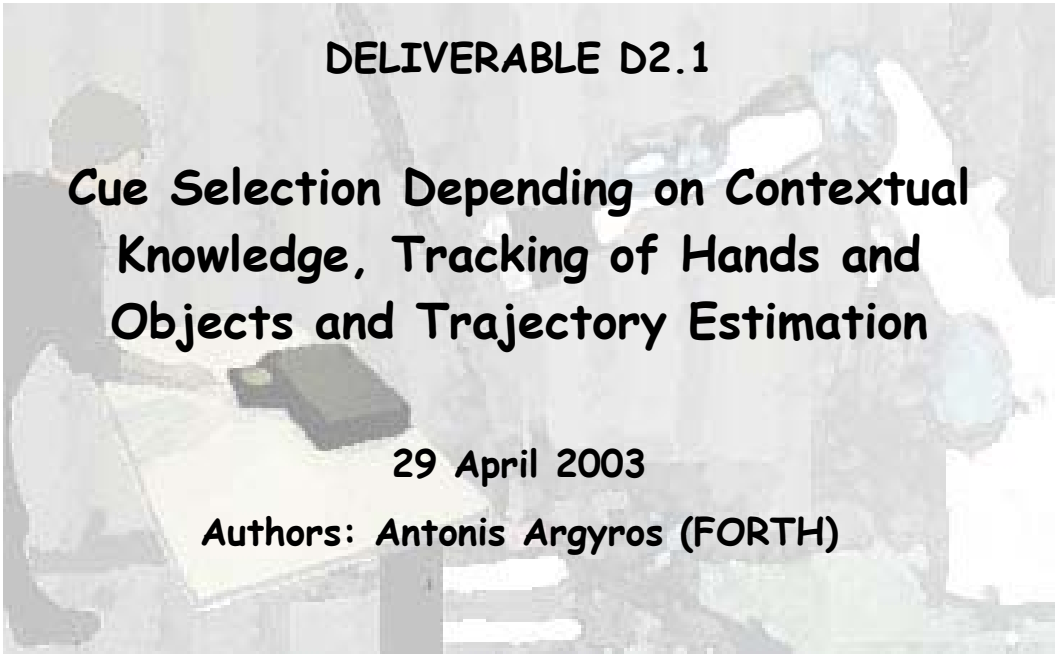


DELIVERABLE D2.1

Cue Selection Depending on Contextual Knowledge, Tracking of Hands and Objects and Trajectory Estimation

29 April 2003

Authors: Antonis Argyros (FORTH)

FOUNDATION FOR RESEARCH AND TECHNOLOGY -HELLAS
PROFACTOR Research for Success
The European Commission Community Research ist information society technologies

TABLE OF CONTENTS

1. Introduction

Deliverable D2.1 is software (demonstration) deliverable. The demonstration will show how cue integration is used to obtain robust hand tracking. It will also demonstrate the quality of the 3D hand trajectories estimated through tracking.

D2.1 is closely linked with D3.2, which presents the cue integration methods and relations of cues and features. The next table summarizes the content originally intended for deliverables D2.1 and D3.2. It also provides the new distribution of content, which results from the wish to present *methods together with results* for each of the two components of the ActIPret framework in *one* deliverable: hand tracking in D2.1 and object tracking in D3.2.

Component	**Hand tracking**	**Object Tracking**
Deliverables	**(new D2.1 = this column)**	**(new D3.2 = this column)**
(original D2.1 = this row)	Cue selection method, trajectory estimation	Cue selection method, trajectory estimation
(original D3.2 = this row)	Relations of image descriptors = cue integration	Relations of image descriptors = cue integration

Table 1: *Old and new content of Deliverables D2.1 and D3.2. The new content results from presenting complete methods and results for the two components of the ActIPret framework in one Deliverable: hand tracking in D2.1 and object tracking in D3.2.*

This document accompanies the software (demonstration) deliverable and its aim is to provide a sort description of the Hand-Tracker component (HT) developed in the context of the ActIPret project.

The rest of the document is organized as follows. Section 2 describes the overall functionality of the HT component. Section 3 describes the interconnection of HT with the rest of the components within the ActIPret framework. In section 4, issues related to the computational performance of the HT component are discussed. Section 5 provides sample results of the operation of the HT component in prerecorded image sequences. Section 6 provides a list of extensions and ideas for improvements that are still under investigation. The main conclusions of this work are summarized in section 7.

2. Hand Tracker (HT) component description

The hand tracker (HT) component that has been developed in the context of the ActIPret project is able to detect multiple hand hypotheses and report the 3D position of each such hypothesis in a scene observed by a moving stereoscopic system, as the one shown in Fig. 1.

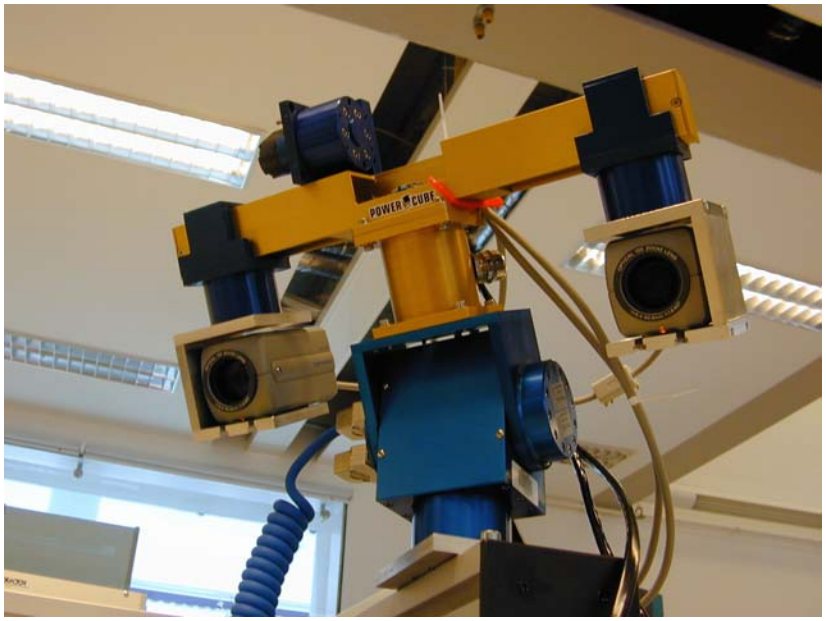


Figure 1: *The stereoscopic head (courtesy Profactor GmbH) that is used to acquire image stereo-pairs that feed the HT component.*

The stereoscopic vision system provides, except from image stereo-pairs, the position and the orientation of the two cameras with respect to a world-centered coordinate system. A timestamp-based mechanism guarantees the required synchronization of image acquisition and camera position estimation processes.

The developed HT component exploits multiple cues towards hand tracking. Example cues include color information, motion and structure information as well as information regarding the known camera positions and epipolar geometry of the stereo system. Figure 2 provides a high-level block-diagram of the developed HT component.

The HT component operates as follows. At each time *t*, the stereoscopic head acquires a synchronized image stereo-pair, $I_L(t)$ and $I_R(t)$. Each of these images is independently fed into a skin color detection (SCD) module. SCD involves (a) measurement of the probability of a pixel being skin colored (b) hysteresis thresholding on the derived probabilities (c) connected components

labeling to come up with skin-colored blobs constituting hand hypotheses (HH) and (d) computation of statistic information for each HH (up to 2nd order moments for each HH). The derived HHs, together with the hand hypotheses derived in the previous time instance *t-1*, are then fed into the "association of hands in time" (AHHT) module. The aim of this module is (a) to assign a new, unique hand ID to each new hand hypothesis (i.e. a HH that appears in the field of view (f.o.v.) for the first time) and (b) to propagate the hand ID of already detected hand hypotheses in time, guaranteeing this way that the same physical hand hypothesis is assigned always the same hand ID. Then, the left and the right hand hypotheses, together with the associated hand IDs, are fed into a module that associates hand hypotheses between the two images of the stereo-pair (AHHS). In fact, the hand hypotheses of the right image of the stereo-pair are assigned the hand-IDs of their corresponding hand hypotheses in the left image of the stereo-pair. As soon as this type of association is completed, the centroids of the corresponding hand hypotheses are refined (Centroid Matching, CM module) based on a correlation-based technique to guarantee that these points correspond to the same 3D point in the scene. The refined matches are then fed into a 3D reconstruction (3DR) module which, taking into account the known geometry of the stereoscopic system as well as the intrinsic calibration parameters of each of the cameras, computes the 3D location of each hand hypothesis. Finally, the reported position of each hand hypothesis is a weighted sum of 3D measurements in a sliding time window. This functionality is provided by the temporal smoothing (TS) module.

