Probabilistic Structure from Camera Location using Straight Segments

J.M.M. Montiel [a] and L. Montano [a]

[a] *Departamento Informática e Ingeniería de Sistemas*
C.P.S. Universidad de Zaragoza
Maria de Luna 3 E-50015 Zaragoza (Spain)
{*josemari,montano*}*@posta.unizar.es*
http://www.cps.unizar.es/deps/DIIS/robot/

Abstract

A method to determine both the correspondences and the structure from the camera location is presented. Straight image segments are used as features. The location uncertainty is coded using a probabilistic model. The finite length of the image segments is considered, so a more restrictive equation (respect the usage of infinite straight lines) is used, and hence the spurious rejection is improved. The probabilistic modelling derives all the location uncertainty from image error and from camera location error. Thus, the uncertainty is fixed from a physical basis, simplifying the tuning for the matching thresholds. Furthermore, covariance matrices representing the reconstruction location error are also computed. Experimental results with real images for a trinocular system, and for a sequence of images are presented.

Key words: structure from camera location, straight segment, probabilistic methods, trinocular stereo, robust feature matching.

1 Introduction

The determination of correspondent features along an image sequence, together with the computation of the underlying scene structure is a classical problem in computer vision. This paper proposes a solution considering as input the camera location known up to some uncertainty. The feature used is the straight segment; i.e. this paper is devoted to solve the correspondences and the structure from the camera location, using straight segments as features. The two main contributions are the consideration of the finite length for the segment, and the use of a probabilistic model for matching and fusion. High rejection of spurious correspondences is achieved. Furthermore, an estimate

of the structure error is reported in the form of a covariance matrix for each reconstructed 3D segment. Experiments with a trinocular triplet and with a 15 images sequence are presented.

The proposal uses a predict-match-update loop which computes correspondences and structure sequentially, so structure and correspondences are available after each image processing; the precision is improved as new images are considered. The predict and update steps are computed by means of recursive least-squares (a Kalman filter with constant state). The match step is computed using the split-track data association for multiple target in clutter [3]. The next image segments are matched with the computed structure from previous images.

An image straight segment is defined by its infinite supporting line, location along this supporting line, and length. The supporting line is detected reliably. However, the length and the location along the supporting line are unreliable because of extraction defects and occlusions. Our contribution is a pairing constraint for straight segments that includes both supporting line pairing (collinearity condition) and location along the supporting line (overlapping condition). The overlapping is coded as the matching between the image segment midpoints. Midpoint matching is only approximate because they are not invariant under perspective projection, and because extreme points are detected unreliably. Because of that midpoint matching (overlapping) is considered with a lower weight than the collinearity. The proposed constraint considers both collinearity and some degree of overlapping so the segment finite length are considered. In any case, the proposed representation follows a general model for coding uncertain geometrical information, so the segments can be treated like any other geometrical feature.

Points and straight segments are commonly used as features in geometrical computer vision. Points impose more restrictive constraints than rectilinear features in problems involving structure and motion [20,21], however the matching for straight segments is easier because lines encodes information of wider image areas in a single feature; on the other hand, in artificial environments, straight image segments correspond with relevant 3D features (doors, corners, industrial parts boundaries). Several methods are available for straight segment detection, one hand methods that compute poligonal approximations to image contours, see [14] for a comparison, on the other hand methods that determine segments directly from image gradient [4].

The prediction-match-update scheme has been proposed for straight segment image matching [8]; the matching is computed for proximal images with unknown motion. [7] uses image tracking for matching and then the 3D structure segments are computed fusing the matched image segments, the camera location is considered known. Jezouin and Ayache [9] consider the computed

structure from the previous images for matching with the next image. They process image sequences considering points and lines separtely. This paper presents a similar approach, but we use the straight segment as a unique feature composed of its midpoint and its supporting line. In [12], a structure from camera location algorithm based on the same ideas is presented, however, the pairing constraint presented in this paper has been modified to consider the camera location independently. Experimental results have been extended to include ground true solution for stereo reconstruction and the processing of a sequence of images.

The use of probabilistic methods to recognize and fuse uncertain geometrical information is a classical technique in multisensor fusion [2,17]. Probabilistic methods profit from the well-established optimal estimation theory for fusion, and from the tracking and data association theory for matching [3]. In [6], Cox reviews data association algorithms to determine the correspondences in feature-based computer vision.

One of the experiments with real imaginery is a trinocular stereo system based on the proposed structure from camera location. Trinocular stereo systems represent a trade off between low spurious rate and matching complexity. Ayache proposed in [1] a trinocular stereo for straight segments based on the epipolar constraint. Shen and Paillou [15] proposed a trinocular stereo for straight segments using Hough transform. Our contribution to trinocular stereo systems is the ability to tune all the thresholds as a function of physical parameters such as camera position and orientation covariance, image error in pixels, and false negative probability for the statistical tests; this is due to the usage of probabilistic methods. Our system can be easily extended to consider more images, as shown in the experimental results (Sec.6.2).

The proposed system only uses geometrical information from CCD cameras. It can be easily extended to fuse information from other geometrical sensors modeled with probabilistic methods, e.g. laser range finders. Additional constraints such as parallelism, verticallity; or parametric information (color, average gray level, ...) can be used to improve matching. The applications cover robot navigation in indoor environments, dimensional control quality, object recognition and urban scene reconstruction from aerial images.

Next, Section 2 is devoted to present the geometrical model; the representation of the geometrical entities used are also presented. Section 3 introduces notation and presents the sequential processing for the images. Section 4 defines the pairing constraint between and image segment and its corresponding 3D segment. Next, Section 5 describes in detail the statistical tests used for matching. Finally, Section 6 presents the experimental results and Section 7 the conclusions. Detailed form for some complicated expressions are included as appendixes at the end of the paper.

2 Modelling Geometric Information

A probabilistic model, named SPmodel [17], has been selected to represent the uncertain geometrical information. It is a general model for multisensor fusion whose main qualities are: homogeneous representation for every feature irrespective of its number of d.o.f; and that the error is represented locally around the feature location estimate. The error is not represented additively, but as transformation composition. These qualities for multisensor fusion are also recognized as important for computer vision in [13].

Additionally, we have added a modification in the original SPmodel, so can it combine uncertain relations with deterministic relations. Sensorial information is uncertain due to measurement errors, however, some relations are known with probability 1, e.g. a projection ray is known to pass through the camera optical center. Next the modified SPmodel is presented.

The SPmodel is a probabilistic model that associates a reference G to locate each geometric element $\mathcal{G}$. The reference location is given by the transformation t_{WG} relative to a world reference W. To represent this transformation, a *location vector* $\mathbf{x}_{WG}$, composed of three Cartesian coordinates and three Roll-Pitch-Yaw angles is used:

$$\mathbf{x}_{WG} = (x, y, z, \psi, \theta, \phi)^T$$

$$t_{WG} = \mathrm{Trans}(x, y, z) \cdot \mathrm{Rot}(z, \phi) \cdot \mathrm{Rot}(y, \theta) \cdot \mathrm{Rot}(x, \psi) \qquad (1)$$

The estimation of the location of an element is denoted by $\hat{\mathbf{x}}_{WG}$, and the estimation error is represented *locally by a differential location vector* $\mathbf{d}_G$ relative to the reference attached to the element. Thus, the true location of the element is:

$$\mathbf{x}_{WG} = \hat{\mathbf{x}}_{WG} \oplus \mathbf{d}_G$$

where $\oplus$ represents the composition of location vectors (the inversion is represented with $\ominus$). Notice that the error is not added, but composed with the location estimate. The differential location error $\mathbf{d}_G$ is a dimension 6 normally distributed random vector. Although $\mathbf{d}_G$ has 6 components, the model forces some components to zero in two cases:

symmetries .- Symmetries are the set of transformations that preserve the element. The location vector $\mathbf{x}_{WG}$ represents the same element location irrespective of that $\mathbf{d}_G$ component value. For example, consider the reference S which locates a 3D segment (Fig. 1 (a)); rotations around the X direction

yield references that represent the same 3D segment. Theoretically those components could take any value; however, a zero value is forced.

deterministic components .- There are components of $\mathbf{x}_{WG}$ known with probability 1. Among all the equivalent references for the element, one whose deterministic component is null is selected; then, the corresponding component is forced to be zero. For example (Fig. 1 (b)) the reference, R, associated to a projection ray; R can be attached to the optical center, expressing its location with respect to the optical center frame C, $\mathbf{x}_{CR}$ (and hence $\mathbf{d}_G$) always has its X, Y, and Z components null.

To mathematically represent which components are null, $\mathbf{d}_G$ is expressed as:

$$\mathbf{d}_G = B_G^T \mathbf{p}_G$$

where $\mathbf{p}_G$, the *perturbation vector*, is a vector containing only non-null components of $\mathbf{d}_G$. B_G, the *self-binding matrix*, is a row selection matrix which selects the non-null components of $\mathbf{d}_G$.

Based on these ideas, the information about the location of geometric element $\mathcal{G}$ is represented by a quadruple, the element *uncertain location*:

$$\mathbf{L}_{WG} = [\hat{\mathbf{x}}_{WG}, \hat{\mathbf{p}}_G, C_G, B_G]$$

So, the random vector defining the element location is expressed as:

$$\mathbf{x}_{WG} = \hat{\mathbf{x}}_{WG} \oplus \mathbf{B}_G^T \mathbf{p}_G \tag{2}$$

$$\hat{\mathbf{p}}_G = E[\mathbf{p}_G]\,;\; C_G = Cov(\mathbf{p}_G)$$

where $\mathbf{p}_G$ is a normal random vector, whose mean is $\hat{\mathbf{p}}_G$ and whose covariance matrix is $\mathbf{C}_G$. When $\hat{\mathbf{p}}_G = 0$ we say that the estimation is centered.

There are geometrical elements whose location is input data for the problem: the camera, and the image segments. On the other hand, the 3D segment location is output of the algorithm and is computed from the input data; the 3D segment is the geometrical element used to represent the scene structure. Additionally, we define an intermediate geometrical element, the *2D segment*, to define the constraint which relates an image segment with a 3D segment. The 2D segment comprises the projection ray for the image segment midpoint, and the projection plane for the image segment supporting line. Next, the model of these four elements is presented in detail.

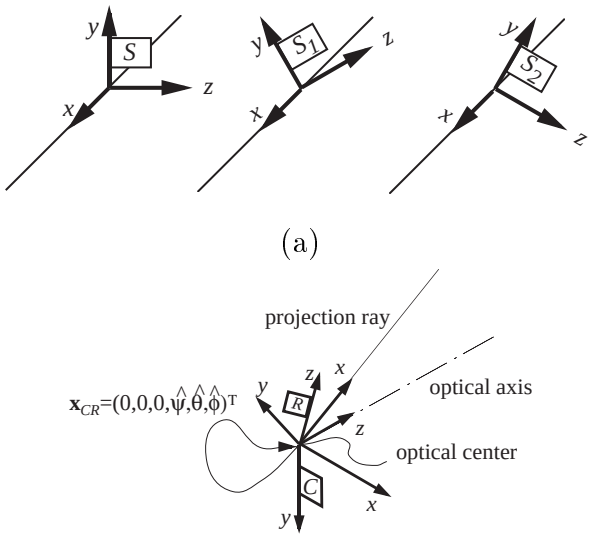


Fig. 1. (a) A 3D segment several equivalent associated references S. (b) A projecting ray R, located with respect to the optical center C.

