Dynamic Foveal 3D Sensing Using Affine Models

Diane Lingrand, Thierry Viéville

INRIA
INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Dynamic Foveal 3D Sensing Using Affine Models

Diane Lingrand and Thierry Viéville

N° 2687

Octobre 1995

PROGRAMME 4

ISSN 0249-6399

SOPHIA ANTIPOLIS

Dynamic Foveal 3D Sensing Using Affine Models

Diane Lingrand and Thierry Viéville

Programme 4 — Robotique, image et vision
Projet RobotVis

Rapport de recherche n° 2687 — Octobre 1995 — 47 pages

Abstract:

This study is aimed at developing a method of analysis of the 3D structure of a scene considering a monocular image sequence, with an uncalibrated camera -as for an active visual system- and using a continuous model of motion.

Surprisingly perhaps, this problem has not been studied much in literature except [25], but only preliminarly, and without any reference to active vision. This difficulty might have its source in the intrinsic complexity of the underlying equations, which yields a heavy implementation and are thus a-priori not robust. Moreover important developments of analytic equations are not possible as it is the case for calibrated systems [24, 5, 2], because of the algebraic complexity of the equations.

In order to overcome this difficulty, we have attempted to develop a simplified parameterization of the problem in the case of two or more views, considering a scene with a set of stationary objects and applying an orthographic model of the projection. In this case, fusion along the image sequence is trivial.

Thanks to the integration of active visual perception, we demonstrate that it is always possible to generate a displacement so that the previous model is valid, and we can then very easily reconstruct the observed scene. In the case where the motion constraints are approximately verified, we can show that the model is still approximately valid close to the retina.

At an experimental level, we report a small implementation taking an image sequence as input, which allows us to compute the retinal motion fields and calculate the reconstruction up to a particular affine transform of the scene.

Key-words: Structure and Motion, Active Vision, Affine Models

(Résumé : tsvp)

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

Perception 3D dynamique au niveau de la fovéa
Utilisation de Modèles Affines

Résumé :

On se propose d'étudier et de mettre au point une méthode d'analyse active de la structure tridimensionelle d'une scène au sein d'une séquence monoculaire d'images dans le cas d'un système non calibré et d'un modèle continu du mouvement.

Paradoxalement peut-être, ce problème a été peu traité dans la littérature, hormis dans [25], mais ceci de manière relativement préliminaire. Dans tous les cas, on n'y fait pas usage de stratégie de vision active. Une revue complète a déjà été établie dans [21]. Cette difficulté vient de la forte complexité des équations qui impose une mise en oeuvre lourde et donc a priori peu robuste et interdit des développements importants de formes analytiques comme c'est le cas pour les systèmes calibrés [24, 5, 2].

Afin de contourner cette difficulté, nous avons cherché à mettre en place une paramétrisation simplifiée du problème dans le cas de deux ou plusieurs vues, considérant une scène contenant un ensemble d'objets fixes et mettant en oeuvre une projection orthographique sur la rétine. Dans ce cas, la fusion entre deux vues est immédiate.

Grâce à la mise en place de stratégies actives de perception, nous démontrons qu'il est toujours possible d'effectuer un mouvement tel que le précédent modèle reste valable, et reconstruisons ainsi très simplement la scène observée. Dans le cas où le mouvement est approximatif, i.e. ne vérifie que partiellement les contraintes voulues, nous démontrons que le modèle reste approximativement valide au voisinage de la fovéa.

Au niveau expérimental une petite maquette est proposée qui prend en entrée une séquence d'image et y calcule les champs de mouvements définis précédemment et la reconstruction à une tranformation affine de l'espace près.

Mots-clé : Structure et Mouvement, Vision Active, Modèles Affines

Contents

1 Introduction

Let us consider the problem of reconstruction a 3D visual scene from a monocular image sequence with a uncalibrated camera and using a continuous model of motion.

Surprisingly perhaps, this problem has not been studied much in literature except [25], but preliminarly, and without any reference to active vision. This difficulty might have its source in the intrinsic complexity of the underlying equations which yields a heavy implementation and are thus a-priori not robust. Moreover important developments of analytic equations are not possible at it is the case for calibrated systems [24, 5, 2].

One work to overcome this difficulty,and considering the important volume of images to process, it is obvious that the task must be performed using active visual strategies as already used in [24] but using some information about the calibration of the system.

Another way to overcome this difficulty, is to develop a simplified parameterization of the problem in the case of two or more views, for instance applying an orthographic model of the projection.

In fact, orthographic projections have been often used in the past. Koenderink [1] used a three stages algorithm, with a small field of view and smooth transformation to obtain the spatial configuration up to an affine transform and then, with metric information, up to a relief transformation, and, at last, with a third view, to settle the calibration.

More recently, in the calibrated case, some methods combine affine approximation in an iterative algorithm [6, 15].

First, Christy and Radu [6], present an algorithm which converges to a unique set of 3D-Euclidean shapes and motion parameters that is consistent with a perspective model. They use two kinds of affine approximation of the perspective : weak perspective (zero-order approximation : $1/(1+\epsilon) \approx 1$) and para perspective (first-order approximation : $1/(1+\epsilon) \approx 1-\epsilon$). As expected, the precision associated with these two algorithms converge to the same value as the distance between the object center and the camera center of projection divided by the object size increases. Furthermore, Boufama etal [15] have compared the use of scaled orthography and perspective when computing an invariant shape from motion. The scaled orthographic projection gives better results for a small field of view, comparable results for a medium field of view and worse results for a large field of view. However, as the weak perspective assumption simplify the computation, but degrades the quality of the reconstruction when we have a large field of view, they combine the two approaches by using a weak perspective reconstruction as an initialization for the perspective iterative algorithm.

The parameterization of motion in the uncalibrated case has been studied in the discrete case [26] or in the continuous case [25] by several authors [11], [18] and [14]. In [26], the three kinds of geometry are studied : Euclidean, affine and projective with points or lines correspondences in a monocular image sequence.

In [25], small displacements are studied and approximate the equations of the discrete case using a first order expansion while they reconstruct the structure and motion of the scene, up to a projectivity. This approach has lead to promising results and we would like to follow this track here.

Presentation of the paper. We want to reconstruct a scene from a video sequence or as output by an active visual system, using an orthographic model of camera, and approximating the displacement by a first order expansion between two frames. We verify in which conditions an affine model can be used for 3D reconstruction.

We propose an affine model of the retinal displacements to determine the depth, by integration of the normals of the surface, considered locally planar patches. Then, we implement the method on a synthetic scene to validate the approximations, and use several classes of motions to validate the model and explore its limits. We also will describe how to use this simplified formalism with a suitable active vision strategy.

Notations: We note vectors in bold letters and matrices in capital letters. The duals of vectors are represented as the transpose of a vector and scalars in italic. The notation $\mathbf{x} \wedge \mathbf{y} = \tilde{\mathbf{x}}\,\mathbf{y}$ corresponds to the cross-product, the dot-product being written as $\mathbf{x}^T \cdot \mathbf{y}$. $\tilde{\mathbf{x}}$ is a 3×3 antisymmetric matrix. The identity matrix is written I. Geometric objects such as points, lines, planes are written with capital letters in 3D, and small letters in 2D. We represent the components of a matrix or a vector using upper subscripts from 0 to 2, e.g. : $\mathbf{x} = (x^0, x^1, x^2)^T$.

2 Using a simplified model of the retinal projection.

2.1 Retinal motion in a small foveal window

It has been known for a long time (see for instance [24] for a recent discussion in the field of active vision) that, as in nature, it is convenient to maintain the observed object close to the estimated principal point of the retina (foveation) since :

1. The usual calibration model relies on the pin-hole model which is valid only close to the optical axis (a biological equivalent is the fovea) [12, 4].

2. The image distortion remains negligible [3] and it is a valid approximation to assume, for the camera calibration model, that the pixel geometry is stable [24, 9].

3. A moving object with unexpected motion is easily maintained in the visual field if it stays around the center of the retina. Moreover the motion-field structure is much simpler in a neighborhood of the principal point [10].

4. The induced disparity remains minimal when zooming, if we are close to the principal point [16].

Moreover, close to the optical axis and considering remote objects it is also known (see for instance [19] for a recent discussion) that an orthographic model of the camera is a valid approximation even if reconstruction issues are addressed.

Let us review this point here.

If we consider a point $M = (X, Y, Z)$ in the scene, taken in a frame of reference attached to the retina, and its projection $m = (u, v')$, the classical model of projection [10, 22] is :

$$\left\{ \begin{array}{rcl} u & = & u_0 + \alpha_u\, X/(Z_0 + z) + \gamma\, Y/(Z_0 + z) \\ v' & = & v'_0 + \alpha_v\, Y/(Z_0 + z) \end{array} \right. \tag{1}$$

in which :

- the depth Z has be written
$$Z = Z_0 + z \tag{2}$$
considering any "average" depth of the remote observed object Z_0 and assuming $z << Z_0$,

- the position of the principal point is (u_0, v_0),

- the term γ describing the fact a pixel appears as non-orthogonal can be taken as 0 [24, 9], for most cameras,
- the ratio $k_{uv} = \frac{\alpha_v}{\alpha_u}$ of the two horizontal and vertical scale factors α_u and α_v is fixed and known in advance [24, 9] so that we can consider $v' = k_{uv}\, v$, $v'_0 = k_{uv}\, v_0$ and write :

$$v = v_0 + \alpha_u\, Y/(Z_0 + z) \tag{3}$$

We obtain, after some algebra, and using the notation $f = \alpha_u/Z_0$:

$$\left\{ \begin{array}{rcl} u & = & u_0 + f\, X + \epsilon_u \\ v & = & v_0 + f\, Y + \epsilon_v \end{array} \right. \tag{4}$$

with :

$$\left\{ \begin{array}{rcccl} \epsilon_u & = & -f\, X \left[\frac{z}{Z_0+z}\right] & = & (u_0 - u) \left[\frac{z}{Z_0} + o(\frac{z}{Z_0})\right] \simeq 0 \\ \epsilon_v & = & -f\, Y \left[\frac{z}{Z_0+z}\right] & = & (v_0 - v) \left[\frac{z}{Z_0} + o(\frac{z}{Z_0})\right] \simeq 0 \end{array} \right. \tag{5}$$

as a very simple model of the retina projection, in the uncalibrated case.

It is thus clear that, if we are

- either close to the optical axis ($u \simeq u_0$ and $v \simeq v_0$)
- far from the remote object $z << Z_0$,

we can use this orthographic projection.

We will follow this track here, since we accept to consider only small foveal windows in an image.

2.2 Considering small rigid motions of local planar patches

In order to analyze the 3D structure of the scene, we assume that the observed surface is locally planar and undergoes a relative rigid motion with respect to the retina.

Moreover, the plane equation can be written :

$$Z = N_x\, X + N_y\, Y + Z_0 \tag{6}$$

This equation is well defined if the plane does not project as a line, but in such a degenerated case, the plane would not be visible and we thus can avoid taking this situation into account.

The local orientation of the surface is defined by the normal of the local planar patch, i.e. :

$$\mathbf{n} = k \left(\begin{array}{c} N_x \\ N_y \\ -1 \end{array} \right) \tag{7}$$

where we can take for instance $k = \frac{1}{\sqrt{1+N_x^2+N_y^2}}$ to have $||\mathbf{n}|| = 1$. In fact, only the direction of $\mathbf{n}$ is meaning-full so that k can have any value.

A step further, we consider that we analyze an image sequence for which the rigid motion, parameterized by a rotation matrix R and a translation vector $\mathbf{t}$, can be considered as small

between two consecutive frames and the rotation matrix R can be approximated by a first order expansion as follows [10] :

$$R = I + \tilde{\mathbf{w}} + \epsilon_R \quad \epsilon_R = \frac{\tilde{\mathbf{w}}^2}{2} + o(||\mathbf{w}||^2) \tag{8}$$

The displacement equations that we have to consider are thus $dM = \mathbf{t} + \mathbf{w} \wedge M + o(||\mathbf{w}||)$ or in a more explicit form :

$$\left\{ \begin{array}{rcl} dX & = & t_x + w_y Z - w_z Y \\ dY & = & t_y + w_z X - w_x Z \\ dZ & = & t_z + w_x Y - w_y X \end{array} \right. \tag{9}$$

2.3 An affine model of the retinal displacements

Combining equations (4) with $\epsilon_u = \epsilon_v = 0$, with (6) and (9) we very easily obtain the following *2D-affine* model of the retinal displacement between two frames :

$$\left\{ \begin{array}{rcl} du & = & C_u + a\,u + b\,v \\ dv & = & C_v + c\,u + d\,v \end{array} \right. \tag{10}$$

with :

$$\left\{ \begin{array}{rcl} C_u & = & f\,t_x + f\,w_y\,Z_0 - w_y\,N_x\,u_0 + (w_z - w_y\,N_y)\,v_0 \\ C_v & = & f\,t_y - f\,w_x\,Z_0 + (w_x\,N_x - w_z)\,u_0 + w_x\,N_y\,v_0 \\ a & = & w_y\,N_x \\ b & = & w_y\,N_y - w_z \\ c & = & w_z - w_x\,N_x \\ d & = & -w_x\,N_y \end{array} \right. \tag{11}$$

Considering (a, b, c, d), we can partially recover the rotation $\mathbf{w}$ and the local surface orientation (N_x, N_y) *even if the camera is not calibrated and the motion is unknown.* Performing some algebra, we observe that the components of the rotation are related by two quadratic equations :

$$\left\{ \begin{array}{rcl} w_x\,b + w_y\,d + w_x\,w_z & = & 0 \\ w_y\,c + w_x\,a - w_y\,w_z & = & 0 \end{array} \right. \tag{12}$$

and further, the components of the rotation and the surface normal by two quadratic constraints :

$$a\,w_y^2 + d\,w_x^2 + w_x\,w_y\,(b + c) = 0 \tag{13}$$

$$a\,N_y^2 + d\,N_x^2 - N_x\,N_y\,(b + c) = 0 \tag{14}$$

while the system of equations is, in fact, of degree four and can be solved up to a scalar constant λ (resp. μ): it is clear that if w_x and w_y (resp. N_x and N_y) are solutions of (13) (resp. (14)), $\lambda\,w_x$ and $\lambda\,w_y$ (resp. $\mu\,N_x$ and $\mu\,N_y$) are also solutions of (13) (resp. (14)).

Since (w_x, w_y, w_z) is constant in an image, we first calculate (w_x, w_y, w_z).