What follows, is a more detailed description of each of these modules.

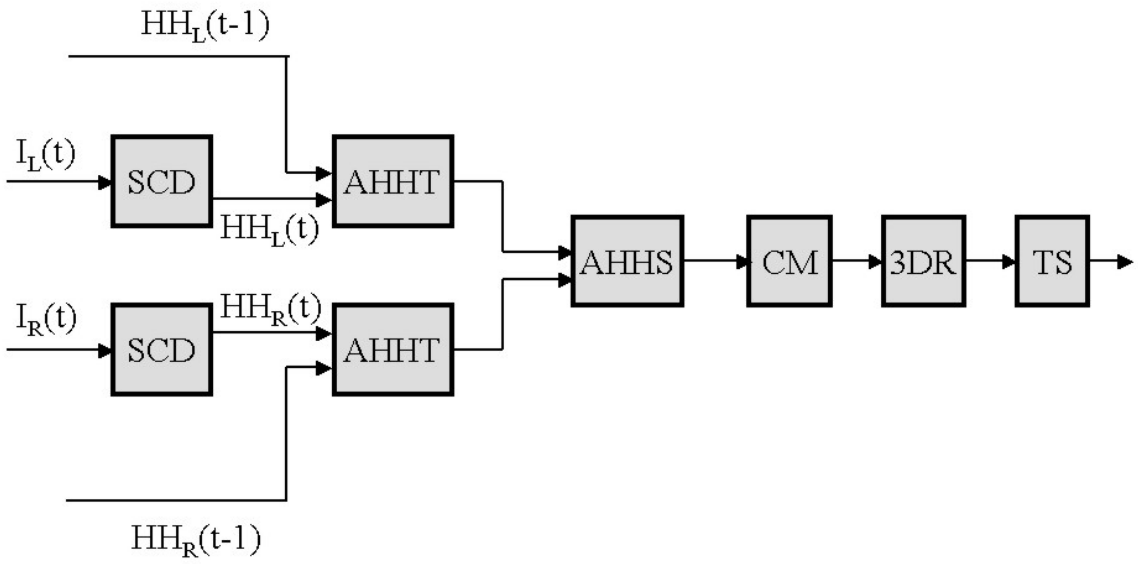


Figure 2: *Block diagram of the FORTH Hand Tracker component.*

2.1. Skin color detection (SCD)

Skin color detection (SCD) is one of the fundamental building blocks of the developed HT component. The goal of the SCD module is to detect skin colored blobs in an image, each corresponding to a different hand hypothesis. SCD is based on a Bayesian approach. It involves (a) an adaptive, off-line training phase and an adaptive, on-line detection phase.

2.1.1 Basic training and detection mechanisms

A set of training images is selected on which a human operator marks skin-colored regions. The color representation used in this process is the YUV 4:2:2 color representation that is the direct output of the firewire cameras used in the stereoscopic system of Fig. 1. However, the Y-component of this representation is not employed for two reasons: (a) the Y-component codes the illumination of an image point and therefore, by omitting it the developed classifier gains illumination-independence characteristics, (b) by employing a 2D color representation (UV), as opposed to a 3D one (YUV), the dimensionality of the problem is reduced and therefore the computational performance of the overall system is improved.

Assuming that image points *I(x, y)* have a color C(x, y) = (u, v), the training set is used to compute:

- The probability *P(S)* of having skin color in an image. This is the ratio of the skin-colored image points in the training set over the total number of image points.
- The probability *P(C)* of occurrence of each color *C* in the training set. This is computed as the ratio of occurrences of each color *C* over the total number of image points in the training set.
- The probability *P(C|S)* of a color *C* being a skin color. This is defined as the ratio of occurrences of a color C within the skin-colored areas over the number of skin-colored image points in the training set.

As soon as training has been performed, the probability *P(S|C)* of an image point with color *C* to be skin-colored can be computed by employing the Bayes rule [1]:

$$P(S \mid C) = \frac{P(C \mid S)P(S)}{P(C)} \tag{1}$$

For each point I(x, y) of the input image having color C = C(x, y) = (u, v), eq. (1) is employed to estimate the probability *P(S|C)* based on the prior probabilities

computed from the training set. The probability map that results from this operation is then thresholded and all image points with probability $P(S|C) > T_{max}$ are considered as skin-colored points. These points constitute the seeds of potential hand hypotheses. More precisely, pixels with probability $P(S|C) > T_{min}$, $T_{min} < T_{max}$, that are immediate neighbors of skin-colored image points are recursively added to the set of skin-colored points. This hysteresis-thresholding type of operation proves extremely robust in identifying skin-colored blobs. Indicative values for the thresholds T_{max} and T_{min} are 0.5 and 0.15, respectively. A connected components labeling algorithm is then responsible for assigning different labels to the pixels of different skin-colored blobs. Blobs that consist of less than T_{size} image points are rejected from further consideration. Each of the remaining blobs corresponds to a different hand hypothesis, HH. The final step in skin color detection is the computation of the 2D position of each hand hypothesis in the image (defined as the center of mass of all skin-colored image points of the corresponding blob) and other statistics that correspond to shape properties of each blob.

2.1.2 Adaptability

The basic scheme for SCD described in the previous section has two major drawbacks:

- **Training:** Training is an off-line procedure that does not affect the on-line performance of the hand tracker. Nevertheless, it is a very time-consuming spadework, in the sense that a human operator should mark all skin-colored pixels in the selected training set. Moreover, in order to come-up with a training set that is capable of supporting tracking of various skin-tones in images acquired from different cameras, a large training set is required. Therefore, a method that facilitates the process of acquiring training data is considered quite important.
- **Detection:** In the case of varying illumination conditions, the SCD module may produce poor results, even if the used color representation has illumination-independence characteristics. Therefore, a method is required that adapts the notion of skin-colored image points according to the recent history of detected skin-colored points.

To cope with the first problem, an adaptive training procedure has been developed. Training is performed on an initial, small set of images for which the human operator provides ground truth by defining skin-colored areas. Then, detection, together with hysteresis thresholding is used to continuously update the prior probabilities $P(S)$, $P(C)$ and $P(C|S)$ in new images. The updated prior probabilities are then used to re-classify the full data set into skin-colored and non-skin colored pixels. In cases where the classifier produces wrong results (false positives / false negatives) the role of the user is just to correct these errors; still, the classifier has already done by itself much of the spadework. The

final training of the classifier is then performed on the large training set that results after editing. The process of adaptation can either be disabled as soon as it is decided that the achieved training is sufficient for the purposes of the HT component or continue as more input images are fed into the system. In fact, the only reason for disabling the adaptation process is to release the system from the associated computational costs.

It is important to note that hysteresis-thresholding is very crucial for achieving the previously described adaptation of probabilities. This is because, due to hysteresis thresholding, image points with relatively low probability of being skin-colored are considered as skin colored, which permits the adaptation of their probabilities.

A basic advantage of the proposed scheme lies in its simplicity. Other methods for adaptation have been proposed in the literature [2]. However, these methods require much more complex modeling of the color characteristics of skin (i.e. modeling based on mixtures of Gaussians). A quantitative comparison of the two approaches is an ongoing task.

To solve the second problem, the SCD module maintains two sets of prior probabilities *P(S)*, *P(C)*, *P(C|S)* (corresponding to the training set) and $P_W(S)$, $P_W(C)$, $P_W(C|S)$ corresponding to the evidence that the system gathers during the *W* most recent frames. Evidently, the second set better reflects the "recent" appearance of hand hypotheses and is better adapted to the current illumination conditions. SCD is then performed based on:

$$P_A(S \mid C) = aP(S \mid C) + (1-a)P_W(S \mid C) \tag{2}$$

In eq. (2), a is a parameter that controls the influence of the training set in the detection process ($0 < a \leq 1$). If $a = 1$ then SCD takes into account only the training set and no adaptation takes place; if a is close to zero, then the SCD becomes very "reactive", taking into account mostly the recent past as a model of the immediate future. A value of $a = 0.8$ gives very good results in the preliminary tests that have been carried out.

2.2. Associating hand hypotheses in time (AHHT)

As soon as a hand hypothesis is detected, it has to be tracked over time. This is a crucial functionality of the HT component since it provides the temporal continuity of hand hypotheses observations.

