T. Viéville, E. Clergue, R. Enciso, and H. Mathieu. Experimentating with 3d vision on a robotic head. Robotics and Autonomous Systems*, 1994. Vol14, No1, Jan/Feb1995.*

Experimenting with 3D Vision on a Robotic Head

Thierry Viéville, Emmanuelle Clergue, Reyes Enciso and Hervé Mathieu
INRIA, Sophia, BP93, 06902 Valbonne, France
vthierry@sophia.inria.fr

Abstract

We intend to build a vision system that will allow dynamic 3D-perception of objects of interest. More specifically, we discuss the idea of using 3D visual cues when tracking a visual target, in order to recover some of its 3D characteristics (depth, size, kinematic information). The basic requirements for such a 3D vision module to be embedded on a robotic head are discussed.

The experimentation reported here corresponds to an implementation of these general ideas, considering a calibrated robotic head. We analyse how to make use of such a system for (1) detecting 3D-objects of interest, (2) recovering the average depth and size of the tracked objects, (3) fixating and tracking such objects, to facilitate their observation.

Keywords : 3D Tracking, Structure and Motion, Robotic Head.

1 Introduction

The idea that all vision algorithms can be improved by performing visuo-motor tasks with an adaptive and mobile visual sensor is sometimes well accepted but far from being obvious. On the contrary the increase in system complexity might be a source of degradation in terms of performance, especially for 3D vision modules [1, 2, 3]. In this paper we discuss how to avoid such a situation and how to take advantage of 3D vision in a visual loop.

Dealing with 3D visual cues on a robotic head. More precisely, we want to address the problem of having a robotic head performing object observation, considering 3D representation [1] of the scene. Most of the earlier or recent studies in the field are based on mechanisms involving only bi-dimensional representations, that is representations in which three-dimensional parameters are implicit [4, 5, 6, 7, 8, 9, 10, 11]. Obviously the relations between the target 3D depth and the visual target have already been introduced

[1] We make the distinction between visual processes which deal either with internal representation involving only 2D parameters (3D parameters being implicit) or internal representation involving also 3D parameters (Camera calibration, 3D location, 3D orientation and 3D motion) explicitly.

for the control of vergence, focus or zoom [12, 9, 13] but in this "3D representation" only one point target in the scene is taken into account, except [14]. More recently, for an eye-arm system, the control of the three-dimensional motion of the camera has been related to the 3D structure of a rather complex object under observation [15, 16], but this result has not yet been extended to the case of robotic heads.

The use of head movements to help solving stereo correspondences has also been recently considered [17] but the precision of the mechanics for such a paradigm to be explored is, according to the authors, far from what is obtained with actual realizations [5, 18, 9, 19] and the only system with such a very high precision is only at a development stage [20]. Therefore, if we want to experiment 3D vision on standard robotic heads, we must not consider the rigid displacement between the two images to be perfectly known as in the case of stereo paradigms, but estimate it as in the case of motion paradigms[2]. The facility in a motion paradigm is that we can assume the disparity between two frames to be small, leading to easy solutions for the correspondence problem [21, 22]. These correspondences must however be established (token-tracking).

In addition to these difficulties, a crucial, mandatory and very sensitive problem is calibration (both calibration of the robotic head and calibration of the visual sensor). Now, in the case of a robotic head vision, the extrinsic parameters and the intrinsic parameters of the visual sensor are modified dynamically. For instance, when tuning the zoom and focus of a lens, these parameters are modified and must be recomputed. It was thus a new challenge to determine dynamic calibration parameters by a simple observation of an unknown stationary scene, when performing a rigid motion as studied by [23, 24, 25, 26]. It has also been demonstrated that certain motions are more suitable to estimate a given set of calibration parameters [26], i.e., that we can perform "adaptive motions for calibration". Other alternatives, considering auto-calibration, include either to limit the visual analysis to some invariants [27, 28] or image based parameters not requiring calibration parameters [29, 9]. But we will not follow one of these tracks and - on the contrary - will show that we can rely on calibration on a robotic head.

Considering situations where we must use 3D vision. We know that, using appropriate camera movements, a reactive visual system can stabilize the image on the sensor retina [30], participate in visual exploration of the surroundings [14], or track an object in movement [31], but it can also provide a certain estimation of the ego-motion, and of the structure and motion of certain parts of the visual neighborhood [16, 9].

Many visual tasks such as target tracking, image stabilization, motion detection, beacon detection, object recognition, etc. do not require 3D visual cues, whereas 2D (or image based) cues are sufficient and often more robust and efficient. In such frameworks [16, 9, 10, 11], the 3D parameters are only implicit.

On the contrary, when introducing 3D parameters in a visual representation, we must be aware that, as usual when increasing the dimension of the state of a system, computation time will increase, system observability and stability might be affected and the estimation is likely to be less robust.

But on the other hand, *when is it mandatory to introduce 3D parameters in the internal representation of a visual system ?* The answer is "when obtaining 3D data is the goal of the visual system". It is very easy to list many actions in which 3D vision must be introduced : knowing the exact value of the focus, calibrating the vergence of a

[2]In such a framework, since 3D information is available for each camera, the stereo problem is to be attacked as a "sensor fusion problem."

binocular system, calibrating the system's visual metric to find the relationship between the angular position of the mount and the retinal displacement in pixel when a translation occurs, computing relative depth, evaluating the depth of an object, positioning with respect to a 3D map, etc. Related applications are : vehicle positioning and manoeuvre, object observation and recognition, moveable or fixed obstacle avoidance, 3D mapping for surveillance, etc. We thus must sometimes overcome the previous difficulties and also introduce 3D computation, in such systems.

Basic modules to perform 3D vision on a robotic head. Let us now discuss, what are the "mechanisms" required for a robotic head, to support 3D-visual modules ?

1. *Early vision and lens parameter tuning.* In this situation, it is not possible to use an early vision system for which one must manually adjust internal parameters (there are up to 16 such parameters in a real-time vision machine [19]), while these parametric adjustments are mandatory because they allow the system to be adaptive and usable in varying conditions (photogrammetric variations, etc.).

   In the present implementation, we make use of auto-focus, automatic adjustment of the iris, in synergy with gain and offset adjustment, as for any other system of this kind. In addition, smooth factors and contrast thresholds are also automatically adjusted, as described below.

2. *Auto-calibration of visual system intrinsic parameters.* We have already explained that, if one changes any parameters of the mount, or any lens parameters, the visual calibration is to be recomputed.

   However, in our implementation, because of the relative simplicity of the mechanical design, we avoid using such sophisticated mechanisms but have *pre-calibrated* the mount in a large number of possible positions and can obtain the extrinsic and intrinsic calibration parameters at any time, by interpolating the pre-recorded values. Experimental results are given below.

3. *Picture stabilization processes.*

   Picture stabilization is a crucial task when performing 3D dynamic vision. It allows the reduction of the ambiguities in the token-tracking problem, because the expected disparity between two occurrences of a token is reduced by this mechanism [32]. It also simplifies the computation of structure and motion since it can be shown, if the system is calibrated, that one can cancel the rotational disparity between two frames, either using visual or inertial cues [30, 33].

   Such mechanisms act either on the mount (control of gaze direction), or simply control the internal metric of the system (picture reprojection). Moreover, since a rotation around the optical center corresponds to a particular homographic transform of the image [23, 28], we can design "virtual" degrees of freedom, for instance eye torsion, and combine them with mechanical degrees of freedom for the control of image stabilization. In fact, using homographic transformations of the image plane for stabilization allows the system to work without calibration and compensates for complex projected motion, as discussed in [34].

   In our implementation, because (1) the system is calibrated and (2) inertial sensors computing the angular velocity of the camera are available, the rotational disparity is automatically computed in a straightforward way, using odometric cues only, and performed by reprojection.

Two basic functions when observing a 3D visual object. Within the present framework, we consider a "robotic head" made of a "stand," a "neck" performing off-centered rotations, and an "eye" performing rotations around the optical center[3]. Instantiations of such a mechanism are now very common [5, 6, 9, 19, 20]. However, whereas almost all algorithms in the field deal with target detection and tracking [4, 5, 6, 15, 16, 9, 10, 11, 31] we would like to design a more sophisticated behavior.

As explained in the previous section, when considering 3D vision as a goal, the visual strategies can be divided into two classes :

[A] : Strategies to detect visual targets to be submitted to 3D visual perception, i.e., **where to look next ?**

[B] : Strategies to maintain and improve the 3D perception of the visual targets, i.e., **how to track a visual target ?**

Let us analyse these two problems. We first develop the vision modules to realize these two functions (section 2 and 3) and then discuss how to control the vision system in order to help 3D visual perception (section 4).

2 Detecting visual targets on a calibrated mount

2.1 Using a calibrated robotic head

We consider a Euclidean frame of reference $\mathcal{M}$ attached to the mount and a 3D point $M = (X, Y, Z)^T$ in this frame of reference. In the retinal frame of reference $\mathcal{R}_i$ attached to the camera at time i, the coordinates $M_i = (X_i, Y_i, Z_i)^T$ of this point are related to the previous one by the usual equation :

$$M_i = R(\mathbf{\Theta}_i) \cdot M + \mathbf{t}(\mathbf{\Theta}_i)$$

where $R(\mathbf{\Theta}_i)$ is a 3×3 rotation matrix and matrix function and $\mathbf{t}(\mathbf{\Theta}_i)$ a 3 dimensional vector which describes the Euclidean translation between these two frames. They simply correspond to the geometric model of the robotic system carrying the cameras, as shown in figure 1.

These quantities are, of course, functions of the mount parameters : angular positions (pan, tilt, vergence, etc.), zoom, focus. We collect all these control variables in a state vector written $\mathbf{\Theta}_i$.

Since we use *the standard pinhole model* for a camera, assuming the camera performs a perfect perspective transform with center C_i (the camera optic center) but *is not calibrated*, the retinal projection $m_i = (u_i, v_i)^T$ of the point M_i, in homogeneous coordinates is, by [28, 35] :

$$\begin{pmatrix} Z_i\, u_i \\ Z_i\, v_i \\ Z_i \end{pmatrix} = \underbrace{\begin{pmatrix} \alpha_u & -\alpha_v \cot(\theta) & u_0 \\ 0 & \alpha_v / sin(\theta) & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{A(\mathbf{\Theta}_i)} \cdot \underbrace{\begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix}}_{M_i}$$

This is a standard camera model [36, 37, 38] which corresponds, for $\theta = 0$, to the usual equations $u = \alpha_u\, X_i/Z_i + u_0$, $v = \alpha_v\, Y_i/Z_i + v_0$. More precisely, α_u and α_v are the

[3]It has been established [9, 20] than one can design systems which really perform pure rotations with respect to the optical center of the camera (more precisely the object nodal point) the related errors are negligible for objects with a reasonable depth. These degrees of freedom often include vergence.

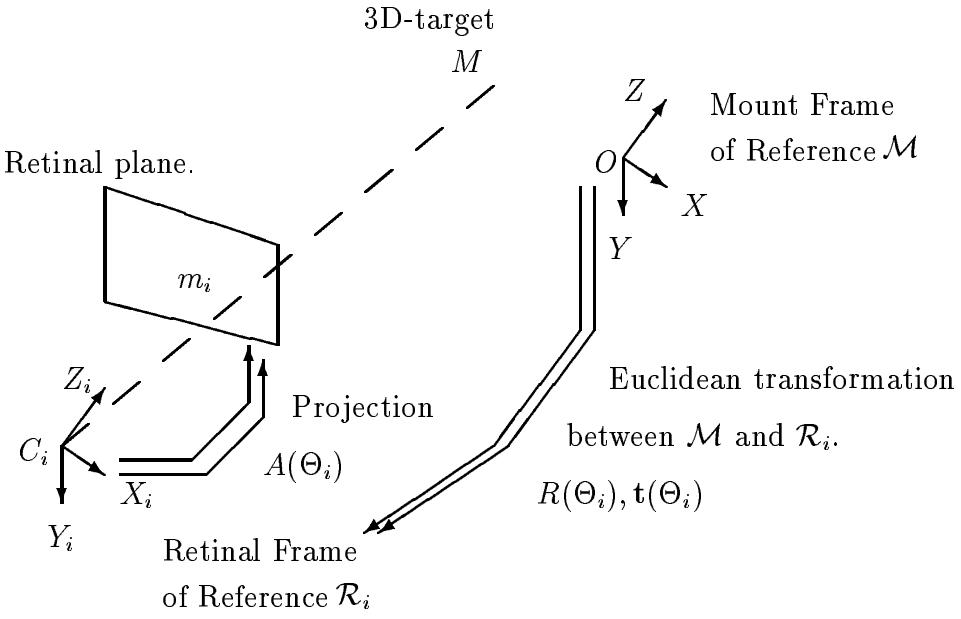

Figure 1: Notation for the geometric model of the camera projection.

horizontal and vertical magnitude factor (or scales, including the focal length) and (u_0, v_0) corresponds to the location of the optical center (principal point). Using the parameter θ we have a model of the fact that the apparent aspect of a pixel might not be rectangular [28].

Note that with this model, the scale factor Z_i is the depth of the point in the frame of reference $\mathcal{R}_i$ attached to camera $\mathcal{C}_i$, and is very close to the distance from the point to the optical center C_i.