Let us look for a solution of the following form:

$$\left\{ \begin{array}{rcl} w_x & = & \lambda \cos(\alpha) \\ w_y & = & \lambda \sin(\alpha) \end{array} \right. \tag{15}$$

In order to analyze these equations, we introduce r, s and t:

$$\left\{ \begin{array}{rcl} b+c & = & 2\,r\,\cos(s) \\ a-d & = & 2\,r\,\sin(s) \\ a+d & = & 2\,t \end{array} \right. \tag{16}$$

Then,

$$\left\{ \begin{array}{rcl} r & = & \frac{1}{2}\sqrt{(b+c)^2+(a-d)^2} \\ s & = & \arctan(\frac{a-d}{b+c}) \\ t & = & \frac{a+d}{2} \end{array} \right. \tag{17}$$

We can simplify, with some trigonometry, equation (13), and obtain :

$$r\,\sin(2\,\alpha+s)+t=0$$

where we have two real solutions if $|\frac{t}{r}| \leq 1$:

$$\left\{ \begin{array}{rcl} \alpha_1 & = & -\frac{\arcsin(\frac{t}{r})+s}{2} \\ \alpha_2 & = & -\frac{\pi-\arcsin(\frac{t}{r})+s}{2} \end{array} \right. \tag{18}$$

else complex conjugated solutions.

We have two other solutions, $\alpha_1+\pi$ and $\alpha_2+\pi$ but they are included in the factor λ. The validity of the solution is :

$$|\frac{t}{r}| \leq 1 \Leftrightarrow \frac{t^2}{r^2} \leq 1 \Leftrightarrow (b+c)^2-4\,a\,d \geq 0 \tag{19}$$

which is already the case when the model is valid since

$$(w_x\,N_x+w_y\,N_y)^2=(b+c)^2-4\,a\,d$$

Thus, we obtain w_z from equation (12), by minimization of :

$$\mathcal{L}=\min_{w_z}\left[(a\,w_x+c\,w_y-w_x\,w_z)^2+(b\,w_x+d\,w_y+w_x\,w_z)^2\right] \tag{20}$$

$$\frac{\partial \mathcal{L}}{\partial w_z}=2\,\left[-a\,w_x\,w_y-cw_y^2+w_z\,w_x^2+b\,w_x^2+d\,w_x\,w_y+w_z\,w_x^2\right]=0$$

We can estimate w_z if either $w_x \neq 0$ or $w_y \neq 0$ and obtain :

$$w_z=\frac{(a-d)\,w_x\,w_y-b\,w_x^2+c\,w_y^2}{w_x^2+w_y^2} \tag{21}$$

Then, by two simple relations we can calculate N_x and N_y if $w_x \neq 0$ and $w_y \neq 0$:

$$\left\{ \begin{array}{rcccl} N_x & = & \dfrac{a}{w_y} & = & \dfrac{w_z - c}{w_x} \\ N_y & = & \dfrac{-d}{w_x} & = & \dfrac{w_z + b}{w_y} \end{array} \right. \tag{22}$$

But using a least-square combination of (22), as before, we obtain :

$$\mathcal{M} = \min_{N_x, N_y} \left[\left[(a - w_y\, N_x)^2 + (c - w_z + w_x\, N_x)^2 \right] + \left[(d + w_x\, N_y)^2 + (b + w_z - w_y\, N_y)^2 \right] \right] \tag{23}$$

$$\left\{ \begin{array}{rcccl} \dfrac{\partial \mathcal{M}}{\partial N_x} & = & 2\,(w_x^2\, N_x + w_y^2\, N_x + c\, w_x - a\, w_y - w_x\, w_z) & = & 0 \\ \dfrac{\partial \mathcal{M}}{\partial N_y} & = & 2\,(w_x^2\, N_y + w_y^2\, N_y - b\, w_y + d\, w_x - w_y\, w_z) & = & 0 \end{array} \right.$$

and can estimate N_x and N_y as soon as either $w_x \neq 0$ or $w_y \neq 0$ and obtain :

$$\left\{ \begin{array}{rcl} N_x & = & \dfrac{w_x\, w_z + a\, w_y - c\, w_x}{w_x^2 + w_y^2} \\ N_y & = & \dfrac{w_y\, w_z + b\, w_y - d\, w_x}{w_x^2 + w_y^2} \end{array} \right. \tag{24}$$

Reciprocally, if we replace w_x and w_y in (11), we obtain, knowing α, :

$$\left\{ \begin{array}{rcl} a & = & \sin(\alpha)\,[\lambda\, N_x] \\ b & = & \sin(\alpha)\,[\lambda\, N_y] - w_z \\ c & = & w_z - \cos(\alpha)\,[\lambda\, N_x] \\ d & = & -\cos(\alpha)[\lambda\, N_y] \end{array} \right. \tag{25}$$

It is clear from these equations that we cannot recover (N_x, N_y) but only $(\lambda\, N_x, \lambda\, N_y)$ and obtain, for each α, only one solution in the general case since these equations are linear.

Geometrical interpretation of the two solutions. As already found by Tsai [23] and Longuet-Higgins [17], we have two solutions for the reconstructed scene. One of these two solutions is to be eliminated. The two values of α corresponds to two configurations of the scene with respect to the angle s (that is if we are in front of or at the back of the scene).

As a **conclusion**, collecting these different results, we *clearly demonstrate that, if we are interested in recovering the local structure of the 3D scene, we can compute the local orientation of the observed surface, for an orthographic projection* :

- as soon as a (small) rotation is performed but not around the optical axis (since we must have either w_x or w_y not null),
- from a set of linear equations if the rotation is known,
- from a set of quartic equations and up to a scale-factor λ (fixed by assuming $N_x^2 + N_y^2 = \lambda^2$ in the previous equations) if the rotation is not known, and obtain two real solutions if we have $(b + c)^2 \geq 4ad$.

An optimal framework to estimate w is to minimize, considering equation (12) for all facets :

$$(\alpha, \bar{w_z}) = argmin_{(\alpha,w_z)} \sum_k \lambda_k \left[\left[\cos(\alpha)\,(b^k + w_z) + \sin(\alpha)\,d^k \right]^2 + \left[\cos(\alpha)\,a^k + \sin(\alpha)\,(c^k - w_z) \right]^2 \right] \tag{26}$$

where :

$$\left\{ \begin{array}{rcl} w_x & = & \cos(\alpha) \\ w_y & = & \sin(\alpha) \end{array} \right. \tag{27}$$

and λ_k is a weight for each match.

We will use this mechanism in our implementation.

2.4 Relation with the perspective projection.

Let us consider our perspective projection model again, i.e. :

$$\left\{ \begin{array}{rcl} u & = & u_0 + \alpha_u\, X/Z \\ v & = & v_0 + \alpha_u\, Y/Z \end{array} \right. \tag{28}$$

Following [27] the retinal motion field of a planar patch defined with the notations of equation (6) is now:

$$\left\{ \begin{array}{rcl} du & = & C_u + a\,u + b\,v + u\,(e\,u + g\,v) \\ dv & = & C_v + c\,u + d\,v + v\,(e\,u + g\,v) \end{array} \right. \tag{29}$$

with :

$$\left\{ \begin{array}{rcl} a & = & \underbrace{\left[-\frac{t_x}{Z_0}\right] N_x - \frac{t_z}{Z_0}}_{a^*} - \frac{t_z}{Z_0}\frac{2\,N_x\,u_0 + N_y\,v_0}{f\,Z_0} + \frac{-2\,w_y\,u_0 + w_x\,v_0}{f\,Z_0} \\ b & = & \underbrace{\left[-\frac{t_x}{Z_0}\right] N_y - w_z}_{b^*} - \frac{t_z}{Z_0}\frac{N_y\,u_0}{f\,Z_0} + \frac{w_x\,u_0}{f\,Z_0} \\ c & = & \underbrace{w_z - \left[\frac{t_y}{Z_0}\right] N_x}_{c^*} - \frac{t_z}{Z_0}\frac{N_x\,v_0}{f\,Z_0} - \frac{w_y\,v_0}{f\,Z_0} \\ d & = & \underbrace{-\left[\frac{t_y}{Z_0}\right] N_y - \frac{t_z}{Z_0}}_{d^*} - \frac{t_z}{Z_0}\frac{N_x\,u_0 + 2\,N_y\,v_0}{f\,Z_0} + \frac{2\,w_x\,v_0 - w_y\,u_0}{f\,Z_0} \\ e & = & \frac{w_y}{f\,Z_0} + \frac{t_z\,N_x}{f\,Z_0^2} \\ g & = & -\frac{w_x}{f\,Z_0} + \frac{t_z\,N_y}{f\,Z_0^2} \end{array} \right. \tag{30}$$

while C_u and C_v are huge expressions of the motion and structure parameters but can be simplified with some algebra :

$$\left\{ \begin{array}{rcl} C_u & = & f\,Z_0\,(\frac{t_x}{Z_0} + w_y) - a\,u_0 - b\,v_0 - e\,u_0^2 - g\,u_0\,v_0 \\ C_v & = & f\,Z_0\,(\frac{t_y}{Z_0} - w_x) - c\,u_0 - d\,v_0 - e\,u_0\,v_0 - g\,v_0^2 \end{array} \right. \tag{31}$$

In the case of $w_x = w_y = 0$, by elimination of f in the previous equations ($f = \frac{\alpha_u}{Z_0}$ is not constant over the whole frame, only locally), we can obtain a error function between u_0, v_0,

$\frac{t_y}{t_x}$ if $t_x \neq 0$ and the parameters from which, by minimization, we can compute the location of the optical center and, up to a scalar factor, the translation, as detailed in the sequel.

As expected, we obtain -for a planar patch under a perspective projection- a quadratic model of the retinal motion field, which parameters do not correspond to the affine model obtained in equation (10), and must be estimated using at least four points.

However, these two models can be related in two situations:

- If we are undergoing a translation parallel to the retinal plane, and if the rotation axis is parallel to the optical axis, i.e. if we have

$$w_x = w_y = 0 \quad and \quad t_z = 0 \tag{32}$$

the two sets of equations are in correspondence through the transformation:

$$w_x \rightarrow \left[\frac{t_y}{Z_0}\right] \quad and \quad w_y \rightarrow \left[-\frac{t_x}{Z_0}\right] \tag{33}$$

It might be noted that Z_0 depends on the points depth but this scale factor disappears in all equations such as in (12) so that it will not perturbate the solution.

There is a geometric interpretation of this situation. We are simply in a case where the 3D-translation of the perspective model "is seen as a rotation" by the orthographic model. Indeed, a rotation around the **y** axis, a "pan", induces an horizontal retinal displacement as an horizontal translation would do, while a rotation around the **x** axis, a "tilt", induces a vertical retinal displacement as a vertical translation up to a sign.

- If the plane is a fronto-parallel plane and if the rotation axis is parallel to the optical axis, i.e. if we have $w_x = w_y = 0$ and $N_x = N_y = 0$ the motion field is still an affine motion field. However, its correspondences with equation (10) does not exist unless $t_z = 0$.

This situation will not be considered here, i.e. we will always impose $t_z = 0$.

A step further, let us consider that we estimate an affine motion field applying equation (10) on three points $\mathbf{m}_i = (u_i, v_i, 1)^T, i = \{1, 2, 3\}$ under a perspective projection.

This yields a set of 6 linear equations (2 for each point) in six unknowns with a unique solution if and only if the three points are not collinear.

In the general case, this solution will neither correspond to the affine model obtained considering an orthographic projection, nor correspond to the first order terms of the quadratic model obtained considering a perspective projection, because the estimation is biased.

However, if we assume that these three points are close to the principal point, i.e. $u_i = u_0 + \nu\,\delta_i^u$ and $v_i = v_0 + \nu\,\delta_i^v$ the estimated parameters $\bar{a}, \bar{b}, \bar{c}, \bar{d}$ will verify the following approximation :

$$\begin{aligned} &\bar{a} = a^* + \upsilon_a, \quad \bar{b} = b^* + \upsilon_b, \quad \bar{c} = c^* + \upsilon_c, \quad \bar{d} = d^* + \upsilon_d \\ with: \quad &\upsilon = o(\epsilon)\; o(|w_x| + |w_y| + |t_z|\,(1 + |N_x| + |N_y|)) \end{aligned} \tag{34}$$

so that, in this case, and even if the rotation is not exactly parallel to the optical axis while the translation is not perfectly parallel to the retinal plane, *the model is approximately related to model proposed previously by the transformation of equation (33).*

We thus can **conclude** that *we can relate the two models, by an active vision strategy in which we attempt to perform a translation parallel to the retinal plane and cancel the rotational components in pan and tilt. In order to cope with the fact that these constraints are only approximately matched, we consider only a foveal window, i.e. an area of the retinal plane close to the principal point.*

2.5 Autocalibration of the visual sensor.

2.5.1 Computing the optical center.

From equation (31), with simplification in the case of motion where $w_x = w_y = 0$ and $t_z = 0$, we obtain :

$$\left\{ \begin{array}{lcl} C_u & = & f\,t_x - a\,u_0 - b\,v_0 \\ C_v & = & f\,t_y - c\,u_0 - d\,v_0 \end{array} \right. \tag{35}$$

since $e = g = 0$.

By elimination of f in the previous equations if $t_x \neq 0$, we obtain :

$$C_v = C_u \frac{t_y}{t_x} + a\,u_0 \frac{t_y}{t_x} + b\,v_0 \frac{t_y}{t_x} - c\,u_0 - d\,v_0 \tag{36}$$

which can be interpreted as a measurement equation. Let us write :

$$x = (x_0 = \frac{t_y}{t_x},\, x_1 = u_0 \frac{t_y}{t_x},\, x_2 = v_0 \frac{t_y}{t_x},\, x_3 = u_0,\, x_4 = v_0) \tag{37}$$

Equation (36) can be written :

$$f(x) = C_v - C_u\,x_0 - a\,x_1 - b\,x_2 + c\,x_3 + d\,x_4 = 0 \tag{38}$$

and we can minimize the quadratic error function :

$$\sum_i \lambda_i f_i^2(x)$$

over the set of matches, as in equation (26).

Having this initial estimate, since it is quadratic, we can refine this previous result by considering :

$$y = (y_0 = \frac{t_y}{t_x},\, y_1 = u_0,\, y_2 = v_0) \tag{39}$$

$$g(y) = C_v - C_u\,y_0 - a\,y_0\,y_1 - b\,y_0\,y_2 + c\,y_1 + d\,y_2 = 0 \tag{40}$$

and minimize the non-linear error function :

$$\sum_i g_i^2(y)$$

over the set of matches, as in equation (26).