We denote with H_t all hand hypotheses detected at time *t* and with $H_t(i)$ a specific hand hypothesis detected at time t, $1 \le i \le N(t)$. A distance measure $D_T(H_{t-1}(i), H_t(j))$ is defined between two hand hypotheses $H_{t-1}(i)$ and $H_t(j)$ that have been detected at times t-1 and t respectively:

$$D_T(H_{t-1}(i), H_t(j)) = \sqrt{\left(cx_i - cx_j\right)^2 + \left(cy_i - cy_j\right)^2} \tag{3}$$

In eq. (3), (cx_i, cy_i) and (cx_j, cy_j) are the centroids of hand hypotheses $H_{t-1}(i)$ and $H_t(j)$. Equation (3) states that the distance between two hand hypotheses is the Euclidean distance of their centroids. A hand hypothesis $H_{t-1}(i)$ matches a hand hypothesis $H_t(j)$ if:

$$D_T(H_{t-1}(i), H_t(j)) = \min_{1 \le k \le N(t)} \left\{ D_T(H_{t-1}(i), H_t(k)) \right\} \tag{4}$$

Two hand hypotheses $H_{t-1}(i)$ and $H_t(j)$ are assumed to correspond to the same physical object if

- $H_{t-1}(i)$ matches $H_t(j)$
- $H_t(j)$ matches $H_{t-1}(i)$
- $D_T(H_{t-1}(i), H_t(j)) < T_D$, where T_D is a predefined threshold depending on the image acquisition frame-rate and on the speed of the hands.

For all corresponding hypotheses, their IDs from the previous time step are propagated to the current step. All hand hypotheses at time *t* that have not been corresponded with hand hypotheses at time *t-1* are assigned new IDs, since these are hand hypotheses observed for the first time.

AHHT is performed independently on the left and the right image of the stereo-pair. The process is simple and computationally cheap. Moreover, it proves very robust in all cases where hand hypotheses do not overlap due to occlusions. In the cases of occlusions, AHHT cannot disambiguate hand hypotheses. Current developments consider the possibility of initializing a Kalman tracker for each new hand hypothesis, which will lead to increased robustness in case of occlusions (see section 6 for more details).

2.3. Associating hand hypotheses in a stereo-pair (AHHS)

In order to provide information regarding the 3D position of each hand hypothesis, the tracker should also be able to associate hands between the two images of a stereo-pair. The AHHS module serves this purpose.

As it has already been stated earlier in this document, it is assumed that the position and orientation of each camera of the stereo pair is known with respect to a world-centered coordinate system. Based on this information, it is possible to compute the rotation matrix *R* and the translation vector *t* of the relative motion between the coordinate systems of the cameras of the stereoscopic system. This, in turn, provides the means to compute the fundamental matrix *F* that codes the epipolar geometry of the stereo-pair:

$$F = \frac{1}{\det(A_1)}[e_1]_x H_\infty \tag{5}$$

where

$$e_1 = A_1 t \tag{6}$$

$$H_\infty = A_1 R A_0^{-1} \tag{7}$$

In the above equations, A_0 and A_1 are the intrinsic calibration matrices of the left and the right cameras respectively, e_1 is the right epipole, $[e_1]_x$ is the skew symmetric matrix of the right epipole and H_∞ is the homography at infinity. Then, it is known [3] that if m_0 and m_1 are two corresponding points in the left and the right images of the stereo pair, then m_1 is constrained to lie on the line defined as Fm_0.

The AHHS module is based on this result to associate hand hypotheses between the two images of the stereo pair. Similarly to the case of the AHHT module, we denote with H_L all hand hypotheses detected at time *t* in the left image of the stereo pair and with H_R all hand hypotheses detected at time *t* in the right image of the stereo pair. Moreover, $H_L(i)$ and $H_R(j)$ denote specific hand hypotheses *i* and *j* detected at time t, in the left and right images, $1 \le i \le N(L)$, $1 \le j \le N(R)$. A distance measure $D_S(H_L(i), H_R(j))$ is defined between two hand hypotheses as:

$$D_S(H_L(i), H_R(j)) = \max\left\{d\left(Fm_i, m_j\right), d\left(F^T m_j, m_i\right)\right\} \quad (8)$$

In the above equation, m_i and m_j are the centroids of hypotheses $H_L(i)$ and $H_R(j)$, respectively, and $d(l, p)$ denotes the distance of point p from the line l. A hand hypothesis $H_L(i)$ matches a hand hypothesis $H_R(j)$ if:

$$D_S(H_L(i), H_R(j)) = \min_{1 \le k \le N(R)} \{D_S(H_L(i), H_R(k))\} \quad (9)$$

Symmetrically, a hand hypothesis $H_R(j)$ matches a hand hypothesis $H_L(i)$ if:

$$D_S(H_R(j), H_L(i)) = \min_{1 \le k \le N(L)} \{D_S(H_R(j), H_L(k))\} \quad (10)$$

Two hand hypotheses $H_L(i)$ and $H_R(j)$ are assumed to correspond to the same physical object if

- $H_L(i)$ matches $H_R(j)$
- $H_R(j)$ matches $H_L(i)$
- $D_S(H_L(i), H_R(j)) < T_S$, where T_S is a predefined threshold depending on the accuracy in the computation of the epipolar geometry.

For all corresponding hand hypotheses, the ID of the hand hypothesis in the left image are propagated to the corresponding hand hypothesis in the right image of the stereo pair. All other non-corresponded hand hypotheses are excluded from further consideration in the subsequent process of 3D position estimation. Such hypotheses can be due to hands that are visible in only one of the two cameras of the stereo-pair.

The above method for associating hands may fail in the case where epipolar geometry is not accurately computed. In this case, the threshold T_S has to be set conservatively to a quite high value, which leaves room for errors in the association of hand hypotheses. For this reason, 3D position information from previous time instances is used, if available. More specifically, when computing distances $D_S(H_L(i), H_R(j))$, the 3D position of the hand hypothesis that results form the assumption that hand hypothesis $H_L(i)$ really corresponds to the hand hypothesis $H_R(j)$, is computed. If the resulting 3D position is invalid (in the sense

that either this position is not plausible or it differs substantially from the hand position in the previous time instant) a penalty term is added in the corresponding distance measure to guarantee that hand hypothesis $H_L(i)$ will not be considered as corresponding to hand hypothesis $H_R(j)$.

In general, the camera position and orientation information that is computed from the encoders of the stereoscopic system is not accurate enough to enable the accurate estimation of the epipolar geometry of the stereo pair. In experiments carried out with prerecorded image sequences, the average error of image points from their epipolar lines is in the order of 15 pixels. This, in turn, affects the robustness of the AHHS module. To overcome this problem, AHHS is applied only to hands that appear in the field of view for the first time. As soon as this is achieved, AHHT module, which is more robust compared to AHHS, assumes the role of propagating the correct hand IDs in both images of the stereo pair. This is further exemplified in Fig. 3.

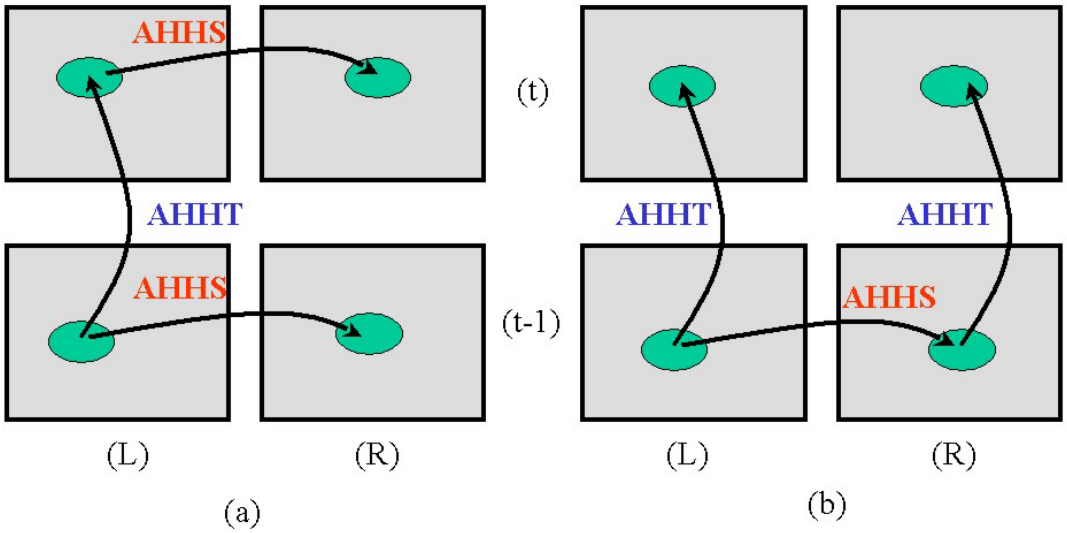


Figure 3: *Two scenarios ((a), (b)) for achieving the propagation of IDs of hand hypotheses both in time and between the two views of a stereo pair. (a) The AHHS module is used to associate hands between the left and the right images of the stereo pair, at each moment in time. The AHHT module is used to propagate labels in time, only in the image sequence of the left camera. (b) The AHHS module is used to associate hands only when a new hand hypothesis appears in the field of view. Two instances of the AHHT module are then used to propagate the hand IDs in time, independently in the left and right image sequences. Since AHHT is more robust compared to AHHS, the second approach is adopted.*

A more accurate computation of the epipolar geometry of the stereo system that is derived through image measurements (as opposed to the encoders based estimation currently employed) will greatly improve the robustness of the AHHS module. Related research at FORTH [4, 5] has resulted in an accurate and efficient image-based camera tracking system. Appendix A provides some further information regarding this method that can prove extremely useful to the HT component.