2.1 Camera Uncertain Location

We use letter C to designate camera reference. We model the camera as a normalized one. The associated reference origin is attached to the camera optical center. The z axis is parallel to the optical axis, and pointing towards the scene. The x axis pointing to the right. The y axis is defined to form a direct reference (see Fig. 1(b)). Camera location has neither any symmetry nor any deterministic component in its differential location vector, so it has no null components (its self-binding matrix is the identity):

$$B_C = I$$

Camera location estimate $\hat{\mathbf{x}}_{WC}$ can be obtained from camera calibration or from another sensorial information. Covariance matrix C_C is input data for the problem and should represent the camera location estimate uncertainty.

2.2 Image Segment Uncertain Location

We use letter P to designate references attached to image segments. The associated reference (Fig. 2) is attached to the image segment midpoint; its y axis is normal to the supporting line and pointing to the "light" side of the segment; so it codes the gray level gradient of the segment; z axis is parallel to the camera z axis. The x axis is defined to form a direct reference.

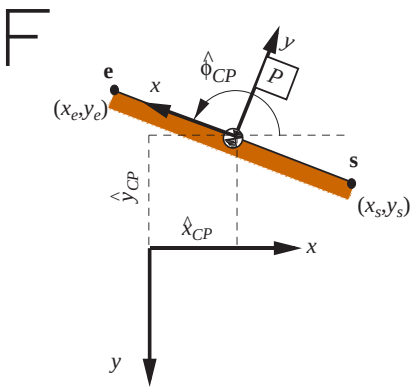


Fig. 2. Image segment in the normalized camera. Letter "F" is used to determine which image plane side we are referring to.

As an image segment belongs to the image plane, its z, ψ, and θ components are deterministic. So, its self binding matrix is:

$$B_P = \begin{pmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1 \end{pmatrix}$$

The image segment location centered estimate is defined, with respect to the camera location, from the extreme points coordinates in the normalized image as (see Fig. 2):

$$\hat{\mathbf{x}}_{CP} = \left(\hat{x}_{CP}, \hat{y}_{CP}, 1, 0, 0, \hat{\phi}_{CP}\right)^T \tag{3}$$

$$\hat{\phi}_{CP} = \operatorname{atan2}\left(y_e - y_s, x_e - x_s\right); \quad \hat{x}_{CP} = \tfrac{x_s + x_e}{2}; \quad \hat{y}_{CP} = \tfrac{y_s + y_e}{2}$$

The covariance assignment for image segment is one of the central points of this work. The covariance in the ϕ and y components are taken from the infinite supporting line. The standard deviation for the x component is defined as proportional to the segment length. According to the proportionallity constant, the allowed deviations of the midpoint along the segment supporting line can be fixed; the values are fixed to allow deviations up to 40%-80% the segment length. This covariance assignment mimics the one proposed in [22], however there it is used for 3D segments, while in this paper it is used for image segments.

Figure 3 sketches a comparison between the 95% acceptance regions for the origin of the reference that locates the image segment: considering it as an infinite line, or considering it with the proposed model. Modelled as a line, the region is unbounded along the line because every point can represent

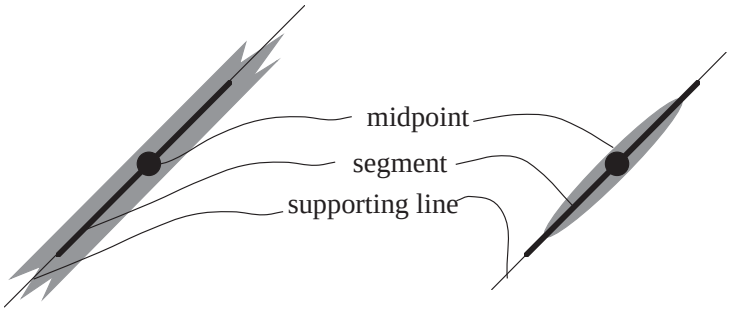


Fig. 3. 95% acceptance regions for the image segment reference origin. Left, when modeled as infinite line; right, when modeled as proposed in this paper.

the line; however, in our model, an ellipse along the segment represents the acceptance region.

It should be noticed how segment length is not considered as a geometrical parameter, but it is used to define the element covariance. In fact, the segment is only located by its midpoint and its orientation. Intuitively, the image segment has been modeled as "a point with orientation," and its standard deviation along the segment supporting line is set proportional to its length.

Next the quantitative expression for the covariance matrix is given:

$$C_P = N C'_P N^T \tag{4}$$

where C'_P is the covariance for the image segment in pixels; N is the Jacobian matrix for the transformation which converts from image segment in pixels to the image segment in the normalized camera. We have chosen that expression in order to deal with the pixel aspect ratio. Appendix B gives a detailed expression for N as function of the image segment location estimate (3), and the camera intrinsic parameters.

The form for C'_P is:

$$C'_P = \mathrm{diag}\left(\sigma_x^2, \sigma_y^2, \sigma_\phi^2\right)$$

- σ_x is set proportional to the image segment length, n (in pixels).

$$\sigma_x = \kappa n \tag{5}$$

  The experimental values for κ have been tuned in $[0.2, 0.4]$.
- σ_y^2 and σ_ϕ^2 are computed from the covariances of the image segment extreme points. Due to systematic errors some correlation between the extreme points location noise exists; the correlation effect is dealt splitting the extreme points covariance in two terms: σ_{cc}^2 completely correlated covariance ($0 - 2$px.), and σ_{nc}^2 non-correlated covariance ($0.25 - 0.5$px.). In

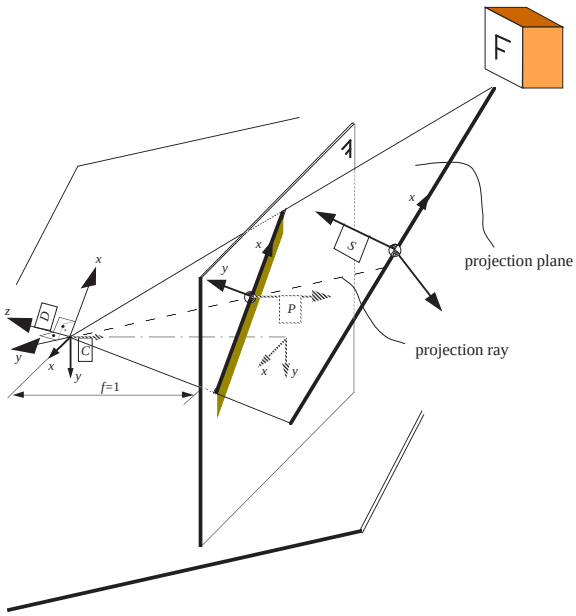


Fig. 4. The 2D segment (D) is an intermediate element used to relate the image segment (P) with the 3D segment (S); it includes both the projection ray for the midpoint and the projection plane for the supporting line. Letter "F" is used to determine which image plane side we are referring to.

[10] is detailed this expression and the tuning

$$\sigma_y^2 = \sigma_{cc}^2 + \frac{\sigma_{nc}^2}{2}, \quad \sigma_\phi^2 = \frac{2\sigma_{nc}}{n^2} \tag{6}$$

2.3 2D Segment

We use letter D to designate references attached to 2D segments. This geometrical element is used as an intermediate element to define the relation between an image segment and a 3D segment (Fig. 4). A 2D segment is composed of the projection elements of the corresponding image segment: the projection ray for the image segment midpoint, and the projection plane for the infinite supporting line. Its covariance is also directly derived from that of the image segment.

The associated reference is attached to the camera optical center; the optical center belongs to every projection element. Its $-y$ axis points towards the image segment midpoint. The z axis is normal to the supporting line projection plane. The x axis form a direct reference. An additional remark: the z direction is defined so that it also codes the image segment gray level gradient. As it is attached to the optical center, its general location vector of a 2D segment is

(with respect to the camera frame):

$$\hat{\mathbf{x}}_{CD} = \left(0, 0, 0, \hat{\psi}_{CD}, \hat{\theta}_{CD}, \hat{\phi}_{CD}\right)^T$$

See Appendix C for $\hat{\psi}_{CD}, \hat{\theta}_{CD}$ and $\hat{\phi}_{CD}$ expressions as function of image segment location (3).

As the translation components are deterministically null, the self-binding matrix only selects ψ, θ and ϕ components:

$$B_D = \begin{pmatrix} 0\ 0\ 0\ 1\ 0\ 0 \\ 0\ 0\ 0\ 0\ 1\ 0 \\ 0\ 0\ 0\ 0\ 0\ 1 \end{pmatrix}$$

The 2D segment covariance matrix is related with that of the image segment:

$$C_D = K_{DP} C_P K_{DP}^T$$

where C_P was presented in (4). A detailed expression for the K_{DP} matrix as function of the image segment location vector is given in Appendix C.

2.4 3D Segment

The letter S is used to designate references associated to 3D segments. The reference is attached (Fig. 1 (a)) to a segment point, which approximately corresponds to the segment midpoint. The reference x axis is aligned with the segment direction. In this work, the 3D segment location estimate is always computed from the integration of several 2D segments (corresponding to different points of view). The integration also yields as result the 5×5 covariance matrix C_S.

The only symmetry for this element is the rotation around its direction, so its self-binding matrix is:

$$B_S = \begin{pmatrix} 1\ 0\ 0\ 0\ 0\ 0 \\ 0\ 1\ 0\ 0\ 0\ 0 \\ 0\ 0\ 1\ 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 1\ 0 \\ 0\ 0\ 0\ 0\ 0\ 1 \end{pmatrix}$$

3 Sequential Processing for the Images

This section is devoted to the predict-match-update loop used to process the sequence. We consider the scene static, so the update step is a recursive weighted least-squares estimation. It is equivalent to consider a Kalman filter with constant state and without state noise. The matching step is based on the Split-Track [3] filter for matching.

3.1 Notation

This section states the notation used to define the matches, focusing in the indexes.

Cameras: $\{\mathbf{L}_{WC_k}\}$ $k = 1 \dots n$. Defines the camera k location with respect to the world frame, n stands for the total number of images.

3D Segments: $\left\{\mathbf{L}_{WS_i^k}\right\}$ $i = 1 \dots m_k$. Defines the location for segment i after processing k images. The total number of kept 3D segments, m_k, varies with k. m_k increases when several matches are possible, decreases when a pairing hypothesis marked as spurious.

Detected 2D Segments: $\left\{\mathbf{L}_{C_kD_l^k}\right\}$ $l = 1 \dots p_k$. Defines the location for 2D segment l, of image k with respect to the camera k. The total number of image segments in image k is p_k. This notation is intended to represent the 2D segment detected, irrespective of the correspondences.

Correspondent 2D segments: $\left\{\mathbf{L}_{C_kD_i^{(k)}}\right\}$ $i = 1 \dots m_k, \quad k = 1 \dots n$. Locates the 2D segment in image k which corresponds to 3D segment $\mathbf{L}_{WS_i^k}$. Notice the difference with respect to the previous notation.