We thus obtain u_0, v_0 and $\frac{t_y}{t_x}$ by this method.

2.5.2 What cannot be calibrated

Considering the model of equation (4), we easily see that the intrinsic calibration is only defined by three parameters u_0, v_0 and f. Similarly the extrinsic parameters as defined in equation (9) are t_x, t_y, t_z and w_x, w_y, w_z, but with $w_x = w_y = t_z = 0$ in our case.

Now, considering equation (29), (30) and (31) we can easily verify that t_x and t_y appear always with $1/Z_0$ as scale factor, so that this quantity being unknown, we cannot compute the absolute value of t_x and t_y but only their ratio. This corresponds to the scale factor indetermination of a monocular image sequence but in the case of an orthographic projection.

Similarly, f never appears as an independent quantity but is always multiplied by the Z_0 unknown factor. As a consequence, it cannot be recovered. Again this corresponds to the same scale factor with is that the depths Z cannot be recovered as absolute values but only relative to a global scale factor.

Finally, the previous equations correspond to all what can be computed and we do not need to try to evaluate anything else from these equations.

2.6 Implementation and experimental verification.

2.6.1 Early-vision module: corner detection and matching.

Feature points corresponding to high curvature points are extracted from each image, the "Harris" corner detector [13].

Given a high curvature point in one image, we use a correlation window centered at this point, and select a rectangular search area around this point in the second image. We perform a correlation operation and select those locations for which the correlation score is high. If the above constraint is fulfilled, we say that the pair of points considered is mutually consistent and forms a candidate match. We eliminate these ambiguities using a relaxation mechanism. Since we are dealing with image sequences with very low disparities, the relaxation mechanism has to deal with only a very small set of ambiguous matches. A sub-pixel interpolation of the retinal disparity is obtained.

2.6.2 Defining a triangulation of the scene.

First, we want to determine the parameters (C_u, C_v, a, b, c, d) defined by the 2 equations (10). A set of 3 points give us a system of 6 linear equations with 6 unknowns, and a unique solution if the points are non-collinear. Thus, we make use of the Delaunay triangulation as implemented by [7] on the first frame and triangulate the second frame by points-corresponding. Furthermore we assume that (C_u, C_v, a, b, c, d) are constant in each triangle.

2.6.3 Computing the surface normals.

We first initialize the two values of α and w_z by (18) and (21) on one triangle where it is computable and refine these values by minimization of (26).

We obtain from the two solutions of α and w_z one unique solution after minimization. We then compute the normals from (24), up to a scalar constant.

We do not consider triangles that are too flat for the computation (in our case, they do not correspond to physical parts of the scene).

2.6.4 Experimentation.

We experiment on a sequence of a calibration grid (see figure (1)). We know the 3-D coor-

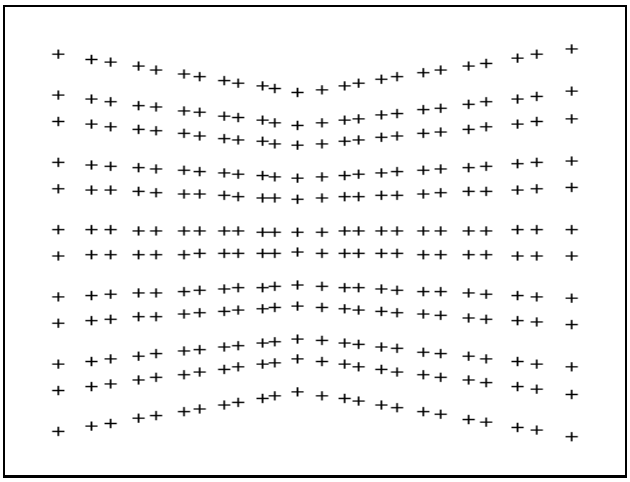

Figure 1: Calibration grid (points of interest).

dinates of 264 points of interest of a grid located at one meter from the optical center. We simulate a perspective projection before and after a small motion (rotation and translation) of the grid and obtain two images (512x512 pixels) where the optical center is in the center of the image. We triangularize the scene in 520 triangles and determine then the optical flow (in pixels) by computing the parameters (C_u, C_v, a, b, c, d), the solutions for α and w_z, and the normals (normalized). Knowing the 3D-coordinates and the motion, we can compute the errors on the optical flow (the norm of the difference between theoretical and practical optical flow related to the distance between the point 2-D and the optical center), the angle α, and the normals (the norm of the difference between theoretical and practical normals related to the distance between the point 2-D and the optical center).

As seen previously, if we have no translation parallel to the z-axis (corresponding to the optical axis) and have rotation only around the z-axis, we can recover the motion and the normals by considering the correspondences of equation (33). We will then introduce some rotation around the x and y-axis and translation parallel to the z-axis to see the influence on the quality of the result. At last, we will study the addition of noise on the matchings and then will deal with real data.

Translation on x and y. See figures (2) and (3).

We have chosen a motion with $t_x = 1.5$ pixels and $t_y = 1.3$ pixels.

The initialization yields two solutions : on one hand $\alpha_1 = 0.856666$ with $w_{z_1} = -3.52263\,10^{-8}$ and on the other hand $\alpha_2 = -1.570826$ with $w_{z_2} = 1.03\,10^{-3}$.

With the minimization, we obtain the same solution for α_1 and α_2 and the same for w_{z_1} and w_{z_2} :

$\alpha = -0.856707$ rad with $|\frac{\Delta\alpha}{\alpha_{th}}| = 1.8\,10^{-4}\%$ and $w_z = -2.2\,10^{-9}$ with $|\Delta w_z| = 2.2\,10^{-9}$. The errors on optical flow are less than $5\,10^{-5}$ pixel, and the errors on normals, less than $4\,10^{-5}$.

In this case, the errors are only due to the approximation that the parameters (C_u, C_v, a, b, c, d) are constant on the triangles i.e. that the surface is locally planar.

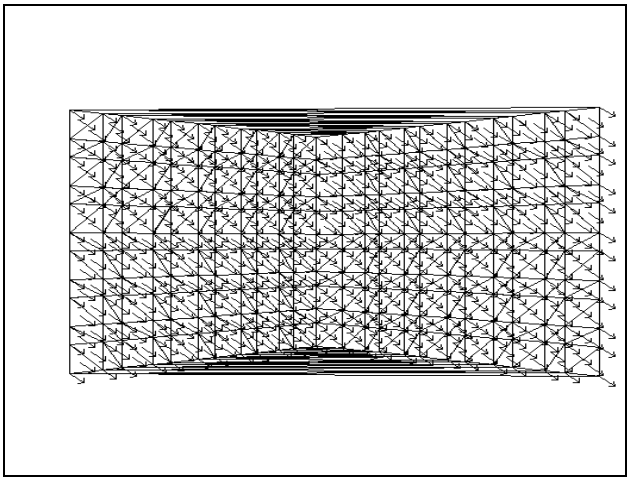

Figure 2: Affine optical flow (in pixels x 10) for a translation of 1.5 pixels on x and 1.3 pixels on y.

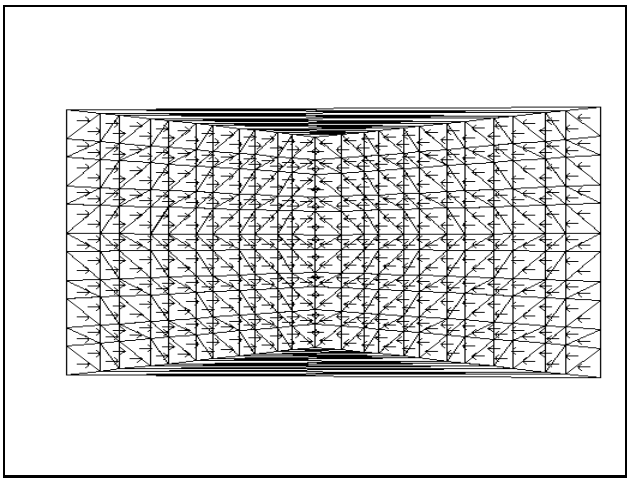

Figure 3: Normals (x 10) for a translation of 1.5 pixels on x and 1.3 pixels on y.

Rotation around the z-axis combined with translation on x and y. See figures 4, 5 and 6.

We have chosen for the motion : $t_x = 1.5$ pixels, $t_y = 1.3$ pixels and $w_z = 0.04$ rad.

The initialization yields two solutions : $\alpha_1 = -0.29$ with $w_{z_1} = 3.97\,10^{-2}$ and $\alpha_2 = -2.13$ with $w_{z_2} = 4.13\,10^{-2}$.

With the minimization, we obtain the same solution for α_1 and α_2 and the same for w_{z_1} and w_{z_2} :
$\alpha = -0.8517$ rad. with $|\frac{\Delta\alpha}{\alpha_{th}}| = 0.59\%$ and $w_z = 0.0399892$ with $|\Delta w_z| = 2.7\,10^{-2}\%$.

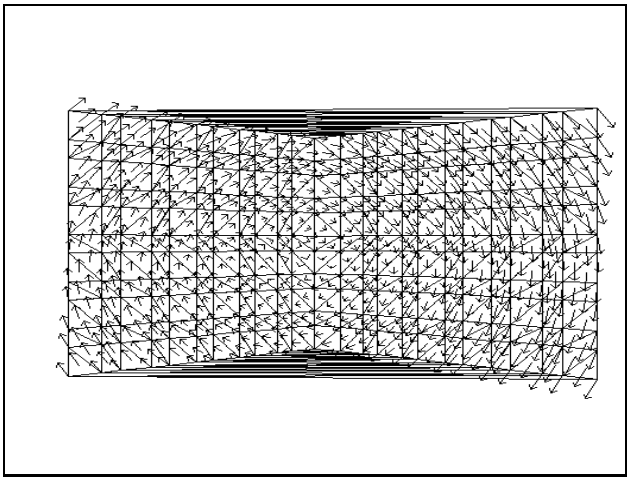

Figure 4: Affine optical flow (in pixels x 2) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis.

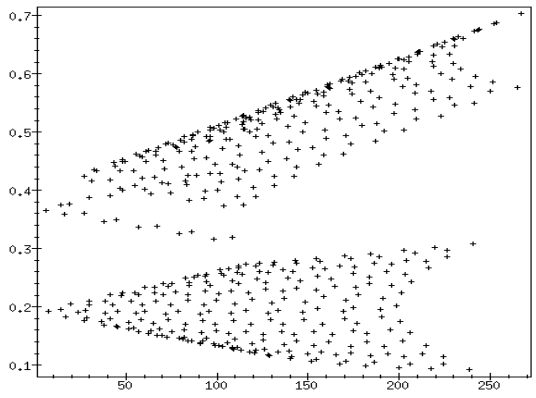

Figure 5: Errors on affine optical flow (in pixels) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis, related to the distance from the optical center (in pixels).

In this case, the errors are mainly due to the approximation of the rotation to his first order. Concerning this source of errors, we have tried with several values of w_z for which it is impossible to initiate α with the method explained previously. With an initialization so that $\alpha = 0$ and $w_z = 0$, we found some results that are collected in table (1). This shows us the angle of rotation must be small to be approximated to its first order but if it is too small, the quality of the result on w_z decreases because of the precision of the computing.

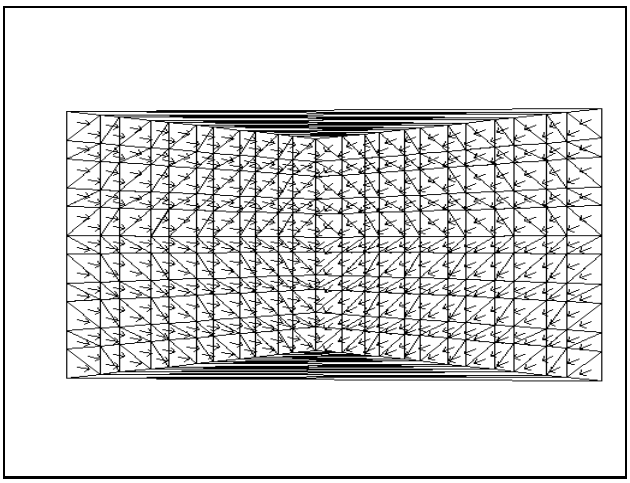

Figure 6: Normals (x 10) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis. (average of errors : 0.35 pixel)

motion	**initialization**				**after minimization**			
w_z	α_1	α_2	w_{z_1}	w_{z_2}	α	w_z	$\Delta\alpha$	Δw_z
$4\,10^{-5}$	-0.86	-1.57	$4.00\,10^{-5}$	$1.1\,10^{-3}$	-0.86	$4.00\,10^{-5}$	$2.0\,10^{-4}$ %	$2.7\,10^{-3}$ %
$4\,10^{-4}$	-0.86	-1.57	$3.99\,10^{-4}$	$1.4\,10^{-3}$	-0.86	$4.00\,10^{-4}$	$1.6\,10^{-4}$ %	$9.6\,10^{-4}$ %
$4\,10^{-3}$	-0.85	-1.58	$3.99\,10^{-3}$	$5.0\,10^{-3}$	-0.86	$4.00\,10^{-3}$	$5.7\,10^{-3}$ %	$2.9\,10^{-4}$ %
$4\,10^{-2}$	-0.29	-2.13	$3.9\,10^{-2}$	$4.1\,10^{-1}$	-0.85	$3.99\,10^{-2}$	0.6 %	0.03 %
$4\,10^{-1}$	0	0	0	0	0.93	$3.9\,10^{-1}$	208 %	2.7 %
4	0	0	0	0	0.36	-0.76	141 %	119 %

Table 1: Table of results with different values of w_z, with $t_x = 1.5$ pixels and $t_y = 1.3$ pixels.

Introduction of rotation around the x-axis and the y-axis. See figures (7), (8) and (9).

We have chosen for the motion : $t_x = 1.5$ pixels, $t_y = 1.3$ pixels, $w_x = 0.01$ rad., $w_y = 0.01$ rad., and $w_z = 0.04$ rad.

We have collected (see table 2) some results that show us the influence of the values of w_x and w_y on the quality of the result. As expected, the lower the values are less, the better the result is.

But, figure (8) shows us that the result is still good near the fovea.

