An Approach for Target Tracking using Log-polar Images*

V. Javier Traver and Filiberto Pla
Department of Computer Languages and Systems
Campus Riu Sec
Jaume I University
E12080–Castellón, Spain
{vtraver|pla}@uji.es

ABSTRACT
Active vision brings important advantages for physically embodied artificial agents interacting with their environment. Gaze control is one of the important issues in active vision. In this paper, we address one subproblem of gaze control, namely, gaze stabilization, which appears when visually tracking a moving object is required. One approach to tackle this is by solving a motion estimation problem. On the other hand, foveal sensing is known to play an important role within active vision. Log-polar imaging is a biologically motivated foveal model with important benefits for tasks such as tracking. Therefore, here we propose an adaptation of a motion estimation algorithm, initially developed for cartesian images, to log-polar images. Experiments are included to illustrate the application of the approach to estimate the motion of a target in real image sequences, as well as to show how motion estimates can be used to drive a pan-tilt head, which conveys some benefits over the mere *passive* tracking approach.

KEY WORDS
Computer Vision; Log-polar imaging; Motion Estimation; Active tracking.

1 Introduction

Active vision is a powerful paradigm endowing artificial systems with greater possibilities to solve problems which are more difficult —or even impossible— to solve under a *passive* vision perspective [1]. Nonetheless, some new problems need to be addressed in the context of active vision. One of them is gaze control, which is needed, among other situations, in active tracking scenarios (face tracking for human-computer interaction, visual surveillance systems, visual servoing, etc.). The goal is to keep an object of interest in the center of the visual field, as this object moves through the environment. This particular problem, which we tackle in this work, receives different names in the literature (e.g., gaze stabilization, fixation, etc.). Foveal sensing, and *log-polar* vision [2] in particular, has received important attention from researchers in the field of active vision, because it offers some advantages over uniformly sampled images [3, 4].

A proper and robust approach for visual tracking is to formulate it as a motion estimation problem. Research in motion estimation [5] has a long tradition, and has attained remarkable results. Despite this significant progress, there are still challenging open problems. Additionally, the amount of work carried out in motion estimation in log-polar images is, comparatively, very scarce and quite recent [6, 7].

Motion estimation techniques developed for cartesian images cannot always be suitable for the log-polar geometry, or exhibit the high performance required in active vision problems. Thus, in this paper we propose a motion estimation algorithm which is based on an existing framework developed for cartesian images [8], which has several interesting characteristics, such as its efficiency and robustness, which turn out to be attractive for our problem. The behavior of the proposed technique is tested with examples of real image sequences, both with a static and a pan-tilt controlled camera.

A brief review of log-polar mapping is given in Sect. 2. The motion estimation technique and figure-ground segmentation are detailed in Sect. 3 and Sect. 4, respectively. Experimental work is described in Sect. 5. A discussion given in Sect. 6 concludes the paper.

2 Log-polar mapping

Log-polar images have a central area (the *fovea*) with a very high resolution, which decreases with the eccentricity (i.e., radially). Due to its geometry, log-polar mapping allows scalings and rotations about the optic axis become simple translations, thus simplifying many visual tasks. The log-polar model used here defines the log-polar coordinates (ξ, η) as [4]:

$$\left\{ \begin{array}{rcl} \xi & = & \log_a \left(\frac{\rho}{\rho_0} \right) \\ \eta & = & q \cdot \theta \end{array} \right. \quad (1)$$

with (ρ, θ) being the usual polar coordinates.

The parameters of the transform are $q = \frac{S}{2\pi}$ (the angular resolution)), ρ_0 (the radius of the innermost ring), and

*Research supported in part by projects GV97-TI-05-27 from the *Conselleria d'Educació, Cultura i Ciència, Generalitat Valenciana*, and CICYT TIC98-0677-C02-01 from the Spanish *Ministerio de Educación y Cultura*.