Combining the two previous equations we can write :

$$\begin{pmatrix} Z_i\, u_i \\ Z_i\, v_i \\ Z_i \end{pmatrix} = P(\mathbf{\Theta}_i) \cdot M + \mathbf{p}(\mathbf{\Theta}_i) \tag{1}$$

where $P(\mathbf{\Theta}_i) = A(\mathbf{\Theta}_i) \cdot R(\mathbf{\Theta}_i)$ and $\mathbf{p}(\mathbf{\Theta}_i) = A(\mathbf{\Theta}_i) \cdot \mathbf{t}(\mathbf{\Theta}_i)$. The P-matrix and $\mathbf{p}$-vector are known from calibration and contain the intrinsic and extrinsic parameters of the mount and camera.

Since they are functions of the angular positioning of the mount, and of the zoom and focus configuration, the disparities induced by zooming, vergence and mount displacements are integrated in these equations.

Let us now show one example of such a calibrated system and evaluate the precision of the calibration, for this system.

2.1.1 Presentation of the INRIA robotic head

The INRIA robotic head is a device which can control the gaze direction of a binocular stereo system, and can tune the parameters of one lens [19].

Basically, the system uses a minimum number of three degrees of freedom (pan, tilt and vergence on one eye) for directing gaze without any redundancy. The mechanical relation between the three axes is known, pan and tilt correspond to two orthogonal

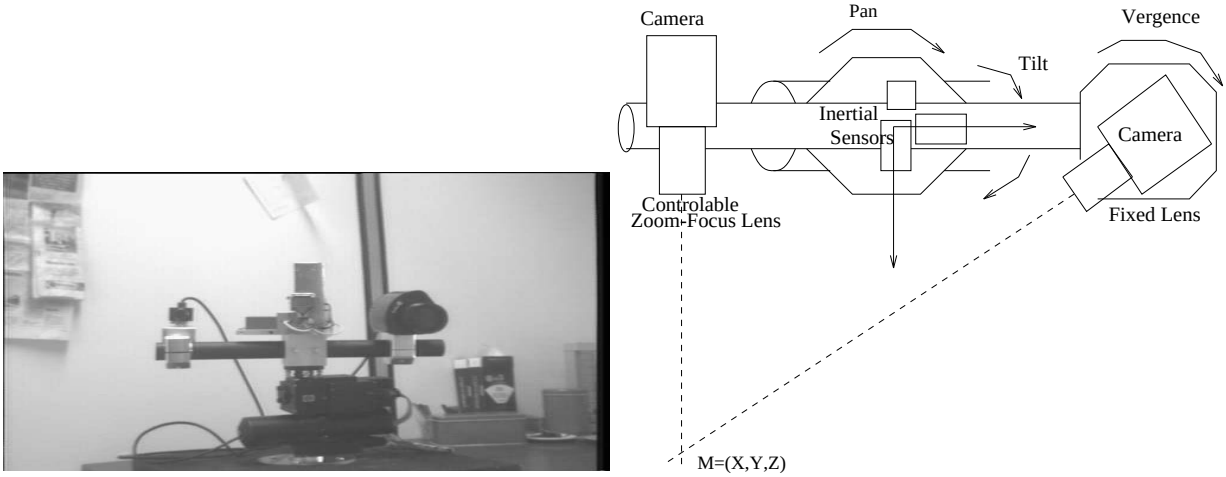

Figure 2: A view of the robotic head and its mechanical architecture

intersecting axes, while the vergence axis is aligned with the pan axis. Pan axis is on tilt. Using such a kinematic chain has two consequences : it allows a very simple computation of inverse kinematics and since we have very few joints, it increases precision positioning. Please refer to [19] for a detailed description of the mechanical principles.

The two visual sensors behave differently and they do not have the same characteristics. The foveal sensor (dominant eye), is fixed with respect to the mount and has a controlled zoom, focus, and iris. The peripheral sensor is capable of vergence. Inertial sensors are also available. This system corresponds to a mechanical instantiation of the proposed framework, and might be considered as the "simplest" robotic head on which 3D vision can be performed.

In particular we have a one to one correspondence between the joint angular positions and the fixated 3D point as detailed in the appendix. The equations are independent from the shift related to the zoom/focus of the lens. There is thus no need to calibrate this parameter at this stage. These equations also provide a direct positioning of the system towards a 3D target. It is thus possible to compute in one step the positioning of the mount. Moreover, we directly define angular joint positions, and the low-level control (feedback, local trajectory generation, etc.) is thus decoupled on each axis.

2.1.2 Head intrinsic calibration

In the first part of the experimental work, we have elaborated a simple model for the variations of the lens intrinsic parameters when the zoom and focus parameters are modified.

Our analysis has demonstrated that a linear model is sufficient to describe these modifications. Experimentally, this approximation is valid, the precision being better than 2-3 pixels in almost every case (i.e. 80 % of the obtained data). The complete set of results is available in [39].

We have obtained these numbers by a set of repetitive static calibrations using a well established interactive modern method of calibration reported in [40].

The main results are the following :

1. We have verified that the ratio $\frac{\alpha_u}{\alpha_v}$ is almost constant and equal to $k_{u/v} = 0.7$ in our case, which is the value given by the constructor. We also have verified that the pixel orthogonality is almost perfect ($\theta \simeq \frac{\Pi}{2}$). This is shown in figure 3. As a

consequence, we do not have to manipulate all five intrinsic calibration parameters, but only three of them, i.e. u_0, v_0 and α_v, while $\theta = \frac{\Pi}{2}$ and $\alpha_u = k_{u/v}\ \alpha_v$.

2. We have verified that linear models of variation of the parameters have enough accuracy, i.e., that the residual error is not predictable (due to uncertainty, not to a bias) so that there is no need for very sophisticated models of calibration. These results are in agreement with what has been found by another team in the field [41, 42] considering that our precision is about one pixel. This shown in figure 4 and figure 5.
3. These results are stable in the sense that they do not significantly vary when another experiment is done on the same system. In figure 5, results from two experimental sessions have been combined.

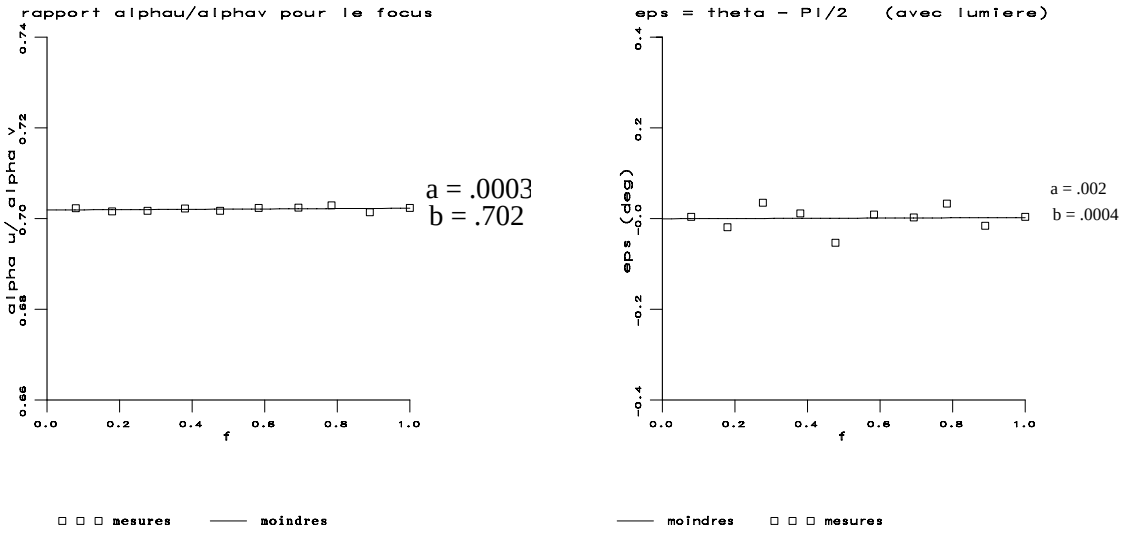

Figure 3: Stability of the $\frac{\alpha_u}{\alpha_v}$ ratio (left view) and the variation of the pixel orthogonality $\epsilon = \theta - \frac{\Pi}{2}$ in function of the focus. It is clear that these two parameters are almost constant which simplifies the calibration model. Focus variations correspond to uncalibrated values, i.e., what is measured on the lens output. Similar results have been obtained for the zoom.

2.1.3 Looking at a 3D point

We have verified the quality of the head calibration by gazing at several locations in space. We have compared the location at which the head was really gazing with what was expected. We have manually measured the 3D locations; therefore the precision of this experiment is no more that $0.5cm$.

We also have noted the retinal error induced by the uncertainty in the calibration, a sample is given in the following table :

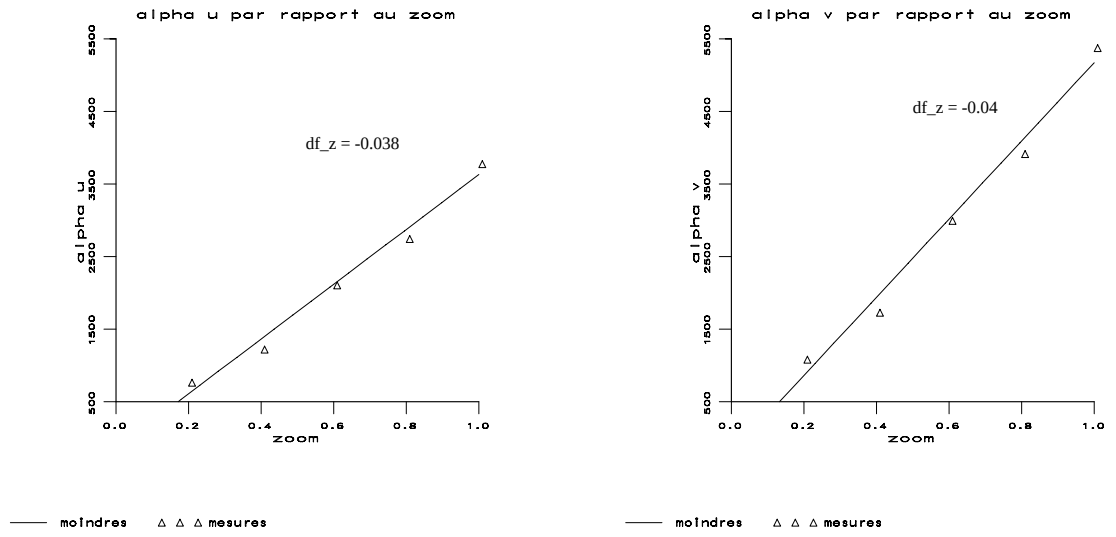

Figure 4: Variation of the horizontal (α_u) and vertical (α_v) scale factor when zooming. Zoom variations correspond to uncalibrated values, i.e., what is measured on the lens output.It is clear that a linear model is quite accurate.

Measured (X, Y, Z) (meter)	Expected (X_0, Y_0, Z_0) (meter)	Retinal Error (ϵ_u, ϵ_v) (pixel)
(0.000, 0.000, 1.062)	(0.000, 0.000, 1.000)	(0,1)
(0.112, 0.005, 1.013)	(0.100, 0.000, 1.000)	(4,2)
(-0.004, 0.098, 1.013)	(0.000, 0.100, 1.000)	(4,3)
(-0.980, 0.120, 1.998)	(-0.100, 0.100, 2.000)	(2,3)
(-0.960, 0.110, 2.898)	(-0.100, 0.100, 3.000)	(2,1)

Although rather approximate this small experiment yields two conclusions :

1. the precision of the head calibration is quite reliable, the precision being of about 1-2 cm, except for one point;
2. such uncertainty corresponds to retinal errors close to the precision of the lens calibration.

This experiment has been repeated considering another set of 20 fixations, and we have obtained an average precision better than "2cm" and "4pixels" (standard deviation).

2.2 Using image stabilization to detect area of interest

Considering attention focusing, we can very easily give a list of which objects in a scene are to be preferred when observed by a visual system :

1. Moving objects might trigger potential alarms ("prey", "predator," moving obstacles to avoid).

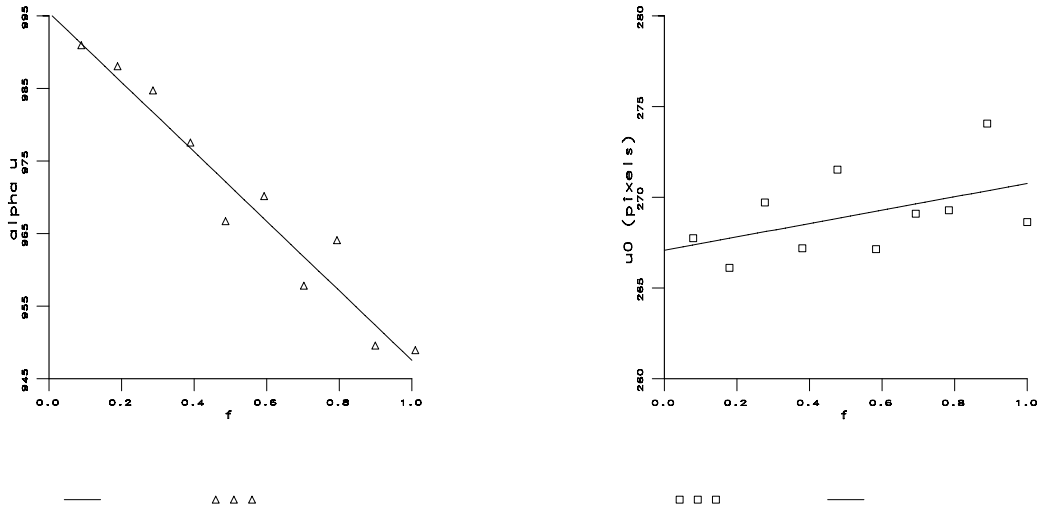

Figure 5: Stability of the horizontal intrinsic calibration parameters during a change of focus. The first trace is the horizontal scale factor and the second trace is the horizontal location of the principal point. Both traces are blown-up to observe the errors, essentially random. Focus variations correspond to uncalibrated values, i.e., what is measured on the lens output. Very similar results have been obtained for vertical parameters and during a zoom.