More precisely, we will study in the next paragraph which of the following approximation is involved :

- orthographic projection ($Z = Z_0$)
- small rotation ($R = e^{\tilde{\mathbf{w}}} = I + \tilde{\mathbf{w}} + o(||\mathbf{w}||)$)

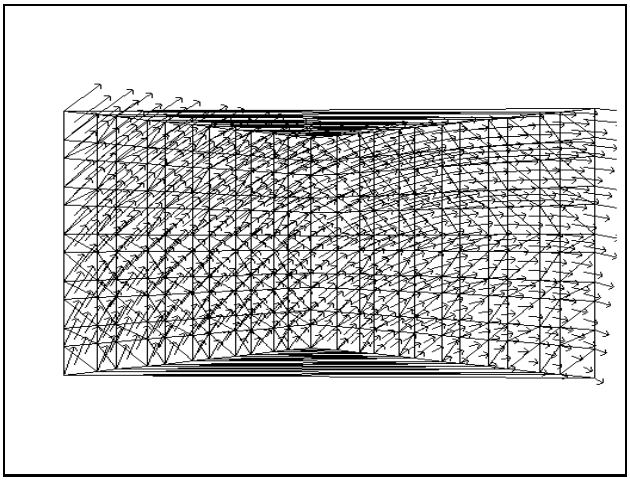

Figure 7: Affine optical flow (in pixels x 2) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.01 rad. around the x and y-axis, 0.01 rad. around the z-axis.

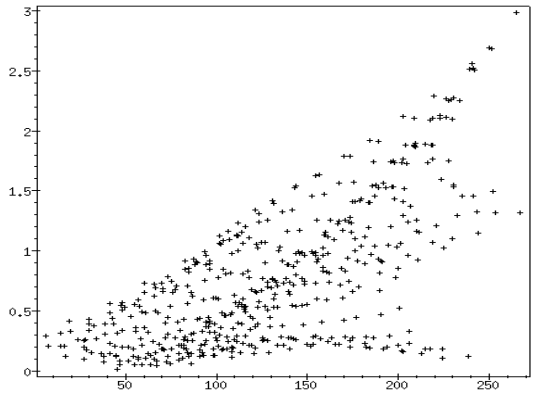

Figure 8: Errors on affine optical flow (in pixels) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.01 rad. around the x and y-axis, 0.04 rad. around the z-axis, related to the distance from the optical center (in pixels).

Study of approximations on rotation and projection. Let us simulate from known 3D-points, the orthographic and perspective model of projection and generate exact rotation ($R = e^{\tilde{\mathbf{w}}}$) and approximate rotation ($R = I + \tilde{\mathbf{w}}$) between 2 frames.

In the case of orthographic projection, the angle α corresponds to the angle between w_x and w_y, and in the case of perspective projection, to the angle between t_y and $-t_x$.

Let us see the results for the following motion : $t_x = 1.5$ pixels, $t_y = 1.3$ pixels, $w_x = w_y = 0.001$ rad., $w_z = 0.04$ rad. as shown in table (3).

We have also demonstrated that for small values of rotation, the errors are principally due to the approximation of the orthographic model of projection, and for big values of the angle of rotation, to the approximation of the motion to its first order.

See figure (16)) for a summary.

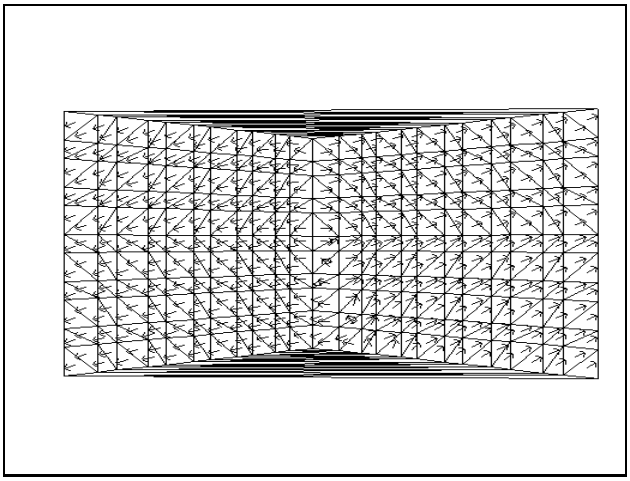

Figure 9: Normals (x 10) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.01 rad. around the x and y-axis, 0.04 rad. around the z-axis (average of errors on normals : 0.01).

motion		**initialization**				**after minimization**			
w_x	w_y	α_1	α_2	w_{z_1}	w_{z_2}	α	w_z	$\Delta\alpha$	Δw_z
10^{-5}	10^{-5}	-0.30	-2.13	0.04	0.04	-0.85	0.04	0.4 %	0.03 %
10^{-4}	10^{-4}	-0.30	-2.12	0.04	0.04	-0.87	0.04	1.3 %	0.03 %
10^{-3}	10^{-3}	-0.89	-1.83	0.04	0.04	-0.10	0.04	16 %	0.08 %
10^{-2}	10^{-2}	-1.07	-2.10	0.04	0.05	1.48	0.04	273 %	0.06 %
10^{-1}	10^{-1}	-1.32	-2.02	0.04	0.0 6	1.08	0.04	256 %	0.16 %

Table 2: Table of results with different values of w_x and w_y, for $w_z = 0.04$ rad, $t_x = 1.5$ pixels and $t_y = 1.3$ pixels.

simulation		**initialization**				**after minimization**			
projection	rotation	α_1	α_2	w_{z_1}	w_{z_2}	α	w_z	$\frac{\Delta\alpha}{\alpha}$	$\frac{\Delta w_z}{w_z}$
perspective	exact	-0.89	-1.83	0.04	0.04	-0.10	0.04	16 %	0.08 %
perspective	first order	-1.18	-1.45	0.04	0.04	-1.00	0.04	17 %	0.01 %
orthographic	exact	-0.95	-2.95	0.04	0.04	0.81	0.04	14 %	$2.6\,10^{-2}$ %
orthographic	first order	0.79	-1.57	0.04	0.04	0.79	0.04	0.1 %	$1.3\,10^{-5}$ %

Table 3: Table of results by simulation of projection and rotation, for $w_x = w_y = 0.001$ radians, $w_z = 0.04$ rad., $t_x = 1.5$ pixels and $t_y = 1.3$ pixels.

Introduction of translation parallel to the z-axis. See figures (17), (18), (19) and (20).

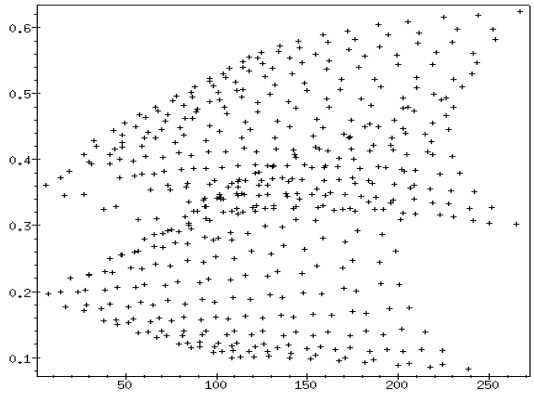

Figure 10: Errors on affine optical flow (in pixels). Simulation of perspective projection coupled with exact rotation.

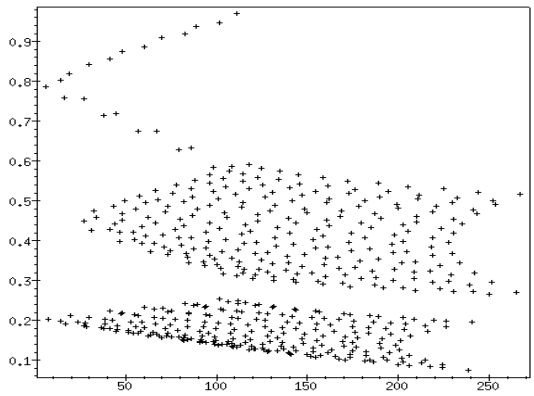

Figure 11: Errors on normals. Simulation of perspective projection coupled with exact rotation.

We have chosen for the motion : $t_x = 1.5$ pixels, $t_y = 1.3$ pixels, $t_z = 0.5$ pixel and $w_z = 0.04$ radians

We have collected (see table 4) some results that show us the influence of the values of t_z on the quality of the result. As expected, the lower the values are, the better the result is. But, figure (21) shows us that the result continues to be good near the fovea.

2.6.5 Discussion

Considering the previous results, several conclusions can be drawn :

- The model is very precisely verified when considering "affine" displacements, i.e. $w_x = w_y = t_z = 0$ which an accuracy lower than one pixel, even at the border of the image.

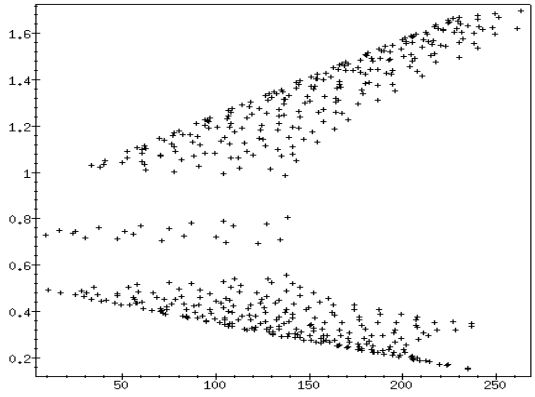

Figure 12: Errors on affine optical flow (in pixels). Simulation of orthographic projection coupled with exact rotation (average of errors on normals : 0.45).

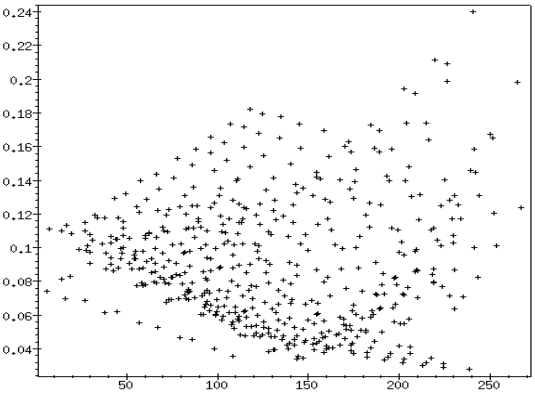

Figure 13: Errors on affine optical flow (in pixels). Simulation of perspective projection coupled with first order rotation.

motion	**initialization**				**after minimization**			
t_z	α_1	α_2	w_{z_1}	w_{z_2}	α	w_z	$\lvert\frac{\Delta\alpha}{\alpha}\rvert$	$\lvert\frac{\Delta w_z}{w_z}\rvert$
$5\,10^{-2}$	-0.25	-2.2	0.039	0.041	-0.85	0.040	0.6 %	$2.5\,10^{-2}$ %
$5\,10^{-1}$	0.29	-2.6	0.040	0.041	-0.86	0.040	0.5 %	$6.5\,10^{-3}$ %
1	0	0	0	0	-0.89	0.040	3.1 %	$2.0\,10^{-2}$ %
5	0	0	0	0	-1.43	0.040	66 %	0.2 %

Table 4: Table of results with different values of t_z, for $w_z = 0.04$ rad, $t_x = 1.5$ pixels and $t_y = 1.3$ pixels.

- Having a "counter-rotation", i.e. $w_z \neq 0$ does not significantly affect the results unless the amplitude of the rotation is so important that it perturbates the early-vision processes.

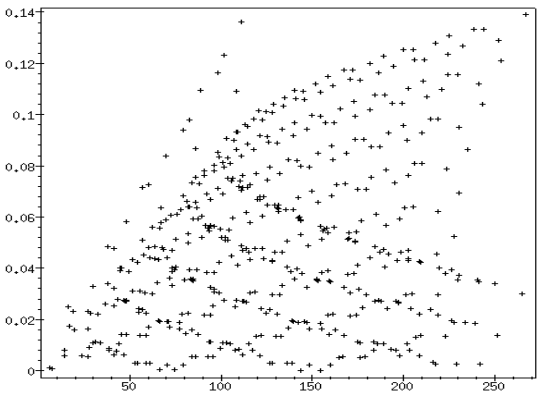

Figure 14: Errors on normals. Simulation of perspective projection coupled with first order rotation.

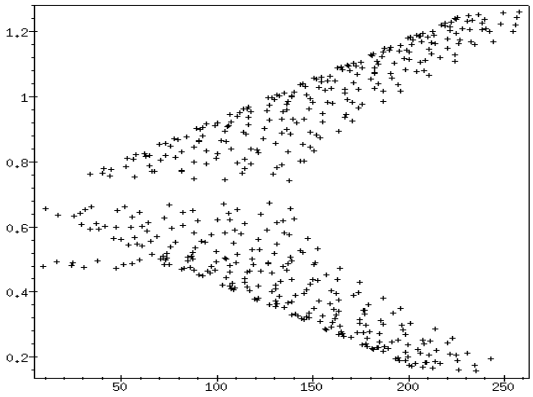

Figure 15: Errors on affine optical flow (in pixels). Simulation of orthographic projection coupled with first order rotation (average of errors on normals : $3\,10^{-6}$).

However, a non-negligible loss of performance exists even for small amplitudes of counter-rotation, so that we better avoid them, even if it is not mandatory to have it cancelled.

Note that this second-order effect is very likely due to the fact we use a first order approximation of the rotation.

- The main source of error is indeed due to the existence of rotations, w_x or w_y which is contradictory with our model. The perturbation is already close to a pixel, for small rotations of 0.01 radians, and the evaluation of the surface normal is not reliable anymore.

  However this effect is highly dependent on the position of the point with respect to the principal point, the error having an hyperbolic profile, as predicted by the model.

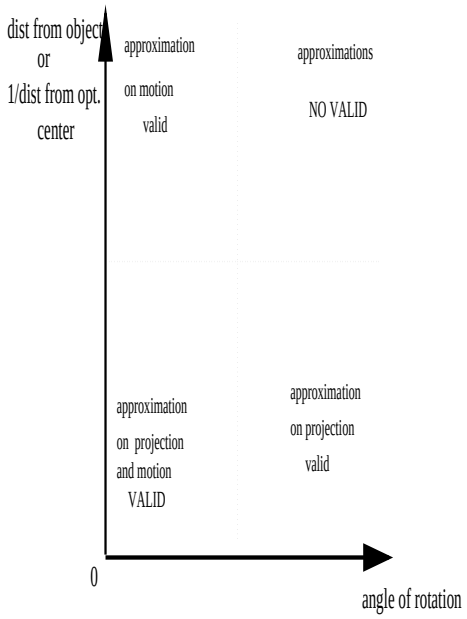

Figure 16: Validity of the approximations.