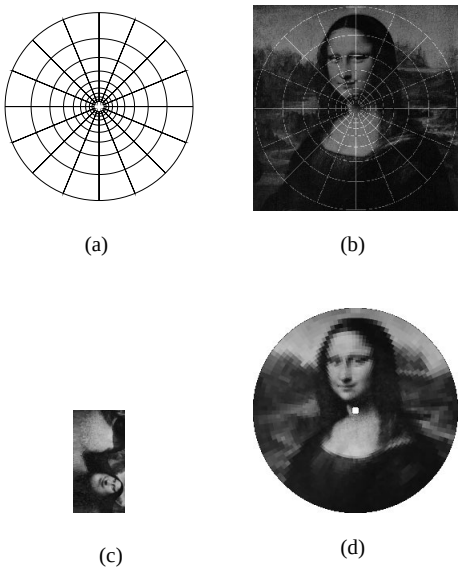


Figure 1. Log-polar mapping: (a) grid layout example (10×16), (b) original cartesian image (256×256), with grid (a) overlapped, (c) cortical image (64×128), (d) retinal image (256×256) reconstructed from (c) by the inverse mapping.

$a = \exp(\ln(\frac{\rho_{\max}}{\rho_0})/R)$ (the rate of radii growing), with R and S being the number of rings and sectors, respectively, of the log-polar image.

From their biological motivation, *retinal* images are those in the usual format, while *cortical* images are those resulting from the log-polar mapping (i.e., the log-polar images themselves). An example is shown in Fig. 1. It is worth noticing the important data reduction achieved by log-polar images, by comparing the sizes of images in Fig. 1(b) and Fig. 1(c).

3 Motion estimation

Let $I(\mathbf{p}, t)$ denote the gray-level value at a given image location $\mathbf{p}$ of an image I acquired at time t. A general parametric *motion model* is defined by $\mathbf{f}(\mathbf{p}; \boldsymbol{\mu})$, with $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_n)^{\mathrm{T}}$ the motion parameter vector. We have that $\mathbf{f}(\mathbf{p}; \mathbf{0}) = \mathbf{p}$. The image at an initial time t_0, I_0, will be denoted by the *reference image*, where a set of $N > n$ locations $\mathcal{R} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N\}$ define a *target region*. Let $\boldsymbol{\mu}^*(t)$ be the ground truth values of $\boldsymbol{\mu}$ at time t, and $\boldsymbol{\mu}(t)$ the corresponding estimate. If changes in subsequent images are only due to $\mathbf{f}$, then for any $t > t_0$, there is a $\boldsymbol{\mu}^*(t)$ such that $I(\mathbf{p}, t_0) = I(\mathbf{f}(\mathbf{p}; \boldsymbol{\mu}^*(t)), t), \forall\, \mathbf{p} \in \mathcal{R}$.

The image of the target region, transformed as of $\boldsymbol{\mu}$, can be written in vector notation as:

$$\mathbf{I}(\boldsymbol{\mu}, t) = \begin{bmatrix} I(\mathbf{f}(\mathbf{p}_1; \boldsymbol{\mu}), t) \\ I(\mathbf{f}(\mathbf{p}_2; \boldsymbol{\mu}), t) \\ \vdots \\ I(\mathbf{f}(\mathbf{p}_N; \boldsymbol{\mu}), t) \end{bmatrix},$$

which will be referred to as the *rectified image*, I_R. The estimation of the motion parameter vector $\boldsymbol{\mu}$ can be found by minimizing a *least squares* objective function which, in vector notation, can be written as:

$$O(\boldsymbol{\mu}) = ||\mathbf{I}(\boldsymbol{\mu}, t) - \mathbf{I}(\mathbf{0}, t_0)||^2. \quad (2)$$