3.2 Problem statement

Using the previous notation, the correspondence problem is stated as:

given: The 2D segments and the cameras location:

$$\left\{\mathbf{L}_{C_kD_l^k}\right\} \quad l = 1 \dots p_k, k = 1 \dots n$$
$$\{\mathbf{L}_{WC_k}\} \quad k = 1 \dots n$$

determine: The correspondences and the 3D segments location:

$$\left\{\mathbf{L}_{C_kD_i^{(k)}}\right\} \quad k = 1 \dots n, i = 1 \dots m_k$$
$$\left\{\mathbf{L}_{WS_i^k}\right\} \quad k = 1 \dots n, i = 1 \dots m_k$$

BEGIN

$\left\{\mathbf{L}_{WS_i^1}\right\}$ = `initial_guess_from_first_image` $\left(\mathbf{L}_{WC_1}, \left\{\mathbf{L}_{C_1 D_l^1}\right\}\right)$

FOR $k = 2$ **TO** n **DO**; every 3D segment from $k-1$ images

FOR $i = 1$ **TO** m_k **DO**; with respect to all 2D segments image k

$\left(\left\{\mathbf{L}_{WS_i^k}\right\}, \left\{\mathbf{L}_{C_k D_i^{(k)}}\right\}\right)$ =`matches_i_segment_wrp_k_image` $\left(\mathbf{L}_{WS_i^{k-1}}, \mathbf{L}_{WC_k}, \left\{\mathbf{L}_{C_k D_l^k}\right\}\right)$

END FOR

IF $\left(k \bmod n_{\text{uniq}}\right) == 0$ **THEN**

$\left(\left\{\mathbf{L}_{WS_i^k}\right\}, \left\{\mathbf{L}_{C_k D_i^{(k)}}\right\}\right)$ = `uniqueness_test` $\left(\left\{\mathbf{L}_{WS_i^k}\right\}, \left\{\mathbf{L}_{C_k D_i^{(k)}}\right\}\right)$

END IF

END FOR

END

Fig. 5. Correspondences and reconstruction algorithm.

3.3 Correspondence Computing

A sequential processing is proposed; every image except the first one is processed in the same way. Algorithm in Figure 5 presents the overall framework. Initially, a scene structure is computed from the first image; this structure is used only to compute the correspondences between the first and the second image; the algorithm is detailed in Sec. 5.3. Afterwards, for each image the correspondences between the 3D segments detected up to the previous iteration and the current image 2D segments are computed. Formally expressed:

given: $\left\{\mathbf{L}_{WS_i^{k-1}}\right\}, \left\{\mathbf{L}_{C_k D_l^k}\right\}, \{\mathbf{L}_{WC_k}\}$

determine: $\left\{\mathbf{L}_{C_k D_i^{(k)}}\right\}, \left\{\mathbf{L}_{WS_i^k}\right\}$

Figure 6 shows the processing performed for each 3D segment, $\mathbf{L}_{WS_i^{k-1}}$, with respect to every 2D segment $\left\{\mathbf{L}_{C_k D_l^k}\right\}$ in the k image:

1.- One Step innovation test. The innovation is computed using equation 8 in Sec. 4, considering the 3D segment location as the estimation, and the 2D segment and the camera location as the measurements. Afterwards the innovation test presented in Section 5.1 is applied. Three cases can happen:

a.- Only one 2D segment fulfills the test. The 2D segment is added to the list of observations for the 3D segment.

b.- More than one 2D segment fulfill the test. A new 3D segment is created for each additional pairing. The observation list up to $k-1$ observation is the same for all of them. For the k observation, each of the

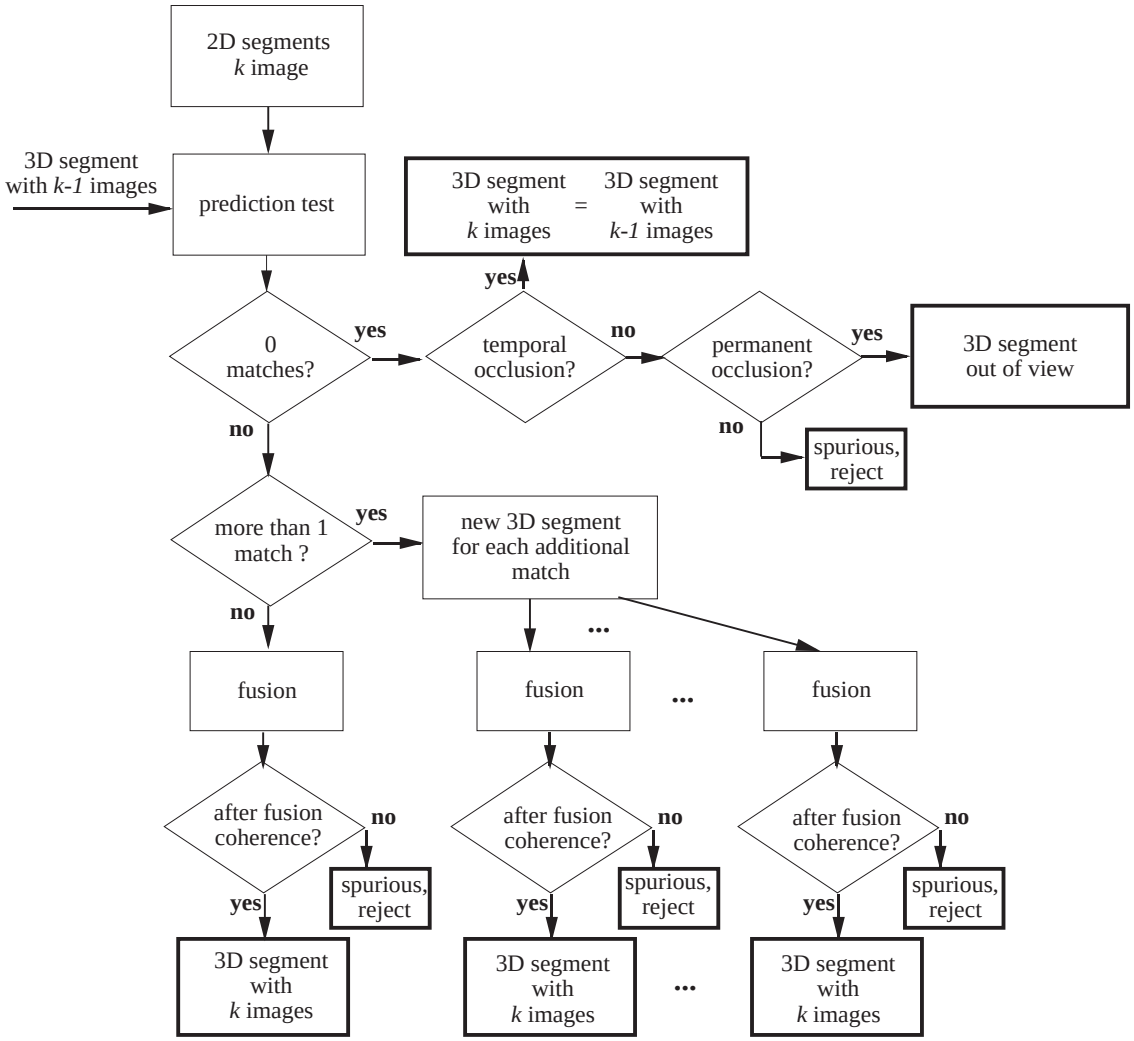


Fig. 6. Detailed algorithm for `correspondences_i_segment_respect_k_image`. i.e Correspondence computing for each 3D segment whit respect to all the 2D segments in image k.

accepted 2D segment is used.

c.- None of the 2D segments fulfills the test. Three cases are considered:

Spurious .- The 3D segment cannot be detected in the current image. In the previous images it was detected only once or twice. It is removed from $\left\{\mathbf{L}_{WS_i^k}\right\}$.

Temporally occluded .- The 3D segment was matched in 2 or more images previously, but not in the current one. It might be detected in the forthcoming images.

Permanently occluded .- The 3D segment was detected in more than 3 images previously; it has not been detected in the last 3 images. It is a reconstructed segment but it is not consider for matching any more.

2.- Fusion. All the 2D segments corresponding to a 3D segment, $\left\{\mathbf{L}_{C_k D_i^{(k)}}\right\}$,

are fused to determine the new location estimate for the 3D segment, $\mathbf{L}_{WS_i^k}$. It is computed using recursive weighted least-squares [3]; each observation is weighted according the inverse of its covariance. All the computations are equivalent to consider a Kalman filter whose state is constant and without state noise, the state would be the 3D segment location which is constant. As the estimated state is constant we cannot properly refer to it as a Kalman filter. In order to deal with non-linearities, and to consider not initial information about the 3D segment location, a extended and iterated information filter formulation is considered.

Using recursive least-squares, the more images are considered the smaller the fused segment covariance is; this indefinite covariance reduction is not realistic. Dealing with long sequences is important to "forget" old observations. It can be done using a sliding window; i.e. fusing only the most recent (6-10) images. We apply recursive fusion of the considered images because reduces the complexity and the memory overhead with respect to the batch fusion.

3.- Coherence after fusion test. The coherence after fusion among all the correspondent observations $\left\{\mathbf{L}_{C_k D_i^{(k)}}\right\}$, and the new location estimate, $\mathbf{L}_{WS_i^k}$, is tested (see Sec. 5.2).

As the previous processing is performed independently for each 3D segment, the same image segment can be paired with more than one 3D the segment. Several authors [1,6] state the ability of the uniqueness constraint to reduce the spuriousness; because of that every n_{uniq} images a uniqueness test is applied. Section 5.2 is devoted to present this test.

4 Measurement Equation

This section is devoted to formalize the pairing constraint between an image segment (P) and a 3D segment (S), see Figure 4. The camera detects the image segment (P); however, the proposed pairing constraint does not use the image segment (P) but the 2D segment (D). As mentioned in Section 2.3, the 2D segment uncertain location, $\mathbf{L}_{CD}$, is derived directly from that of the image segment, $\mathbf{L}_{CP}$.

The SPmodel method to define pairing constraints is used. The pairing constraint is defined in terms of the location vector $\mathbf{x}_{DS}$ of the 3D segment with respect to the 2D segment. Let

$$\mathbf{x}_{DS} = (x_{DS}, y_{DS}, z_{DS}, \psi_{DS}, \theta_{DS}, \phi_{DS})^T$$

The pairing constraint is an implicit equation that states which $\mathbf{x}_{DS}$ compo-

nents should be zero; these null components are:

- z_{DS}. Otherwise, the 3D segment would not belong the projection plane.
- θ_{DS}, rotation around y axis. It should be zero, otherwise, the 3D segment would not be in the projection plane.
- x_{DS}. Otherwise, the 3D segment midpoint would not belong to the image segment midpoint projection ray. Theoretically, *the segment midpoint is not invariant under perspective projection; however we consider it as invariant but this constraint, as shown later, is considered with a low weight.*

To sum up, the nullity of z_{DS} and θ_{DS} considers the collinearity in the image between the image segment and the 3D segment. The nullity of x_{DS} considers the overlapping condition; due to the covariance assignment for image segment along the segment direction, the overlapping constraint has normally lower weight than the collinearity ones. The low weight for this constraint is justified by the unreliable segment extreme points extraction, and by the approximate consideration that the midpoint is invariant under perspective projection. Experimental results confirm the validity of this assignment.