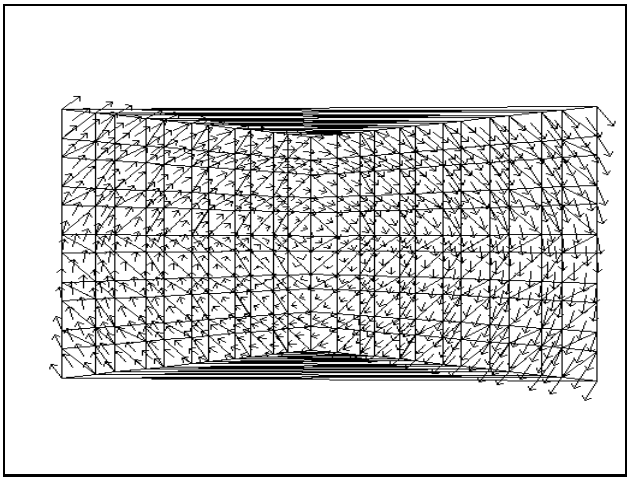

Figure 17: Affine optical flow (in pixels x 2) for a translation of 1.5 pixels in x, 1.3 pixels in y, and 0.5 pixels on z and a rotation of 0.04 rad. around the z-axis.

As a consequence, considering a small window of say, 50 x 50 pixels, this bias is reduced by a factor close to 5. This will be the purpose of the next subsection.

Furthermore, the error can be evaluated and it might be possible to detect such a bias, and control the system to avoid this bias as discussed in the last section.

- As already discussed for the counter-rotation, the fact we use a first-order approximation of the rotation, is not the main source of error, especially because we attempt, in fact, to cancel them, and thus to maintain rotation as small as possible.

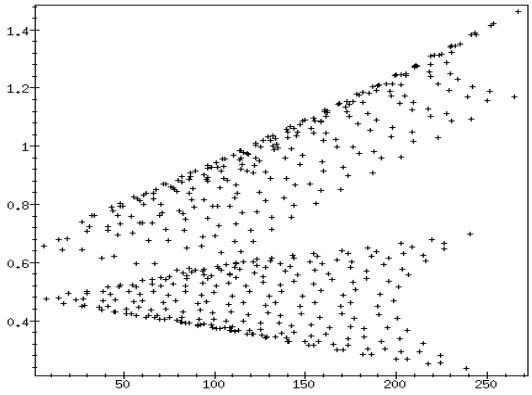

Figure 18: Errors on affine optical flow (in pixels) for a translation of 1.5 pixels in x, 1.3 pixels in y, and 0.5 pixels on z and a rotation of 0.04 rad. around the z-axis, related to the distance from the optical center (in pixels).

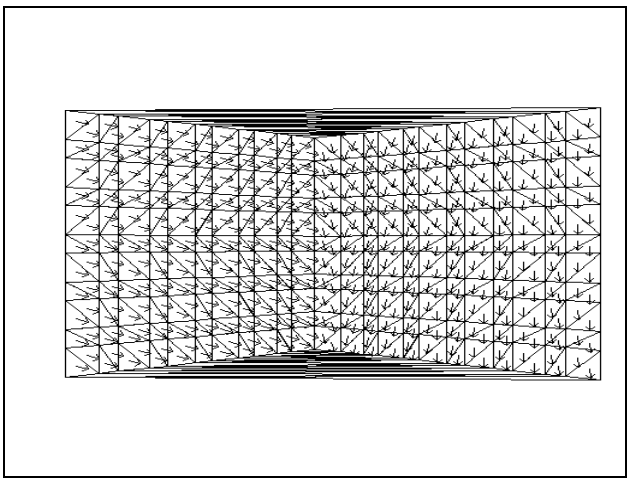

Figure 19: Normals (x 10) for a translation of 1.5 pixels in x, 1.3 pixels in y, and 0.5 pixel in z and a rotation of 0.04 rad. around the z-axis.

2.6.6 Improving the performances.

We have obtained in the two previous paragraphs that when the quality of the global result decreases, it is still near the fovea. If we know the location of the optical center, we are able to introduce on the weights for the minimization on which we obtain α and w_z the following quantity :

$$\frac{1}{1+\phi\,||(u,v)-(u_0,v_0)||^{\psi}} \tag{41}$$

We will try several values of ϕ and ψ and collect the results in table (5). Since the influence of ϕ is not significant, we have principally reported the results concerning ψ.

This heuristic is in fact, of common use in the domain, see for instance [10], [28].

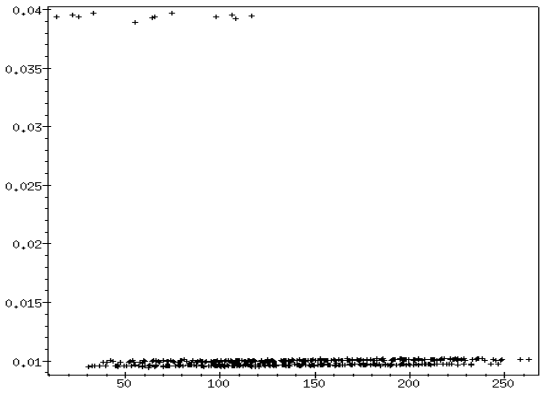

Figure 20: Errors on normals for a translation of 1.5 pixels in x, 1.3 pixels in y, and 0.5 pixel in z and a rotation of 0.04 rad. around the z-axis, related to the distance from the optical center (in pixels).

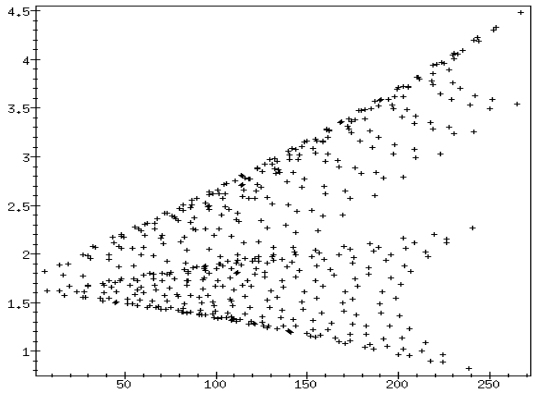

Figure 21: Errors on affine optical flow (in pixels) for a translation of 1.5 pixels in x, 1.3 pixels in y, and 0.5 pixel in z and a rotation of 0.04 rad. around the z-axis, related to the distance from the optical center (in pixels).

The motion considered here is: $t_x = 1.5$ pixels, $t_y = 1.3$ pixels, $t_z = 0.5$ pixel $w_x = 0.005$ pixel, $w_y = 0.005$ pixel and $w_z = 0.04$ pixel.

From table (5), we see that for $\phi = 1$ and $\psi = 2$, we obtain a amelioration on the error on α with a not too important degradation of the value of w_z.

But, in the general case, we do not know the location of the optical center. So that we must compute it from the parameters C_u and C_v (see equations (38) and (39)).

We compute $y = (\frac{t_y}{t_x}, u_0, v_0)$ for some motions and collect the results in table 6. We observe that the precision for $\frac{t_y}{t_x}$ is very good, and the precision for u_0 and v_0 not as good. But, when introducing these values in the weights of equation (41), we obtain the same results as the results obtained with exact values of u_0 and v_0, collected in table 5. We can

ϕ	ψ	$\lvert\frac{\Delta\alpha}{\alpha}\rvert$	$\lvert\frac{\Delta w_z}{w_z}\rvert$
0	1	64 %	0.16 %
1	1	42 %	0.12 %
10	1	41 %	0.11 %
1	1.5	13 %	0.17 %
10	1.5	12 %	0.17 %
1	2	0.1 %	0.21 %
10	2	0.4 %	0.21 %
1	2.5	5.6 %	0.24 %
1	3	9.7 %	0.26 %

Table 5: Table of results since introduction of weights related to the distance of points from the fovea in the minimization

explain this by the fact that the errors considering the normals are less in the foveal part of the retina and not exactly in the center of the fovea.

motion			**results**					
w_x	w_y	w_z	u_0	$\frac{\lvert\Delta u_0\rvert}{u_0}$	v_0	$\frac{\lvert\Delta v_0\rvert}{v_0}$	$\frac{t_y}{t_x}$	$\frac{\lvert\Delta\frac{t_y}{t_x}\rvert}{\frac{t_y}{t_x}}$
0	0	0	253	1.8 %	256	0.2 %	0.866667	$4\,10^{-5}$ %
0	0	0.004	255	0.4 %	257	0.5 %	0.866753	$9\,10^{-3}$ %
0	0	0.04	272	6 %	237	7 %	0.866644	$3\,10^{-3}$ %
0	0	0.4	252	1.6 %	264	3.2 %	0.866699	$3\,10^{-3}$ %
0.001	0.001	0.04	254	0.6 %	264	3 %	-0.017414	102 %
0.01	0.01	0.04	269	5 %	242 %	5 %	-0.676686	178 %

Table 6: Table of results with $t_x = 1.5$ pixels, $t_y = 1.3$ pixels and $t_z = 0$ for computing u_0, v_0 and $\frac{t_y}{t_x}$ from C_u and $C - v$.

Then, we compute the angle α from $\frac{t_y}{t_x}$ and the result obtained by minimization (26) with this initialization is the same as with the previous method defined by (18) and (21).

As a conclusion,, we observe that we obtain a much better precision on u_0 and v_0 but that the good precision on $\frac{t_y}{t_x}$ in the case of $\mathbf{t} = (t_x, t_y, 0)$ and $\mathbf{w} = (0, 0, w_z)$ decrease very fast with introducing of other components in the motion.

Furthermore, we easily verify that considering a weighted estimation which integrate the fact that foveal information is more reliable that peripheral information significantly improve the performances of the results.

2.6.7 Testing the robustness to noise.

For this test, we introduce Gaussian noise on the 2-D projected points on the first frame with a mean value equal to zero, and various variance.

As for the minimization for α and w_z, we introduce weights on each triangle : $\lambda_k = \frac{1}{1+e}$ where e is the average of the errors of each of the three points determining the triangle k.

We have tried some other expressions for the weights in the following form :

$$\lambda_k = \frac{1}{1 + \mu\, e^{\nu}}$$

but they do not significantly increase the quality of the result.

We have taken the same motion as previously, i.e. $t_x = 1.5$ pixels, $t_y = 1.3$ pixels, and $w_z = 0.04$ radians

The results are collected in table (7).

noise	**after minimization**			
σ	α	w_z	$\lvert\frac{\Delta\alpha}{\alpha}\rvert$	$\lvert\frac{\Delta w_z}{w_z}\rvert$
0	-0.83	0.04	3.7 %	0.03 %
0.1	-0.83	0.04	3.2 %	0.03 %
0.14	-0.84	0.04	1.5 %	0.02 %
0.22	-1.0	0.04	20 %	0.3 %
0.31	-0.87	0.04	2.1 %	0.4 %
0.44	-0.78	0.04	8.6 %	0.2 %
0.70	-0.77	0.04	10.3 %	0.8 %
1	-0.77	0.04	10.6 %	2.5 %
1.41	-0.76	0.05	10.9 %	23 %
2.23	-0.73	-0.1	15 %	288 %

Table 7: Table of results using a Gaussian noise with a constant motion ($t_x = 1.5, t_y = 1.3, w_z = 0.04$), and $\phi = 1$, $\psi = 2$.

Figure (22) shows us that addition of noise perturb the normals and figure (23) that the errors do not depend on the location of the point, as expected.

2.6.8 Using a non-planar object.

In the the ideal case, see figures (24) and (25), and with gaussian noise ($\sigma = 0.32$ pixel), see figures (26) and (27).

Conclusion It is clear that the algorithm tolerates a reasonable amount of uncertainty in the feature localization. It seems however important, as very often for such methods, to be able to extract features with a subpixel accuracy. A level of precision of 0.3 pixel seems to be an upper limit to obtain very precise evaluations, i.e. better than 5 %.

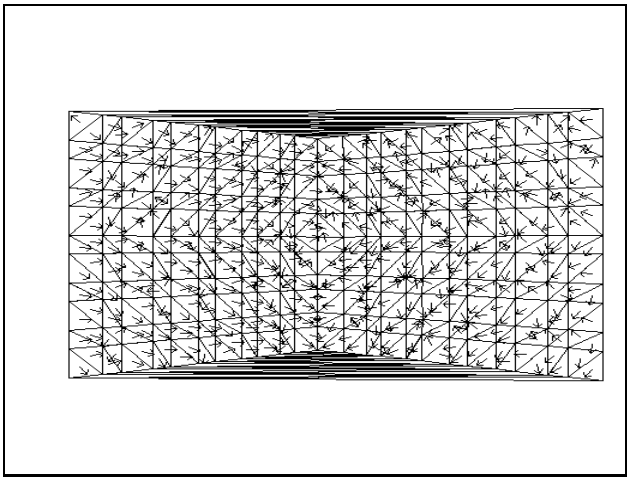

Figure 22: Normals (x 10) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis with noise of variance 0.1 pixels^2.

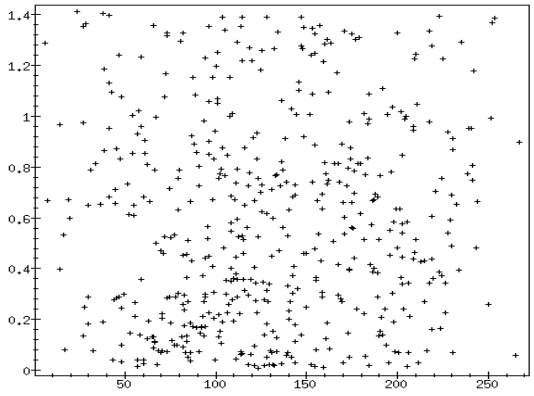

Figure 23: Errors on normals for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis with noise of $\sigma = 0.32$ pixel, related to the distance from the optical center (in pixels).

As a **conclusion**, in order to obtain a precision of 10 % on the normals, a precision of 0.5 to 2 pixels on the features is to be achieved, which is a common precision for such early vision modules. For a noise up to 1 pixel, the estimation of the displacement is quite reliable (of about 2 %).

Furthermore, it has been verified as for other authors, that the precision on the affine optical flow is very similar to the precision on the pixel location.

We thus can carry on using our method event if we are in the presence of noise.

Let us make use of these results now.

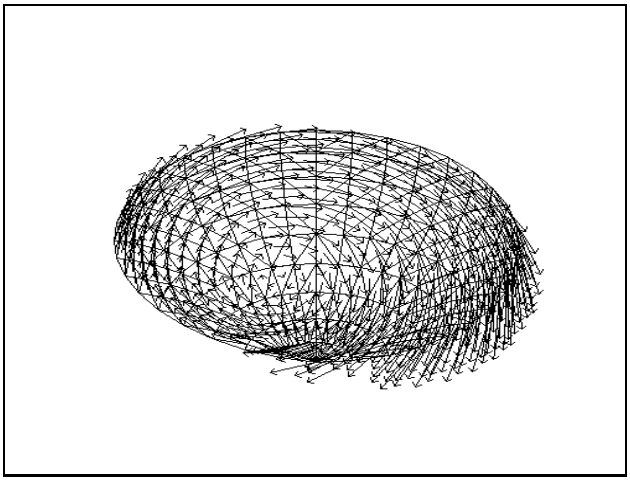

Figure 24: Optical flow (in pixels x 10) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis without noise.

