥ 3 in the graph. The last pose retrieved is shown.

7.3 Pose Estimation

Reaching to a solution with a set of adjacent points, the pose estimation is the procedure to determine the transformation (translation and rotation) between the runtime solution and the "reference pose" (of the model points) computed during the Artefact Calibration step.

We have developed the 3D-3D least square pose estimation presented by [Haralick89].

The pose estimation problem is to infer a rotation matrix, **R**, and a translation vector, **T**, for solution points which transform them into the model points. Three non-collinear correspondences are required to estimate a 6DOF pose.

Assuming our model points which define the "reference pose" denoted by {x_1, x_2, ..., x_N} and the corresponding points denoted by {y_1, y_2, ..., y_N}, where **N** define the number of correspondences. The least square problem can be expressed to minimize:

$$\sum_{n=1}^{N} w_n \| y_n \square R x_n \square T \|^2 \qquad (1)$$

Where w_i represent the weight given to the i^{th} point based on its re-projection error. To simplify the problem we can compute the mean values of each set of points (centroid) and translate them to the reference coordinate system origin, eliminate the translation **T**:

$$\sum_{n=1}^{N} w_n \| y_n \square R x_n \|^2 \qquad (2)$$

Expanding the equation (2) we have:

$$\begin{aligned}&\sum_{n=1}^{N} w_n \square \| y_n \|^2 \square 2(y_n, Rx_n) \square \| Rx_n \|^2 \square = \\ &\sum_{n=1}^{N} w_n \| y_n \|^2 \square 2trace\left(R^T \sum_{n=1}^{N} w_n y_n x_n{}^T \right) \square \\ &\square \sum_{n=1}^{N} w_n \| Rx_n \|^2\end{aligned} \qquad (3)^1$$

In order to minimize the general equation, we want to maximize the second term.

Defining K as a correlation matrix:

$$K = \sum_{n=1}^{N} w_n y_n x_n{}^T \qquad (4)$$

The problem can be now stated as:

$$trace(R^T K) \rightarrow maximum \qquad (5)$$

The solution to the correlation matrix can be found by singular value decomposition (SVD), where the correlation matrix can be decomposed into the form:

$$K = W \Lambda V \qquad (6)$$

Here Λ represents the singular values. The rank of K is equal to the number of linearly independent columns or rows of K. Since $\mathbf{RR^T} = 1$, the equation (5) is maximized if:

$$R = V \begin{pmatrix} 1 & & \\ & 1 & \\ & & \det(VU^T) \end{pmatrix} U^T \qquad (7)$$

This gives a unique solution to rotation matrix.

The translation vector, T, is given by:

$$T = \bar{y} \square T \bar{x} \qquad (8)$$

Where $\bar{y}$ and $\bar{x}$ represent the centroids of each set of points computed previously.

8. RESULTS AND DISCUSSION

In order to assess the system accuracy and precision we have assembled a preliminary setup in our Computer Graphics lab at ISCTE, of size 4m x 2m x 2m.

8.1 Frame Rate and Latency

To determine the system performance we have executed our application during 10 minutes moving two artefacts inside the tracking volume. The mean frame rate measured was 24.80 fps which gives a mean latency of 40.32 milliseconds.

8.2 Feature Segmentation analysis

In Section 5 we have presented a combination of three different techniques and one metric to solve feature segmentation problem – Connect-Components (CC), Minimum Enclosing Circle Fitting (MECF), Hough Circle Transform (HCT) and Occlusion Metric (OM). Now we base our choices through algorithms comparison, summarized on Table 2. All results presented on this sub-section result from an analysis of 10000 frames.

[1] (A,B) = *trace*(A,B)^T denotes the inner Euclidean product. See [Haralick89] for a detailed explanation.

Technique	Mean Time [ms]
HCT applied to the whole image	37.59
Our technique with CC+HCT only	21.13
Our final technique using HCT or MECF depending in the OM	11.07

Table 2: Different feature segmentation techniques and its computational time required to process one image.