The above ideas are formalized mathematically as follows:

$$\mathbf{f}(\mathbf{x}_{DS}) = B_{DS}\mathbf{x}_{DS} = 0 \tag{7}$$

$$B_{DS} = \begin{pmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0 \end{pmatrix}$$

considering that $\mathbf{x}_{DS} = \ominus\mathbf{x}_{CD} \ominus \mathbf{x}_{WC} \oplus \mathbf{x}_{WS}$ and (2), equation (7) can be expressed as:

$$\mathbf{f}(\mathbf{x}_{DS}) = \mathbf{f}(\mathbf{p}_D, \mathbf{p}_C, \mathbf{p}_S) = B_{DS}\left(\ominus B_D^T\mathbf{p}_D \ominus \hat{\mathbf{x}}_{CD} \ominus B_C^T\mathbf{p}_C \ominus \hat{\mathbf{x}}_{WC} \oplus \hat{\mathbf{x}}_{WS} \oplus B_S^T\mathbf{p}_S\right) = 0 \tag{8}$$

So we have an implicit function which relates three perturbation vectors, $\mathbf{p}_D$, $\mathbf{p}_C$, and $\mathbf{p}_S$ corresponding to the 2D segment, the camera, and the 3D segment respectively, i.e. the normal random vectors involved in the problem. The camera and 2D the segment perturbation vectors act as measurement error, while the 3D segment perturbation is the vector whose estimation is improved.

Fusion and matching is based on a linear measurement equation. Thus, it is necessary to have a linearization for equation (7). Besides, we are using the equation to compute the correspondences and the structure as stated in Section 3. Let us consider the equation which relates the location estimate for

$\mathbf{L}_{WS_i^{(k-1)}}$ with a 2D segment detected in image k, $\mathbf{L}_{C_k D_i^{(k)}}$, and the camera k location, $\mathbf{L}_{WC_k}$. The linearized equation using the subindex notation presented in Section 3.1 is:

$$\mathbf{f}\left(\mathbf{p}_{D_i^{(k)}}, \mathbf{p}_{C_k}, \mathbf{p}_{S_i^{(k-1)}}\right) \approx \mathbf{f}_i^{(k)} + H_i^{(k)} \mathbf{p}_{S_i^{(k-1)}} + G_i^{(k)} \begin{pmatrix} \mathbf{p}_{D_i^{(k)}} \\ \mathbf{p}_{C_k} \end{pmatrix} = 0$$

expressed as the explicit linear measurement equation normally used in optimal estimation:

$$\mathbf{z}_i^{(k)} = H_i^{(k)} \mathbf{p}_{S_i^{(k-1)}} + G_i^{(k)} \mathbf{v}, \quad \mathbf{p}_{S_i^{(k-1)}} \sim N\left(0, C_{S_i^{(k-1)}}\right), \quad \mathbf{v} \sim N\left(0, R_i^{(k)}\right) \quad (9)$$

where:

$$\begin{aligned}
\mathbf{z}_i^{(k)} &= -\mathbf{f}_i^{(k)} = -B_{DS} \hat{\mathbf{x}}_{D_i^{(k)} S_i^{(k-1)}} \\
H_i^{(k)} &= \left. \frac{\partial \mathbf{f}}{\partial \mathbf{p}_S} \right|_{\left(\mathbf{p}_{D_i^{(k)}}=0, \mathbf{p}_{C_k}=0, \mathbf{p}_{S_i^{(k-1)}}=0\right)} \\
G_i^{(k)} &= \begin{pmatrix} \left. \frac{\partial \mathbf{f}}{\partial \mathbf{p}_D} \right|_{\left(\mathbf{p}_{D_i^{(k)}}=0, \mathbf{p}_{C_k}=0, \mathbf{p}_{S_i^{(k-1)}}=0\right)} & \left. \frac{\partial \mathbf{f}}{\partial \mathbf{q}_C} \right|_{\left(\mathbf{p}_{D_i^{(k)}}=0, \mathbf{p}_{C_k}=0, \mathbf{p}_{S_i^{(k-1)}}=0\right)} \end{pmatrix} \\
\mathbf{v} &= \begin{pmatrix} \mathbf{p}_{D_i^{(k)}} \\ \mathbf{p}_{C_k} \end{pmatrix} \\
R_i^{(k)} &= \begin{pmatrix} C_{D_i^{(k)}} & 0 \\ 0 & C_{C_k} \end{pmatrix}
\end{aligned}$$

$C_{D_i^{(k)}}$ is computed as shown in Sec.2.3, C_{C_k} computed from calibration, and $C_{S_i^{(k-1)}}$ comes from the previous iteration. Detailed expressions for the previous equations, as function of the location estimates for the camera, 2D segment, and 3D segment are available in Appendix D.

5 Matching

This section is devoted to present the detailed algorithms used for prediction, coherence after fusion, and uniqueness tests. The initial structure guess for the first image is also explained.

5.1 Prediction Test

This test verifies the compatibility between a 3D segment, $\mathbf{L}_{S_i^{(k-1)}}$ location estimate with $k-1$ images, and a 2D segment detected in image k, $\mathbf{L}_{C_k D_i^{(k)}}$. It is the classical [3] χ^2 test applied to equation (9).

$$\nu^T_{D_i^{(k)} S_i^{k-1}} C_\nu^{-1} \nu^T_{D_i^{(k)} S_i^{k-1}} \leq \chi^2_{3,\alpha} \tag{10}$$

$$\nu_{D_i^{(k)} S_i^{k-1}} = \mathbf{z}_i^{(k)} - H_i^{(k)} \mathbf{p}_{S_i^{(k)}} - G_i^{(k)} \mathbf{v} \tag{11}$$

$$C_\nu = H_i^{(k)} C_{S_i^{k-1}} H_i^{(k)^T} + G_i^{(k)} R_i^{(k)} G_i^{(k)^T} \tag{12}$$

where $\nu_{D_i^{(k)} S_i^{k-1}}$ is the innovation when considering the 2D segment $\mathbf{L}_{C_k D_i^{(k)}}$ as an observation of the 3D segment $\mathbf{L}_{S_i^{(k-1)}}$. $\chi^2_{3,\alpha}$ is the percentile $1-\alpha$ for a χ^2 distribution with 3 d.o.f (3 is the vector $\nu_{D_i^{(k)} S_i^{k-1}}$ dimension) and α is the false negative probability.

At this stage the gray level compatibility among all the correspondent image segments is also verified.

5.2 Coherence After Fusion and Uniqueness Test

This test is used to verify the coherence among all the observations, $\left\{\mathbf{L}_{C_k D_i^{(k)}}\right\}$, $k = 1 \ldots n_k$ corresponding to a 3D segment $\mathbf{L}_{W S_i^{n_k}}$; the 3D segment location has been fused from the n_k 2D segments. We use the batch test proposed by Tardós in [16], also proposed in [3]:

$$\sum_{k=1}^{n_k} \nu^T_{D_i^{(k)} S_i^{n_k}} \left(G_i^{*(k)} R_i^{(k)} G_i^{*(k)^T} \right)^{-1} \nu_{D_i^{(k)} S_i^{n_k}} \leq \chi^2_{3n_k-5,\alpha} \tag{13}$$

The d.o.f for χ^2 are $3n_k - 5$: the measurement equation dimension is 3, n_k measurements are considered, and 5 because the dimension for the estimated value (the 3D segment) is 5. The matrix $G_i^{*(k)}$ is a linearization matrix as $G_i^{(k)}$; however, the linearization is done using the location estimate after fusing all n_k images. $G_i^{(k)}$ is linearized using the location estimate with $k-1$ images. Notice also that $\nu_{D_i^{(k)} S_i^{n_k}}$ is the residual considering 3D segment after fusing n_k images with respect to each of the 2D segments used to fuse it; because of that, the n_k superindex is used.

Unlike the usual recursive test [3], we use the previous batch test. Theoretically both test are equivalent for a linear system; however our problem is non-linear.

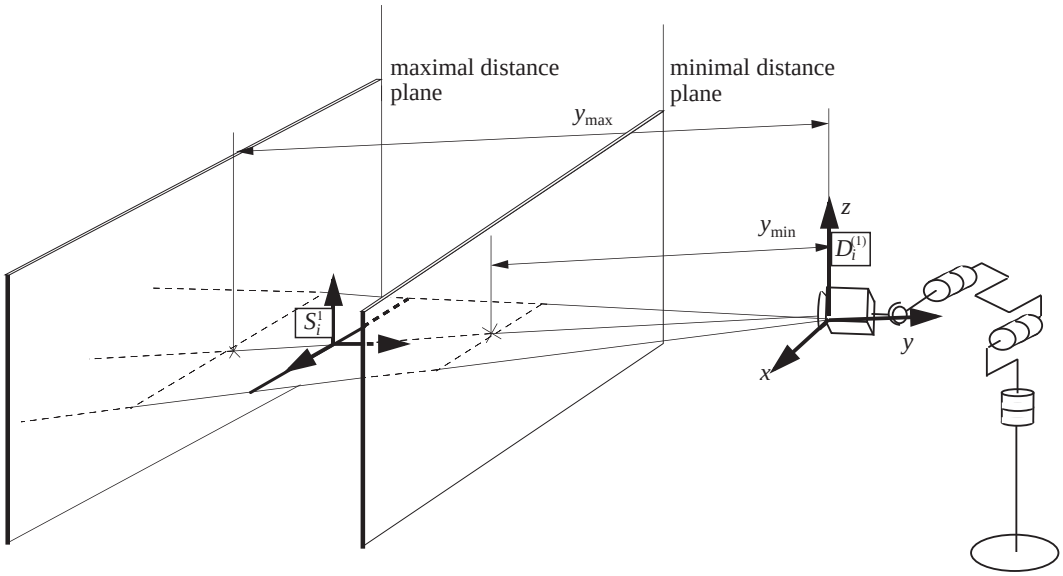


Fig. 7. Working space sketch. $D_i^{(1)}$ represents the 2D segment detected by the first camera, $S_i^{(1)}$ represents the initial guess for the corresponding 3D segment.

The batch test overperforms because all the linearizations are made around the last estimate.

Finally, the score of test (13) is used for testing uniqueness. When more than one 3D segment is in correspondence with the same 2D segment, then only the 3D whose score is the lowest is kept, the rest are considered as spurious. This test is applied after processing several images.