2.4. Centroid matching (CM)

The correspondence of hand hypotheses between the left and the right images of the stereo pair that has been achieved up to now, leads to a rough correspondence between hand centroids. This could be directly used for deriving the 3D position of hand hypotheses. However, centroids have been computed by the SCD module, as the mean *x*- and *y*-coordinate of each skin-colored blob, therefore it is not guaranteed that the left and right centroids of the same hand hypothesis correspond to the same 3D point. In order to refine this initial rough correspondence, a correlation-based matching algorithm is employed. Let *m* denote the centroid of a hand hypothesis in the left image of the stereo pair and *m'* denote the centroid of the same hand hypothesis in the right image of the stereo pair. Then, a model region *M* around m and a search window *S* around *m'* are defined. *M* is placed within all possible positions in *S* and a correlation measure is computed. The location *m''* in *S* where the correlation measure *C* is maximized (C_{MAX}) is considered as the refined right centroid of the specific hand hypothesis. The process is repeated symmetrically, by defining a model region around *m'* and a search region around *m*. If this search gives rise to a correlation score greater than C_{MAX} for some point *m'''* in the left image, then we consider the (*m''', m'*) pair of centroid correspondences instead of the (*m, m''*) pair. This centroid refinement process is repeated for all pairs of corresponding hand hypotheses. Note that if epipolar geometry was accurate enough, search lines (epipolar lines) could be used instead of search regions. This could result in substantial reduction of the required computations. However, since the epipolar geometry is not accurate enough, this type of optimization has not been taken into account.

The correlation measure used in the CM module is inspired by the work of Hirschmüller [6] on dense stereo matching. The basic idea behind this selection is that the model window M is divided into five overlapping sub-windows, a central (C), an upper-left (UL), an upper-right (UR), a bottom-left (BL) and a bottom-right (BR). These five windows have a certain overlap each other, as shown in Fig. 4. At each placement of the model window *M* in the search window *S*, five correlation values C_C, C_{UL}, C_{UR}, C_{BL} and C_{BR}, are computed independently. These values measure the correlation of each sub-window with the corresponding image part in the search window. Then, the correlation value C

for this particular placement can be computed by adding the values of the two best surrounding correlation windows C_{max1} and C_{max2} to the middle one:

$$C = C_C + C_{max1} + C_{max2} \quad (11)$$

This approach uses, in fact, a small central window and supports the correlation decision by four nearby windows. This formulation enables the refinement process to cope very well with depth discontinuities and occluded/revealed regions that introduce errors when conventional correlation is employed between the whole model window and the corresponding part of the search window. It should be noted that the cases of depth discontinuities and occlusions are very common in the particular hand-tracking scenario, where the hand figure is typically a small image region, quite closer to the cameras compared to its immediate surroundings.

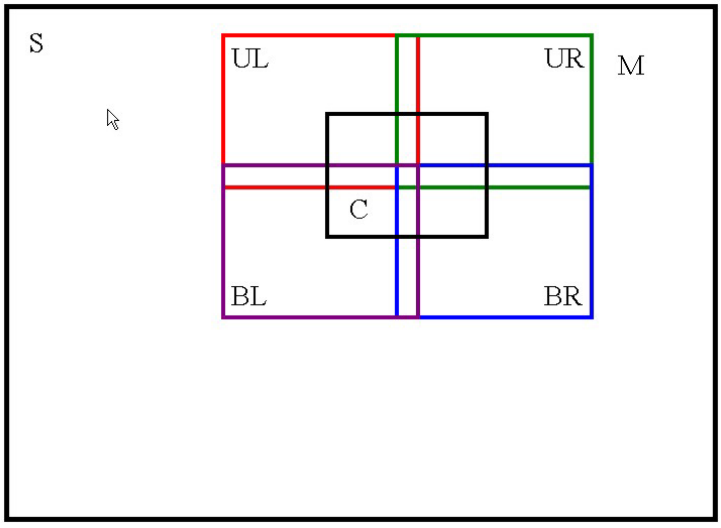


Figure 4: *The configuration of overlapping windows used in the correlation method proposed by Hirschmüller [6].*

2.5. 3D recovery of the position of hands (3DR)

The refined centroid correspondences of the hand hypotheses are fed into a 3D reconstruction module, which computes the 3D position of each hand hypothesis. Two different reconstruction methods have been tested.

The first method [7] computes the 3D position *(X, Y, Z)* of a point *P*, given its projections m_0 and m_1 in the left and the right image of a stereo pair, as follows:

$$Z = -\frac{(m_1 \times e_1)\cdot(m_1 \times H_\infty m_0)}{\|m_1 \times H_\infty m_0\|^2}$$
$$X = Z[A_0^{-1}(0)]m_0 \qquad (12)$$
$$Y = Z[A_0^{-1}(1)]m_0$$

In eq. (12), m_0 and m_1 are vectors in homogeneous coordinates, and e_1 and H_∞ are defined as in eqs. (6), (7), respectively. Moreover, $[A_0^{-1}(i)]$ denotes the vector that corresponds to the *i*th row of the inverse of the intrinsic camera parameters matrix of the left camera. Equation (12) gives the 3D position *(X,Y,Z)* of a point *P* with respect to the coordinate system of the left camera. The 3D position of this point with respect to the world-centered coordinate system can easily be estimated through a rigid 3D transformation.

The second method considers directly the intersection of two 3D lines. More specifically, one 3D line is defined by 3D points p_1 and q_1, where p_1 is the origin of the coordinate system of the left camera and q_1 is the 3D position of centroid of a hand hypothesis on the left image plane. Similarly, a second 3D line is defined by 3D points p_2 and q_2 where p_2 is the origin of the coordinate system of the right camera and q_2 is the 3D position of the centroid of the same hand hypothesis on the right image. Then, the 3D location *P* of the hand hypothesis is:

$$P = \frac{1}{2}(p_1 + \hat{v}_1 s_1 + p_2 + \hat{v}_2 s_2) \qquad (13)$$

where

$$s_1 = \frac{\det(p_2 - p_1 \quad \hat{v}_2 \quad v_{12})}{|v_{12}|^2} \qquad s_2 = \frac{\det(p_2 - p_1 \quad \hat{v}_1 \quad v_{12})}{|v_{12}|^2} \qquad (14)$$

and

$$\hat{v}_1 = \frac{q_1 - p_1}{|q_1 - p_1|} \qquad \hat{v}_2 = \frac{q_2 - p_2}{|q_2 - p_2|} \qquad v_{12} = v_1 \times v_2 \qquad (15)$$

If the 3D lines really intersect, then *P* in eq. (13), is the point of their intersection. If the 3D lines are skew, then *P* is the midpoint of the minimum-length line segment that connects the two 3D lines.

This second method provides more accurate 3D reconstruction results compared to the first reconstruction approach and therefore, has been adopted in the 3DR module of the HT component.

2.6. Temporal smoothing (TS)

The temporal smoothing (TS) module performs temporal smoothing of the derived 3D position of each hand hypothesis, based on the assumption that the hand trajectory is smooth as a function of time. The current implementation considers 3D positions P_{t-2}, P_{t-1} and P_t of a hand as they have been computed in the last three time instances *t-2*, *t-1* and *t*, and reports the 3D position *P* defined as:

$$P = 0.6P_t + 0.3P_{t-1} + 0.1P_{t-2} \tag{16}$$

Weights are appropriately adapted in case that the 3D position measurements in time instances *t-2* and/or *t-1,* are not available.

3. HT in the context of the ActIPret framework

The HT component described in the previous section has been initially implemented in standard C, as a stand-alone application that takes as input sequences of stereoscopic images. This permits extensive, off-line experimentation and testing. However, in the context of the ActIPret project, HT is on out of many components of a cognitive vision framework. A schematic presentation of this framework, which shows the interaction of HT with the rest of the components, is shown in Fig. 5.