2. Nearby objects will be the first to interact with the system as potential obstacles. Their proximity should ease their 3D observation.

3. Objects with a high density of edges correspond to informative parts of the visual field and might be worth a closer look.

These three categories might appear as very natural; they are, but in addition, they correspond to a very precise 3D motion property : *considering that the 3D rotational disparity has been cancelled between two consecutive frames (i.e., the motion disparity between them is only induced by a translational motion), the residual disparity is only due to (1) object in motion, (2) object with a non-negligible depth* [4]*, (3) object with complex texture or shapes, likely detected with some error, and whose related disparity is not correctly detected.*

In other words, these structures correspond to points with a non-negligible residual disparity; the global rotational disparity of the scene been cancelled. The mechanism developed by [31] is implicitly based on this principle.

Let us now formalize this idea. We consider two configurations of the mount $\mathbf{\Theta}_0$ and $\mathbf{\Theta}_1$ and using equation (1), we can relate the projections $m_0 = (u_0, v_0)$ and $m_1 = (u_1, v_1)$

[4]It is well known that the retinal motion amplitude for a given target due to a translational motion is proportional to the ratio between the translation velocity and the target depth.

of a *stationary* point M with the equation :

$$Z_1 \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix} = Z_0 \underbrace{P(\mathbf{\Theta}_1) \cdot P(\mathbf{\Theta}_0)^{-1}}_{Q_{10}} \cdot \begin{pmatrix} u_0 \\ v_0 \\ 1 \end{pmatrix} + \underbrace{\mathbf{p}(\mathbf{\Theta}_1) - P(\mathbf{\Theta}_1) \cdot P(\mathbf{\Theta}_0)^{-1} \cdot \mathbf{p}(\mathbf{\Theta}_0)}_{\mathbf{s}_{10}} \quad (2)$$

The matrix $Q_{10} = A(\mathbf{\Theta}_1) \cdot R(\mathbf{\Theta}_1) \cdot R(\mathbf{\Theta}_0)^T \cdot A(\mathbf{\Theta}_0)^{-1}$ is also called the "uncalibrated rotation" and can be interpreted as the collineation of the plane at infinity [35]. It can be estimated directly from the mount parameters, as soon as it is calibrated.

We can now transform image 0 by the following transformation, called *rotational stabilization* in the sequel :

$$\begin{pmatrix} u_0 \\ v_0 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} u'_0 \\ v'_0 \\ 1 \end{pmatrix} = Q_{10} \cdot \begin{pmatrix} u_0 \\ v_0 \\ 1 \end{pmatrix}$$

which corresponds to a collineation of the retinal plane. Now considering the new transformed image of points $m'_0 = (u'_0, v'_0)$ we can write the previous relation, in a much simpler form[5] :

$$\begin{cases} u_1 = \frac{u'_0 + s^0_{10}/Z_0}{1 + s^2_{10}/Z_0} \\ v_1 = \frac{v'_0 + s^1_{10}/Z_0}{1 + s^2_{10}/Z_0} \end{cases}$$

It is now clear that if the point is stationary and has a huge depth with $1/Z_0 \simeq 0$ we have $m_1 = m'_0$, i.e., it is stabilized. On the contrary, if Z_0 is small, a residual disparity is observed. If, in addition, the point M is not stationary but mobile an additional disparity is observed.

2.2.1 Experimentation on rotational stabilization

Considering the previous discussion, using odometric cues, for our calibrated system, we cancel the rotational disparity between two frames. More generally, the rotational displacement is measurable using inertial and odometric cues [33, 30], and can also be obtained from visual cues [35]. The input is two consecutive frames, the rotational displacement between those two frames, and the intrinsic calibration parameters of these two frames. The output is two consecutive frames, the previous frame being stabilized with respect to the present one.

We have shown a few examples in figures 6, 7 and 8, illustrating some applications of this mechanism, considering different visual tasks.

2.3 Auto-tuned early-vision : fast region detector

If we want to go a step further, we have to analyse each image and try to detect which region could be considered as a potential target. What we want here is a *coarse but dense* analysis of the image (we do not want to "forget" a visual alarm even if our measure is approximate). We thus detect areas of homogeneous intensity. Using a very rapid algorithm of region segmentation, the toboggan method [43], we group and extract regions

[5] We represent the components of a matrix or a vector, such as $\mathbf{s}_{10}$, using upper subscripts from 0 to 2.

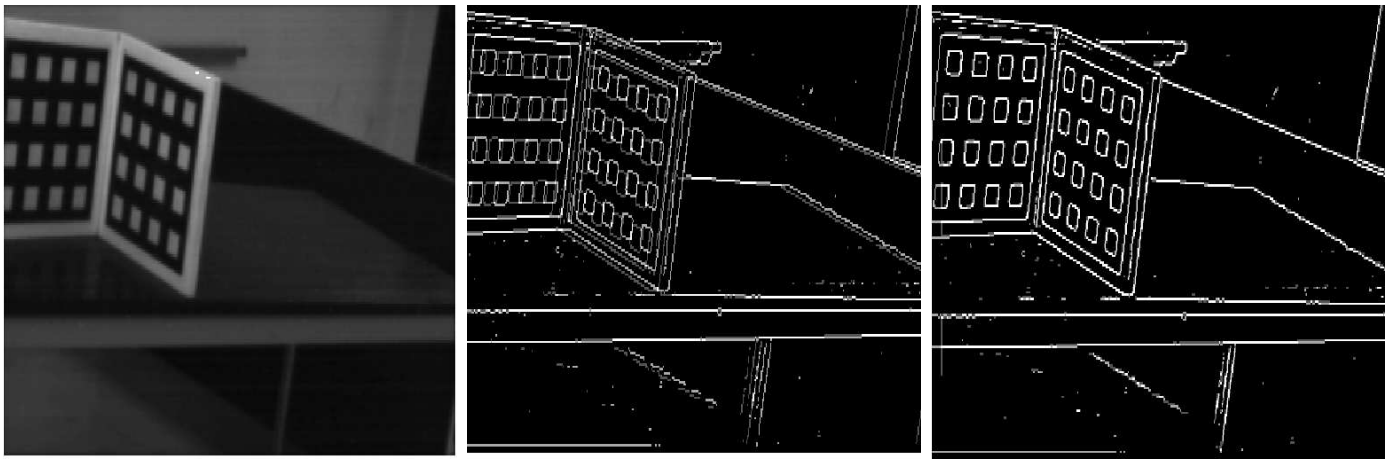

Figure 6: Rotational stabilization, in the case where the system performs a pure rotation : the image (left), the unstabilized edges (middle) and the stabilized edges (right) are shown. As expected in this case, the stabilization is perfect almost everywhere, and no residual disparity occurs.

with consistent intensity. This will not be reviewed here since we have just reimplemented the work of this author, while improvements in terms of speed have been reported in [44].

The input of this module is one stabilized frame, a parameter to choose the smoothing factor and a parameter to adjust the threshold under which intensity variation is taken as negligible. The output is a list of regions with homogeneous intensity, each region being characterized by its average retinal (1) horizontal and (2) vertical locations, (3) its size, and (4) the average intensity. These four numbers will be used in what follows.

As discussed in the previous section, any parameter defined in a system must be either automatically tuned or kept fixed, since no manual adjustment can occur. In this module we have two parameters : (a) a smoothing factor, which is expressed as a retinal window size, and (b) a gradient threshold. Let us discuss how we can automatically compute these two parameters.

1. For **the smoothing factor**, we have determined that :
   - On the one hand, when the smoothing factor is very small the number of regions increases because some noisy parts of the disparity map are not filtered and appear as regions.
   - On the other hand, when the smoothing factor gets to be very high, the number of regions decreases also, because some borders between regions are being flattened by the smoothing and, if below the contrast threshold, could be partially canceled.

   In between, the number of regions is expected to be stationary. We thus expect these two complementary effects to yield an optimal value of the smoothing factor. This value simply corresponds to the smoothing factor for which the variation of the number of regions is minimum, i.e., a "plateau" in the number of regions.
2. **The gradient threshold**, can be considered as the limit between two statistical distributions: one contains the points with a negligible contrast and is expected to be zero up to a certain level of noise, the second contains points with a residual disparity. Using classical methods [45, 46] we have implemented an automatic detection of this limit.

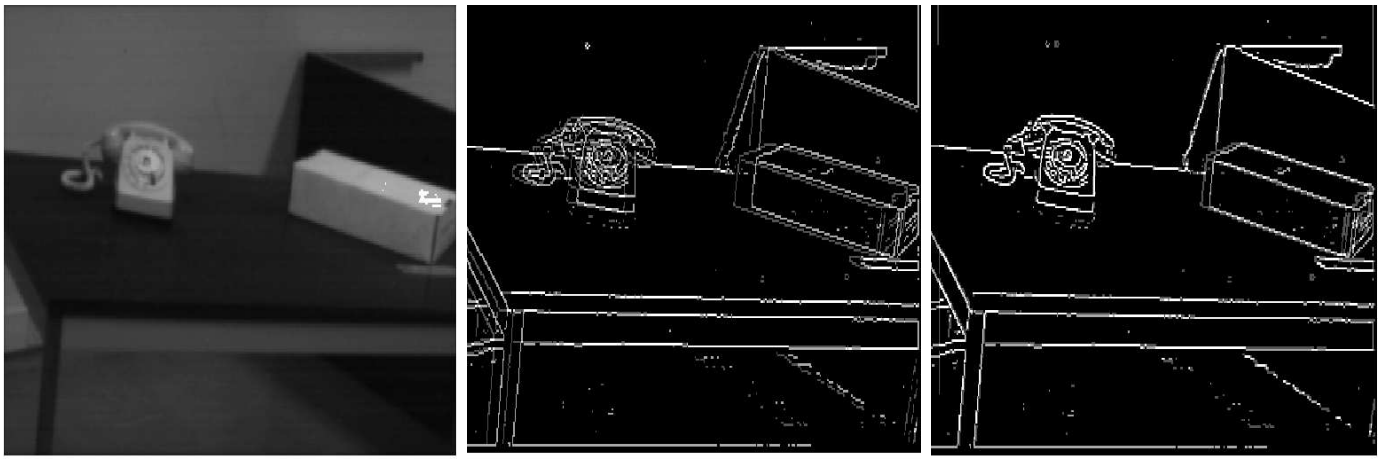

Figure 7: Rotational stabilization, in the presence of a moving object (the white box). In the unstable image the motion of the box is not visible because all edges are moving, whereas it is in the stabilized case. Rotational stabilization thus helps in the detection of moving objects. The unstabilized average disparity was of about 8 pixels, and less than 1 pixel after stabilization.

These mechanisms are run once at the beginning of a session, during a boot-strapping phase. They are not run continuously because they can perturb the other estimators since they modify the number of regions and their shapes.

2.3.1 Experimentation on the fast region detector.

As shown in figure 9, we have experimentally shown that the automatic adjustment of the smoothing factor indeed yields a realistic value of smoothing and is close to what an experienced user would have chosen. The region clustering is not very efficient, because quite a lot of very small regions are detected (see figure 9), but a simple fixed threshold of the region minimal size, allows the elimination of this problem (we have used a 50 pixels threshold in our implementation). This will be visible in the left part of figure 16.

In order to illustrate the mechanism used for auto-tuning, we show two examples of the evolution of the number of regions, when different smoothing factors (size of the filter window in pixels) are taken into account, in figure 10. It is clear that there is a region of the curve which is an inflection. The result of the region detection is shown in figure 11 for both examples.