5.3 Initial Guess from First Image

The correspondences computing for the second image needs an initial guess for the scene structure; this guess is computed from the first image. Some assumptions are made about the working space where the 3D segments can be located. This region, defined by $y_{\min}$ and $y_{\max}$, is depicted in figure 7; e.g. in the experimental results, $y_{\min} = 500$mm. and $y_{\min} = 8000$mm.. The assumptions for the initial location are: its reference is parallel to the 2D segment in image 1, the 3D segment belongs to the projection plane, the 3D segment midpoint belongs to the image segment midpoint projection ray, and it is located in the middle of the working space. Mathematically expressed:

$$\hat{\mathbf{x}}_{WS_i^1} = \hat{\mathbf{x}}_{WD_i^{(1)}} \oplus \hat{\mathbf{x}}_{D_i^{(1)}S_i^1}$$

$$\hat{\mathbf{x}}_{D_i^{(1)}S_i^1} = \left(0,\, y_{DS},\, 0,\, 0,\, 0,\, 0,\right)^T, \quad y_{DS} = \frac{y_{\min} + y_{\max}}{2}$$

The covariance for the initial guess is defined as:

$$C_{S_i^1} = H_i^{(1)^T} G_i^{(1)} R_i^{(1)} G_i^{(1)^T} H_i^{(1)} + \text{diag}\left(0, \sigma_y^2, 0, 0, \sigma_\phi^2\right)$$
$$\sigma_y = \frac{|y_{\min} - y_{\max}|}{2 \times 1.96}, \quad \sigma_\phi = \frac{|\pi|}{2 \times 1.96}$$

where $H_i^{(1)}$ and $G_i^{(1)}$ come from the linearized measurement equation, considering that the 3D segment is in the proposed initial location. The covariances in y and ϕ components have been defined in such a way that the acceptance region for a 95% χ^2 test are $[y_{\min}, y_{\max}]$ and $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ respectively. The first matricial addend represents the covariance due to the observation with the camera. The second addend represents the covariance for depth and for orientation inside the projection plane, so that the acceptance region for a χ^2 test is contained in the working space depicted if Fig. 7.

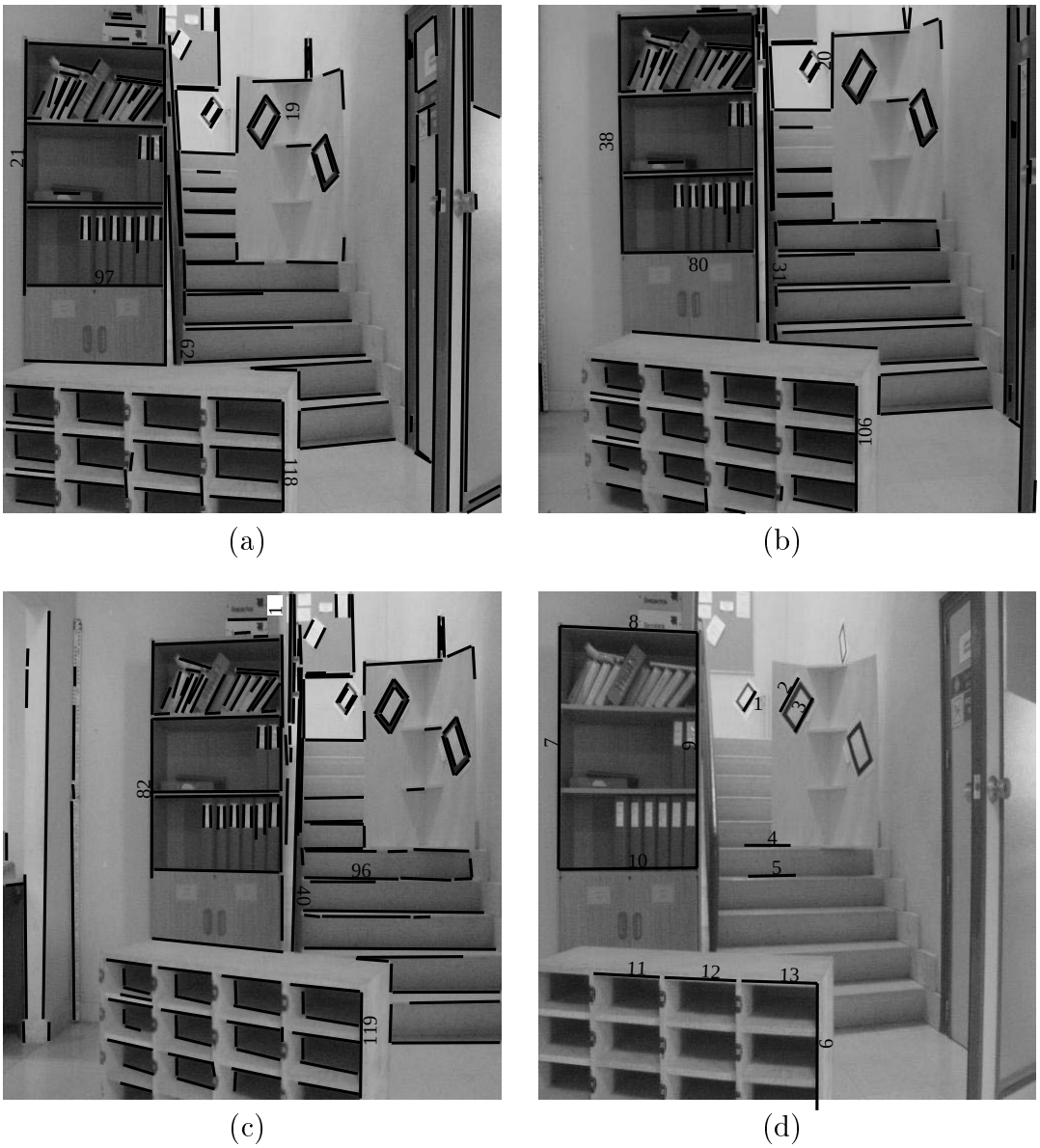
6 Experimental Results

This section presents two experiments with real indoor images. The first experiment is a trinocular stereo reconstruction based on the proposed ideas; we focus on two aspects: the matches, and the reconstruction quality compared with a ground true solution. The second experiment is the sequential processing of 15 images (5 trinocular images) of a robot moving along a corridor, to show the covariance evolution as the image sequence is processed.

6.1 Trinocular Reconstruction

The three images were taken with a trinocular rig, and were processed sequentially as proposed. The experiment considered as input the camera calibration parameters (Tsai camera model [19]). The sizes of the triangle formed by the optical centers were 400mm., 375mm., and 700mm. The gray level images were $512 \times 500 \times 8$, focal length was 6mm; the radial distortion was 0.003mm.^{-2} (maximal distortion $\approx$ 5Px.). Segments were extracted using Burns method [4], segments shorter than 15 Px. or with gray level gradient smaller than 20 gray levels per Px. were removed. The number of segments for the 1st, 2nd and 3rd images were 188, 172, and 182 respectively. Figure 8 (a), (b) and (c) shows the extracted segments in each image. Labels identify the segments in each image, so no relation exists among segments with the same label in different images. We will refer a segment in an figure, for example segment 13 in figure 8(a), as 13(8a); i.e., the 13th segment in figure 8a.

The ground true location for some 3D segments was computed using two theodolites. Figure 8 (d) shows the ground segments backprojected in camera



(a) (b)

(c) (d)

Fig. 8. Trinocular reconstruction input image segments for cameras 1 (a), 2 (b), and 3 (c). (d) shows the ground true solution projected on camera 1.

1. The reconstruction is compared with the ground true computing the Mahalanobis distance (using the observed segment covariance matrix) between the computed 3D location and the ground true solution.

The image segments in figures 8(a),8(b) and 8(c) were used to compute 3 reconstructions. Each of them with a different value for κ (see eq. (5)): 0.0002, 0.2 and 10. The 0.0002 value encoded the perfect correspondence between image segment midpoints, the 0.2 value represented a matching constraint that allowed deviations of the midpoint along the segment direction up to the image segment length; the 10 value encoded a situation which only considered the

κ	α	image 2		image 3		total		CPU
		pred.	aft. fusion	pred.	aft. fusion	match	spurious	sec.
0.2	0.75	470	215	491	159	114	1	2.2
0.0002	0.90	3057	253	108	100	90	3	4.2
10	0.5	978	810	13444	1436	76	> 10	21.5

Table 1
Summary for each reconstruction complexity.

κ	ground true segment label												
	1	2	3	4	5	6	7	8	9	10	11	12	13
0.2	1224.2	22.4	18.8	14.9	13.4	4.5	1.1	4.6	3.2	9.2	7.1	6.8	7.9
0.0002	9158.2	197.4	32.9	337.7	1916.5	-	-	5.1	2.6	8.1	430.3	158.1	177.0

	percentiles for χ^2 with 5 d.o.f.					
α	0.99	0.95	0.90	0.75	0.50	0.25
χ^2	15.1	11.1	9.24	6.63	4.35	2.67

Table 2
Mahalanobis distance between the ground true and the reconstructed segment. Labels identifying segments correspond with those in Fig.8(d). Percentiles for χ^2 with 5 d.o.f. are also shown

collinearity for matching. Camera covariance values (tuned experimentally) were:

$$C_{C_k} = \mathrm{diag}\left(\sigma_P^2, \sigma_P^2, \sigma_P^2, \sigma_O^2, \sigma_O^2, \sigma_O^2\right)$$
$$\sigma_P = 1.0\mathrm{mm}. \quad \sigma_O = 0.1^\circ$$

The assignment for the image segment covariances (see eq.6) were:

$$\sigma_{cc} = 2.0 \text{ Px}. \quad \sigma_{nc} = 1.0 \text{ Px}.$$

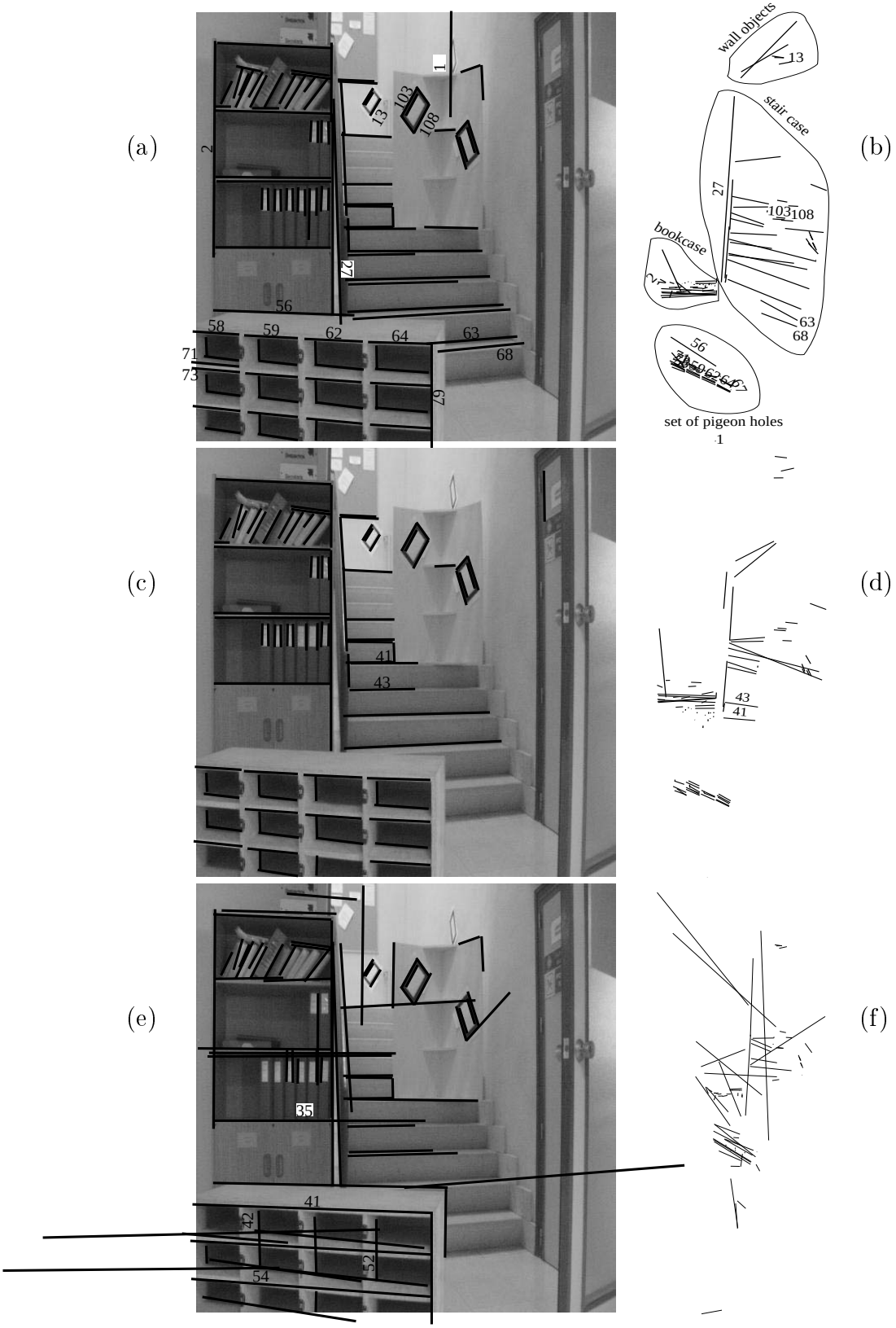


Fig. 9. (a), (c) and (e) camera 1 backprojection for reconstructions with $k = 0.2$, $k = 0.0002$, and $k = 10$ respectively. (b), (d) and (f) corresponding top views.