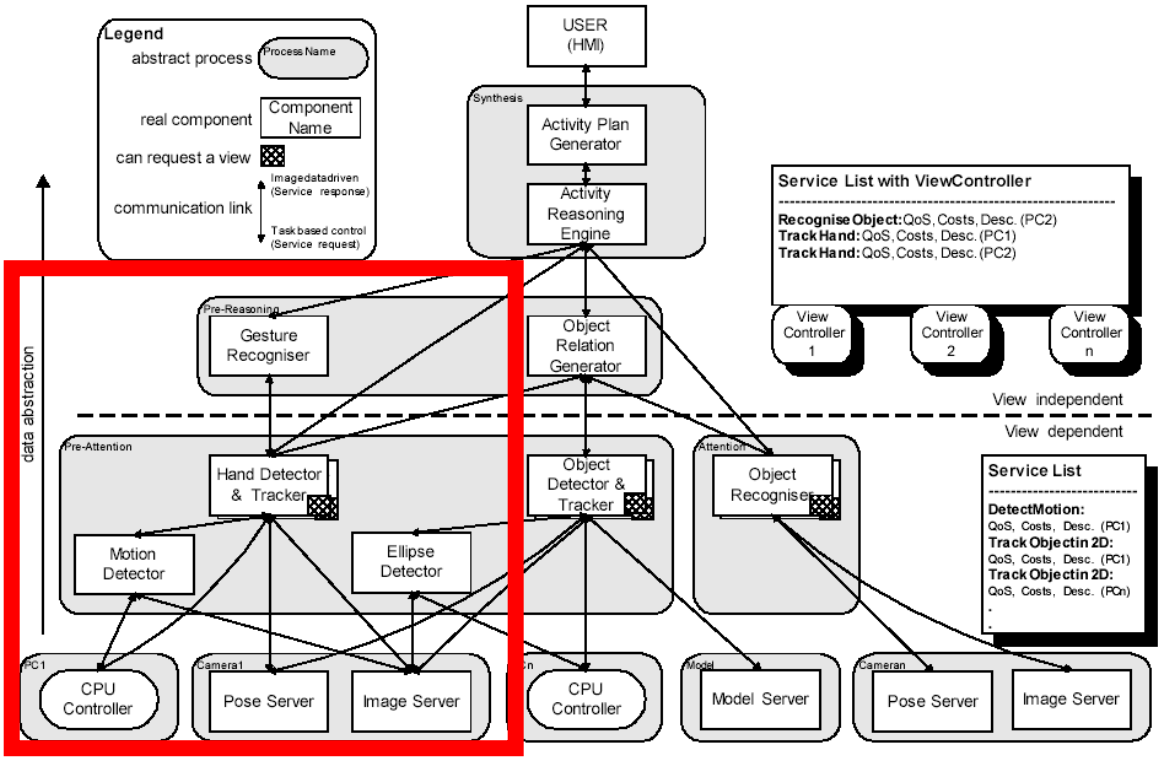


Figure 5: *The HT component within the ActIPret framework. The HT component interacts with the Image Server and Pose Server components in order to acquire the required image, camera geometry and camera calibration data. Moreover, it has to communicate with the Gesture Recognizer component to deliver the necessary information regarding the position of the tracked hands.*

As it is shown in Fig. 5, the HT component needs to communicate with the Image Server and Pose Server components in order to acquire the required image, camera geometry and camera intrinsic calibration data. Moreover, it has to communicate with the Gesture Recognizer component to provide the necessary information regarding the position of the tracked hands. The activation of the components is accomplished in a top-to-bottom manner. More specifically, the Gesture Recognizer activates the HT component, asking for possible hand hypotheses within a 3D space of interest (SOI). Then, the HT component activates the Image Server and Pose Server components requesting image stereo pairs, camera position information and camera calibration data. As soon as Image Server and Pose Server respond to the HT component with the requested data, detection and tracking of hand hypotheses is initiated and the HT component reports the 3D positions of the detected hand hypotheses to the Gesture Recognizer.

To support the necessary interaction of the HT component with the rest of the components in the ActIPret framework, the original, framework-independent, core HT functionality has been extended with a framework-dependent layer, which supports the necessary communication structures. The interaction of the HT component with the rest of the components in the ActIPret framework has been verified in several project integration meetings.

4. HT performance considerations

Several tests have been carried out aiming at assessing the functionality and the performance of the HT component. Both off-line experiments (involving pre-recorded image sequences) and on-line experiments have been carried out. Off-line experiments employed the framework-independent version of the HT component, while on-line experiments have been carried-out by employing the framework-dependent version of it. The performance of the HT component within the framework is relatively difficult to assess because of the complex interaction of components. However, it is straightforward to measure the performance of HT in off-line experiments. It turns out that one-cycle of operations of the HT component takes approximately 75 milliseconds on an Intel P4@1.8 GHz running Linux. The cycle includes all HT functionality plus reading a stereo pair of 640x480 images from disk. The HT may be forced to operate in sub-sampled versions of the original images. If the input images are sub-sampled by a factor of two (i.e. 320x240 images are employed), then the HT-cycle time becomes 35 milliseconds. The reason why the performance gain is not directly proportional to the input data reduction (i.e. a factor of four) is that the HT always reads full-resolution images from disk and therefore, disk I/O for image reading takes constant time, independent of the operational image resolution.

An important observation is that the hand trajectories computed in full image resolution are in close resemblance with the hand trajectories computed at half resolution. This means that, at least in the conducted experiments, significant speedup can be achieved without sacrificing much of accuracy. Still, a rate of 13 Hz (full resolution images) or 28 Hz (half resolution images) is considered sufficient for the purposes of ActIPret. Detailed quantitative analysis as well as monitoring of the performance of the HT component within the ActIPret framework is still ongoing tasks.

5. Sample results

In this section, characteristic results are provided from the application of the HT component to sequences of stereo-pair images that have been acquired off-line. These sequences have been recorded at the premises of Profactor GmbH.

In the first sequence ("Sequence17_head"), a human operator manipulates a CD player (opens the tray, picks-up a CD, places it in the tray and closes the tray). The stereoscopic system does not move in this experiment. The full sequence consists of 146 left and 146 right frames. Figure 6 (top to bottom, left to right) shows characteristic snapshots from the obtained tracking results. Every 10^{th} frame is shown in this figure. For visualization purposes, each hand hypothesis appears as a color blob superimposed on the original right image of the stereo pair. A red cross marks the centroid of each hand hypothesis. Moreover, an ellipse derived from the statistics of each hand hypothesis is shown around each color blob. It can be seen that the system identifies three hand hypotheses (1) the head of the human operator, (2) the skin-colored arm of the armchair and (3) the hand of the human operator. It can also be seen that the coloring of the hypotheses is consistent through out the whole experiment, which means that hypotheses are correctly tracked both in time and between the images of the stereo-pair.

Figure 8, shows the 3D trajectories computed by the system for the three hand hypotheses. The upper facet of the CD player has also been reconstructed, as a reference. The trajectory of the operator's hand seems qualitatively correct. In this particular sequence, the arm of the armchair does not move, while the head of the operator moves slightly. To measure the stability of the derived 3D coordinates, the 3D bounding box of all estimated 3D positions for a hand hypothesis has been computed. For the case of the static arm of the armchair, the dimensions of this bounding box are 1.2cm x 1.5cm x 1.6cm.

In the second sequence ("Sequence18_arm"), a human operator again manipulates a CD player. In this sequence the cameras are mounted on a robotic arm that moves, so cameras are also moving in time. The sequence consists of 134 left and 134 right frames. Figure 7 (top to bottom, left to right) shows characteristic snapshots from the obtained tracking results. Every 10^{th} frame is again shown in this figure. It can be seen that the system identifies two hand hypotheses (1) the head of the human operator and (2) the hand of the human operator. It can also be seen that the coloring of the hypotheses is consistent through out the whole experiment, which means that hypotheses are correctly tracked both in time and between the images of the stereo-pair. Figure 9, shows the 3D trajectory computed by the system for the hand hypothesis corresponding to the operator's hand. Since the CD player is not fully visible in this sequence, the line segment corresponding to the CD tray has been reconstructed and is displayed in Fig. 9 as reference.