In the absence of a good initial guess of $\boldsymbol{\mu}$, a costly global optimization procedure would be needed to optimize (2). However, in a visual tracking scenario, the continuity of motion provides this starting point. Thus, the problem can be reformulated to that of determining a vector of offsets $\delta\boldsymbol{\mu}$, such that $\boldsymbol{\mu}(t+\tau) = \boldsymbol{\mu}(t) + \delta(\boldsymbol{\mu})$. If the components of $\delta\boldsymbol{\mu}$ has a small magnitude, continuous optimization can be applied to a linearized version of the problem. Taking this into account, and with the additional approximation $\tau\mathbf{I}_t \approx \mathbf{I}(\boldsymbol{\mu}, t+\tau) - \mathbf{I}(\boldsymbol{\mu}, t)$, the solution is [8]:

$$\delta\boldsymbol{\mu} = -(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{M}^{\mathrm{T}}[\mathbf{I}(\boldsymbol{\mu}, t+\tau) - \mathbf{I}(\mathbf{0}, t_0)], \quad (3)$$

where $\mathbf{M}$ is the $N \times n$ Jacobian matrix of $\mathbf{I}$ with respect to $\boldsymbol{\mu}$.

As we are dealing with log-polar images rather than cartesian ones, I denotes a log-polar image, and its coordinates are $\mathbf{p} = (\xi, \eta)$. Although other motion models would be possible, in this paper we focus on a similarity motion model (i.e., translation, rotation and scaling), as it is reasonable simple and yet useful:

$$\mathbf{f}(\mathbf{p}; \boldsymbol{\mu}) = \mathbf{p} + \mathbf{t}_l(\boldsymbol{\mu}) + \mathbf{J}(\mathbf{p}) \cdot \mathbf{t}_c(\boldsymbol{\mu}), \quad (4)$$

where $\mathbf{t}_l = (r, s)^{\mathrm{T}}$ is a translation in the log-polar domain, and $\mathbf{t}_c = (b, c)^{\mathrm{T}}$ is the common cartesian translation. Therefore, our 4-parameter motion vector is $\boldsymbol{\mu} = (r, s, b, c)$. Note how the use of the log-polar geometry simplifies the expression of the motion model regarding rotation and scaling (they are just a translation; the actual rotation angle is $\phi = r/q$, and the actual scaling factor is found as $\alpha = a^s$). However, the usual translation is modified by the log-polar Jacobian matrix

$$\mathbf{J} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} = \frac{1}{\rho} \begin{bmatrix} \frac{\cos\theta}{\ln a} & \frac{\sin\theta}{\ln a} \\ -q\sin\theta & q\cos\theta \end{bmatrix}. \quad (5)$$

Note that the usual notation for partial derivatives, $A_b = \partial A/\partial b$, is used here, and will also be used below.

The matrix $\mathbf{M}$ is built taking into account the log-polar transform (Eq. 1) and the motion model (Eq. 4). Each element in $\mathbf{M}$, m_{ij}, is computed as:

$$m_{ij} = \begin{bmatrix} I'_{\xi'} & I'_{\eta'} \end{bmatrix} \cdot \begin{bmatrix} \xi'_{\mu_j} \\ \\ \eta'_{\mu_j} \end{bmatrix} \quad (6)$$

where $I' = I(\mathbf{p}')$, with $\mathbf{p}' = \mathbf{f}(\mathbf{p}; \boldsymbol{\mu}) = (\xi', \eta')$. Additionally, we have:

$$\begin{array}{ll} \xi'_{\mu_1} = \xi'_r = 1 & \eta'_{\mu_1} = \eta'_r = 0 \\ \xi'_{\mu_2} = \xi'_s = 0 & \eta'_{\mu_2} = \eta'_s = 1 \\ \xi'_{\mu_3} = \xi'_b = \xi_x & \eta'_{\mu_3} = \eta'_b = \eta_x \\ \xi'_{\mu_4} = \xi'_c = \xi_y & \eta'_{\mu_4} = \eta'_c = \eta_y \end{array}$$

where the partial derivatives ξ_x, η_x, ξ_y, and η_y are given by the Jacobian $\mathbf{J}$ (Eq. 5).

Notice that, in Eq. 6, $\mathbf{p}$ represents a different image location at each row $i \in [1, \ldots, N]$ of matrix $\mathbf{M}$, while column $j \in [1, \ldots, n]$ denotes each of the four motion parameters (here, $n = 4$).