The reconstruction top view and its backprojection on camera 1 are shown, for each κ value, in Fig. 9. Table 1 columns summarize the tuning parameters and the number of hypotheses evolution along the sequential processing; there is one row per each κ value. α column shows the significance level for both the prediction (see Sec. 5.1) and after fusion (see Sec. 5.2) tests. Columns "pred." and "aft. fusion" shows hypotheses number after the corresponding tests. Column "total match" shows number of reconstructed segments, after the uniqness test. "Spurious" shows the spurious matches, checked by hand. CPU column shows the execution time on a Sparc 20 workstation (segment extraction not included).

Table 2 shows the Mahalanobis distance between reconstructed and ground true segments. There is one column per each ground true segment; labels correspond with labels in Figure 8(d). There is a row for the experiment with $\kappa = 0.2$ and another for the experiment with $\kappa = 0.0002$; the experiment with $\kappa = 10$ was not considered. Theoretically this distance should follow a χ^2 distribution with 5 d.o.f; a table for this distributions is included in 2. So, the magnitude of the error can be evaluated considering the corresponding percentile.

First we will focus on the $\kappa = 0.2$ value. It was a trade-off solution between problem complexity, number of spurious and solution accuracy. The matching is satisfactory. Segments that were collinear but did not overlap were not matched as unique segment, for example: segments in the set of pigeon holes: 58(9a), 59(9a), 62(9a) and 64(9a). Image segments that overlapped but with different length were also matched. e.g. the segment 2(9a) in the bookcase corresponded to image segments 21(8a), 38(8b), and 82(8c). Also 67(9a), in the set of pigeon holes, corresponded to image segments 118(8a), 106(8b), and 119(8c). The spurious was the reconstructed 1(9a) which corresponded with: 19(8a) in the right stair wall, 20(8b) in the pattern on the stairs, and 1(8c) in the left stair wall. The estimated reconstruction quality can be appreciated qualitatively in top view 9 (b). The set of pigeon holes, the staircase, and the bookcase can be easily identified. The reconstructed 27(9b) is seen in top view as too long, because the matched segments 62(8a), 31(8b) and 40(8c) had their extreme points poorly extracted because several near image segments were extracted as only one. The estimated reconstruction was coherent with the computed covariance because the corresponding residuals in table 2 were compatible with those of the χ^2 with 5 d.o.f. The biggest residuals appeared in ground true segments 1(8d), 2(8d), and 3(8d) which corresponded with 13(9a), 103(9a) and 108(9a) respectively. All of them were 3D segments far away from where the pattern for camera calibration were fixed; the pattern were stationed more or less where the set of pigeon holes appears in Fig. 9 (a).

Reconstruction with $\kappa = 0.0002$ detected fewer segments than with $\kappa = 0.2$, despite greater α, 0.9 instead of 0.75 (see table 1). For example, 71(9a), 73(9a),

67(9a) and 56(9a), in the set of pigeon holes, were not detected in 9 (c); the same for 2(9a) in the bookcase, and 63(9a), 68(9a) in the stairs. These matches were not detected because the midpoint pairing were not fulfilled. This showed that midpoint match was a too much strict condition. The quality of the reconstructed segments was worse, because the pairing between the midpoints was considered exact while it is only approximated. It can be seen in the top view reconstruction how segments 41(9d) and 43(9d) were located at the start of the stairs (near the bookcase), despite their location is in the middle of the stairs block (see Fig. 9 (c)). Residuals in table 2 were too much big to be compatible, so the computed covariance was not able to represent the location error.

Reconstruction with $\kappa = 10$ produced a lot of spurious matches; e.g.: segment 35(9e) matches 97(8a), 80(8b) and 96(8c). It were not able to match separately segments that were collinear but did not overlap: 41(9e), 54(9e) and 52(9e) included several collinear but not overlapping segments. Reconstruction in top view (Fig. 9 (f)) was very poor. It can be also seen in table 1 that a lot of hypotheses were dealt. It was because the matching constraint considering image segments with nearly infinite length were not very restrictive. This implied not only CPU overhead, but also memory overhead.

6.2 Sequence Processing

To show the reconstruction covariance evolution along a sequence of images, a second experiment was performed. A mobile robot moved along a corridor taking 5 trinocular frames, i.e. 15 images. Figure 10 shows images 3,7 and 15; figure 11 shows a corridor plane and the robot locations. These 15 images were processed sequentially as proposed in Section 3; the uniqness test was applied after processing images 3,7,11 and 15. In order consider only the recent observations for the 3D segment location, a 8 images sliding window is applied; i.e for the reconstruction and after fusion tests, only the 8 most recent images were utilized.

The camera location with respect to the robot was available from camera calibration. A precise robot location computed using 2 theodolites was available. The trinocular rig used to take the images was the same as the experiment in Section 6.1. Covariance values for image segments and camera locations were $\sigma_P = 3.0$mm., $\sigma_O = 0.2°$. All this information were taken from the "Cpsunizar Experiment" [5].

In Figure 10 the images 3rd, 7th and 15th are shown with the reconstructed scene backprojected; a view of the reconstructed scene is also shown. It can be noticed the quality of the reconstruction. Only segments 4,6 and 7 were

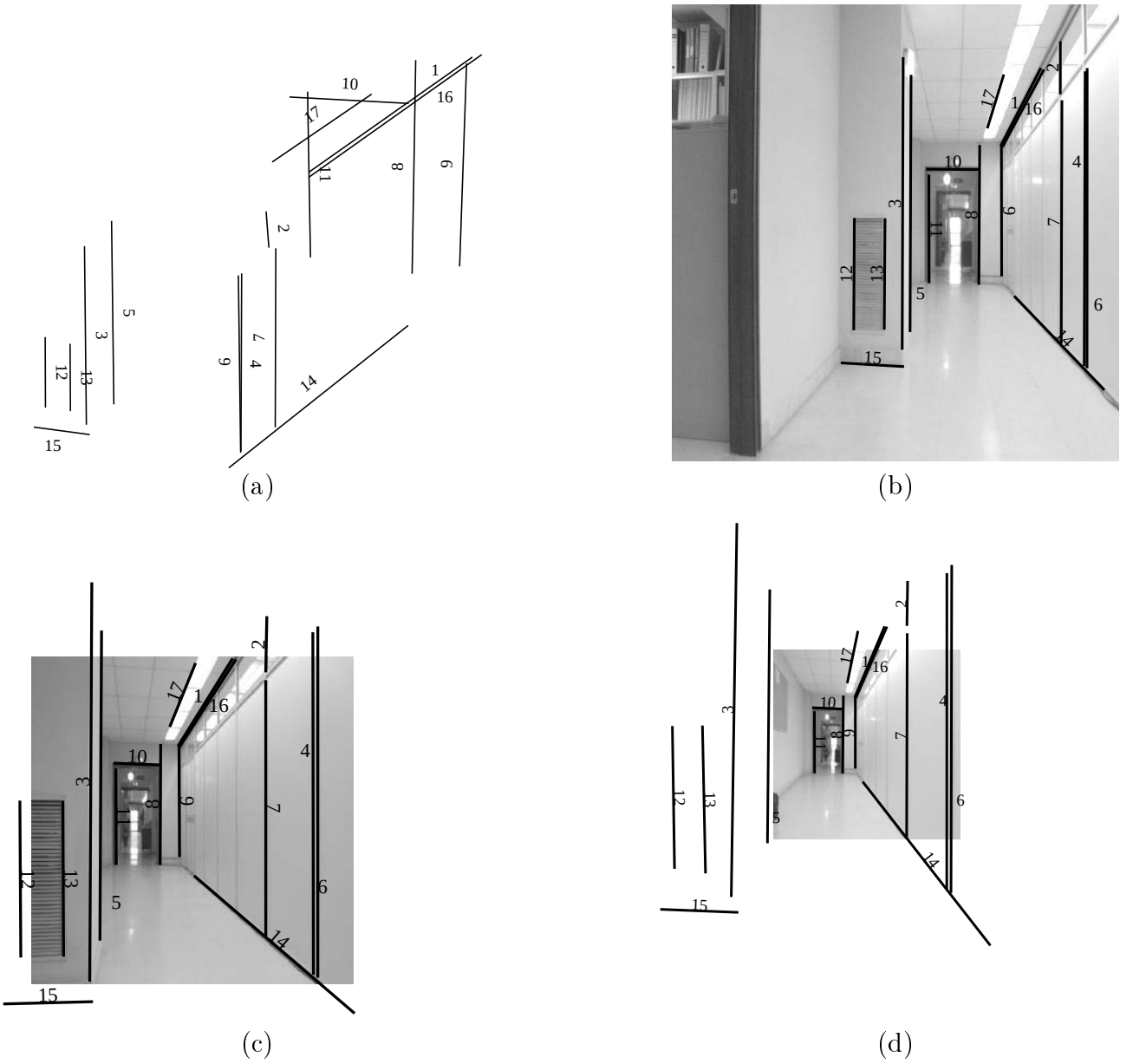


Fig. 10. The reconstructed 3D segments after processing 15 images(a). Gray level images 3(b),7(c) and 15(d); the reconstructed 3D scene from 15 images has been backprojected. Correspondent segments have the same label. Note that in figures (b),(c) and (d) all reconstructed segments are backprojected, so as the robot approched to the door only a part of reconstructed scene was sensed by the camera.

detected in the corridor right side. It was because in that area there were reflections and most of the other vertical segments were broken at the extraction stage. Figure 11 shows the evolution of the reconstructed segments covariance. In order to simplify the figure, only the covariance for the midpoint of the vertical segments is plotted after processing images 3,7 and 15. It is also shown the the corridor plane i.e the ground true solution.