6. Extensions under consideration

3D tracking of multiple hand hypotheses in scenes observed by a moving stereoscopic system is a difficult research problem in cases where a robust

performance under general conditions is required. The HT component that has been developed within the ActIPret project has several attractive features. Nevertheless, there are still important improvements that can be made and certain research and development activities at FORTH aim towards realizing such improvements. Major such improvements are the following:

- **Handling of occlusions:** The AHHT and AHHS modules have difficulties in associating hand hypotheses in time and between the images of a stereo-pair, when hand hypotheses occlude each other in the field of view of either camera. Tracking of each new hand hypothesis with a Kalman tracker is under development. It is expected that this will have a significant impact in the performance of the HT component in cases of occluded hand hypotheses.
- **Providing hand pose information:** Currently, each hand hypothesis is represented as a point in 3D space. However, in the ActIPret cognitive vision framework, it would be also desirable to provide information regarding the 3D pose of each hand hypothesis as well as information regarding the 3D positions of fingertips. Towards this goal, an alternative model of a hand is currently being investigated. A 2D hand blob is modeled as a circular palm region. Then, fingers can be modeled as long, skin-colored structures emanating from this circular region. The center of the palm can be estimated by computing the median coordinates of the image points constituting a hand blob, and the radius of the palm-circle as the median of the distances of each skin-colored pixel of the blob from the palm center. This simple model is a coarse approximation of the shape of the hand and can provide the necessary information to compute the 3D position of fingertips and evidence on the hand pose.
- **Image-based camera tracking:** In the current version of the HT component, the camera positions and orientations are provided by the Pose Server component which relies on information derived from the encoders of the stereoscopic system. This information is not accurate enough, especially in the case of moving cameras. Research at FORTH has resulted in a camera-tracking system that employs point correspondences to robustly compute the 3D position and orientation of a moving camera. This novel method [4, 5] is briefly described in Appendix A and is expected, when employed, to improve substantially the camera position and orientation estimation processes, which will in turn, improve the accuracy in estimating the 3D position of hand hypotheses.

7. Summary

In this document, a brief description of the Hand Tracker component has been provided. The current version of the system is capable of detecting and tracking multiple hand hypotheses in scenes viewed by a moving stereoscopic system in which each camera has independent pan, tilt and vergence control. The HT component operates at approximately 13 Hz in full resolution, 640x480 images

on an Intel P4@1.8 GHz running Linux. The computational performance of the system can be improved by employing subsampled versions of the original images. By reducing the input images to a size of 320x240, the HT component operates at 28 Hz, without a considerable degradation of the quality of the computed hand trajectories. Ongoing research and development activities are focused on (a) improving the robustness of hand tracking in cases of overlapping hand hypotheses (b) modelling of a hand hypothesis to permit the computation of hand pose information (c) development of an efficient and accurate image-based camera tracking system, (d) extensive qualitative and quantitative testing of the developed hand tracker and (e) performance optimisations, especially in the case of the framework-dependent version of the HT component.

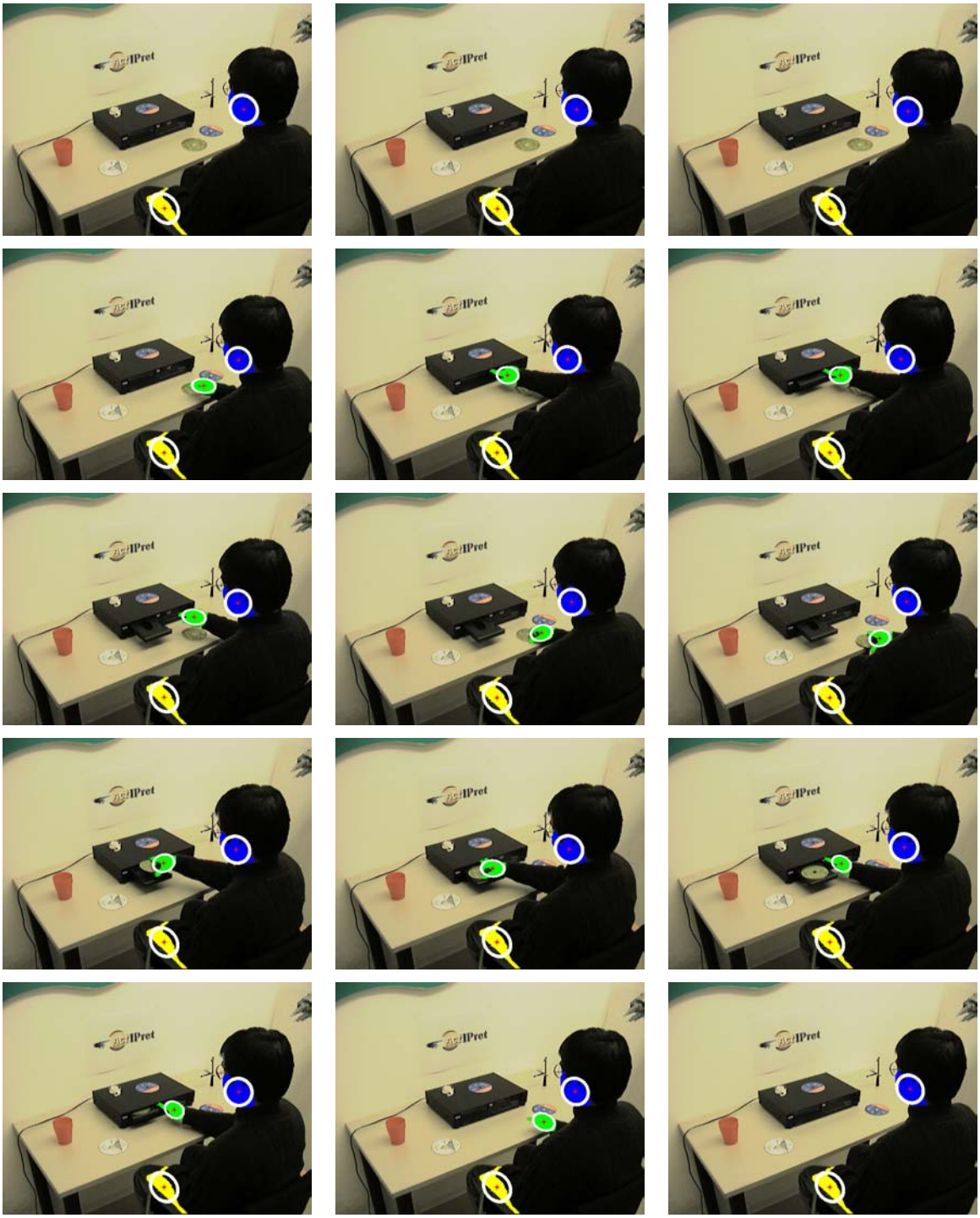


Figure 6: *Tracking results for the "Sequence17_head" sequence. Each hand hypothesis appears as a color blob superimposed on the original right image of the stereo-pair.*

