

Disparity Estimation on Log-Polar Images and Vergence Control

R. MANZOTTI, A. GASTERATOS, G. METTA, G. SANDINI

Laboratory for Integrated Advanced Robotics (LIRA-Lab)
Department of Communications Computer and System Science
University of Genoa
Viale Causa 13, I-16145, Italy
{manzotti, antonis, pasa, [giulio](mailto:giulio@lira.dist.unige.it)}@lira.dist.unige.it

Abstract

An important issue in the realization of an autonomous robot with stereoscopic vision is the control of vergence. Together with version, it determines uniquely the position of the fixation point in space. Vergence control is directly related to both depth perception and *binocular fusion*. Previous works in this field employed either a measure of correlation of stereo images or some kind of disparity related estimate. In this paper, we present a new method to extract a global disparity measure for vergence control, which does not require a priori segmentation of the object of interest. Our method uses images acquired by retina-like sensors and, therefore, the computation is performed in the log-polar plane. The technique we present here is: i) global, in the sense that it is an integral measure over the whole image, ii) computationally inexpensive, considering that the goal was to use it in the robot control loop rather than to accurately measure some 3D world features. Moreover, the proposed technique is robust and independent of the average illumination as well as of other features of the target such as size, shape and direction of motion. It provides a precise and linear estimate of the vergence error, which is the only requirement from the control point of view. Several experimental results on a real robotic setup demonstrate the effectiveness of the proposed technique.

Index terms: active vision, log-polar, disparity, vergence.

List of symbols

θ, θ'	<i>Coordinates in cortical plane</i>	<i>Eta, csi</i>
ρ, θ	<i>Coordinates in retinal plane</i>	<i>Rho, theta</i>
x, y	<i>Coordinates in Cartesian plane</i>	
θ_0	<i>Innermost circle</i>	
θ', θ	<i>First derivatives of vergence angle, vergence angle</i>	<i>theta</i>
θ	<i>Eye angle</i>	
C	<i>Global index of binocular fusion</i>	
$I, I_l, I_r,$	<i>Images</i>	
$I(,)$	<i>Point of image</i>	
K	<i>Constant gain</i>	
C	<i>First derivatives of the vergence angle</i>	
θ_r, θ_l	<i>Mean values of images</i>	<i>mi</i>
t	<i>Time</i>	
r	<i>Distance of the fixation point from baseline</i>	
B	<i>Baseline length</i>	
d	<i>Disparity</i>	
χ	<i>Discrete function of disparity</i>	<i>chi</i>
δ	<i>Minimum disparity</i>	<i>delta</i>

1. Introduction

In building a robotic stereo head an extremely important degree of freedom is represented by vergence. Given a stereoscopic vision system (figure 1), the vergence angle, together with version and tilt angles, describes uniquely the fixation point in space. The problem we are dealing with here is controlling the vergence angle only with the assumption that other subsystems maintain the object of interest close to the image center. For instance, we could imagine a tracking module, which deals with the problem of following the target in space (by controlling version and tilt) or a saccade-like control to quickly foveate a possibly interesting target [1].

If vergence could be controlled effectively, several advantages would arise in subsequent image processing. These include easier fusion of stereo images in one "cyclopean image", easier figure-ground segmentation (e.g. by means of zero disparity filters [2]). Of course, if a tracked object is stable in the retinas, further image processing is facilitated. Indeed, it is not just a coincidence that most of the biological stereoscopic vision systems in the higher branches of the evolutionary tree possess a developed and specialized vergence control system [3]. If we consider biological systems, several kinds of disparity exist (horizontal, vertical, and rotational) in order to implement horizontal and cyclo-vergence [4]. In this paper only the horizontal component, (by far the most relevant for processing and control) is considered. Maintaining a correct vergence angle should not be seen as a goal in itself but as a way to improve the performance and robustness of successive visual computation. Particularly relevant in this respect is the intrinsic limit introduced by a correct binocular fusion in the computation of binocular disparity and 3D feature extraction. Vergence angle, moreover, provides a measure of absolute distance, even if limited to a point in space, as well as a reference point in the environment [5]. Therefore, a general principle is that vergence control should be implemented keeping in mind its further use by the whole system [6].