By analysing the table above it is easy to conclude that our final technique is faster than others. The comparison between the implemented approach and the similar version only with CC and HCT is relevant to prove than Occlusion Metric is important to choose the algorithm to identify a feature in a certain ROI. The difference between the two last approaches presented on Table 2 in terms of computational time needed, is largely due the fact that MECF and OM only requires an average of 0.02 ms to process a ROI while HCT takes about 0.82 ms to analyse a ROI. As mentioned, HCT is sensible to occlusions.

8.3 Precision in artefact pose retrieval

To measure the system's precision, where we want to sense the deviation of the estimated pose of the artefact relatively to the average estimation, we have placed a static single marker on the working volume and recorded 10000 samples of its 3D reconstruct position. We have then measured its deviation in relation to the average reconstructed position across all samples. The histogram in Figure 11 shows this deviation. The mean deviation computed was 0.35 mm and the maximum deviation in the experiment was 0.82 mm.

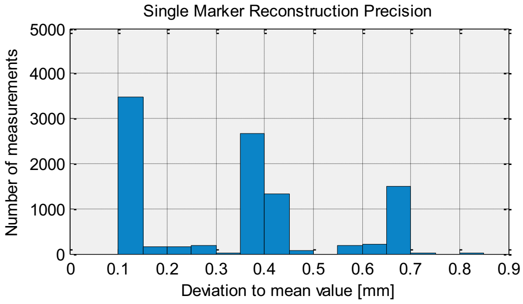

Figure 11: Single marker deviation to mean value.

Following the same reasoning, we have placed a four-marker artefact in the working volume (Figure 9). Next, we have calibrated the pose and estimated the same pose 10000 times, using our technique, for a unique pose in the real space. The histogram on Figure 12 shows the deviation of the estimated translation (in this case, the Euclidian distance between the calibrated geometric centre of the artefact and the estimated centre). In Figure 13 we present the overall rotation error (average rotation error over the three axes X, Y, Z) between the calibrated and estimated pose. The mean and maximum translation deviations were 0.08 mm and 0.55 mm. The mean and maximum overall rotation errors were, respectively, 0.04º and 0.11º.

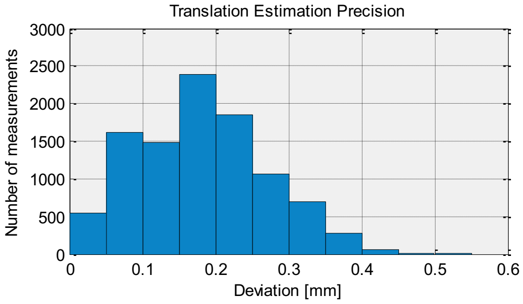

Figure 12: Deviation between the calibrated and estimated translation.

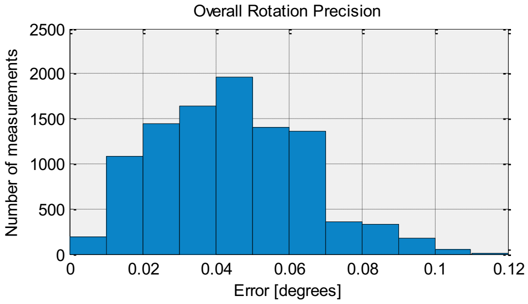

Figure 13: Overall rotation error between calibrated and estimated rotation.

8.4 Accuracy in artefact pose retrieval

With the accuracy tests we want to evaluate the mean error in translation and rotation of the estimated pose and for that, we have designed several experiments. The first experiment to estimate the system accuracy provided us with the computation of the re-projection error of a single marker in all images. We have moved the object during 10000 frames across the working volume. The re-projection error histogram is shown in Figure 14. The mean and maximum re-projection error measured were, respectively, 3.49 mm and 6.86 mm.

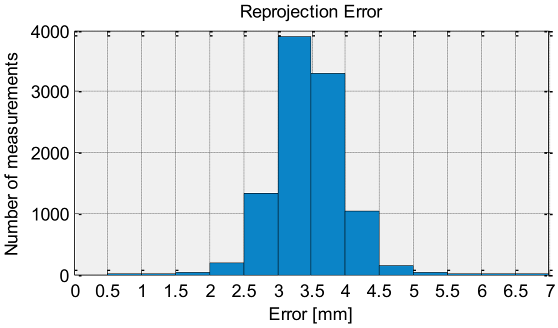

Figure 14: Histogram of the reprojection error.