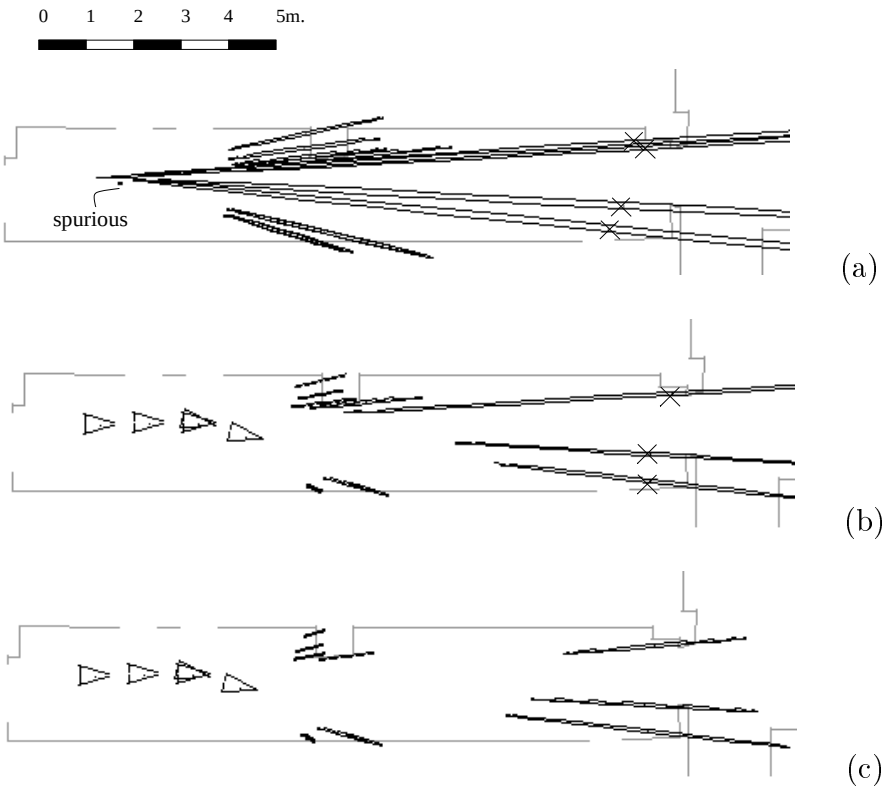


Fig. 11. Top view of the vertical reconstructed segments; ground true solution (dot lines) is also represented. The robot locations are shown as triangles. The covariance for the midpoint point of the vertical reconstructed segments is shown (solid lines). After processing 3 images (a), 7 images (b) and 15 images (c). Note that some ellipses has been cut in the right hand side of the figure, in such ellipses the center is marked with an "x"

From Figure 11 it can be seen how as the number of processed images was increased, the uncertainty in the fused features is reduced. Another effect is that the nearer a segment is to the camera the more precise is the corresponding observation and hence more precise is the computed reconstruction. So, as the robot advanced along the corridor, the farther segments were observed more precisely because two reasons: one because more observations were considered, and second because the new observations covariance were smaller. Due to the weighted least mean squares algorithm, lower covariance observations weight more in the final solution. The covariance for the farther segments 10(10a), 11(10a) 8(10a) is so big in depth because of the linearization. Actually, the area of possible locations for a segment far from the camera is asymetric: it is smaller between the segment and the camera than between the segment and infinity. However, the model proposed assigns a eliptical region symetric around the detected segment.

The robustness of the matching increases as more images are processed. It can be seen how the spurious match detected after processing 3 images was

removed when more images were considered; after processing 7 and 15 images, none a match was spurious.

7 Conclusions

A method to compute the correspondences and the structure for an image sequence from the camera location has been presented. The features used in the image are the straight segments. The location uncertainty is represented with a probabilistic model. The constraint for matching and fusion considers the collinearity between the segments supporting line and the overlapping of the segments. The overlapping is modeled considering the image segment midpoints as correspondent, but this correspondence has a lower weight than collinearity.

The use of a probabilistic model for the image segment produces matches with a low spurious rate, and sequentially increasing precision reconstruction. The fusion weights more the more precise observations. The parameter tuning, i.e covariance tuning, can be done from a physical basis such as camera calibration error and image error in the extracted features. The standard deviation along the image segment direction is assigned heuristically as proportional (κ is the proportionally constant) to the image segment length, in order to consider the segment finite length. Experiments showed how a too much small value for κ (0.0002) or a too much big value (10.0) yielded poor results, however an intermediate value as $\kappa = 0.2$ produce quality in the reconstruction and matches while keeps the complexity low. The input covariance are propagated and all the acceptance tests are computed as χ^2 tests, so the acceptance regions are defined as function of α, the false negative risk for the test. In our opinion, the good performance is because probabilistic models profit from the well established theories of optimal fusion and data association.

The proposed method has good performance with short and long sequences. The use of a probabilistic sequential processing, allows to combine the vision with another sensors. The presented trinocular system has a performance, both time and spurious rate, comparable with that of a classical trinocular systems as [1]. Besides, it can be easily extended to consider more images; the threshold tuning can be done on physical basis .

The 3D segment location from 2 images is an overconstrained problem if the finite segment length is considered. If the segment is considered as the corresponding supporting line, at least 3 images are necessary for the problem to be overconstrained. Some works consider that the finite length consideration is also important for the structure and motion problem [21,18] using straight segments. All this goes to show that it is important to consider the

segment length when dealing with segments. The straight segment representation proposed in this paper has been successfully used to compute structure and motion from image correspondences [11].

Implementation

The implementation of the stereo trinocular algorithm and the data used in the experiments is available for noncommercial use. The can be accessed at `http://www.cps.unizar.es/~josemari/StereoDemo.html`

Acknowledgement

This work has been partially supported by CICYT-TAP-94-0390 and CICYT-TAP97-0992-C02-01. Authors wish to express their gratitude to Dr. Zhengyou Zhang for the fruitfull discussions, for some of the images, and for the software for image and reconstruction visualization.

Software for simulation and visualization were developed by D. Berna Sanjuán, M.A. Cuartero Maestro, J.L. Martinez Delgado and C. Calvo.

References

[1] N. Ayache. *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*. MIT Press, Cambridge, MA, 1991.

[2] N. Ayache and O.D. Faugeras. Mantaining representations of the environment of a mobile robot. In R. Bolles and B. Roth, editors, *Robotics Research: The Fourth International Symposium*. MIT Press, 1987.

[3] Y. Bar-Shalom and T.E. Fortmann. *Tracking and Data Association*, volume 179 of *Mathematics in Science and Engineering*. Academic Press, INC., San Diego, 1988.

[4] J.B. Burns, A.R. Hanson, and E.M. Riseman. Extracting straight lines. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(4):425–455, 1986.

[5] J.A. Castellanos, J.M. Martínez, J. Neira, and J.D. Tardós. Experiments in multisensor mobile robot localization and map building. In *3rd IFAC Symposium on Intelligent Autonomous Vehicles*, Madrid, Spain, March 1998.

[6] J.C. Cox. A review of statistical data association techniques for motion correspondence. *Int. Journal of Computer Vision*, 10(1):53–66, 1993.

[7] J.L. Crowley, P. Stelmaszyk, T. Skordas, and P. Puget. Measurement and integration of 3-d structures by tracking edge lines. *International Journal of Computer Vision*, 8(1):29–59, 1992.

[8] R. Deriche and O. Faugeras. Tracking line segments. In *First European Conference on Computer Vision*, pages 259–268, Antibes, France, 1990.

[9] J.L. Jezouin and N. Ayache. 3d structure from a monocular sequence of images. In *3th. Int. Conf. on Computer Vision*, pages 441–445, Osaka, 1990.

[10] J.M Martínez and L. Montano. The effect of the image imperfections of a segment on its orientation uncertainty. In *7th. Int. Conf. on Advanced Robotics*, pages 156–162, Spain, September 1995.

[11] J.D. Martínez Montiel. *Visión Tridimensional Basada en Segmentos.* PhD thesis, Dpto. Informática e Ingeniería de Sistemas University of Zaragoza, Spain, September 1996.

[12] J.M Martínez Montiel, Z. Zhang, and L. Montano. Segment-based structure from an imprecisely located moving camera. In *IEEE Int. Symposium on Computer Vision*, pages 182–187, Florida, November 1995.

[13] X. Pennec and Thirion J.P. Validation of 3-d registration methods based on points and frames. In *V Int. Conf. on Computer Vision*, MIT. USA, 1995.

[14] P.L Rosin. Techniques for assessing polygonal approximations of curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):659–666, 1997.

[15] J. Shen and Paillou P. Trinocular stereovision by generalized hough transform. *Pattern Recognition*, 29(10):1661–1672, October 1996.

[16] J.D. Tardós. *Integración Multisensorial para Reconocimiento y Localización de Objetos en Robótica.* PhD thesis, Dpto. Inge. Eléctrica e Informática, University of Zaragoza, Spain, Febrero 1991.

[17] J.D. Tardós. Representing partial and uncertain sensorial information using the theory of symmetries. In *IEEE Int. Conf. on Robotics and Automation*, pages 1799–1804, Nice, France, May 1992.

[18] C.J. Taylor and D.J. Kriegman. Structure and motion from line segments in multiple images. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 17(11):1021–1032, November 1995.

[19] R.Y. Tsai. A versatile camera calibration technique for high accuaracy 3d machine vision metrology using Off-the-Shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, August 1987.

[20] J. Weng, T.S. Huang, and N. Ahuja. *Motion and Structure from Image Sequences.* Springer-Verlag, Heidelberg, 1993.

[21] G. Xu and Z. Zhang. *Epipolar Geometry in Stereo, Motion and Object Recognition: A Unified Approach.* Kluwer Academic Publishers, 1996.

[22] Z. Zhang and O. Faugeras. A 3d world model builder with a mobile robot. *Int. Journal of Robotics Research*, 11(4):269–284, August 1992.