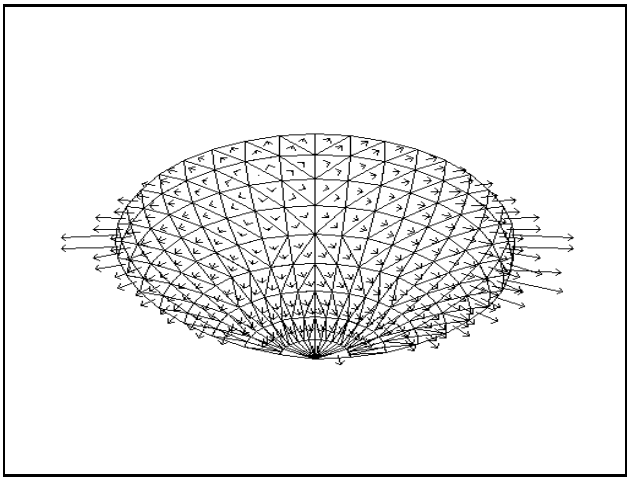

Figure 25: Normals (x 10) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis without noise.

3 Surface reconstruction.

3.1 From surface normal field to surface equation.

Indetermination in the reconstruction From the previous section, it is clear that, using equation (22) we can compute N_x and N_y up to a global scale factor as soon as the estimation of either w_x or w_y is not zero. These informations should help us to reconstruct the scene structure, i.e. *the 3D surface* of the objects in front of the camera.

Furthermore, from the previous section, it is clear that, N_x and N_y up to a global scale factor constitute *all that can be recovered from the scene structure*, i.e. :

$$(f\,N_x, f\,N_y, 1)^T \equiv \left(f\,\frac{\partial s(\mathbf{M})}{\partial X}, f\,\frac{\partial s(\mathbf{M})}{\partial Y}, \frac{\partial s(\mathbf{M})}{\partial Z} \right)^T \tag{42}$$

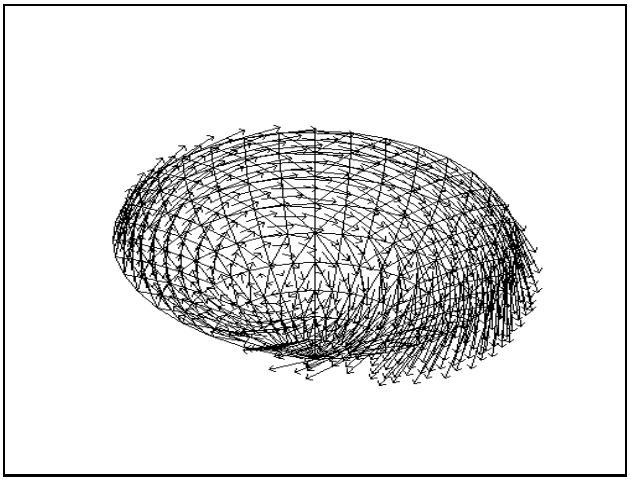

Figure 26: Optical flow (in pixels x 10) for a translation of 1.5 pixels in x, 1.3 pixels on y and a rotation of 0.04 rad. around the z-axis with noise of $\sigma = 0.32$ pixel.

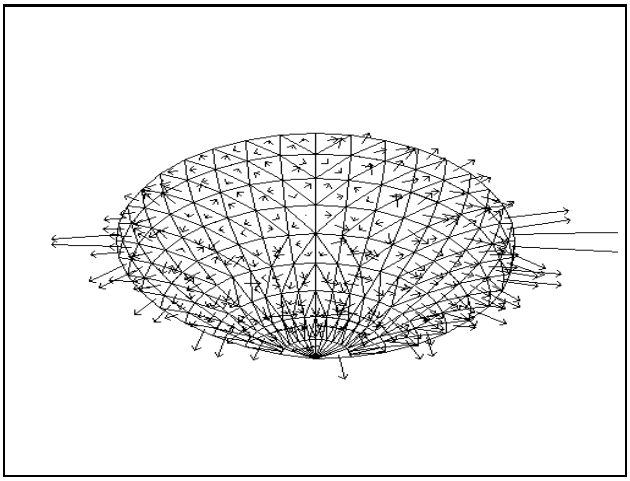

Figure 27: Normals (x 10) for a translation of 1.5 pixels in x, 1.3 pixels in y and a rotation of 0.04 rad. around the z-axis with noise of $\sigma = 0.32$ pixel.

where f is an unknown global scale factor and $s(\mathbf{M}) = 0$ the surface equation.

As a consequence, we can not recover the surface location, but its derivative with a scale factor determination parameterized by f. On the reverse, having the surface normal for each point, we can "integrate" this field and recover the surface up to a fixed transformation of the space.

This double indetermination can be formalized as follows. If we consider:

1. the translation $\tau : M \rightarrow M + (u_0/f, v_0/f, -Z_0)^T$ and
2. the expansion $\phi : M \rightarrow \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot M$

they define a linear constant transform :

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \overset{\phi \circ \tau}{\rightarrow} \begin{pmatrix} u \\ v \\ z \end{pmatrix}$$

from the retinal coordinates to the camera coordinates, which corresponds to equations (2) and (4), and *the scene structure can be recovered up to this linear transform.*

- The fact we have this translation indetermination is due to the fact that we have chosen a frame of reference attached to the optical center who's location is not known with respect to the scene.
- The fact we have this factor of expansion indetermination is due to the fact that we are using monocular cues, and have thus no way of recovering the absolute scale of the scene, which corresponds -with an affine model of the camera- to a scale factor indetermination along the Z-axis.

Through this transform, we can parameterize locally the surface using(u, v) and we have $N_u = f\, N_x$ and $N_v = f\, N_y$. Consequently, *since N_x and N_y are defined up to a scale factor we will not distinguish them from N_u and N_v in the sequel.*

Using a frame of reference attached to the retina. Considering a frame of reference attached to the retina. We use (u, v, z) as space coordinates for point M.

We define the retinal depth $d(u, v)$ of point $M(u, v)$ such that :

$$\mathbf{M}(u, v) = d(u, v) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \tag{43}$$

We thus parameterize the surface by (u, v, z), and have :

$$M \in \mathcal{S} \Leftrightarrow s(u, v, z) = -z + d(u, v) = 0 \tag{44}$$

This retinal depth is defined, from (6), by :

$$d(u, v) = f^{\bullet} \underbrace{[N_x\, u + N_y\, v]}_{\delta(u,v)} + Z_0^k \tag{45}$$

where $f^{\bullet}$ is a global undeterminated factor, whereas Z_0^k depends on the local plane equation, for a planar patch of index k.

Note that, although we use a local planar model of the surface, we now assume that the surface is not necessarily a plane but simply a regular surface $\mathcal{S}$.

3.2 Performing the reconstruction from the retinal displacement.

Constraints from the Delaunay triangulation. Let us consider now the problem of reconstructing a coherent surface from the estimated retinal displacement.

Considering the Delaunay triangulation, our surface is now defined by a set of triangular facets, and all we have to do is (i) either recover the planar equation of these facets, as defined in equation (6), or, (ii) compute the depth of the three vertices of the facet.

However, in order to build a coherent surface from this data, we have to introduce a new aspect of the problem : *not all set of orientations are coherent with a real rigid surface.* In order to figure out this last point, the reader just has to remember that in the continuous case, a field of normal $\mathbf{N}(u,v)$ can be integrated in terms of a depth map $d(u,v)$ such that $\mathbf{N}(u,v) = \vec{grad}(d(u,v))$ if and only if the rotational $\vec{rot}(\mathbf{N}(u,v)) = 0$. Obviously this constraint will be translated in a very different form in our case.

Considering two consecutive facets a and b as shown figure 28, we consider three cases:

1. The two facets do not correspond to the same object, i.e. there is a discontinuity in depth. In this case we must not relate the two plane equations. One other possibility is that a facet corresponds to a non-physical surface, or is a spurious element of data.

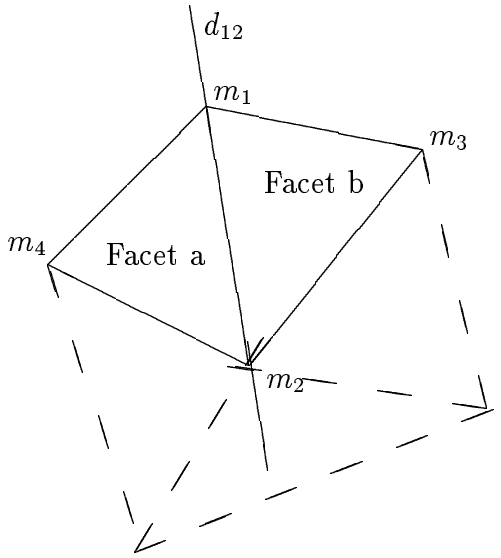

Figure 28: Continuity constraint between two facets.

2. The two facets correspond to the same non-planar object. In this case we require that for each point of the edge d_{12} the depth computed from the facet a equals the depth computed from the facet b. Since this is a line, it is sufficient to take two points, for instance the two vertices m_1 and m_2. So that the two facets correspond to the same non-planar object if and only if the depth of the two vertices are equal.

3. The two facets correspond to the same planar object. In this last case the constraint must simply either state that both facet equations are the same or that the four points of the two consecutive facets are coplanar.

Moreover, as represented in figure (28), it appears that the different facets are related to each others by a complex set of constraints depending on the topology of the Delaunay triangulation. A vertex can be common to several facets.

The theory [20] tells us that, a vertex belongs to 6 facets, in average, and at least 3. However, it can depend much on the configuration of the points. When at the "border" of the triangulation this property still holds if we consider the triangles with an infinite point. If we do not consider these triangles, we have, in the general case, two facets for a vertex, but, again, it could be one or more.

Fusion of reconstructions along an image sequence. The problem of fusing different views along an image sequence becomes trivial as soon as we verify the constraints of equation (32). In such a case, we only perform a rotation of the retinal plane, and a translation parallel to the retinal plane, so that *the depth remains constant from one frame to another.*

First, we wanted to fusion the depths Z but it is not possible in the case of real data since the quantity N_x and N_y are known up to a scale factor which depend an α_u and $||\mathbf{t}||$, which is not constant along an image sequence.

So, we decided to fusion the components N_x and N_y of the normals and to use a Kalman-filtering (very trivial in this case). We note α the measure with the associated variance V_α, $\bar{\alpha}$ the old measure with the associated variance $V_{\bar{\alpha}}$, and the result of the fusion α^* with the associated variance V_{α^*}.

$$\left\{ \begin{array}{lcl} V_{\alpha^*}^{-1} & = & V_\alpha^{-1} + V_{\bar{\alpha}}^{-1} \\ \alpha^* & = & V\alpha^* \, [V_\alpha^{-1} \, \alpha + V_{\bar{\alpha}}^{-1} \, \bar{\alpha}] \end{array} \right. \tag{46}$$

This result corresponds to a first order adaptive low-pass filter.

The measures used in our case are $N_x^2 + N_y^2$ and $\alpha = arctan(\frac{N_y}{N_x})$ which is parameterized by $\cos(\alpha)$ and $\sin(\alpha)$.

Introducing regularization. Moreover, we require the depth not to vary too much for consecutive points in order to maintain the continuity in the depths map. More precisely, even if we are in the case of a discontinuity, or if there is an occlusion, we can assume the continuity of the depth map.

Furthermore, if we do not have any other information on two depths d^{h_1} and d^{h_2}, a conservative constraint is to assume that the scene is, by default, a fronto-parallel structure, i.e. $d^{h_1} - d^{h_2} \simeq 0$.

In the presence of noise, we might maintain the idea that the depth must not vary too much, and have a "regularization" factor which allows the depth to vary considering the measure if and only if the measure is reliable.

All these ideas will be implemented as reported now.

3.3 An algorithm to perform the reconstruction.

Defining the notations. Let us translate the previous discussion in an algorithmic framework. We need the following notations :

- We consider $K = \{1, 2, \ldots, k_{max}\}$ a set of indices for the facets, and $H = \{1, 2, \ldots, h_{max}\}$ a set of indices for the vertices.
- For a given vertex of index h, we write K_h the set of facets which vertex belongs to.
- For a given facet of index k, we write H_k the set of the three vertices which define the facet.
- Now, considering equation (45), we define :

$$\begin{array}{rcccl} d^h & = & d(u^h, v^h) & = & f^\bullet\ \delta^{hk} + Z_0^k \\ \delta^{hk} & = & \frac{1}{f^\bullet}\left[d(u^h, v^h) - Z_0^k\right] & = & N_x^k\ u^h + N_y^k\ v^h \end{array} \tag{47}$$

  where

  - d^h is the *depth* of the point and
  - δ^{hk} is the *normalized depth* as defined by considering that the vertex of index h belongs to the facet of index k.

- We write

  - ϖ_{d^h} the inverse of the variance of the depth d^h
  - ϖ_k the inverse of the variance associated to a facet of index k

Choosing a criterion. In order to eliminate the unknown Z_0^k of each facet we use :

$$(d^{h_1} - d^{h_2}) = f^\bullet\ (\delta^{h_1 k} - \delta^{h_2 k}) \tag{48}$$

as measurement equation. If $f = 0$, we only require the two depths to be equal, or else, we estimate some variation in depth, as discussed before.

In order to overcome the global offset indetermination for the depth we introduce the following linear constraint :

$$\sum_{h \in H} d^h = card(H) Z_0^\bullet \tag{49}$$

We thus can integrate all these pieces of information in a comprehensive criterion:

$$\mathcal{N} = \frac{1}{2} \sum_{k \in K} \varpi_k \sum_{\{h_1, h_2\} \in H_k, h_1 > h_2} \left[(d^{h_1} - d^{h_2}) - f^\bullet\ (\delta^{h_1 k} - \delta^{h_2 k})\right]^2 \tag{50}$$

minimized with respect to the depths d^h, parameterized by f and constrained by equation (49).