2.4 Detecting visual targets for 3D visual perception

Now, in order to compute the 3D characteristics of these "regions", some other parameters must be estimated. First, since we want to compute depth from focus we must estimate the relative amount of blur for a given region. This will be estimated from (1) the average of the intensity gradient magnitude as in [14], the summation being taken over the N pixels of the region. Then, since we want to estimate depth from motion, we also need to find (2) the corresponding region in the previous frame. We thus have implemented a region tracker, but because of the stabilization mechanism between two consecutive frames, we have limited our implementation to finding the closest region, with a similar average intensity. This restrained but fast implementation was sufficient in practice for

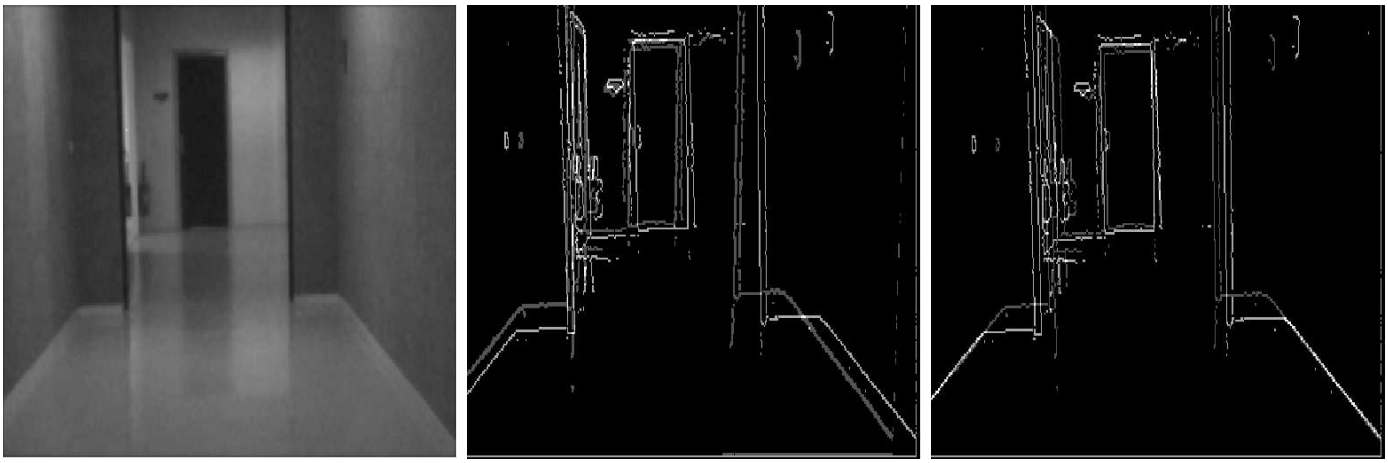

Figure 8: Rotational stabilization, during a forward translation, with some rotation also. After stabilization, the closer the edges, the higher the residual disparity. This can help detecting close obstacles.

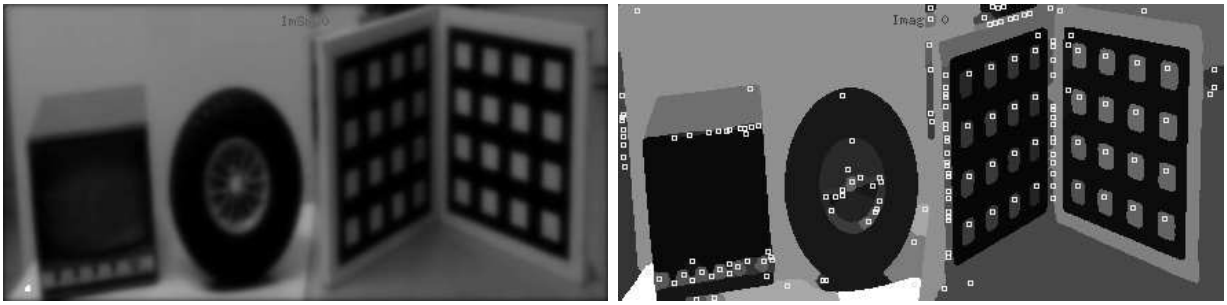

Figure 9: Optimal smoothing of a real moving scene observed during a motion. The objects are on a table with wheels which has been moved during the picture acquisition. The region segmentation is shown on the right image, one tip for each region.

images not very textured and relatively simple. Similarly, considering a binocular system, (3) the corresponding region in the other image is detected, using a similarity criterion along epipolar lines. Because we only match gross regions, the matching is rather simple to implement and we do not need a sophisticated implementation.

Finally, we detect whether the residual retinal disparity of the center of each region is less than a given threshold. As before, this threshold is computed automatically by considering non-stationary regions are detected as "outliers" of the statistical distribution of stationary regions. In our implementation we use a very simple Gaussian model as in [46].

If more than one region has been detected, the system must decide which one is to be observed first. This decision is task-dependent and is often to be taken considering 3D characteristics, such as the depth or the 3D motion. This is going to be computed next.

2.4.1 Experimentation on visual target detection

In our experiments, since we do not have to perform any specific task we have simply chosen the area with the highest residual disparity to be observed first.

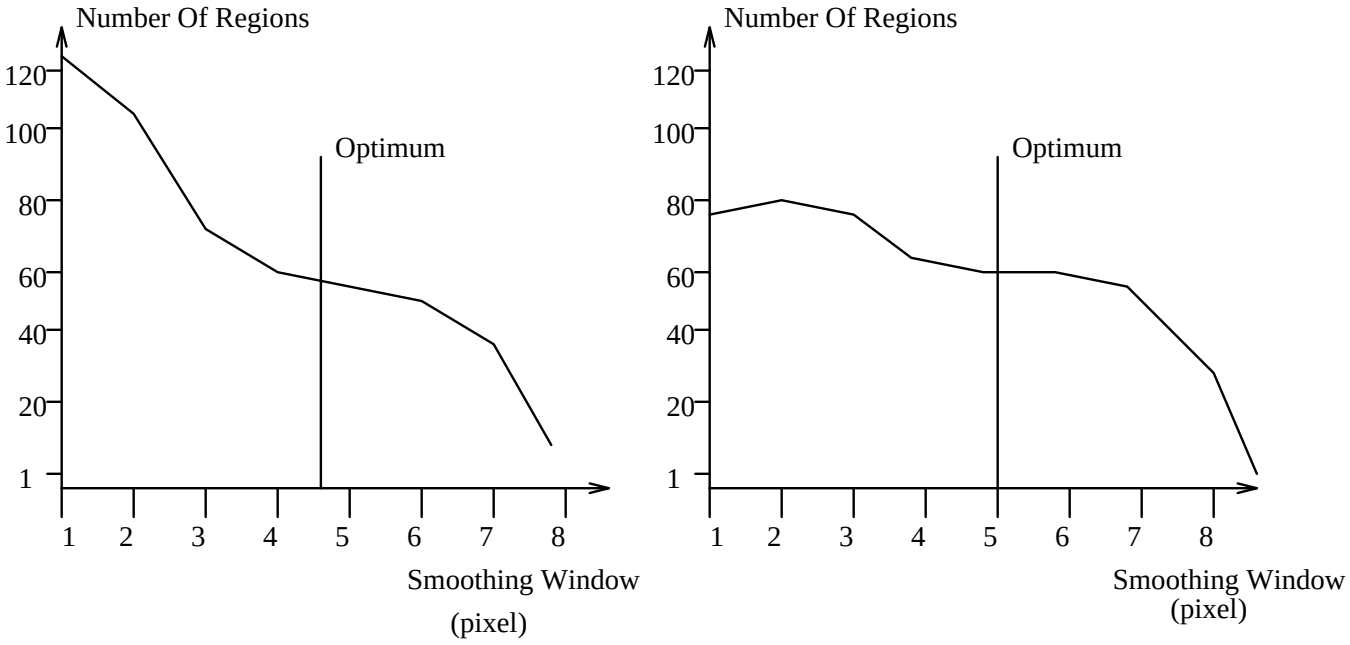

Figure 10: Variation of the number of regions as a function of the smoothing factor, for the image with the calibration grid (left) and the image with the telephone (right). The optimum values are shown on the curve.

We have analysed the behavior of this part of the system, when an object of unexpected residual disparity was detected. Such an example is shown in figure 12.

The target was at about 2 meters from the camera (no zoom) and translation was of about 4-5 cm between each pair of images.

We have analysed the performance of this module by measuring the retinal motion thresholds corresponding to the detection of an unexpected moving object, as shown here :

Object apparent size (pixel)	279	110	52	23
Motion Threshold (pixel/frame)	4	6	5	4

We also have analysed the retinal motion thresholds corresponding to the detection of a close stationary object, as shown here :

Object apparent size (pixel)	221	113	72	27
Disparity Threshold (pixel/frame)	3	5	4	4

These two results are quite important because they demonstrate two things :
(1) the thresholds are close to the actual precision of the system and
(2) they do not depend on the object size, which is important for applications.

We also have checked our modules considering more complicated scenes. For these two examples, we have worked on pre-recorded image sequences. The results are shown in figure 13. They illustrate the fact that this "low-level" mechanism of focus of attention can be used in various situations.

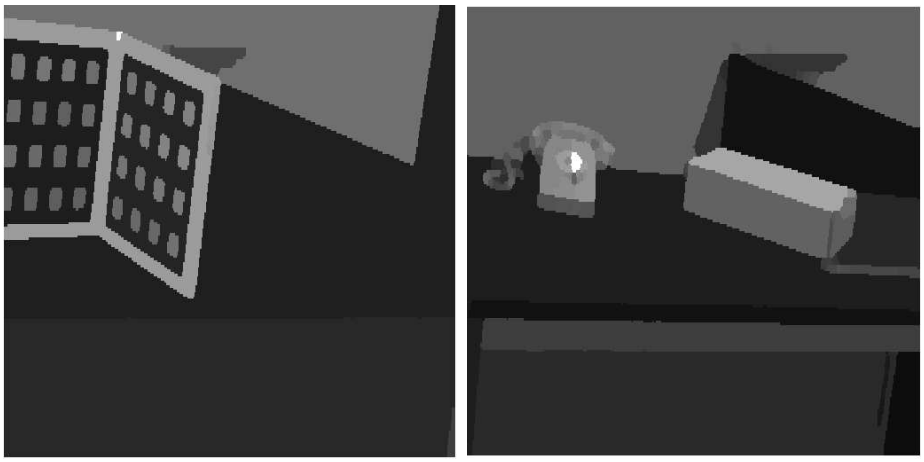

Figure 11: Image of the regions at the optimum values of smoothing factor and contrast threshold. Each region has been drawn with a constant intensity. It is clear that the main regions of each image are preserved.

3 Computing the 3D characteristics of a visual target

In such a system, depth can be obtained from several cues : stereo disparity, motion disparity, zoom disparity, blur variations and focus. These different cues do not have the same precision [47]. Moreover, we expect the two first ones to provide only an information about the "average depth" of the object. In order to deal with this situation we propose to analyse the 3D structure in two steps : we first consider the object as a flat fronto-parallel shallow of constant depth and estimate its depth and motion, and then refine this 3D estimation using stereo disparity only, since recovering the 3D structure of an object in motion is often a hard problem. On the contrary, stereo mechanisms can be used during motion [34], since for a moving object, the 3D structure can be estimated at each instant without knowing its motion.

Since the observed object is, here, assumed to be of constant depth, the projection of the center of gravity of the points of the 3D object corresponds to the center of gravity of the projections of the points of the 3D object as easily verified (see figure 14). Therefore, we can compute the depth for the center of gravity of the object only.

Let us first consider focus.

3.1 Using second-order focus variations to compute depth

If the system has a calibrated auto-focus module, each time the system is on focus for a given retinal location we have an estimation of the distance from the point to the optical center of the camera Z_i, because there is a direct relation between the target depth and the distance at which we are on focus (with the notation of figure 14, we have from the lens formula : $\frac{1}{f} = \frac{1}{u} + \frac{1}{w}$, $Z_i = u + t$ as a function of w).

Please note that Z_i corresponds to our notation of equation 1.

But if the system is not in focus, from the geometry of the lens, we can calculate the diameter of the blur circle (see [14]) as $d = D\ v\ \left(\frac{1}{f} - \frac{1}{u} - \frac{1}{v}\right)$. This value can be related to the intensity distribution but the obtained results, according to almost every author,

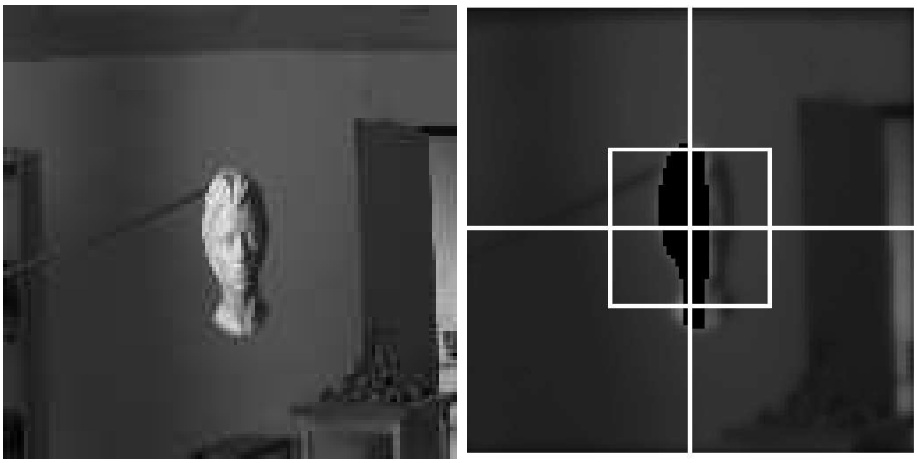

Figure 12: An example of detection of a moving object, while the head was in motion. The left image shows the picture, and the right image the detected region. This is an area of 279 pixels, with an average intensity variation of 21 over the 255 intensity range, and an apparent motion of 2 pixels.

are very qualitative and need some additional calibration procedures. Moreover, as stated by [48] there is a twofold ambiguity, since the blur exists whenever the retina is behind or in front of the image plane. We thus need at least two measures to distinguish these two configurations. It has also been established that the relations on depth from out of focus are deeply dependent upon the calibration parameters of the lens, which are quite expensive to obtain.