An interesting advantage of using log-polar images, instead of cartesian ones, is that we do not explicitly select a target region, as it is done in [8]. Our $\mathcal{R}$ will therefore be the whole image (i.e., $N = R \cdot S$), Because of the small size of the log-polar images, this does not imply a loss in efficiency. The *implicit focus-of-attention* of log-polar images [7] will effectively deal with images with a foveated target, even when it only occupies a small part of the visual field, without the background becoming too distracting.

4 Figure-ground segmentation

As mentioned above, we make no *a priori* selection of what the target region will be. On the contrary, if a moving object (the target) is kept foveated (which is the case with an active tracking mechanism), it is possible to automatically discover the target and segment it from the background. To that end, we propose the following probabilistic approach, in which pixels are classified as either *target* or *background* pixels.

Let $P_c(\mathbf{p})$ be the *current* probability of the pixel at $\mathbf{p}$ being a target pixel, which can be estimated on the basis of the most recently estimated motion. Let $P_h(\mathbf{p}, t)$ be the *historic* probability of the same pixel being a target pixel. This *history* of the target is updated at each time step t as $P_h(\mathbf{p}, t) = \lambda \cdot P_h(\mathbf{p}, t-1) + (1-\lambda) \cdot P_c(\mathbf{p})$, where $\lambda \in [0, 1]$ can be regarded as a *forget/memory* factor, which weights the historic probability against the more recent confidence. Initially, $P_h(\mathbf{p}, t_0) = 0$ (i.e., no target identified yet).

As for $P_c(\mathbf{p})$, we choose to compare the reference and rectified images (I_0 and I_R) on a pixel-by-pixel basis, by using the squared frame difference function $D(\mathbf{p}) = (I_0(\mathbf{p}) - I_R(\mathbf{p}))^2$. The rational behind this is that, if motion estimates are accurate enough, the rectified image will look similar to the initial, reference image, at those pixels which belong to the target (whose motion is being estimated). Then, to get a probability from D, we use $P_c(\mathbf{p}) = \exp\left(-\frac{1}{2}\frac{D(\mathbf{p})}{\sigma^2}\right)$, with σ being a noise estimate.

Often, it is important for the tracking process to supervises its own performance. We suggest to use the normalized cross correlation, as a *stabilization index* κ,

$$\kappa = \frac{1}{2} - \frac{\sum_{\mathbf{p}} (I_0(\mathbf{p}) - m_0) \cdot (I_R(\mathbf{p}) - m_R)}{2\sqrt{\sum_{\mathbf{p}} (I_0(\mathbf{p}) - m_0)^2 \cdot \sum_{\mathbf{p}} (I_R(\mathbf{p}) - m_R)^2}}$$

as a unique measure of how similar the reference and rectified images are, which is a good indication of the tracking performance. Interestingly, $\kappa \in [0, 1]$, with 1 denoting a perfect match. In this expression, m_0 and m_R denote the average gray level in I_0 and I_R, respectively.

5 Experiments

To show the performance of the algorithm, we report two experiments. In both of them, we used a Sony EVI-G21 camera and a Matrox Meteor frame grabber. The code is written in C++, and it runs on a PC computer under Linux as the operating system. As a relevant detail for this work, the camera used is mounted on a pan-tilt mechanism, so that it can be commanded to point to different points in space by controlling its pan and tilt angles. Captured (cartesian) frames are transformed by software to as small as 32×64-sized log-polar images.