Deriving the equations. The normal equations of this quadratic criterion can be written :

$$\frac{\partial \mathcal{N}}{\partial d^l} = 0 \Rightarrow d^l = \frac{1}{2} \sum_{k \in K_l} \varpi_k \left[\sum_{k \in K_l} \varpi_k \left[\sum_{h \in H_k - \{l\}} d^h - f\left(\delta^{hk} - \delta^{lk}\right) \right] \right] \tag{51}$$

which provides an iterative schema to estimate the set $\{d_l\}_{1 \ldots card(H)}$ and from which we can estimate the inverse of the variance :

$$\varpi_{d^l} = 2 \sum_{k \in K_l} \varpi_k \tag{52}$$

at the end of the process.

The constraint (49) can be satisfied by correcting the depths at each steps with :

$$d^l \leftarrow d^l + Z_0^\bullet - \frac{\sum_{h \in H} d^h}{card(H)} \tag{53}$$

An initial value of the depth can be obtained from three sources:

- an average value for Z_0^k, say $Z_0^\bullet$
- an estimation using N_x and N_y computed by propagation (we begin with a vertex which we assign the depth to be equal to $Z_0^\bullet$, compute the depths of neighbors considering $d^l = d^h + f^\bullet\left(\delta^{lk} - \delta^{hk}\right)$ and we mark the vertices with depth known) up to an indetermination on the first depth that we calculate.

Description of the algorithm. Combining the previous equations leads to the following algorithm :

- *Algorithm input:*
  - Any value for $Z_0^\bullet$, say 1.
  - The inverse of the variances for each facets, computed from a previous module.
- *Algorithm steps:*

  Iteratively adjust the depth using equation (51) and (53) at each step, until convergence[1].
- *Algorithm output:*

  The estimation of the depths and the inverse of their related variance.

3.4 Experimentation.

We have first experimented this algorithm on a sequence of a calibration grid (288 points), second on a sequence of a quarter of a sphere (181 points). The motion considered here is: $t_x = 15$ pixels, $t_y = 13$ pixels and $t_z = 0$ pixel, while $w_x = w_y = w_z = 0$ radian.

[1] The convergence is guaranteed by the fact that the series defined by equations (51) and (53) are contracting, thus convergent, as we can easily demonstrate.

Choosing an initialization for the depths. Concerning, the initialization of the depths at the beginning of the algorithm, we try two kinds of initialization in the case of the previous depths are not defined :

- $d_l = Z_0^\bullet$ for all vertices
- we take a vertex randomly which we assign the depth $Z_0^\bullet$, and compute iteratively the depths of the neighbors considering $d^l = d^h + f^\bullet(\delta^{lk} - \delta^{hk})$

But, the choice of the initialization do not influence the result of the algorithm in the case of synthetic data without noise. It has only a consequence on the number of iterations in the algorithm. For example, for a sequence of a quarter of a sphere, with the first initialization, we need 45 iterations, and 14 with the second initialization.

We can see the reconstruction of the two synthetic scenes from two views, without noise, on figures (29) and (30) for a grid of calibration, (31) and (32) for a quarter of a sphere.

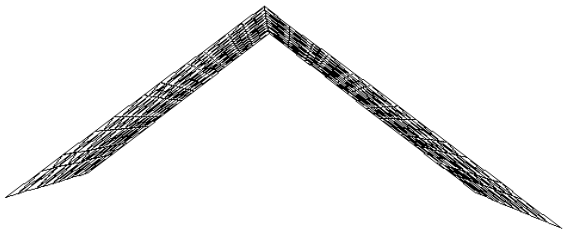

Figure 29: Top view of the 3d-reconstruction of a synthetic grid without noise.

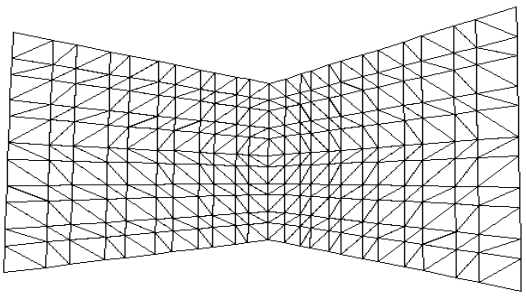

Figure 30: Front view of the 3d-reconstruction of a synthetic grid without noise.

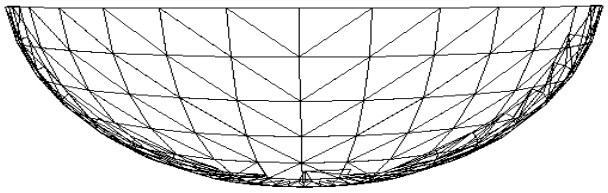

Figure 31: Top view of the 3d-reconstruction of a synthetic quarter of a sphere without noise.

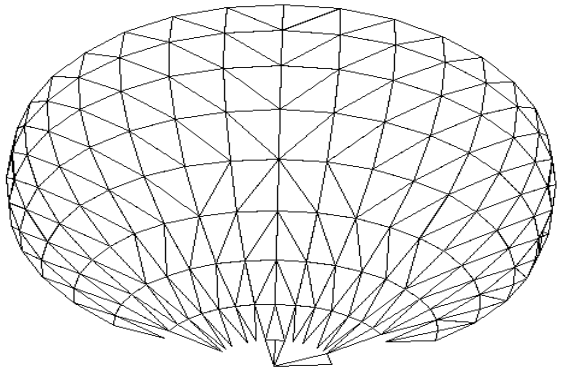

Figure 32: Front view of the 3d-reconstruction of a synthetic quarter of a sphere without noise.

Choosing an initialization for $f^\bullet$. In the previous paragraph, we took as initial value $f^\bullet = 1$. In this paragraph, we took some other values to see the influence of this parameter. We can see the results on figure (33) for the scene of a grid

Introducing of noise. We introduce again a Gaussian noise on the 2D-projected points in all images. The goal that we expected is to fusion the results of several views to reconstruct the scene with more precision. One track to follow here is to introduce the errors of the matching points of each triangle on the inverse of variance ϖ_k in the following form :

$$\frac{1}{1+\mu\,\epsilon^\nu} \tag{54}$$

as we will check in a near future.

We can see the results with gaussian noise ($\sigma = 0.32$) in figures (34), (35), (36), (37), (38), (39),(40) and (41) without considering the terms of errors and using the following motion

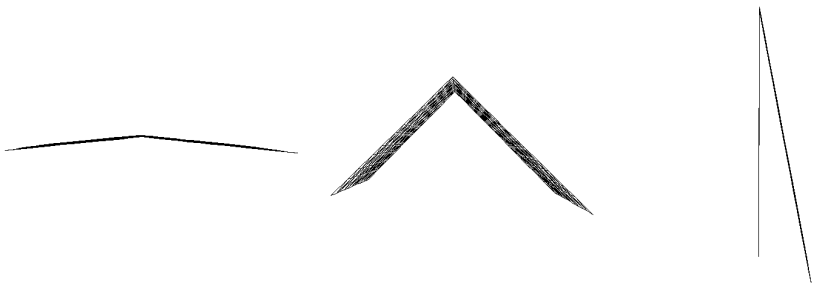

Figure 33: Top views of the 3d-reconstruction of a synthetic grid, without noise, and with an initial value of $f^\bullet = 0.1; 1; 10$.

between two consecutive frames : $t_x = 15$ pixels, $t_y = 13$ pixels and $t_z = 0$ pixel, while $w_x = w_y = w_z = 0$ radians.

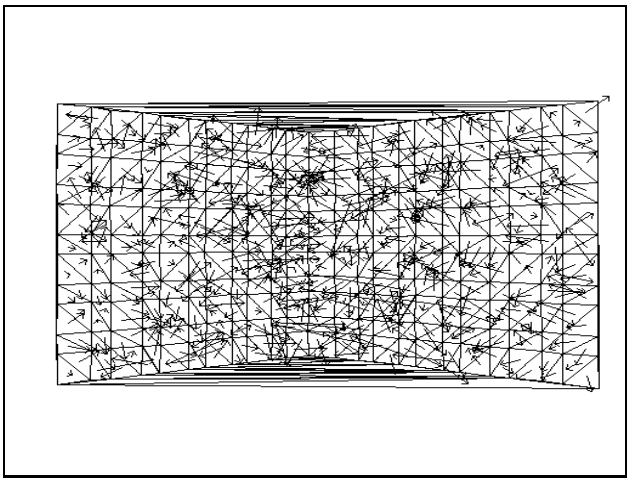

Figure 34: Normals (x 10) for a translation of 15 pixels in x, 13 pixels in y with noise of σ= 0.32 pixel, and considering the 2 first frames in the sequence.

Using real data. We also have reconstructed a real scene composed by the grid of calibration and two planes from a real sequence of 15 views (see figure (42)).

We propose two views of the reconstructed scene : the top view on figure (43), the bottom view in figure (45) and the front view in figure (44).

In order to visualize the results, we can map the texture of the first view on these triangles, knowing that the intensity on the 3d-point (x, y, z) is the intensity on the point (x, y) on the first view.

The result is given in figures (46) and (47).

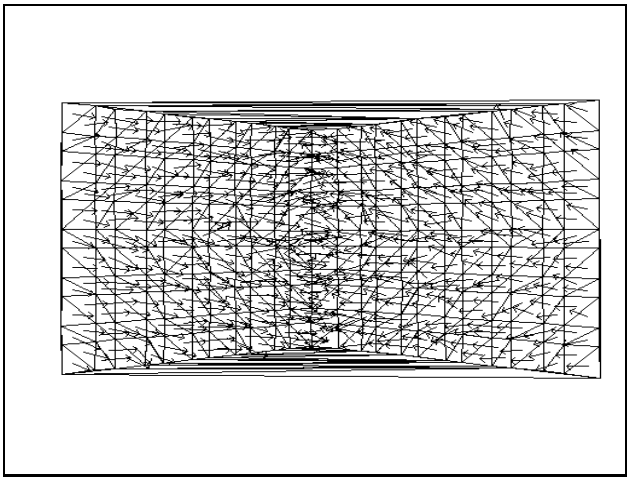

Figure 35: Normals (x 10) for a translation of 15 pixels in x, 13 pixels in y with noise of σ= 0.32 pixel, and considering the 30 frames in the sequence.

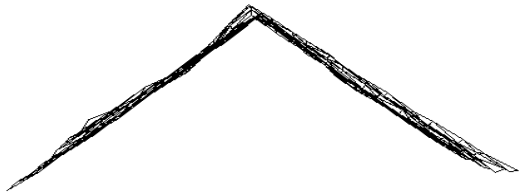

Figure 36: Top view of the 3d-reconstruction of a synthetic grid with noise ($\sigma = 0.32$).

4 Conclusion

We have demonstrated that by using a specific class of displacement, the simple orthographic and affine models presented here are valid, and we can very easily reconstruct the observed scene, in the uncalibrated case. In the case where this motion is approximate, i.e. only partially verifies the required constraints, the model is still valid close to the retina.

Furthermore, we can control if the assumptions are correct, since the validity of the affine model can be verified on the data, as noticed in our developments.

At an experimental level, a small implementation taking an image sequence as input, allows us to compute the predefined motion fields and calculate the reconstruction up to a particular affine transform of the scene.

Finally, it has been demonstrated that it is always possible to determine how to generate such specific displacements, from the error signals output by our estimation algorithms.

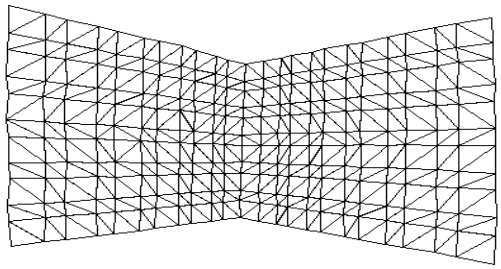

Figure 37: Frontview of the 3d-reconstruction of a synthetic grid with noise ($\sigma = 0.32$).

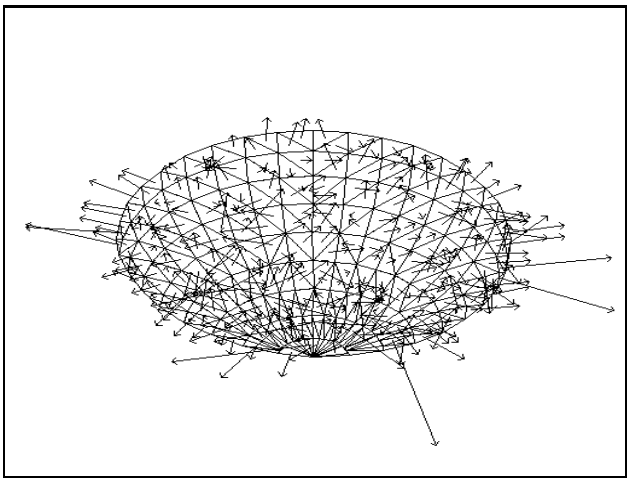

Figure 38: Normals (x 10) for a translation of 15 pixels in x, 13 pixels in y with noise of σ = 0.32 pixel, and considering the 2 first frames in the sequence.

References

[1] B. Boufama, D. Weinshall, and M. Werman. Shape from motion algorithms: a comparative analysis of scaled orthography and perspective. In Eklundh [8], pages 199–204.

[2] S. Boukir. *Reconstruction 3D d'un environnement statique par vision active.* PhD thesis, University of Rennes, Dept of Signal Processing and Telecommunications, 1993.

[3] P. Brand, R. Mohr, and P. Bobet. Distorsions optiques : correction dans un modèle projectif. Technical Report 1933, LIFIA–INRIA Rhône-Alpes, 1993.

[4] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[5] F. Chaumette and S. Boukir. Structure from motion using an active vision paradigm. In *11th Int. Conf. on Pattern Recognition, The Hague, Netherlands*, 1991.

[6] S. Christy and R. Horaud. Euclidian shape and motion from multiple perspective views by affine iterations. Technical Report 2421, INRIA Rhones-Alpes, Dec. 1994.

[7] O. Devillers, S. Meiser, and M. Teillaud. Fully dynamic Delaunay triangulation in logarithmic expected time per operation. *Comput. Geom. Theory Appl.*, 2(2):55–80, 1992.