Following another approach, we have interpolated the point of optimal focus using the following simple **local** model, which is not obtained from the physical model of the lens :

$$L_i(\mathbf{\Theta}) = \log(\sigma_i^2) = A + B\ (W_i(\mathbf{\Theta}) - Z_i)^2$$

where $\sigma_i^2 = 1/N\ \sum_N ||\nabla I||^2$ is the average intensity gradient magnitude for the region taken into account and Z_i is the depth in the retinal frame of reference. The quantity $W_i(\mathbf{\Theta})$ is the distance corresponding to an object in focus, for a mount configuration $\mathbf{\Theta}$. This model is justified by the fact that the intensity variance is approximately related to the focus by a Gaussian function [49]. The quantities A and B are unknown. Now considering three values of the intensity variances, at times 0, 1 and 2, we can eliminate A and B and obtain :

$$Z_i = \frac{1}{2}\ \frac{W_i(\mathbf{\Theta}_0)^2\ (L_i(\mathbf{\Theta}_2) - L_i(\mathbf{\Theta}_1)) + W_i(\mathbf{\Theta}_1)^2\ (L_i(\mathbf{\Theta}_0) - L_i(\mathbf{\Theta}_2)) + W_i(\mathbf{\Theta}_2)^2\ (L_i(\mathbf{\Theta}_1) - L_i(\mathbf{\Theta}_0))}{W_i(\mathbf{\Theta}_0)\ (L_i(\mathbf{\Theta}_2) - L_i(\mathbf{\Theta}_1)) + W_i(\mathbf{\Theta}_1)\ (L_i(\mathbf{\Theta}_0) - L_i(\mathbf{\Theta}_2)) + W_i(\mathbf{\Theta}_2)\ (L_i(\mathbf{\Theta}_1) - L_i(\mathbf{\Theta}_0))}$$

This allows us to estimate Z_i even if not in focus. More precisely if Z_i is not constant we obtain an average value of Z_i at time 1. This model is, of course, only valid when close to the optimal focus.

We have verified this idea experimentally and show one result in figure 15, for two examples.

The variance attached to this equation can be related to the peak of the focus, i.e., the parameter A, also called "depth of focus" and is easily obtained from [47], equation (25). These authors have computed the variance of this quantity as a function of the lens parameters. But if we make the following realistic hypotheses $Z \gg t$, $Z \gg f$ and $D \gg d$

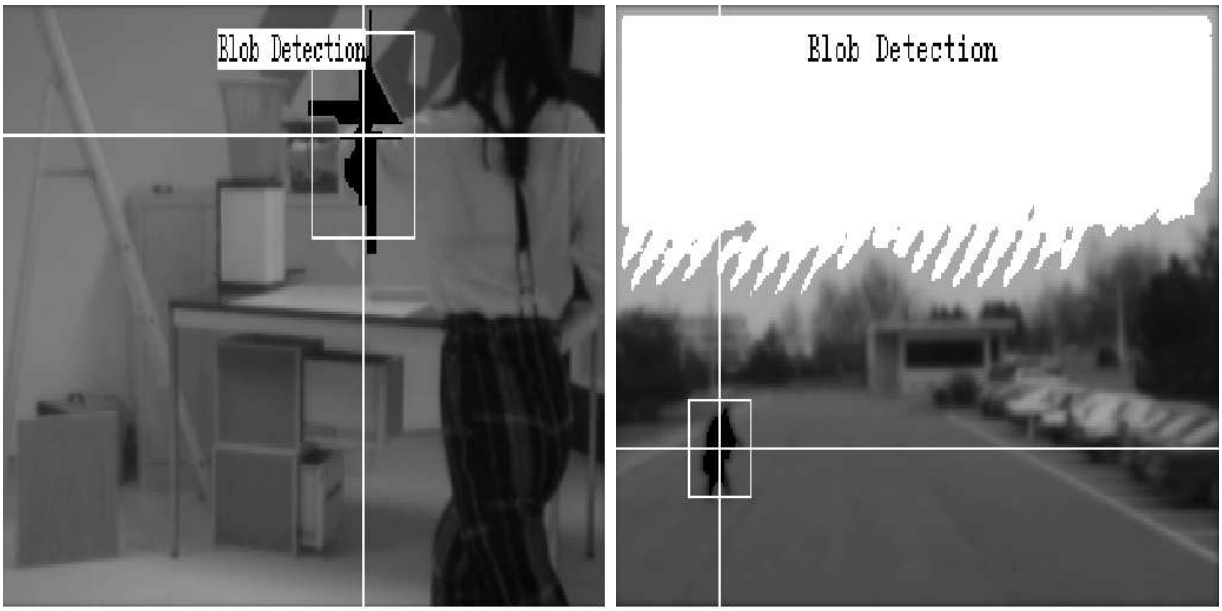

Figure 13: Two examples of target detection. The arm of the Kung-Fu teacher in the left view, and the pedestrian in the right view have been detected as the most important object to focus on, in the scene. Both objects correspond to regions for which residual disparity is maximal. This is consistent with what could be expected, from a naive observation.

which corresponds to assuming a remote object and an image almost in focus, we simply have :

$$V_{Z_i} \simeq k \frac{f^2}{D^2}$$

This means that the precision of the focus is proportional to the focal length and inversely proportional to the aperture. The factor k has been calibrated on the lens.

3.2 Multi-model concurrency to integrate the 3D depth

As soon as the object has been located in an image, equation (1) yields, after Z_i is eliminated, two linear equations as functions of the 3D location of the center of gravity of the object. Having correspondences between two frames in a binocular system provides at each instant 4 linear equations to recover this 3D location.

In addition to that, since we have, for one lens, a measure of Z_i, we also use equation (1) to relate this data to the 3D location of a 3D point.

We thus have, at each instant, 5 measurement equations : 1 coming from focus in one lens, 2×2 coming from the target location in both frames. This includes cues from vergence and motion. Obviously, if one or more of these measures are undefined we can avoid their integration by considering their (co)variance as infinite (the inverse of the (co)variance is zero). In other words, each of these 5 measurement equations are weighted by the inverse of a variance. So if only one camera is used or if no information about depth from focus is available, the estimator can still be run with the corresponding weights equal to zero.

These measurement equations have been used in a set of linear Kalman filters. Four filters are run in parallel. Each filter estimates the center of gravity of a 3D-object

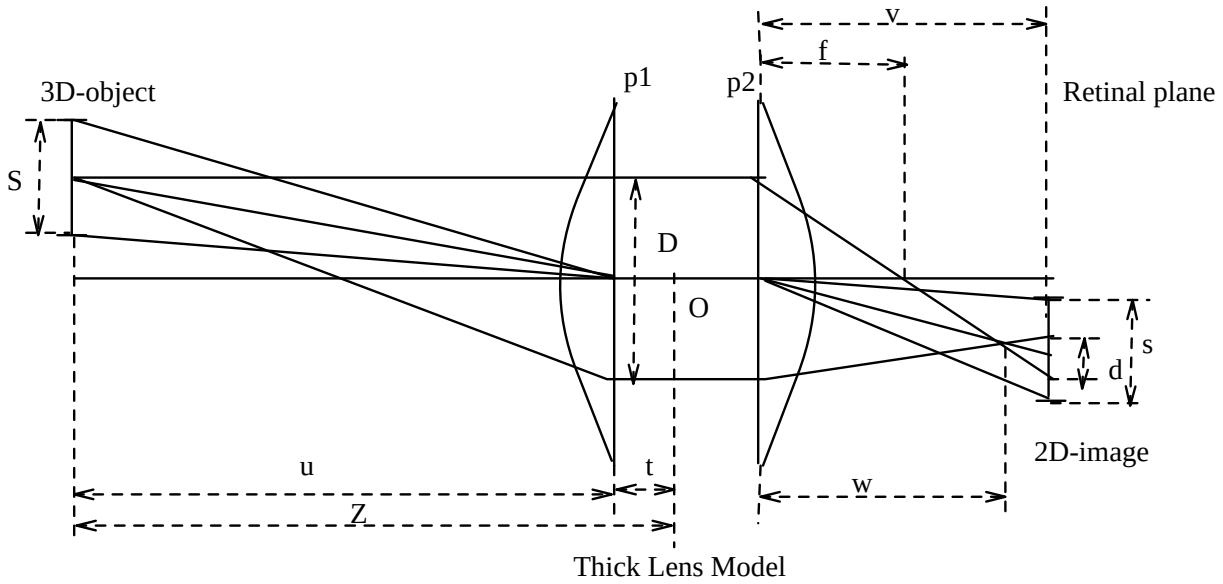

Figure 14: The camera geometry and the object geometry; p1 and p2 are the principal planes, D is the aperture of the lens, f the focal length; the image formation is done at a distance w of the principal plane and the retina is at a distance v of the retina. The 3D-object size is S and its distance to the lens Z, it is the sum of the distance u from the object to the principal plane p1 and distance t from p1 to the origin. The 2D-object size is s and the blur circle diameter d.

$M = (X, Y, Z)$. The first filter assumes M is stationary, i.e., its velocity is null. The second filter assumes M is moving with a constant velocity, and the third filter assumes M is moving with a constant acceleration. An additional filter assumes that the previous location of M is unknown, i.e., only uses binocular disparity and depth from focus, but does not rely on previous information. Assuming Gaussian additive noise, for each filter a probability of error is computed from the residual error, related to a Ξ^2 distribution. The state of the system (M location, velocity and acceleration) is updated considering the best filter output. This well known mechanism of multi-Kalman filter [50] allows detection if the target is either moving at constant acceleration, or moving at constant velocity, or stationary.

Moreover, because of the last filter, we can detect model ruptures. If the internal state is no longer reliable, the last filter, which does not rely on previous information, will be a better estimate than the three first filters, because it does not take previous information into account. On the contrary, the three first filters explicitly use the previous state since they integrate some models for the evolution of the parameters. Then, for a target either moving at constant velocity or acceleration or stationary, one of the first three filters must provide a better fit, because it corresponds to such the situation.

Finally, this adaptive mechanism allows the use of a model with a minimal number of parameters since, for instance, the target acceleration is not estimated if negligible with respect to the system noise.

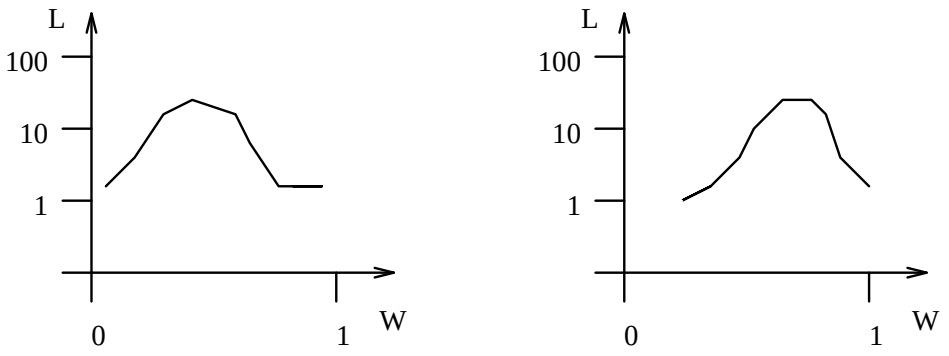

Figure 15: Two examples of focus-adjustment curves. The horizontal axis corresponds to the uncalibrated focus measurement (W), and the vertical axis to the average gradient magnitude in log coordinates (L). These two curves show that we have a measurable maximum, that can be interpolated even if not in focus.

3.3 Computing uncertainty and object size

In summary, for each region of unexpected disparity, considering the center of gravity of the region and the intensity variance, and using both spatial and temporal correspondences, we can output the average depth of the region, detect if the corresponding object is in motion, and estimate this 3D motion.

Considering that each pixel has a variance $V = \sigma^2\ I$, i.e., has a constant and isotropic uncertainty, the variance V_g for the center of gravity of a region of N pixels is obvious to compute : $V_g = \frac{\sigma^2}{N}\ I$.

The average 3D size of the object is obtained immediately from the 2D size of the object since : $S = \frac{Z}{v}\ s$, as visible in figure 14.

Although rather rudimentary, this 3D information is sufficient to drive the mount. Moreover, if required, a better 3D reconstruction can be processed. This is explained in the next section.

3.4 An example of implementation with 3D reconstruction

We now use this average depth to initiate a reconstruction algorithm.

Each time a visual target is detected, its size and depth is computed and the target can be reconstructed as a fronto-parallel object. Moreover, not only the target structure but also the target 3D-motion is analysed. Since this is the result of a Kalman filter, the quality of the results improves as the number of equations increases.

However, the obtained 3D-model is not very efficient, since (a) tilted surfaces are not represented, regions (b) which have not been matched have their depth undefined, etc. But we already have *a coarse dynamic depth and kinematic map of the surroundings.* Moreover, considering stationary structures, we can improve this knowledge as follows.