Passive tracking experiment. The first experiment is aimed to show the effectiveness of the approach to estimate motion in real image sequences, as captured from our camera, which is fixed in its home position (i.e., the pan and tilt degrees of freedom are not used in this case). When ground truth of the motion parameters is not known, as it is the case in many real-world experiments, it is helpful to make some simplifications to build up a set-up with some controlled conditions, so that this help assess the results. To that end, we placed an object in front of the camera, and moved it while describing a circular trajectory (the radius of the circle is 25 mm). The same trajectory was executed three times. As it was moved manually, the trajectories followed are not very accurate. Similarly, the speed at which the object moved is, only approximately, the same. The top view of this set-up is shown in Fig. 2. With the object moving like this, it can be expected that the dominant motion will be the horizontal translation, and also some scaling. The horizontal shift b estimated at each frame, is displayed in Fig. 3. Notice that the same pattern is repeated three times, corresponding to the three turns the object underwent.

Some images of the sequence are shown in Fig. 4. Some points deserve mentioning. First, it is important to note that the object is only occupying part of the whole image, and without any explicit segmentation, the algorithm is able to estimate the target's motion. This is possible because of the so called focus of attention, i.e., the predominance of fovea pixels over the background, a built-in feature of log-polar images because of its space-variance

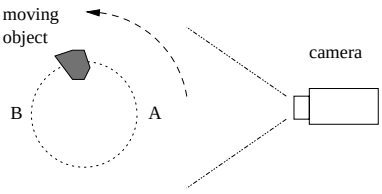


Figure 2. The top view of the set-up for the experiments.

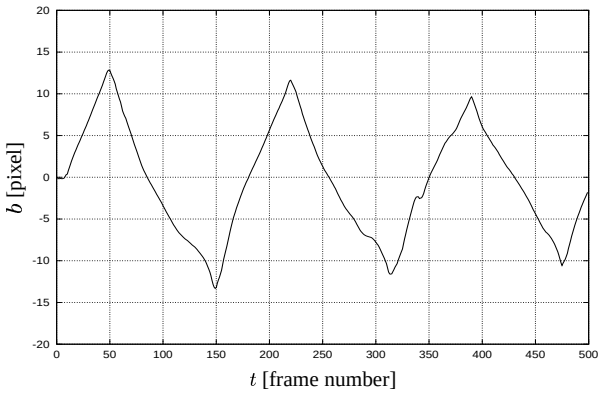


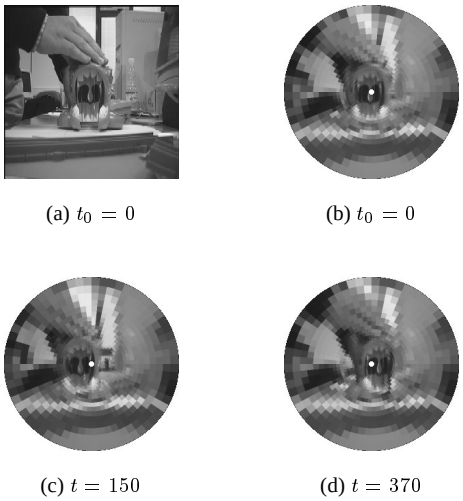
Figure 3. Estimation of the horizontal translation (motion parameter b).

resolution. For tracking purposes this turns out to be a very interesting advantage.

Second, the object passed approximately through the same positions at each of the three turns. However, the extreme values of b are not the same at each turn, as they ideally should. As an example, compare, in Fig. 4, the value of b at $t \approx 50$, at $t \approx 220$, and at $t \approx 390$. This phenomenon is partly due to the "delay" occurring in the motion estimates, probably associated with having $\delta\boldsymbol{\mu}$ values somehow bigger than expected. Even though $\delta\boldsymbol{\mu}$ should be small, the accumulated $\boldsymbol{\mu}$ can be quite large. How big it can be depends on the object size and image resolution.

Also, notice that the object is initially centered in the field of view, but as it is moved around the circle, the eccentricity of its image projection grows, as can be seen in the example frames provided in Fig. 4. This implies that the image projection of the object moves from a high to a lower resolved area in the image. Motion estimates, however, are quite robust to this loss of resolution. This property has an obvious limitation, and moving the object too far from the center would carry a serious inconvenience to the motion estimation algorithm.

This last observation leads naturally to the consideration of the role of active tracking when space-variant images are used. As it happens with the human visual behavior, tracking a moving object requires eyes and/or head motion to reposition the fovea over the object of interest (*foveation*).