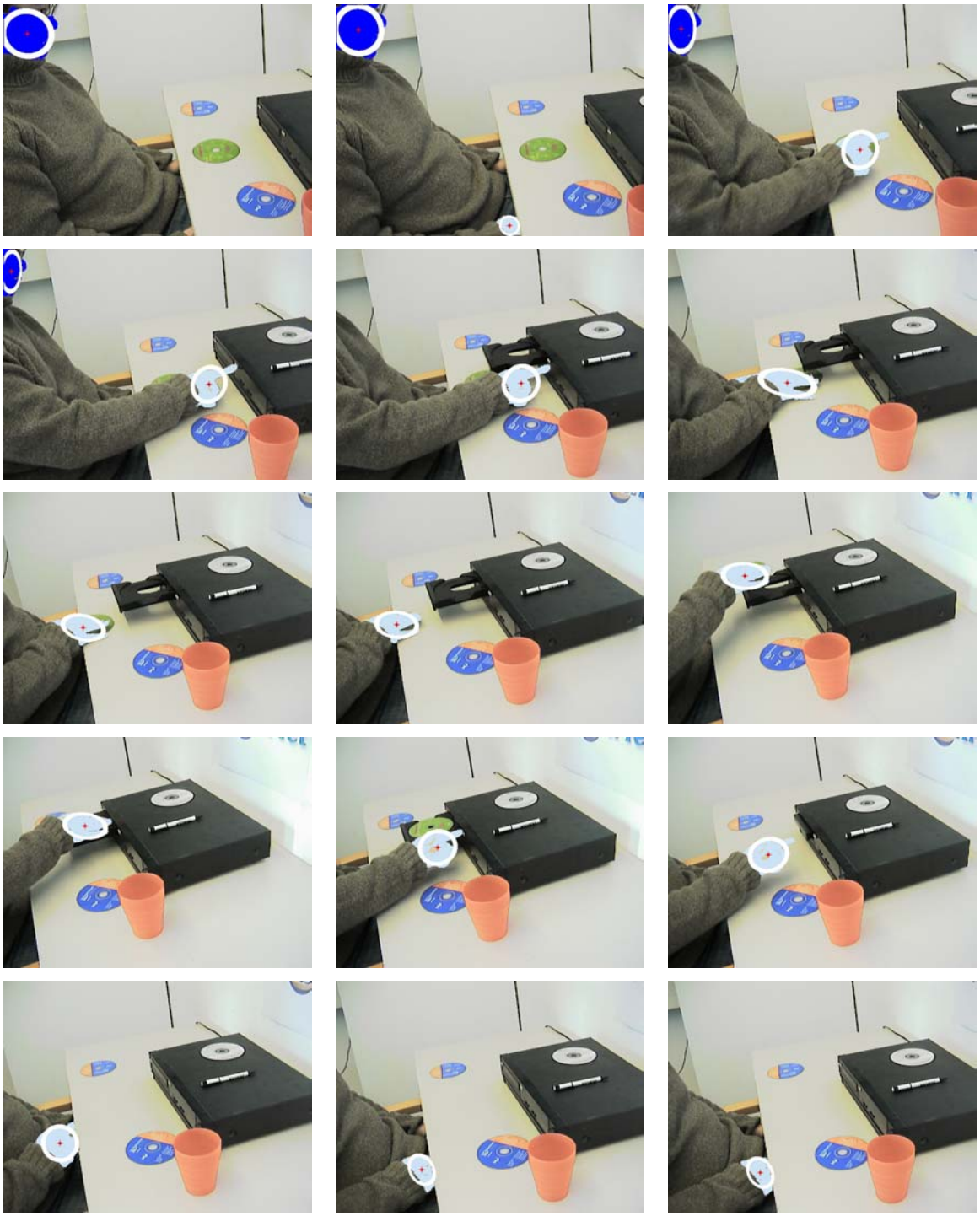


Figure 7: *Tracking results for the "Sequence18_arm" sequence. Each hand hypothesis appears as a color blob superimposed on the original right image of the stereo-pair.*

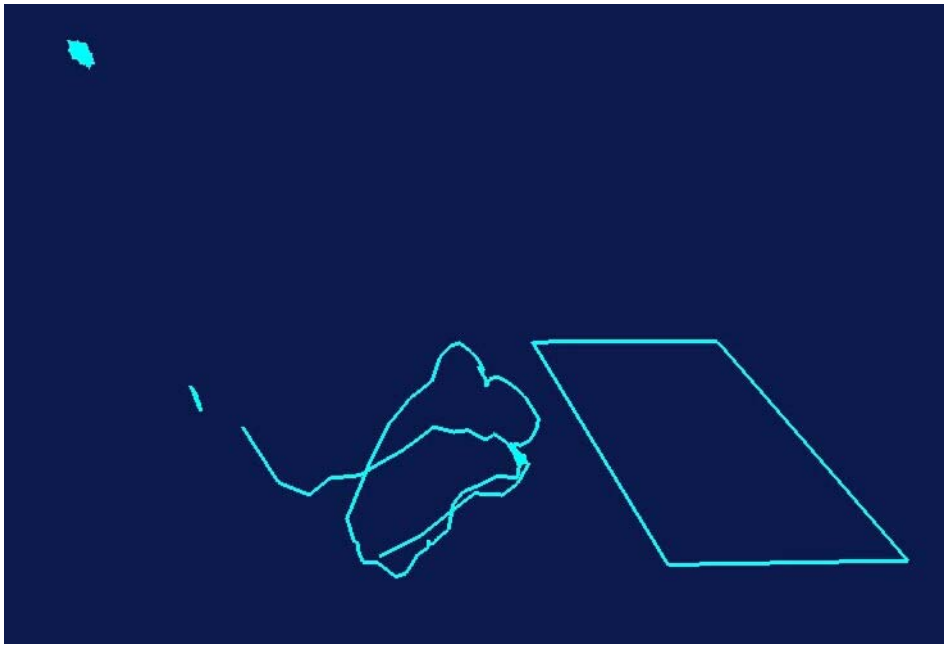


Figure 8: *The 3D trajectories of the hand hypotheses detected in the experiment of Fig. 6. The top-left and middle-left isolated spots correspond to the motion of the operator's head and of the arm of the armchair, respectively. The trajectory in the center of the image corresponds to the hand trajectory. The upper facet of the CD player has also been reconstructed, as a reference.*

Figure 9: *The 3D trajectories of the hand hypotheses detected in the experiment of Fig. 7. The straight line segment appearing in the right part of the figure corresponds to the tray of the CD player.*

8. APPENDIX A: Image-based camera tracking

We address the problem of camera matchmoving over a sequence of images acquired by a freely moving observer, a task that has a broad spectrum of useful applications in domains such as augmented reality and creation of special effects. Camera matchmoving is an application involving synthesis of real scenes and artificial objects, in which the goal is to insert computer-generated graphical 3D objects into live-action footage depicting unmodeled, arbitrary scenes. Graphical objects should be inserted in a way so that they appear to move as if they were a part of the real scene. Seamless, convincing insertion of graphical objects calls for accurate 3D camera motion tracking (i.e. pose estimation), stable enough over extended sequences so as to avoid the problems of jitter and drift in the location and appearance of objects with respect to the real scene. Additionally, the placement of the objects with respect to the real scene often requires the extraction of limited 3D geometry information; for instance, accurate 3D reconstruction of a few guiding control points is in many cases sufficient. Matchmoving finds several important applications in augmented reality as well as the creation of special effects in the post-production industry. To provide the versatility required by such applications, very demanding camera tracking requirements, both in terms of accuracy and speed, are imposed.

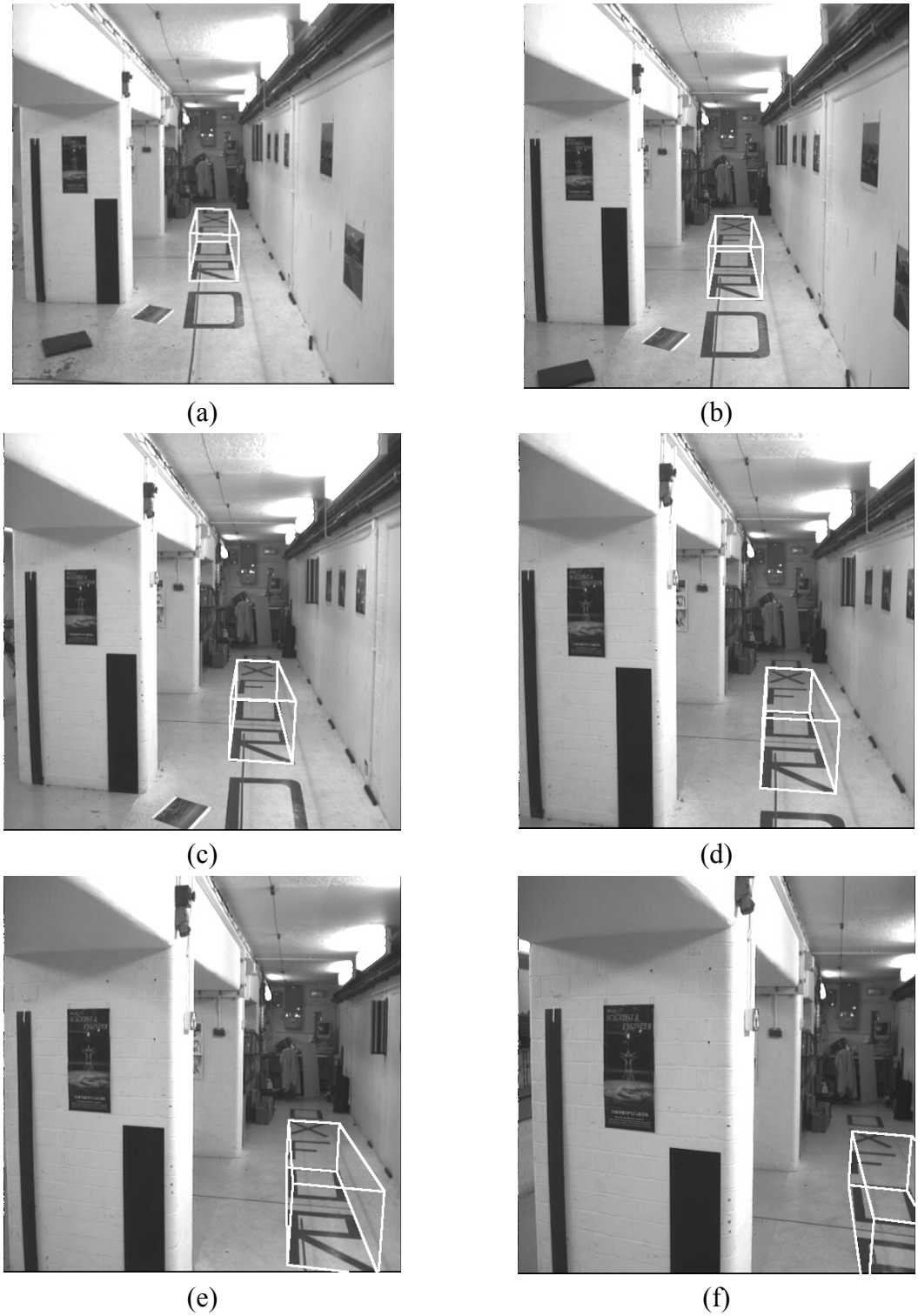
At the core of the proposed approach lies a novel, feature-based 3D plane tracking technique. Given a triplet of consecutive images and a plane homography between the first two of them, the plane tracker is capable of estimating the homography induced by the same plane between the second and third images, without requiring the plane to be segmented from the rest of the scene. In other words, the proposed method operates by “chaining” (i.e. propagating) across frames the image-to-image homographies due to some 3D plane. The chaining operation represents projective space using a ``plane + parallax'' decomposition, which permits the combination of constraints arising from all available point matches, regardless of whether they actually lie on the tracked 3D plane or not. Being straightforward to extend over long image sequences, plane tracking permits the estimation for each image pair in the sequence of the homographies induced by the 3D plane. Knowledge of such homographies allows the corresponding projection matrices encoding camera motion to be expressed in a common projective frame and therefore to be recovered directly. Additional knowledge of intrinsic camera calibration can be used to upgrade projective reconstruction to a Euclidean one. In addition to camera motion, the proposed method can recover a rough representation of 3D structure. Finally, it is shown that the tracked plane can be a virtual one, thus raising the implicit assumption regarding the presence of at least one 3D plane in the viewed scene.