We have chosen a regularization criterion, as reported in [44]. Considering two views, say 0 and 1, we minimize the variations in intensity between view 1, I_1, and view 0, I_0 :

$$\sum_m [I_1(m_1(m_0, Q_{10}, \mathrm{s}_{10}, \Pi(m)) - I_0(m)]^2 + \lambda ||\nabla \Pi||$$

where Q_{10} and s_{10} are defined as before, for each point $\Pi(m) = 1/Z_i$ is the inverse of the depth of the point in the retinal frame of reference, or *proximity*, and the function $m_1()$ is

defined in equation (2). The parameter λ allows to balance the effect of the regularization factor, and has been kept at a fixed value equal to 0.1.

The implementation is done as in [51] and will not be detailed since it corresponds to a classical methods of this kind.

The key point here is that, because the previous visual module provides a rather good initial condition for the 3D map and reliable information on the camera location and calibration, the reconstruction process is fast and quite efficient, and does not requires a complex implementation.

On the other hand, this additional module allows to compensate for the drawbacks of the previous methods (fronto-parallel objects only, "holes" in the 3D map).

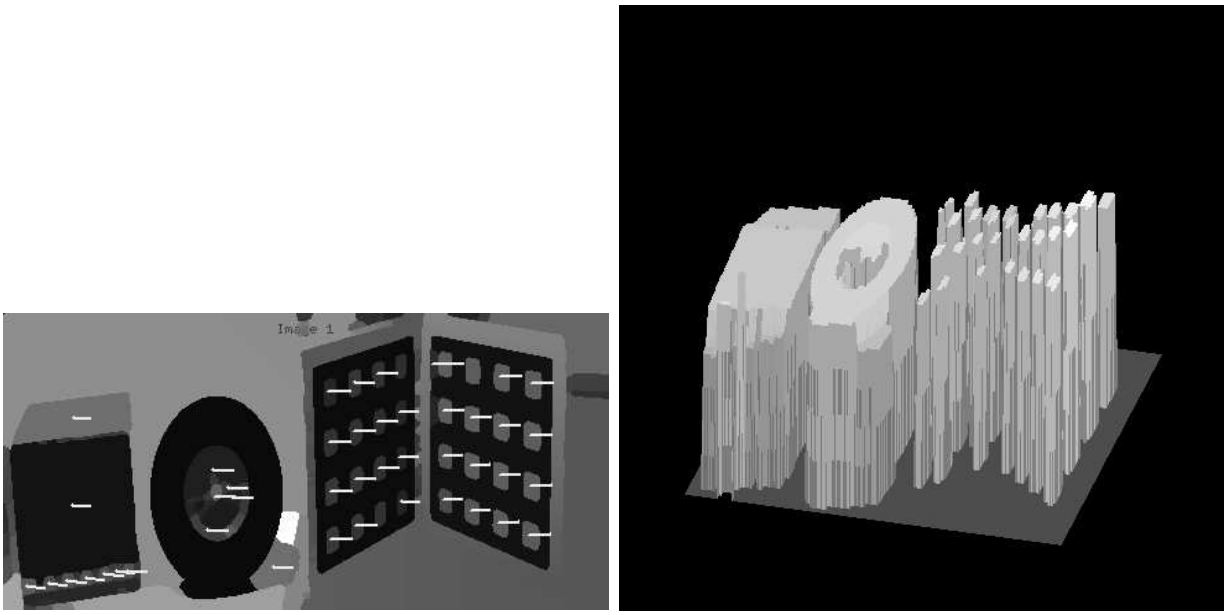

Figure 16: The computation of motion for the best matched regions (left view) and the corresponding proximity map (right view), for the scene shown in figure 9.

In any case, the precision of the overall system is good enough to recover a coarse depth map as shown in figure 16.

Finally, the architecture of the overall system is presented in figure 3.4 with some indications on the performances on a sun station, during simulations, to ease the comparison with other visual modules. The final implementation has been done a VxWorks system using some dedicated hardware [52].

Let us now use all this information to control the mount.

4 Controlling the mount parameters

Obviously, it is possible to control zoom, focus and gaze direction including vergence using only 2D cues. Considering the projection of an object under observation, it is straightforward to tune the zoom such that the object's size has the desired extent, adapt focus to minimize blur, and perform vergence (plus pan and tilt) to view the object around the retinas's center in both cameras [9]. However, the previous method requires feedback control and different object characteristics must be recomputed at each step. On the other

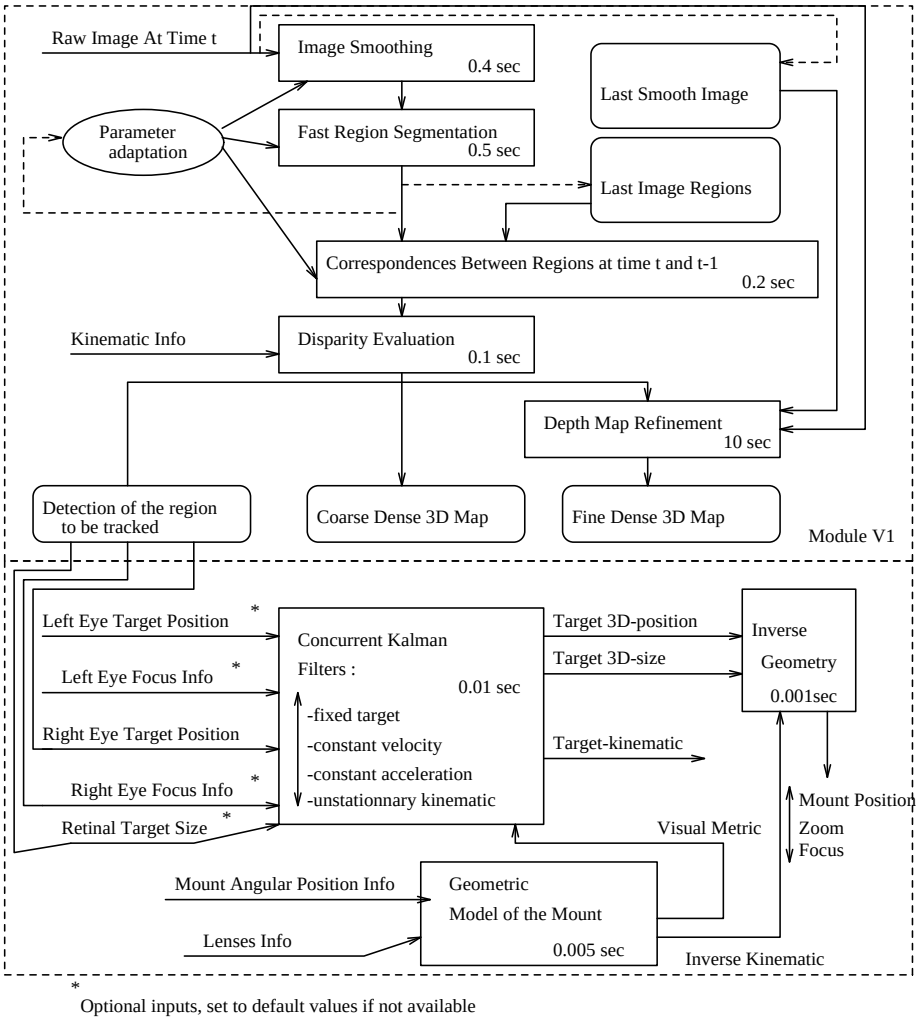

Figure 17: The architecture of the visual module, designed to be used on a robotic head : the reconstruction algorithm has been implemented, considering first a coarse model for the 3D map, and then a fine model in order to obtain both fast and fine data. A binocular implementation has also been experimented simply connecting a second module V1 for the camera to the "Left Eye" inputs. The internal parameters of this algorithm are adjusted automatically. The average computation time is given for a 400×200 image : a coarse 3D map of the retinal view can be issued at about 0.8Hz, using a SparcII CPU. On the contrary, the 3D map refinement is much slower, but it can be run at a different rate.

hand, if the depth and the size of the object are known and if the system is calibrated, these parameters can be estimated in one step.

One very important consequence of this "one-step" estimate is the following : control is not necessarily embedded in real-time feedback, but may simply be limited to static positioning. In other words, the control can be a simple "look and move" paradigm. This restriction is very important in the case of hardware limitations. Thus, using 3D cues, a direct global control is possible, the local feedbacks (implemented using 2D cues) being only used for the residual error correction.

4.1 General strategy to drive the mount

Let us now make explicit some requirements for controlling the mount parameters of a robotic head on which 3D vision is to be performed. This analysis is simply based on some bibliographical elements as made explicit in the sequel.

- $[\mathcal{R}_1]$ We need to reduce (but not necessary cancel) the disparity between two consecutive frames (retinal stabilization) because :

  1. Visual algorithms for motion estimation (token-trackers, tracking-snakes, optical-flow operators) perform better when the disparity between two consecutive frames is minimum [30].
  2. Calibration parameters are valid only for a given area of the visual field and thus only if the observed object has a stable position on the retina [36, 38].
  3. This will maintain the observation of an object during several frames as required by 3D vision algorithms [34].
  4. In terms of discretization errors, it has been demonstrated that when if the observed object is stationary with respect to the retina, these errors are minimal [16].

- $[\mathcal{R}_2]$ As in nature, it is convenient to maintain the observed object close to the estimated principal point (foveation) since :

  1. The calibration model relies on the pin-hole model which is valid only close to the optical axis (a biological equivalent is the fovea) [36, 38].
  2. A moving object with unexpected motion is easily maintained in the visual field if it stays around the center of the retina (obvious).
  3. The induced disparity remains minimal when zooming (see $[\mathcal{R}_1]$), if we are close to the principal point [13].
  4. An off-centered object will drift rapidly due to its eccentricity [28].

- $[\mathcal{R}_3]$ When $[\mathcal{R}_2]$ is not satisfied, to prevent observed objects from drifting out of sight we must reset the gaze direction as quickly as possible.

  A reasonable strategy is to reset the mount position as quickly as possible, cancel visual information during the motion, and restart visual perception when motion ends (saccade), as for biological movements [53].

  However, we must relate the visual information before and after the displacement, as discussed previously. If this displacement is done with a translation, image deformations will occur and the correspondence between the two images is not straightforward (it depends on the depth of objects). On the contrary, if the displacement

is simply a rotation, the related image transformation is a simple and predictable reprojection, hence the system can relate the visual information before and after displacement [32, 35].

Furthermore, if this saccade is very quick, the additional variations in the image due to non-stationary objects between the beginning and the end of the motion are expected to be small (obvious).

This yields to the following mechanism : *rotate the camera as quick as possible such that the angular position of the target is aligned with the optical axis.*

- $[\mathcal{R}_4]$ Mount translational motion is also required to infer structure from motion and to relate the 3D structure parameters to the projected displacement. This can be done with linear joints performing pure translations or off-centered rotations with angular joints [9].

  The orientation of the translation must be "orthogonal to the 2D points" [15, 30, 54], i.e. oriented such the induced disparity is maximal : considering a line segment, for instance the projection of the translation is optimal if it induces a normal displacement of the rectilinear edge.

  Moreover, if the translation is performed such that its component along the Z-axis is zero, the retinal disparity is not dependent upon the location of the principal point [28]. As a consequence, even if this quantity is not known with a high precision (approximate calibration), the result will not be affected. Moreover, the retinal field has no singularity in the image which simplifies its estimation [29]. It is thus a good strategy to try to perform translation parallel to the retinal plane in this case.

  This yields the following mechanism : *induce a translational motion in a direction orthogonal to the average target location in the image, and in the parallel to the retinal plane, and compensate for this disparity using rotational stabilization (eye-neck coordination)* [11, 30].

- $[\mathcal{R}_5]$ Zoom control, when used to cope with variations in disparity, is subject to contradictory requirements because :

  1. To detect unexpected objects, the best conservative configuration for the zoom is to be minimum (the focal length being smallest, the field of view is wider). This extremal configuration corresponds also to a situation where object size and projected displacements are minimal as required from $[\mathcal{R}_1]$ [13].
  2. If the field of view is too wide then so will be the density of edges and an artificial disparity will be induced by matching errors. Zooming into the observed object will overcome insufficient resolution. This leads to a criterion for zoom control : *the focal length is to be increased if and only if this reduces the residual disparity between two frames for the observed object, and it is to be tuned to minimize this disparity.*

     In some cases this should be done in a "saccadic mode" because most visual modules assume the projection matrix is constant [11, 24] when the system is to be dynamically calibrated. This is not a restriction for our implementation, since calibration is always given.

  One way to avoid this contradiction and preserve both requirements 1 and 2 is to use an *unhomogeneous* visual sensor. One with a large field of view, the other

with zooming capabilities. We have implemented this strategy, and will show that stereoscopic perception is still possible with a "peripheral" and a "foveal" sensor, as the INRIA head.

Using the previous discussion we can easily integrate the previous strategies in a comprehensive visual behavior : *The system observes the visual surroundings. If an unexpected residual disparity is detected, the system changes its strategy and foveates the target to analyse until another target is detected.* Only one object is taken into account at a given instant.