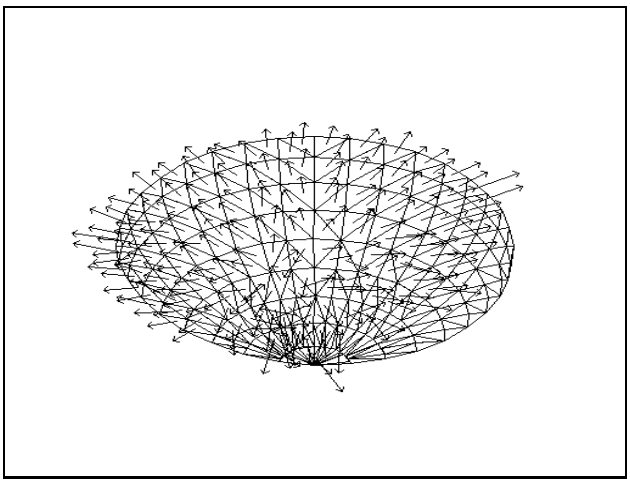

Figure 39: Normals (x 10) for a translation of 15 pixels in x, 13 pixels in y with noise of σ = 0.32 pixel, and the 30 frames in the sequence.

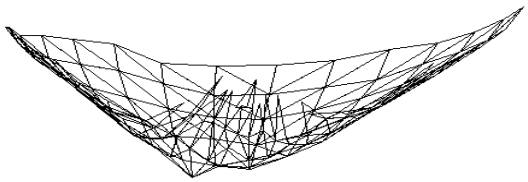

Figure 40: Top view of the 3d-reconstruction of a synthetic quarter of a sphere with noise ($\sigma = 0.32$).

[8] J.-O. Eklundh, editor. volume 800-801 of *Lecture Notes in Computer Science*, Stockholm, Sweden, May 1994. Springer-Verlag.

[9] R. Enciso, T. Viéville, and O. Faugeras. Approximation du changement de focale et de mise au point par une transformation affine à trois paramètres. *Traitement du Signal*, 11(5), 1994.

[10] O. Faugeras. *Three-Dimensional Computer Vision: a Geometric Viewpoint.* The MIT Press, 1993.

[11] O. Faugeras and S. Laveau. Representing three-dimensional data as a collection of images and fundamental matrices for image synthesis. In *Proceedings of the International Conference on Pattern Recognition*, pages 689–691, Jerusalem, Israel, Oct. 1994. Computer Society Press.

[12] O. Faugeras and G. Toscani. Camera Calibration for 3D Computer Vision. In *International Workshop on Machine Vision and Machine Intelligence*, pages 240–247, Tokyo, Feb. 1987.

[13] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conf.*, pages 189–192, 1988.

[14] R. I. Hartley and R. Gupta. Computing matched-epipolar projections. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 549–555, New-York, NY, June 1993. IEEE Computer Society, IEEE.

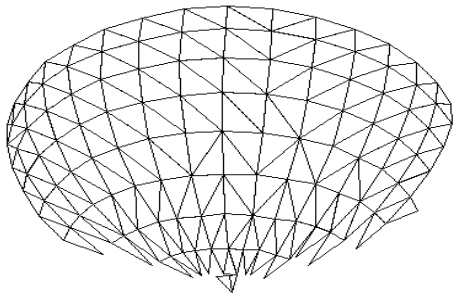

Figure 41: Front view of the 3d-reconstruction of a synthetic quarter of a sphere with noise ($\sigma = 0.32$).

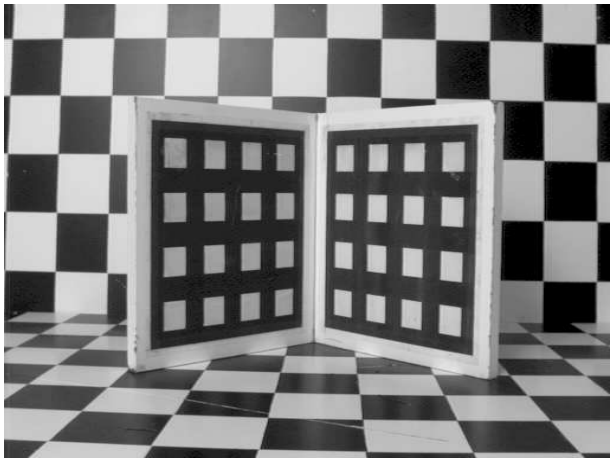

Figure 42: Real scene with a grid of calibration.

[15] J. J. Koenderink and A. J. van Doorn. Affine Structure from Motion. *Journal of the Optical Society of America*, A8:377–385, 1991.

[16] J. Lavest, G. Rives, and M. Dhome. 3-D reconstruction by zooming. *IEEE Trans. on Robotics and Automation*, 9(2):196–207, Apr. 1993.

[17] H. Longuet-Higgins. The visual ambiguity of a moving plane. In *Proceedings of the Royal Society of London*, number 223, pages 165–175, 1984.

[18] Q.-T. Luong and T. Viéville. Canonic representations for the geometries of multiple projective views. In Eklundh [8], pages 589–599, Vol. 1.

[19] R. Mohr, B. Boufama, and P. Brand. Accurate projective reconstruction. In J. Mundy and A. Zisserman, editors, *Applications of Invariance in Computer Vision*, volume 825 of *Lecture Notes in Computer Science*, pages 257–276, Berlin, 1993. Springer-Verlag.

[20] F. Preparata and M. Shamos. *Computational Geometry*. Springer-Verlag, New-York, 1985.

[21] K. A. Tarabanis, P. K. Allen, and R. Y. Tsai. A survey of sensor planning in computer vision. *IEEE Transactions on Robotics and Automation*, 11(1):86–104, Feb. 1995.

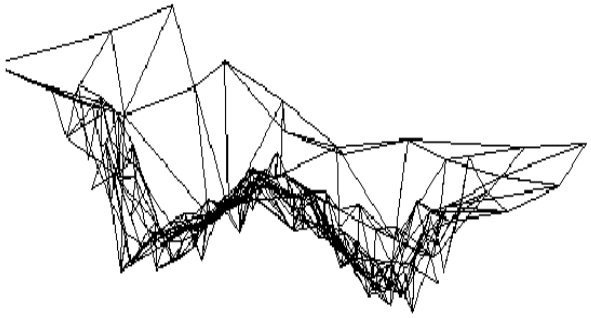

Figure 43: Top view of the 3d-reconstruction of a real scene.

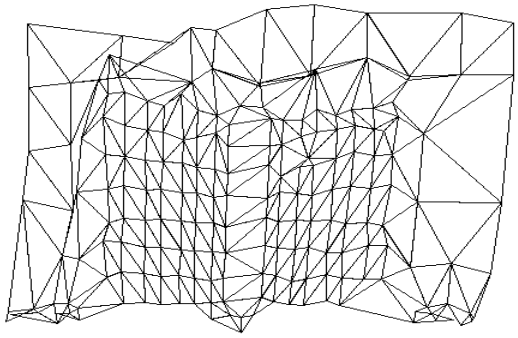

Figure 44: Front view of the 3d-reconstruction of a real scene.

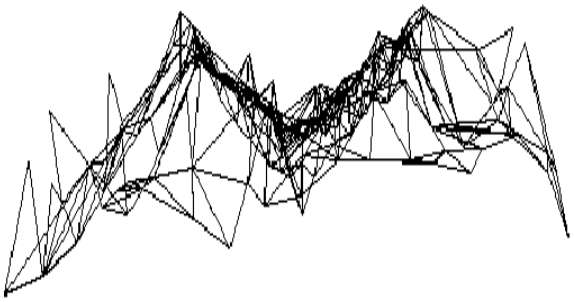

Figure 45: Bottom view of the 3d-reconstruction of a real scene.

[22] R. Tsai. Synopsis of recent progress on camera calibration for 3D machine vision. In O. Khatib, J. J. Craig, and

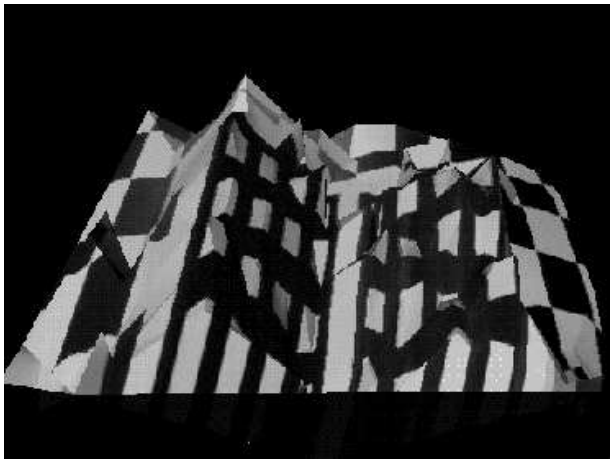

Figure 46: Bottom view of the 3d-reconstruction of a real scene, with texture mapping.

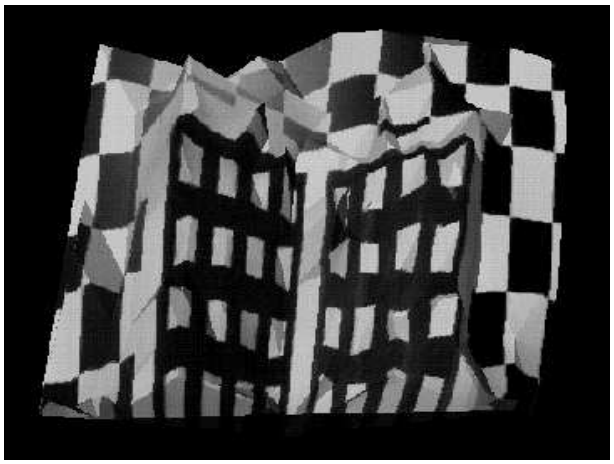

Figure 47: Front view of the 3d-reconstruction of a real scene, with texture mapping.

T. Lozano-Pérez, editors, *The Robotics Review*, pages 147–159. MIT Press, 1989.

[23] R. Tsai, T. Huang, and W. Zhu. Estimating three-dimensional motion parameters of a rigid planar patch, ii: singular value decomposition. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 30(4):525–534, 1982.

[24] T. Viéville, E. Clergue, R. Enciso, and H. Mathieu. Experimentating with 3d vision on a robotic head. *Robotics and Autonomous Systems*, 1995. 14(1).

[25] T. Viéville and O. Faugeras. The first order expansion of motion equations in the uncalibrated case. *Computer Vision and Image Understanding*, 1995. Accepted for publication.

[26] T. Viéville, Q. Luong, and O. Faugeras. Motion of points and lines in the uncalibrated case. *International Journal of Computer Vision*, 1995. To appear.

[27] A. M. Waxman and S. Ullman. Surface structure and three-dimensional motion from imageflow kinematics. *Int. J. of Robot. Res.*, 4, 1985.

[28] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, 1994. to appear. Also INRIA Research Report No.2273, May 1994.

A Perspective : Using active vision to improve the reconstruction.

A.1 Validating the affine model.

The previous formalism is only valid if we verify the motion constraint of equation (32). If we do not verify this constraint, equation (12) will not provide a *constant value for all facets* but will be subject to a bias.

Considering a perspective projection, this bias is an affine function of the normal of the surface of each facet, as made explicit in the following equation:

$$\begin{cases} W_y c + W_x a - W_y W_z = \\ \underbrace{\frac{W_x v_0}{f Z_0} w_x - \frac{(W_y v_0 + 2 W_x u_0)}{f Z_0} w_y - \left[\frac{N_x (W_y v_0 + 2 W_x u_0) + N_y (W_x v_0)}{f Z_0^2} + \frac{W_x}{Z_0} \right] t_z}_{\mathcal{B}_1} \\ W_x b + W_y d + W_x W_z = \\ \underbrace{\frac{(W_x u_0 + 2 W_y v_0)}{f Z_0} w_x - \frac{W_y u_0}{f Z_0} w_y - \left[\frac{N_y (W_x u_0 + 2 W_y v_0) + N_x (W_y u_0)}{f Z_0^2} + \frac{W_y}{Z_0} \right] t_z}_{\mathcal{B}_2} \end{cases} \tag{55}$$

The quantities $W_x \to t_y/Z_0, W_y \to -t_x/Z_0, W_z \to w_z$ are the estimated quantities.
The biased $\mathcal{B}_1$ and $\mathcal{B}_2$ vanish if and only if:

- either $W_x = W_y = 0$ which is a degenerated solution of equation (12),
- else if $u_0 = v_0 = t_z = 0$ while w_x and w_y are undefined, but in this particular case we are still in a situation where the affine model is valid, as visible in equation (30),
- else if $w_x = w_y = t_z = 0$ which is the required situation.

As a consequence, these bias are canceled if and only if equation (32) is verified except if the system is related to a normalized frame of reference which is easily avoided.

Moreover these bias are linear functions of the system parameters and thus act as quadratic factors in the least-square criterion. As a consequence, the minimum of this least-square criterion, *with respect to these control parameters* corresponds to zero values, and we have a convex function to minimize.

This suggest a very simple strategy to control the active visual system, and constraint the displacement to be a pure translation parallel to the retinal plane:

1. Perform any displacement.
2. Compute the parameters using equation (12) though the least-square criterion of equation (26) and calculate the residual.
3. Generate a displacement, considering previous estimates so that the residual decreases and repeat the previous step on this one until a minimum is attained.

From the previous discussion we can see that this algorithm converges if and only if the robotic system which realizes the displacements is able to perform a pure translation parallel to the retinal plane with or without a controllable or uncontrollable counter-rotation given by w_z. So that this method will be applicable on robotic systems with enough degree of freedom (robot arm, robotic head with "eye and neck", mobile robot with a turret, etc...)

A.2 Performing optimal displacements for the reconstruction.

Let us now consider the problem of computing optimal displacements for the reconstruction. It is clear from the previous paragraph that we are looking for displacements for which $w_x = w_y = t_z = 0$.

We thus have to decide on $t_x = t\ \cos(\alpha)$, $t_y = t\ \sin(\alpha)$ and w_z. Following [24] we will use the following qualitative arguments :

- We better reduce the retinal disparity between two consecutive frames in order to ease the matching process.
- We better generate retinal disparity to increase the robustness of the structure from motion equations.

These two - somehow contradictory arguments - will be used as follows in our case:

1. The counter-rotation w_z will be limited or canceled, unless it improves the reconstruction.
2. The amplitude of the translation t will be increased unless the matching performances decreases so that the performances of the reconstruction is affected.
3. The orientation of the translation α will be chosen in order to mostly increase the performance of the reconstruction.

In all cases, the strategy is to look for displacements which tend to decrease the residual of equation (50), so that these mechanisms can be treated globally as follows:

1. Perform any displacement.
2. Compute the parameters though the least-square criterion of equation (50) and calculate the residual.
3. Generate a displacement, considering previous estimates so that the residual decreases and repeat the previous step and this one until a minimum is attained.

We finally end up with a mechanism which tend to minimize conjointly two criterions, but not considering the same degrees of freedom. Let us discuss one implementation now.