Another experiment to determine the system's accuracy consisted in assembling a two-marker artefact with a known Euclidean distance (15 cm). Subsequently, we have recorded the estimated distance between the markers while moving the artefact across the working volume during 10000 frames. The histogram in Figures 15 and shows the quadric error of the 3D Reconstruction (deviation of the estimated distance from 15 cm). The mean and maximum quadratic reconstruction errors in the trial were, respectively, 2.44 mm and 49.75 mm.

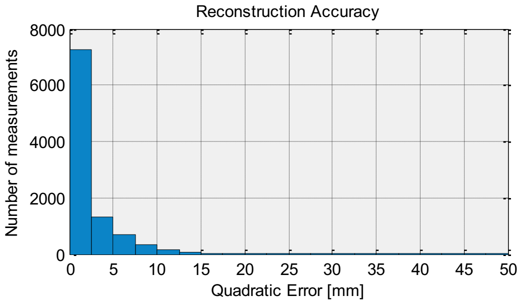

Figure 15: Reconstruction accuracy.

In another experiment aiming at determining the internal accuracy of our pose estimator algorithm, we have performed the following steps. By having a calibrated artefact, we have applied to it a 3D transformation (translation or rotation) as a ground truth. The resulting transformed points and the original points of the calibrated artefact were the input to the pose estimator algorithm. The algorithm output, the 3D transformation estimation between the sets of points, was compared with the ground truth transformation. We have repeated this process 10000 times, setting transformations across all degrees of freedom. The mean translation and rotation errors, i.e. the difference from the estimated and ground truth transformation, were 4.9x10^{-15} for translation and 2.1x10^{-9} for rotation (negligible values). Finally, to compute the absolute accuracy in pose retrieval over the 6DOF, we have performed a similar experiment as the presented above. This time, instead of computing only a 3D transformation in relation to the calibrated artefact and then execute the pose estimation algorithm, we wanted to take into account the tracking algorithms errors (feature segmentation error, re-projection error and reconstruction error). The experiment worked as follows. First we have computed a ground truth 3D transformation and have applied it to the calibrated points, keeping record of the original points. Next we have re-projected all 3D features in each camera image space. The subsequent steps of this experiment followed the regular system workflow: (1) feature segmentation; (2) 3D reconstruction; (3) model fitting, whose output is a pose, i.e. a 3D transformation. The absolute difference between this 3D transformation and the known ground truth transformation, define the pose accuracy error. Experiments over each degree of freedom have been performed. In Figures 16 and 17, we show the observed errors in translation in the Y axis and, in rotation over the Y axis.

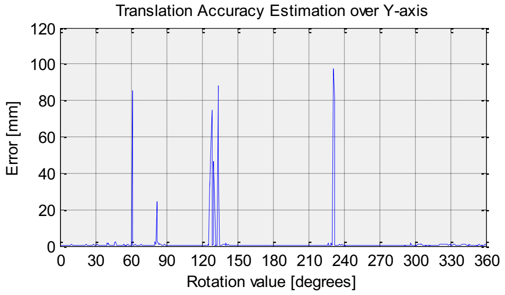

Figure 16: Translation accuracy in Y axis.

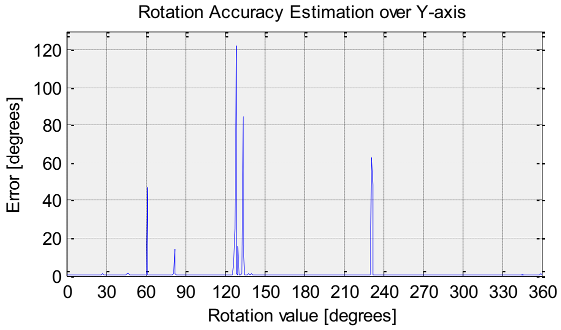

Figure 17: Rotation accuracy over Y axis.