This behavior can, considering a strategy related to human oculomotor behavior [55], be formalized as follows. We use two cameras, one with a peripheral field, one with a foveal field. The first camera is used to detect potential targets, the second tracks one target and analyses it :

- Process for the peripheral camera $\mathcal{P}$: passive observation, and image stabilization.
  - Detect unexpected residual disparity and update the average depth and size of each target, choose the target (if any) for which residual disparity is maximal, as developed previously.
  - If no target detected :
    - Rotate the camera smoothly to minimize the rotational disparity between two frames (*stabilization*) $[\mathcal{R}_1]$.
    - If the camera is too eccentric (a fixed threshold of 25 deg is used in our implementation), reset the camera position in one step, using a pure rotation, and cancel the visual computations during this motion (*resetting saccade*) $[\mathcal{R}_3]$.
  - else, if one target is detected :
    - Rotate the camera smoothly to maintain the target around the principal point of the camera (*tracking*) $[\mathcal{R}_2]$.
- Process for the foveal camera $\mathcal{F}$: foveation, and 3D object observation :
  - If a target is detected :
    - Modify in one step the eye angular position, to relocate it in the central part of the visual field *(capture and correcting saccade)* $[R_3]$.
    - Keep the observed area near the center of the retina, using smooth eye movements and maintain the size of the object to about half of the size of the retina controlling zoom *(smooth-pursuit)* $[R_1]/[R_5]$.
    - Smoothly move the neck so that the neck orientation corresponds to the average gaze position, over an adaptive time window, decreased to zero when the camera eccentricity reaches its maximum *(eye-neck cooperation)* $[R_4]$
  - If no target detected :
    - Zoom back to a minimal focal $[R_5]$.
    - Perform the same displacements as the other sensor.

The implementation of this behavior depends on the mechanical hardware and will be given after the presentation of the corresponding mount.

4.2 Overview of the controller

We finally can describe the complete set of processes of visual control :

- The focus is driven to maximize the gradient magnitude for the observed target if any, or if none, for the whole visual field (in that case subsampled). If a target is given with an estimate of its depth, the auto-focus is driven directly from this depth information.
- If no target, the vergence is driven to minimize the horizontal drift between two consecutive frames, measuring the angular velocity from odometric cues, and reset if too eccentric. Pan and tilt are still.
- If a target is detected, then pan, tilt and vergence are driven to foveate the target. The dynamics of the pan and tilt are limited to induce about 2/5 pixels of disparity per frame max, the relation between angles and pixels being known from calibration.
- The zoom is driven so that the size of the target is approximately equal to half the visual field, and is adjusted to minimize the average retinal disparity of the stabilized image, and is minimum if no target.
- The iris, in synergy with acquisition gain and offset, is driven to maximize the intensity distribution over the measure scale.
- The visual control is done by both the process for peripheral vision $\mathcal{P}$ and the process for foveal vision $\mathcal{F}$, as described before.

Although very simple and straightforward, this implementation matches all the requirements given previously and allows the automatic observation of a set of 3D visual targets, fixed or mobile. This apparent simplicity is due to the choice of the mechanical hardware, and the analysis of the problem done in the previous sections.

4.3 Tracking a 3D target : simulation experiment

In order to verify the validity of our design we have built a simulator to analyse the behavior of the 3D estimator. We have checked that the system indeed not only predicts the 3D location and motion of a point target but also detects the class of motion (no motion, constant velocity, accelerations) and model ruptures. This mechanism is comparable to what has been described in [56] on an arm-eye system.

Considering the simulation reported in figure 18, for instance, it is visible that when the dynamic of the target is changed the system automatically resets and tracks the target with a small phase lag. This was not obvious because the level of noise is quite large (5 pixels) and thus the system has to perform a major filtering operation.

Moreover we have observed another advantage of this implementation. This is related to stability. Considering an internal model including target acceleration (a second-order filter), it is clear that some instabilities can occur, especially when a prediction of the target is used in a closed-loop mode. This also occurs here; but instantaneously, a more stable model (a first-order filter) is chosen by the algorithm, because it minimizes the measurement error, whereas the unstable filter does not. Therefore, an automatic switch is done to recover a stable feedback. This is an important property.

4.4 Tracking a 3D target : real object experiment

We have finally conducted an experiment using the INRIA robotic head and have tracked several targets (objects on table with wheels, humans, manufactured objects, etc.) at a

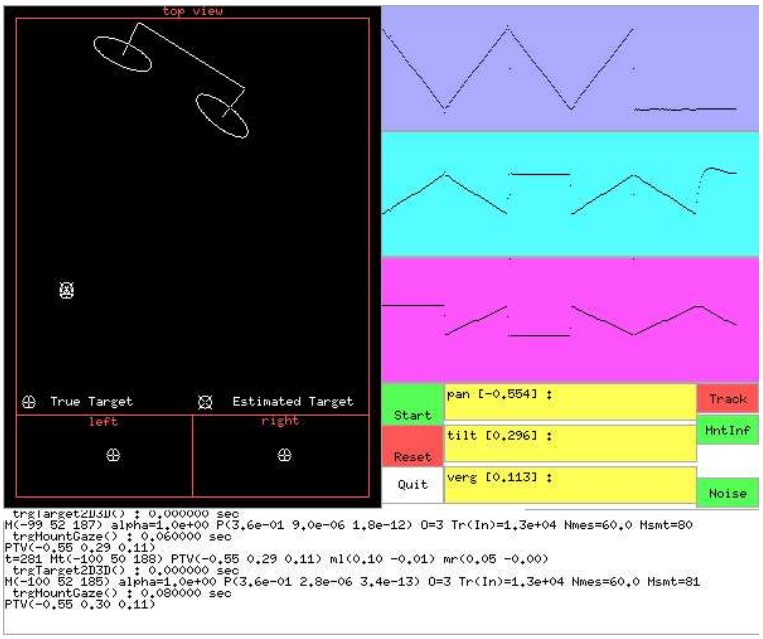

Figure 18: Simulation of the 3D tracking of a noisy target with abrupt variations in the dynamics. The top view represents the binocular mount during the tracking of the target represented as a point. Retinal projections are shown below in left and right cameras. The estimation of the 3D target location is shown on the right for X, Y and Z coordinates (from top to bottom). Each trace corresponds to 10 seconds and 5 $\times$ 5 $\times$ 10 $\times$ meters considering real-time tracking (25Hz).

frequency a bit higher than 1 Hz. Such a sampling rate is indeed a drastic limit of the dynamics of the target, but our goal was only to realize an experiment, not an industrial system. Due to the nature of the image processing, it was always possible to track the moving object even for retinal disparities close to half the retinal size of the target. We also have verified that the qualitative behavior of the 3D estimator corresponds to what has been obtained during simulations.

The main positive feature is the following : the system is able to (a) predict the target motion and (b) re-center in one step, so that we have observed that the tracking is very robust in the sense that it does not "lose" the target, even if we have large delays. It also provides quite efficient tracking as shown in figure 19 and figure 20. It is very difficult to compare this performance with other works in the field, because except for one author [57], the experimental results are limited to a description of the qualitative behavior of the algorithm. In our case, even if this is for limited range of results, we have tried to collect some quantitative data.

5 Conclusion

Let us briefly summarize the results obtained in this study.

- We can control the different degrees of freedom of a mount (zoom, focus, vergence, gaze direction) using 3D visual cues, in a single step. The same control in 2D would have been iterative.

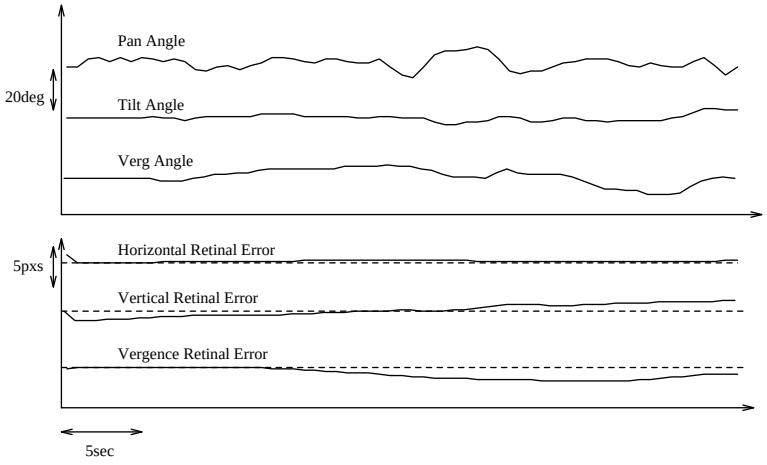

Figure 19: Results of real 3D tracking on the INRIA head. The angular head positions and the corresponding retinal errors are shown. The average error is of 2-3 pixels and the corresponding image sequence is given in the next figure.

- We can provide the relationship between the mount angular position and the retinal displacement in pixels, and keep the observed object on the foveal part of the cameras, even for slow systems. This can only be done using a 3D calibration model (mount+visual sensor), and computing object depth. This relationship is not very precise but has still the order of magnitude of most of the existing algorithms.
- We detect regions of interest considering the retinal disparity related to the translational component of the motion.
- We can track mobile or stationary tokens in an image sequence by considering a realistic model of their rigid motion. We also determine the nature of the 3D motion of the observed object.
- The average depth and size of the observed objects are computed and a coarse 3D map is reconstructed.

Although this was not obvious at first, the additional complexity of using 3D cues on a robotic head not only preserves but also further enhances the capabilities of such a system. It provides a higher level of knowledge representation for task oriented visual systems.

The success of this experimentation is deeply related to the existence of a mechanism which is "tuned to the task." This means that, we have configured the robotic head for this visual application. Moreover, we must profit from the fact that the mechanical precision of the system allows us to know the calibration of the camera (extrinsic and intrinsic) in any configuration.

A The inverse kinematics of the INRIA head

Let us derive the inverse kinematics for the INRIA robotic head. Several mechanical constraints are verified on this system : pan and tilt axes intersect and are orthogonal, vergence axis is parallel to the tilt axis, pan is on tilt, the camera optical centers and the three rotation axes are coplanar. Moreover, for the camera with vergence, because the optics are fixed it is possible to have the optical center contained in the vergence

Figure 20: Results of a real 3D tracking on the INRIA head. One image every four frames is shown, for one camera only. Please note that the system has performed a zoom and has refocused.

rotation axis, and thus this degree of freedom is a pure rotation. However, the relative orientation is not easily controllable, but is known from calibration. The transformation from the mount to the camera, here limited to a rotation, will be noted H_{left} and H_{right} [6]. In practice, for the camera with a fixed lens, $H_{left} \simeq I$. On the other hand, the camera without vergence but having a variable lens has not only its relative orientation but also its relative position not easily controllable although known from calibration. In practice, this transformation is of the form $H_{right} \simeq \begin{pmatrix} I & (0,0,\Delta_f)^T \\ (0,0,0) & 1 \end{pmatrix}$ since the main effect of a zoom is to translate the optical center along the optical axis $\mathbf{z}$ to a quantity equal to the variation of the focal length Δ_f, as demonstrated in [39]. These two transformations are known from calibration, and their values are expected to be small. They are constant over time, except the translation induced by the zoom. Moreover, it is always possible to compensate for the rotational part of H_{left} and H_{right} by applying a reprojection, that is a linear transform of the homogeneous coordinates of the picture [30], and to calibrate these parameters using accurate calibration techniques [40]. Finally

[6] We use the very common notation :

$$\begin{pmatrix} X' \\ Y' \\ Z' \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} R & \mathbf{t} \\ (0,0,0) & 1 \end{pmatrix}}_{H} \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

to relate the coordinates of a point between two frame of references which differ by a rotation R and a translation $\mathbf{t}$.

a fine mechanical adjustment can compensate for the translational components of these static transformations. We thus can consider, from now on, that they can be canceled.

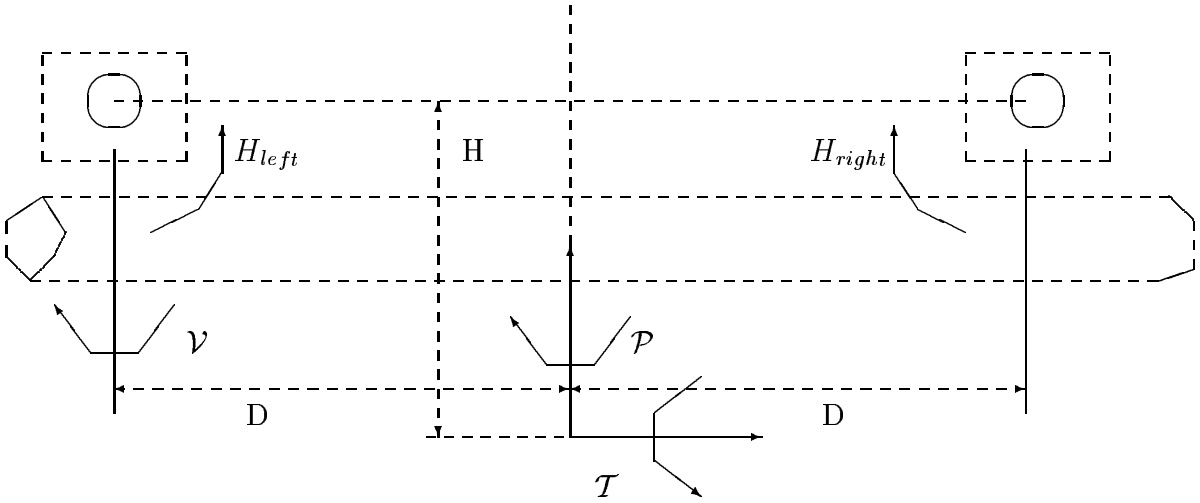

Figure 21: Notation for the head kinematics. $\mathcal{P}$ is the pan angle, $\mathcal{T}$ is the tilt angle, $\mathcal{V}$ is the vergence angle, D the eccentricity of the cameras with respect to the rotation center, H the height of the cameras with respect to the rotation center.