(a) $t_0 = 0$ (b) $t_0 = 0$

(c) $t = 150$ (d) $t = 370$

Figure 4. Some frames at some time steps: (a) the original cartesian image; (b)–(d) some retinal images (obtained by the log-polar transform of the corresponding cartesian images). Note that (b), the first frame captured, is taken as the reference image. Below each image its corresponding frame number is shown.

Active tracking experiment. In this second experiment, the object was made to move also circularly, but along a larger circle (the radius measured 35 mm). In this case the object is made to move mainly along the y cartesian coordinate direction.

To control the pan and tilt angles, φ_p and φ_t, respectively, we use the usual approximation consisting of modeling the relationship between retinal shifts (b, c) and increments in these angles $(\delta\varphi_p, \delta\varphi_t)$, required to cancel these shifts, as a simple linear one:

$$\begin{cases} \delta\varphi_p &= \gamma_x \cdot b \\ \delta\varphi_t &= \gamma_y \cdot c \end{cases} \tag{7}$$

where the parameters γ_x and γ_y depend on the intrinsic parameters of the camera. To find a value for these parameters, we performed this simple calibration procedure. Pan was moved in small increments, and an image was taken and log-polar transformed at each position. Motion was estimated following the algorithm presented in Sect. 3. Then, the parameter γ_x was computed so that $\varphi_p(t) = \gamma_x \cdot b(t)$. The same procedure was followed to find γ_y, but this time by varying the tilt angle.

Two basic control strategies are possible: control in position and in velocity. Control in velocity tends to move the camera more smoothly, but entails some extra complication. At the time being, we stick to the simpler control in position.

Notice that the active tracking mechanism is always trying to cancel the retinal offsets. This scheme does not let the motion estimation algorithm perform what it does

so well —integration over time by exploiting motion continuity. By constantly re-centering the target, we leave little chance for the tracker to improve its estimates. An alternative could consist of letting the tracker work for a while, and only after motion is bigger than a certain amount, use the active tracker to cancel the estimated motion. Unfortunately, this also poses some problem: the camera movements are very jerky.

Upper row in Fig. 5 shows some images in the sequence. The main point to notice here is that even though the target moved over a wider range of distances than in previous experiment, active tracking keeps the target close to the fovea. Having the target at this high resolution area favors the performance of the estimation algorithm.

In lower row in Fig. 5, results of target segmentation are shown. Probabilities $P_h(\mathbf{p})$ are coded as gray levels, the brighter the level the higher the probability. It can be observed how the probabilities evolve: on the one hand, probabilities associated to pixels belonging to the target increase, because the target is kept foveated over time. Meanwhile, background pixels change as the result of the active tracking and, therefore, they are more unlikely to increase their probabilities. Although the segmentation is not perfect, it illustrates the potential of the approach. The approach is based on having a non-uniform, changing background over time. If uniform areas in the background persist in time, it is more difficult to get rid of them. In this sense, active tracking can improve the results —provided the object moves over distinct background over time. On the other hand, better figure-ground segmentations could readily be used to improve the active tracking process (e.g., by taking the position of the target into account).

In Fig. 6 some measures are plotted. Evolution of the camera's tilt angle (upper row) clearly illustrates how the pan-tilt head was driven to actively pursuit the tracked object. It can be verified that the object performed two turns and a half. The basic motion component during the experiment was a vertical shift, but it is virtually canceled by the tracking process. Another important motion component present is change of scale, as the object moves back and forth. Estimates of α, the scale factor, are displayed in the middle row in this same figure. It can be checked, for instance, how $\alpha \approx 1$ at the approximate time moments when $\varphi_t(t) = 0$ and $\partial\varphi_t(t)/\partial t > 0$, which corresponds to the crossings of the target through its initial departure position (point A in Fig. 2). Finally, the stabilization index κ (lower row) is a good measure to evaluate how the tracking is performing. In this experiment, values for κ are always greater than 0.8.