The mean accuracy error can be seen in Table 3.

Mean Error	Over X	Over Y	Over Z	Overall
Translation [mm]	1.09	0.85	0.85	0.93
Rotation [deg]	0.55	0.9	0.60	0.51

Table 3: Absolute pose estimation accuracy over the 6DOF.

9. CONCLUSIONS AND FUTURE WORK

In this paper a complete hardware and software architecture of an infrared tracking system was presented and some advances in the current state-of-the-art have were identifying. The tracking system is appropriate for large scale immersive virtual reality environments or augmented reality settings. A full engineering approach was addressed, from clear user requirements to system specifications, development and testing plans, software and hardware design and engineering, followed by an analysis of system performance, precision and accuracy. Although we have achieved sufficient performance, precision and accuracy according to the requirements previously set, we have still several ideas to improve the system in the future. By observing the accuracy results in translation, one of the challenges is to decrease the maximum observed error in the artefact pose retrieval aiming at better system reliability. A system output assessment needs to be done in order to determine the source of the problem and to address the respective improvements. An alternative approach is the introduction of predicting filters (e.g. Kalman Filter), to perform comparisons between estimated and predicted pose and to develop heuristics to choose the best fit, in order to minimize the pose error. Another envisaged improvement is a better artefact design based on topology assessment. Our lab-made artefacts do not have the sub-millimetre precision required to enhance the model fitting complexity reduction, which force us to have an off-line artefact calibration phase. We hope to achieve a precise artefact construction which will allow us to suppress the artefact calibration phase and introduce artefacts in the system only by its geometric and topological description. Approaches to design artefacts that maximize the minimal Euclidean distance across all markers [Pintaric08] are foreseen and for that, collaboration with the Architecture department of ISCTE is envisaged. By reducing the size of markers, which reduce the probability of occlusion, we could improve the features detection. Another way to develop the feature detection is through hardware segmentation. CCD cameras allow us to regulate the light that enters through the camera lens, or to change the exposure time. These two changes usually create a detectable gradient between two features that were previously occluded, avoiding the utilization of the costly Hough Circle Transform. However, the artefact support infrastructure couldn't reflect light otherwise this would create even more unpredictable occlusions. Alternatively, we could combine information from several cameras. This is especially beneficial when, in a certain view, a feature is occluded and, in another view, the same feature is detected only through the pixel-connected algorithm. This information could be correlated in the Multiple View Correlation phase and used to solve the feature segmentation problem in a robust way. We also plan to investigate the applicability of GPU computing and advanced parallel programming in our system.

10. ACKNOLEGMENTS

We would like to thank Rui Varela and Pedro Martins, of ISCTE for designing and developing the system's electronics and to Luciano Pereira Soares of Petrobras for the lens model simulation.

11. REFERENCES

[Abdel-Aziz71] Abdel-Aziz,Y.I., Karara, H.M., "Direct Linear Transformation into Object Space Coordinates in Close-Range Photogrammetry", *Procedures of Symposium of Close-Range Photogrammetry*, January 1971.

[ART09] Advanced Realtime Tracking GmbH, Website: http://www.ar-tracking.de/, last visited on 19.08.09.

[Arun87] Arun, K. S., Huang, T. S., Blostein, S. D., "Least-squares fitting of two 3-D point sets*", IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1987.

[AVT09] Allied Vision Technologies, Website: http://www.alliedvisiontec.com/, visited at 16.06.2009.

[Cruz-Neira92] Cruz-Neira, C., Sandin, D., DeFanti, T., Kenyon, R., Hart, J., "The CAVE: Audio Visual Experience Automatic Virtual Environment". *Communications of the ACM 35*, 1992 (65–72).

[DeMenthon92] DeMenthon D., Davis, L. S., "Model-Based Object Pose in 25 Lines of Code", European Conference on Computer Vision, 1992.

[Dias05] Dias, J. M. S., Jamal, N., Silva, P., Bastos R., "ARTIC: Augmented Reality Tangible Interface by Colour Evaluation", *Interacção 2004*, Portugal, 2004.