Figures 10 and 11 provide representative results from a conducted experiment that involves augmenting an image sequence with an artificial 3D

object. The point features employed in this experiment have been extracted and matched automatically [8]. The experiment was performed on the well-known Oxford “basement” image sequence. This sequence consists of 11 frames acquired by a camera mounted on a mobile robot as it approached the scene while turning left. The proposed method was applied to the “basement” sequence and the camera 3D locations for each image along with a 3D point model of the scene were recovered. Using a few of the recovered 3D points, a wire-frame rectangular parallelepiped was inserted into the scene. Specifically, aiming to give the impression of an object lying on the floor, the parallelepiped was inserted so that its bottom face extends from the top of the second “O” to the bottom of the letter “R” in the word “OXFORD”. In a real application, a more complex 3D model would have been inserted into the scene with the aid of a 3D graphics package. Figs. 10(a)-(f) are snapshots of the sequence resulting by augmenting the original one. A top view of the VRML 3D model that was recovered, showing also the location of the inserted parallelepiped as well as the camera locations and trajectory is illustrated in Fig. 11. As it is clear from the results, the accuracy of camera matchmoving using the proposed method is satisfactory.

The average running time of the proposed matchmoving method for each image triplet was 102 ms on an Intel P4@1.8 GHz running Linux. This time does not include the time required for matching points among frames; around 350 points were matched between every pair of successive frames.

For more information regarding the developed method, the interested reader is referred to [4, 5].



(a) (b) (c) (d) (e) (f)

Figure 9: *Snapshots of the "basement" sequence (courtesy of the Oxford Visual Geometry Group), corresponding to frames 0, 2, 4, 6, 8 and 10, resulting after augmenting the original sequence with a rectangular parallelepiped drawn in red.*

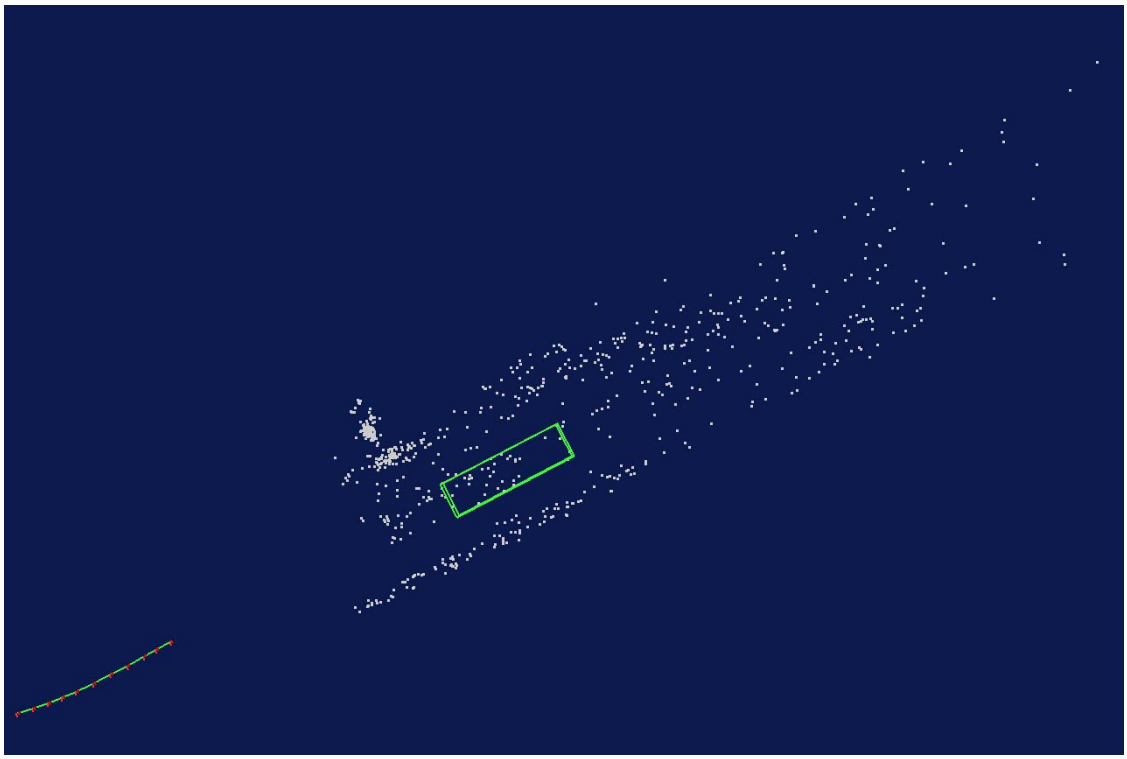


Figure 10: *Top view of the VRML 3D reconstruction for the scene of Fig. 9, showing also the inserted object and the 3D camera locations; see text for explanation.*

9. List of abbreviations

ABBREVIATION	FULL NAME
AHHS	Association of Hand Hypotheses between the images of a stereo-pair.
AHHT	Association of Hand Hypotheses in Time
CM	Centroid Matching
f.o.v	Field of view
HH	Hand Hypothesis
HT	Hand Tracker
SCD	Skin Color Detection
SOI	Space of Interest
TS	Temporal Smoothing
3DR	3D Reconstruction

10. References

[1] David A. Forsyth and Jean Ponce, "Computer Vision: A Modern Approach", Prentice Hall, 2003.

[2] S. McKenna, S. Gong, Y. Raja, "Modeling facial colour and identity with gaussian mixtures", Pattern Recognition, 31(12):1883--1892, 1998

[3] O. Faugeras, Q.-T. Luong, and T. Papadopoulo. “The Geometry of Multiple Images”. MIT Press, 2001.

[4] M.I.A. Lourakis, A.A. Argyros, “Automatic 3D Camera Matchmoving Using Markerless, Segmentation - Free Plane Tracking”, submitted to the Second International Symposium on Mixed and Augmented Reality, (ISMAR 2003), under review.

[5] M.I.A. Lourakis, A.A. Argyros, “Chaining Planar Homographies: Fast and Reliable 3D Plane Tracking”, submitted to the British Machine Vision Conference, (BMVC 2003), under review.

[6] H. Hirschmüller “Improvements in Real-Time Correlation-Based Stereo Vision”, In Proceedings of CVPR 01.

[7] L. Robert, C. Zeller, O. Faugeras, M. Hébert, “Applications of non-metric vision to some visually guided robotics tasks”, INRIA RR-2584, Robotvis, June 1995.

[8] J. Shi and C. Tomasi. Good Features to Track. In Proceedings of CVPR’94, pp. 593–600, 1994.