In both experiments, the object was moved slowly for two reasons. One is that the current frame rate is not very high. The other reason is to take care of not violating the main assumption in [8] of having small values for $\delta\boldsymbol{\mu}$ at each time step. Chances of overcoming this limitation seem possible [8]. The use of multiple levels of image resolution is a typical resort to cope with both small and big image velocities: the lower the speed, the higher the resolution, and

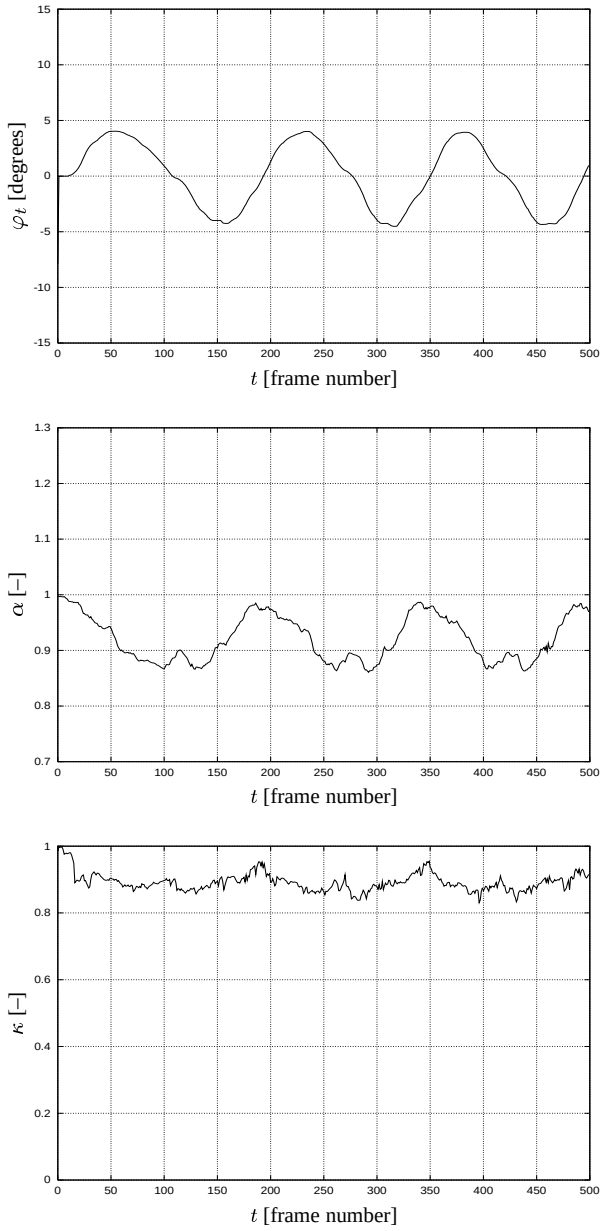


Figure 6. Evolution, while tracking a moving target, of φ_t, α, and κ, respectively.

conversely. Therefore, processing dynamically proceeds at the proper resolution, by adapting to the target's dynamics. How multi-resolution hierarchies would operate in log-polar images, which are multi-resolved by themselves, needs a careful study. Another possibility consists of modeling the target's motion and take this information into account to predict the object's future positions and velocities. This is probably a good choice when combined with an active tracking mechanism. Kalman filters [9] are well-known mathematical tools for these purposes, not without their limitations and problems [10].