[Dias06] Dias, J. M. S., Bastos, R., "An Optimized Marker Tracking System", *Eurographics Symposium on Virtual Environments*, 2006.

[Dias07] Dias, J. M. S. et al., "CAVE-HOLLOWSPACE do Lousal – Príncipios Teóricos e Desenvolvimento", *Curso Curto, 15º Encontro Português de Computação Gráfica*, Microsoft, Tagus Park, Porto Salvo, Portugal, 17th Oct 2007.

[Duda72] Duda, R. O., Hart, P. E., "Use of the Hough transformation to detect lines and curves in pictures", *Communications of the Association for Computing Machinery 15*, Jan 1972.

[Fitzgibbon95] Fitzgibbon, A. W., Fisher, R. B., "A buyer's guide to conic fitting", *Proceedings of the 5th British Machine Vision Conference*, Birmingham, 1995 (513–522).

[Golub93] Golub, G. H., Van Loan, C. F., "Matrix Computations", *Johns Hopkins University Press 2nd edition*, 1993.

[Haralick89] Haralick, R. M., Joo, H., Lee, C.N., Zhuang, X., Vaidya , V.G. , Kim, M.B., "Pose estimation from corresponding point data". *IEEE Trans. Sys. Man. Cybernetics*, 1989 (1426-1446).

[Hartley03] Hartley, R., Zisserman, A., "Multiple View Geometry in computer vision", Cambridge University Press, 2003.

[Heikkila97] Heikkila, J., Silven, O., "A Four-step Camera Calibration Procedure with Implicit Image Correction", *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition*, 1997.

[Kato99] Kato, H., Billinghurst, M., "Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System", *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality*, October 1999.

[Kimme75] Kimme, C., Ballard, D. H., Sklansky, J., "Finding circles by an array of accumulators," *Communications of the Association for Computing Machinery 18*, 1975.

[Konk07] Konc, J., Janežiči, D., "An improved branch and bound algorithm for the maximum clique problem", *MATCH Communications in Mathematical and in Computer Chemistry 58*, 2007.

[Lowe91] Lowe, D. G., "Fitting Parameterized Three-Dimensional Models to Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.13, 1991.

[NI09] National Instruments, Website: http://www.ni.com/, last visited on 18.06.09.

[OpenCV] Open Computer Vision Library, Website: http://sourceforge.net/projects/opencvlibrary/, last visited on 24.08.09.

[Pintaric07] Pintaric, T., Kaufmann, H., "Affordable Infrared-Optical Pose-Tracking for Virtual and Augmented Reality", *Proceedings of Trends and Issues in Tracking for Virtual Environments Workshop*, *IEEE VR*, 2007.

[Pintaric08] Pintaric, T., Kaufmann, H., "A Rigid-Body Target Design Methodology for Optical Pose Tracking Systems", *Proceedings of the 2008 ACM symposium on Virtual reality software and technology*, 2008.

[Santos06] Santos, P., Buanes, A., Jorge, J., "PTrack: Introducing a Novel Iterative Geometric Pose Estimation for a Marker-based Single Camera Tracking System", *IEEE Virtual Reality Conference (VR 2006)*, 2006.

[SK09] Schneider Kreuznach, Website: http://www.schneiderkreuznach.com, last visited on 18.06.09.

[Vicon09] Vicon Motion Systems, Website: http://www.vicon.com, last visited on 19.08.09. [Wang04] Wang, F., "A Simple and Analytical Procedure for Calibrating Extrinsic Camera Parameters", *IEEE Transactions on Robotics and Automation, Vol. 20, No. 1*, Feb 2004.

[Zhang96] Zhang, Z., "Parameter estimation techniques: A tutorial with application to conic fitting", *Image and Vision Computing 15*, 1996 (59-76).

[Zhang99] Zhang, Z., "Flexible Camera Calibration by Viewing a Plane from Unknown Orientations", *International Conference on Computer Vision*, 1999.

[Zhao07] Zhao, J., Yan, D., Men, G., Zhang, Y., "A method of calibrating intrinsic and extrinsic camera parameters separately for multi-camera systems", *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics*, Hong Kong, 2007.