Appendixes

A Transformations

The locations for the references are expressed as transformations. There are two mathematical representations for the transformation t_{WG}: a 6 component location vector $\mathbf{x}_{WG}$, and an homogeneous matrix $\mathbf{H}_{WG}$:

$$\mathbf{x}_{WG} = (x_{WG}, y_{WG}, z_{WG}, \psi_{WG}, \theta_{WG}, \phi_{WG})^T$$

$$\mathbf{H}_{WG} = \begin{pmatrix} n_{WG_x} & o_{WG_x} & a_{WG_x} & p_{WG_x} \\ n_{WG_y} & o_{WG_y} & a_{WG_y} & p_{WG_y} \\ n_{WG_z} & o_{WG_z} & a_{WG_z} & p_{WG_z} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Location vector form is well suited for theoretical discussion and for covariance assignment; however, the mathematical operations such as composition, inversion or derivation is better expressed using the homogeneous matrix. Conversion between them:

$$\mathbf{H}_{WG} = \begin{pmatrix} \mathrm{C}\phi_{WG}\mathrm{C}\theta_{WG} & \mathrm{C}\phi_{WG}\mathrm{S}\theta_{WG}\mathrm{S}\psi_{WG} - \mathrm{S}\phi_{WG}\mathrm{C}\psi_{WG} & \mathrm{C}\phi_{WG}\mathrm{S}\theta_{WG}\mathrm{C}\psi_{WG} + \mathrm{S}\phi_{WG}\mathrm{S}\psi_{WG} & x_{WG} \\ \mathrm{S}\phi_{WG}\mathrm{C}\theta_{WG} & \mathrm{S}\phi_{WG}\mathrm{S}\theta_{WG}\mathrm{S}\psi_{WG} + \mathrm{C}\phi_{WG}\mathrm{C}\psi_{WG} & \mathrm{S}\phi_{WG}\mathrm{S}\theta_{WG}\mathrm{C}\psi_{WG} - \mathrm{C}\phi_{WG}\mathrm{S}\psi_{WG} & y_{WG} \\ -\mathrm{S}\theta_{WG} & \mathrm{C}\theta_{WG}\mathrm{S}\psi_{WG} & \mathrm{C}\theta_{WG}\mathrm{C}\psi_{WG} & z_{WG} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{(A.1)}$$

where C and S stands for cos and sin respectively.

$$\mathbf{x}_{WG} \begin{pmatrix} x_{WG} \\ y_{WG} \\ z_{WG} \\ \psi_{WG} \\ \theta_{WG} \\ \phi_{WG} \end{pmatrix} = \begin{pmatrix} p_{WG_x} \\ p_{WG_y} \\ p_{WG_z} \\ \text{atan2}\left(o_{WG_z}, a_{WG_z}\right) \\ \text{atan2}\left(-n_{WG_z}, +\sqrt{n_{WG_x}^2 + n_{WG_y}^2}\right) \\ \text{atan2}\left(n_{WG_y}, n_{WG_x}\right) \end{pmatrix} \tag{A.2}$$

B Image Normalization Jacobian

$$N = \begin{pmatrix} \frac{\text{C}\hat{\phi}_{CP}\text{C}\hat{\phi}_{MP}}{\alpha_u} + \frac{\text{S}\hat{\phi}_{CP}\text{S}\hat{\phi}_{MP}}{\alpha_v} & \frac{-\text{C}\hat{\phi}_{CP}\text{S}\hat{\phi}_{MP}}{\alpha_u} + \frac{\text{S}\hat{\phi}_{CP}\text{C}\hat{\phi}_{MP}}{\alpha_v} & 0 \\ \frac{-\text{S}\hat{\phi}_{CP}\text{C}\hat{\phi}_{MP}}{\alpha_u} + \frac{\text{C}\hat{\phi}_{CP}\text{S}\hat{\phi}_{MP}}{\alpha_v} & \frac{\text{S}\hat{\phi}_{CP}\text{S}\hat{\phi}_{MP}}{\alpha_u} + \frac{\text{C}\hat{\phi}_{CP}\text{C}\hat{\phi}_{MP}}{\alpha_v} & 0 \\ 0 & 0 & \frac{\frac{\alpha_u}{\alpha_v}}{\left(\frac{\alpha_u}{\alpha_v}\right)^2 + \cos^2\phi'\left(1 - \left(\frac{\alpha_u}{\alpha_v}\right)^2\right)} \end{pmatrix}$$

where $\hat{\phi}_{CP}$ were defined in (3). $\frac{1}{\alpha_u}$ and $\frac{1}{\alpha_v}$ are the pixel sizes in the x and y directions, expressed in mm. S and C stands for the sin() and cos() functions. $\hat{\phi}_{MP}$ is defined as:

$$\hat{\phi}_{MP} = \text{atan2}\left(\alpha_v \sin\hat{\phi}_{CP}, \alpha_u \cos\hat{\phi}_{CP}\right)$$

C 2D Segment Definition

The 2D segment location with respect to the camera frame is expressed as:

$$\hat{\mathbf{x}}_{CD} = \left(0, 0, 0, \hat{\psi}_{CD}, \hat{\theta}_{CD}, \hat{\phi}_{CD}\right)^T$$

where:

$$\hat{\psi}_{CD} = \text{atan2}\left(o_z, a_z\right), \quad \hat{\theta}_{CD} = \text{atan2}\left(n'_z, \sqrt{n'^2_x + n'^2_y}\right), \quad \hat{\phi}_{CD} = \text{atan2}\left(n'_y, n'_x\right)$$

where:

$$
\begin{aligned}
n'_x &= \cos\hat{\phi}_{CP} + \hat{y}_{CP}^2 \cos\hat{\phi}_{CP} - \hat{x}_{CP}\hat{y}_{CP}\sin\hat{\phi}_{CP} \\
n'_y &= \sin\hat{\phi}_{CP} + \hat{x}_{CP}^2 \sin\hat{\phi}_{CP} - \hat{x}_{CP}\hat{y}_{CP}\cos\hat{\phi}_{CP} \\
n'_z &= \hat{y}_{CP}\sin\hat{\phi}_{CP} + \hat{x}_{CP}\cos\hat{\phi}_{CP} \\
o_z &= \frac{-1}{\sqrt{1+\hat{x}_{CP}^2+\hat{y}_{CP}^2}} \\
a_z &= \frac{\hat{x}_{CP}\sin\hat{\phi}_{CP} - \hat{y}_{CP}\cos\hat{\phi}_{CP}}{\sqrt{1+\left(\hat{x}_{CP}\sin\hat{\phi}_{CP} - \hat{y}_{CP}\cos\hat{\phi}_{CP}\right)^2}}
\end{aligned}
\tag{C.1}
\tag{C.2}
$$

and the corresponding covariance is defined as function of K_{DP}:

$$
K_{DP} = \begin{pmatrix}
0 & -\frac{\sin\hat{\psi}_{DP}}{\hat{y}_{DP}} & 0 \\
0 & -\frac{\sin\hat{\phi}_{DP}\sin\hat{\psi}_{DP}}{\hat{y}_{DP}\cos\hat{\phi}_{DP}} & \frac{\sin\hat{\psi}_{DP}}{\cos\hat{\phi}_{DP}} \\
\frac{\cos\hat{\phi}_{DP}}{\hat{y}_{DP}} & -\frac{\sin\hat{\phi}_{DP}\cos\hat{\psi}_{DP}}{\hat{y}_{DP}} & 0
\end{pmatrix}
\tag{C.3}
$$

where:

$$
\hat{y}_{DP} = -\sqrt{\hat{x}_{CP}^2 + \hat{y}_{CP}^2 + 1}, \quad \hat{\psi}_{DP} = \operatorname{atan2}\left(1, a'_z\right), \quad \hat{\phi}_{DP} = \operatorname{atan2}\left(o_x, n_x\right)
$$

and where:

$$
\begin{aligned}
a'_z &= \left(\hat{x}_{CP}\sin\hat{\phi}_{CP} - \hat{y}_{CP}\cos\phi_{CP}\right) \\
n_x &= \frac{1}{\|\mathbf{n}'_D\|}\left(1+\left(\hat{x}_{CP}\sin\hat{\phi}_{CP} - \hat{y}_{CP}\cos\hat{\phi}_{CP}\right)^2\right) \\
o_y &= \frac{-1}{\|\mathbf{o}'_D\|}\left(\hat{y}_{CP}\sin\hat{\phi}_{CP} + \hat{x}_{CP}\cos\hat{\phi}_{CP}\right) \\
\|\mathbf{o}'_D\| &= \sqrt{1+\hat{x}_{CP}^2+\hat{y}_{CP}^2} \\
\|\mathbf{n}'_D\| &= \sqrt{1+\hat{x}_{CP}^2+\hat{y}_{CP}^2+\hat{x}_{CP}^2\hat{y}_{CP}^2+(\hat{y}_{CP}^4+\hat{y}_{CP}^2)\cos^2\hat{\phi}_{CP}+} \\
&\overline{+(\hat{x}_{CP}^4+\hat{x}_{CP}^2)\sin^2\hat{\phi}_{CP} - (\hat{x}_{CP}^3\hat{y}_{CP}+\hat{y}_{CP}^3\hat{x}_{CP}+\hat{x}_{CP}\hat{y}_{CP})\,2\sin\hat{\phi}_{CP}\cos\hat{\phi}_{CP}}
\end{aligned}
$$

The values of $\hat{x}_{CP}$, $\hat{y}_{CP}$, and $\hat{\phi}_{CP}$ are taken form the image segment location with respect to the camera frame (3).

D Measurement equation

The detailed expression for the matrices and vectors used in the linearizations are:

$$\mathbf{f} = \begin{pmatrix} \hat{x}_{DS} \\ \hat{z}_{DS} \\ \text{atan2}\left(-n_{DSz}, \sqrt{n_{DSx}^2 + n_{DSy}^2}\right) \end{pmatrix} \tag{D.1}$$

$$G = \begin{pmatrix} 0 & -\hat{z}_{DS} & \hat{y}_{DS} & n_{DSx} & o_{DSx} & a_{DSx} & 0 & 0 \\ -\hat{y}_{DS} & \hat{x}_{DS} & 0 & n_{DSz} & o_{DSz} & a_{DSz} & 0 & 0 \\ \sin\hat{\phi}_{DS} & -\cos\hat{\phi}_{DS} & 0 & 0 & 0 & 0 & \cos\hat{\psi}_{DS} & -\sin\hat{\psi}_{DS} \end{pmatrix}$$

$$H = \begin{pmatrix} -n_{CD_x} & -n_{CD_y} & -n_{CD_z} & \begin{array}{c} n_{CD_y}\hat{z}_{CS} - \\ +n_{CD_z}\hat{y}_{CS} \end{array} & \begin{array}{c} -n_{CD_x}\hat{z}_{CS} + \\ +n_{CD_z}\hat{x}_{CS} \end{array} & \begin{array}{c} n_{CD_x}\hat{y}_{CS} - \\ +n_{CD_y}\hat{x}_{CS} \end{array} \\ -a_{CD_x} & -a_{CD_y} & -a_{CD_z} & \begin{array}{c} a_{CD_y}\hat{z}_{CS} - \\ +a_{CD_z}\hat{y}_{CS} \end{array} & \begin{array}{c} -a_{CD_x}\hat{z}_{CS} + \\ +a_{CD_z}\hat{x}_{CS} \end{array} & \begin{array}{c} a_{CD_x}\hat{y}_{CS} - \\ +a_{CD_y}\hat{x}_{CS} \end{array} \\ 0 & 0 & 0 & \begin{array}{c} -o_{CS_x}\cos\hat{\psi}_{DS} + \\ +a_{CS_x}\sin\hat{\psi}_{DS} \end{array} & \begin{array}{c} -o_{CS_y}\cos\hat{\psi}_{DS} + \\ +a_{CS_y}\sin\hat{\psi}_{DS} \end{array} & \begin{array}{c} -o_{CS_z}\cos\hat{\psi}_{DS} + \\ +a_{CS_z}\sin\hat{\psi}_{DS} \end{array} \end{pmatrix} \tag{D.2}$$

Previous expressions are given as functions of the homogeneous matrices $\mathbf{H}_{CD}$, $\mathbf{H}_{DS}$ and $\mathbf{H}_{CS}$. These matrices can be computed directly from the location estimated for the 2D segment, $\mathbf{H}_{CD}$, the camera $\mathbf{H}_{WC}$, and the 3D segment location $\mathbf{H}_{WS}$.