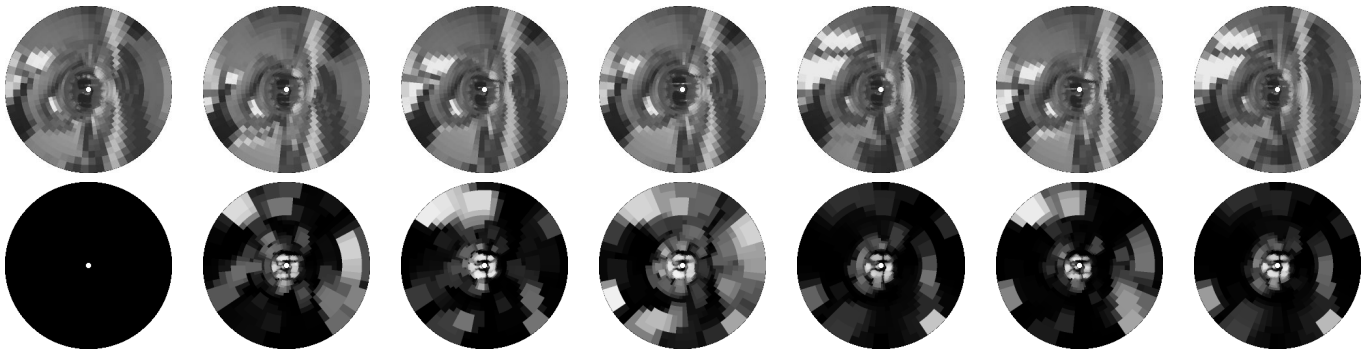


Figure 5. Some frames and target probabilities in the sequence of the second experiment.

6 Conclusions

A motion estimation technique for log-polar images has been presented. The great reduction of image data to be processed and its implicit focus of attention are the main benefits implied by the use of log-polar vision. A simple but quite effective figure-ground segmentation has also been proposed. The effectiveness of the approach are illustrated through two examples, one with a static camera and another with the pan-tilt head actively pursuing a moving target. The benefits of active tracking have been showed, and can be summarized as follows: (1) it keeps the object foveated, thus having better image stabilization, and favoring the motion estimation process; (2) it gets better segmentation results, because the gray level of background pixels are more likely to change due to camera motion. An additional advantage, integration of perception and control, can still be mentioned. For example, the rectification process could be simplified, by performing it along the log-polar coordinates, whereas rectification along the x and y coordinates would automatically be done —and for free!— by active tracking. This would be especially advantageous using the log-polar imaging model.

Further work can be directed to overcome the limitation of the algorithm of requiring small motions from step to step. Chances to do this seem possible, but challenging. An increase of the frame rate in the implementation is also a key issue. Improving the control algorithm would yield a more robust tracking and a smoother camera motion. Enhancements in target segmentation would allow its results to be used to make tracking more effective.

References

[1] John (Yiannis) Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *Intl. Journal of Computer Vision*, pages 333–356, 1988.

[2] Marc Bolduc and Martin D. Levine. A review of biologically motivated space-variant data reduction models for robotic vision. *Computer Vision and Image Understanding (CVIU)*, 69(2):170–184, February 1998.

[3] Massimo Tistarelli and Giulio Sandini. On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 15:401–410, 1993.

[4] C. Capurro, F. Panerai, and G. Sandini. Vergence and tracking fusing log-polar images. In *Intl. Conf. on Pattern Recognition (ICPR)*, pages 740–744. IEEE, 1996.

[5] Christopher Stiller and Janusz Konraad. Estimating motion in image sequences: A tutorial on modeling and computation of 2D motion. *IEEE Signal Processing Magazine*, pages 70–91, July 1999.

[6] Hilary Tunley and David Young. Dynamic fixation of a moving surface using log polar sampling. In *British Machine Vision Conference*, volume 2, pages 579–588, Univ. York, York, September 1994. BMVA Press.

[7] Alexandre Bernardino, José Santos-Victor, and Giulio Sandini. Tracking planar structures with log-polar images. In *Symp. on Intelligent Robotic Systems*, Reading, UK, July 2000. (Also as VisLab TR 06/2000).

[8] Gregory D. Hager and Peter N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 20(10):1025–1039, October 1998.

[9] Peter S. Maybeck. *Stochastic Models, Estimation and Control*. Academic Press, New York, 1979.

[10] Per-Olof Gutman and Mordekhai Velger. Tracking targets using adaptive Kalman filtering. *IEEE Trans. on Aerospace and Electronic Systems*, 26(5):691–698, September 1990.