Using the notation of figure 21, we can compute the kinematic chain in homogeneous coordinates for both cameras[7] :

$$Right \quad Camera \quad H_{right} \quad = \quad \begin{pmatrix} R(\mathbf{y},\mathcal{P}) \cdot R(\mathbf{z},\mathcal{T}) & \begin{pmatrix} -D \\ H \\ \Delta_f \end{pmatrix} \\ (0,0,0) & 1 \end{pmatrix}$$

$$Left \quad Camera \quad H_{left} \quad = \quad \begin{pmatrix} R(\mathbf{y},\mathcal{V}+\mathcal{P}) \cdot R(\mathbf{x},\mathcal{T}) & R(\mathbf{y},\mathcal{V}) \cdot \begin{pmatrix} D \\ H \\ 0 \end{pmatrix} \\ (0,0,0) & 1 \end{pmatrix}$$

Let us now compute the inverse kinematics. It is generally not possible to gaze at a given point $M = (X, Y, Z, 1)$ in space, with the left camera having vergence but not the right camera. This is because the two optical axes are required to be in the same plane. However this is the case in our configuration. First, M should belong to the plane of the two optical axes. This equation of the plane at height H and tilted by an angle $\mathcal{T}$ (see figure 21) is : $\sin(\mathcal{T})Z + \cos(\mathcal{T})Y - H = 0$. The 2D coordinates of the projection m of M onto this plane are $m = (z = \cos(\mathcal{T})Z - \sin(\mathcal{T})Y, X)$. This projection should also belong to the two optical axes. These two axis are at distance $\pm D$ to the origin in this plane and an orientation equal to $\mathcal{P}$ and $\mathcal{P} + \mathcal{V}$. We thus obtain two additional equations : $\sin(\mathcal{P})z - \cos(\mathcal{P})X - D = 0$ and $\sin(\mathcal{P}+\mathcal{V})z - \cos(\mathcal{P}+\mathcal{V})X + D = 0$. For bounded angles

[7] $R(\mathbf{u}, \Theta)$ is the matrix of a 3D-rotation around an axis $\mathbf{u}$ ($||\mathbf{u}|| = 1$) with angle Θ.

we can explicitly solve these equations[8] and find the admissible domain as :

$$\begin{array}{rcl} & if & \left(\sqrt{Y^2+Z^2} > H\right) \\ \mathcal{T} & = & \arcsin(H \,/\, \sqrt{Y^2+Z^2}) - \arctan(\frac{Y}{Z}) \\ & z = & \cos(\mathcal{T})\, Y - \sin(\mathcal{T})\, Z \\ & if & \left(\sqrt{z^2+X^2} > D\right) \\ \mathcal{P} & = & \arcsin(D \,/\, \sqrt{z^2+X^2}) + \arctan(\frac{X}{z}) \\ & if & \left(\sqrt{z^2+X^2} > D\right) \\ \mathcal{V} & = & -\mathcal{P} - \arcsin(D \,/\, \sqrt{z^2+X^2}) + \arctan(\frac{X}{z}) \end{array} \tag{3}$$

References

[1] Y. Aloimonos, I. Weiss, and A. Bandopadhay. Active vision. *Int'l J. Comp. Vision*, 7:333–356, 1988.

[2] D. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[3] R. Bajcsy. Active perception. *Proc IEEE 76*, 8:996–1005, 1988.

[4] K H Cornog. Smooth pursuit and fixation for robot vision., 1985. MSc thesis.

[5] E P Kroktov. Exploratory visual sensing for determining spatial layout with an agile stereo camera system., 1987. PhD thesis.

[6] C M Brown. Gaze controls with interactions and delays. Technical report, OUEL: 1770/89, 1989.

[7] F Chaumette. *La commande référenciée vision*. PhD thesis, University of Rennes, Dept of Comp. Science, 1990. PhD thesis.

[8] M.J. Swain and M. Stricker. Promising directions in active vision. Technical Report CS 91-27, University of Chicago, 1991.

[9] Kourosh Pahlavan, Jan-Olof Ekhlund, and Tomas Uhlin. Integrating primary ocular processes. In *2nd ECCV*, pages 526–541. Springer Verlag, 1992.

[10] N.P. Papanikolopoulos, B. Nelson, and P.K. Khosla. Full 3D tracking using the controlled active vision paradigm. In *The 7th IEEE Symposium on Intelligent Control, Glasgow, August*, 1992.

[11] T. Viéville, J. O. Ekhlund, K. Pahlavan, and T. Uhlin. An example of artificial oculomotor behavior. In T. Henderson, editor, *Seventh IEEE Symposium on Intelligent Control, Glasgow*, pages 348–353. IEEE Computer Society Press, 1992.

[12] T J Olsen and R D Potter. Real time vergence control. Technical report, University of Rochester, Comp Sci: 264, 1988.

[8] Note that the equation : $a*cos(x)+b*sin(x)+c=0$ with $\theta = \arctan(\frac{a}{b})$ can be rewritten as $\sin(x+\theta) = -c/\sqrt{a^2+b^2}$ thus having at most one solution in $]-\frac{\Pi}{2}, \frac{\Pi}{2}[$: $x = -\arcsin(c/\sqrt{a^2+b^2}) - \arctan(\frac{a}{b})$. We have a unique solution if and only if a and b are not null together and if $|c|$ is bounded by $\frac{1}{\sqrt{a^2+b^2}}$, else there is no solution.

[13] J.M. Lavest, G. Rives, and M. Dhome. 3D reconstruction by zooming. In *Intelligent Autonomous System, Pittsburg*, 1993.

[14] Kourosh Pahlavan, Tomas Uhlin, and Jan-Olof Ekhlund. Dynamic fixation. In *4th ICCV*, pages 412–419. IEEE Society, 1993.

[15] B. Espiau and P. Rives. Closed-loop recursive estimation of 3D features for a mobile vision system. In *I.E.E.E. Conference on Robotics and Automation, Raleigh*, 1987.

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

[17] Q.-T. Luong and O.D. Faugeras. Active head movements help solve stereo correspondence ? In *Proc. ECAI 92*, pages 800–802, 1992.

[18] C M Brown. The rochester robot. Technical report, University of Rochester, Comp Sci: 257, 1988.

[19] T. Viéville. Real time gaze control : Architecture for sensing behaviours. In Jan-Olof Eklundh, editor, *The 1991 Stockholm Workshop on Computational Vision, Rosenon, Sweden*. Royal Institute of Technology, Stockholm, Sweden, 1991.

[20] C.W. Urquhart and J.P. Siebert. Development of a precision active stereo system. In *Proc. Int. Syp. on Intelligent Control, Glasgow*, 1992.

[21] R. Deriche and O. D. Faugeras. Tracking Line Segments. In *Proceedings of the 1st ECCV, Antibes*, pages 259–269. Springer-Verlag, Berlin, 1990.

[22] M.J. Stephens, R.J. Blisset, D. Charnley, E.P. Sparks, and J.M. Pike. Outdoor vehicle navigation using passive 3d vision. In *Computer Vision and Pattern Recognition*, pages 556–562. IEEE Computer Society Press, 1989.

[23] H.P. Trivedi. Semi-analytic method for estimating stereo camera geometry from matched points. *Image and Vision Computing*, 9, 1991.

[24] O.D. Faugeras, Q. T. Luong, and S. Maybank. Camera self-calibration : Theory and experiment. In *2nd ECCV*, Genoa, 1992.

[25] N A Thacker. On-line calibration of a 4-dof robot head for stereo vision. In *British Machine Vision Association meeting on Active Vision*, London, 1992.

[26] T. Viéville. Autocalibration of visual sensor parameters on a robotic head. *Image and Vision Computing*, 12, 1994.

[27] J.L.S. Mundy and A. Zisserman. *Geometric Invariance in Computer Vision*. MIT Press, Boston, 1992.

[28] O.D. Faugeras. *Three-dimensional Computer Vision: a geometric viewpoint*. MIT Press, Boston, 1993.

[29] E. Francois and P. Bouthemy. Multiframe-based identification of mobile components of a scene with a moving camera. In *Conf. Computer Vision and Pattern Recognition, Hawai*, pages 166–172. IEEE Computer Society Press, Alamitos, Califiornia, 1991.

[30] T. Viéville, P.E.D.S. Facao, and E. Clergue. Building a depth and kinematic 3D-map from visual and inertial sensors using the vertical cue. In H.H. Nagel, editor, *4th I.C.C.V., Berlin*. IEEE Computer Society Press, Los Alamitos, California, 1993.

[31] D.W. Murray, P.F. MacLauchlan, I.D. Reid, and P.M. Sharkey. Reactions to peripheral image motion using a head/eye platform. In *4th ICCV*, pages 403–411. IEEE Society, 1993.

[32] T. Viéville, P.E.D.S. Facao, and E. Clergue. Computation of ego-motion using the vertical cue. *Machine Vision and Applications*, 1994. To appear.

[33] T. Viéville and O.D. Faugeras. Computation of Inertial Information on a Robot. In Hirofumi Miura and Suguru Arimoto, editor, *Fifth International Symposium on Robotics Research*, pages 57–65. MIT-Press, 1989.

[34] O.D. Faugeras, B. Hotz, H. Mathieu, T. Viéville, Z. Zhang, P. Fua, E. Théron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy. Real time correlation-based stereo: algorithm, implementations and applications. *International Journal of Computer Vision*, 1994. In press.

[35] T. Viéville, Cyril Zeller, and Luc Robert. Using collineations to compute motion and structure in an uncalibrated image sequence, 1994. Accepted after review.

[36] G. Toscani and O.D. Faugeras. Camera calibration for 3D computer vision. In *Proceedings of the International Workshop on Machine Intelligence, Tokyo*, February 1987.

[37] R Y Tsai. An efficient and accurate calibration technique for 3D machine vision. In *IEEE Proc CVPR'86, Miami Beach, Fl., June*, pages 364–374, 1986.

[38] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37, 1971.

[39] Reyes Enciso, Thierry Viéville, and Olivier Faugeras. Approximation du changement de focale et de mise au point par une transformation affine à trois paramètres. Technical Report 2071, INRIA, 1993.

[40] L. Robert. *Perception Stéréoscopique de Courbes et de Surfaces Tridimensionnelles, Application à la Robotique Mobile.* PhD thesis, Ecole Polytechnique, Palaiseau. France, 1992. PhD thesis.

[41] Reg Willson. *Modeling and Calibration of Automated Zoom Lenses.* PhD thesis, Department of Electical and Computer Engineering, Carnegie Mellon University, 1994.

[42] R.G. Willson and S.A. Shafer. What is the center of the image ? In *IEEE Proc CVPR'93, New-York, June*, pages 670–671, 1993.

[43] J. Fairfield. Toboggan contrast enhencement. In *Applications of Artificial Intelligence, Machine Vision and Robotics*, volume 1708. Proceedings of S.P.I.E., 1990.

[44] E. Clergue. Méthodes de reconstruction denses pour la vision active. Technical report, Université de Nice, Septembre 1993. Rapport de Stage de DEA,.

[45] T. Voorhees and H. Poggio. Detecting textons and texture boundaries in natural images. In *Proceedings of the First International Conference on Computer Vision, London*, pages 250–258, June 1987.

[46] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* Wiley Interscience, New-York, 1973.

[47] S. Das and N. Ahuja. A comparative study of stereo, vergence and focus as depth cues for active vision. In *IEEE Proc CVPR'93, New-York, June*, pages 194–199, 1993.

[48] M.J. Anderson. Range from out-of-focus blur. Technical Report LIU–TEK–LIC–1992:17, Linköping University, 1992.

[49] Y. Xiong and S.A. Shafer. Depth from focusing and defocusing. In *IEEE Proc CVPR'93, New-York, June*, pages 68–73, 1993.

[50] Y. Bar Shalom and T. E. Fortmann. *Tracking and Data Association.* Academic-Press, Boston, 1988.

[51] Emmanuelle Clergue and Thierry Vieville. Methods for dense reconstruction in active vision. In *Proc. 17th European Conference on Visual Perception, Eindhoven*, 1994.

[52] H. Mathieu. The Multi-DSP 96002 board. Technical Report Technical Report, 153, INRIA, 1993.

[53] T. Viéville, S. Ron, and J. Droulez. Two dimensional saccadic and smooth pursuit response to an extrafoveal smooth movement. In Levy-Shoen A. and O'Reagan K., editors, *Proceedings of the 3rd European Conference on Eye Movements*, 1986.

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

[55] A. Berthoz and G.Melvill Jones. *Adpative Mechanism in Gaze Control.* Elsevier, Amsterdam, 1985.

[56] F. Chaumette and A. S. Santos. Tracking a moving object by visual servoing. In *Proceedings of 12th World Congress IFAC, Sydney, Autralia*, 1993.

[57] J.E.W. Mayhew, Y. Zheng, and S.A. Billings. Layered architecture for the control of micro saccadic tracking of a stereo camera head. Technical Report AIVRU 72, University of Sheffield, 1992.