In the implementation presented here we "only" consider disparity measure, which even if not the only source of information useful for vergence control [7], is possibly the most direct and relevant. Disparity estimation has been performed according to several techniques. These are based either on correlation [8-10], matching [11, 12], phase difference [13, 14] or Bayesian methods [15, 16]. Despite this big variety of disparity estimation methods, examples employing log-polar images are quite occasional [17]. The reason is the complex geometrical layout of log-polar images, which apparently is not well suited for disparity computation. Log-polar images, however, are ideal for vergence control tasks. They provide high resolution in the fovea, where the target should to be located, and a wide field of view at the same time [18]. These features, along with a computationally simple mapping technique used for disparity estimation, allows real-time performance to be achieved with high accuracy.

The experimental results reported in order to show the feasibility of the approach have been carried out with one main goal in mind: to demonstrate that horizontal disparity alone allows an efficient and robust vergence control. The paper is organized as follows. The essentials of log-polar images are given in section 2. Several techniques, which have been applied for vergence control, are discussed in section 3. The proposed disparity estimate technique is extensively described in section 4. In section 5, the application of the proposed technique for vergence

control is presented. Section 6 contains the experimental results and, finally, section 7 contains some concluding remarks.

2. Log-Polar Images

Studies on the primate visual pathways from the retina to the visual cortex have shown that the geometrical layout follows an almost regular topographic arrangement [19-21]. These results can be summarized as follows:

- ?? The distribution of the photoreceptors in the retina is not uniform. They lay more densely in the central region called fovea, while they are sparser in the periphery. Consequently the resolution also decreases, moving away from the fovea toward the periphery. It has a radial symmetry, which can be approximated by a polar distribution.
- ?? The projection of the photoreceptors array to the primary visual cortex can be well approximated by a logarithmic-polar (log-polar) distribution mapped onto a rectangular-like surface (the cortex). Here the representation of the fovea is quite expanded, i.e. more neurons are devoted to it, and the periphery is represented using a coarser resolution.

From the mathematical point of view the log-polar mapping can be expressed as a transformation between the polar plane (r, θ) (retinal plane), the log-polar plane (ρ, θ) (cortical plane) and the Cartesian plane (x, y) (image plane), as follows:

$$\begin{aligned} \rho &= r/q \\ \theta &= \theta \end{aligned} \quad [1.1]$$

where r_0 is the radius of the innermost circle, $1/q$ is the minimum angular resolution of the log-polar layout and (r, θ) are the polar co-ordinates. These are related to the conventional Cartesian reference system by:

$$\begin{aligned} x &= r \cos \theta \\ y &= r \sin \theta \end{aligned} \quad [1.2]$$

Figure 2 illustrates the log-polar layout as derived by equations (1.1) and (1.2). In particular, the grid on the left represents a standard Cartesian image mapped according to equation (1.1). The plot on the right shows the corresponding cortical image.

3. Vergence Control

Vergence control issues have been addressed by means of several different techniques. It is worth noting that all of them are somehow related to the estimation of 3D features. These techniques can be classified as follows:

- 1) Segmentation techniques [11, 12, 17, 22, 23].
- 2) Fusion index [24, 25].
- 3) Direct disparity estimation [9, 26].

The first group requires some sort of heuristics to identify the object of interest (segmentation). The main problem, in this case, is their lack of flexibility. We would like to stress the fact that the segmentation of the object from its background is not an easy task itself. Many of the proposed systems do not use any direct control of the vergence angle, but they rather control each eye separately. These approaches may, for instance, fail in the presence of a false matching, i.e. the robot might try to follow two different targets. Moreover, in biological systems, vergence control is a relatively low-level functionality, which does not require an actual segmentation or recognition of the target object. On the contrary, these techniques act at a higher level of abstraction, i.e. the object need to be segmented and/or recognized.

The second group exploits the fact that if an object is correctly verged the stereo images should be very similar, at least around the fixation point. Of course, this does not hold exactly as the two images are never the same. However, on standard conditions, such as typical fixation distance, optical parameters and kinematics of robot heads, the difference between the images is rather small. In that case the images are said to be binocularly fused. The goal of a vergence control system is to minimize this difference. The global index of binocular fusion is an example of such a difference estimate. It can be computed using the normalized correlation technique [27]:

$$C(I_l, I_r) = \frac{\int_{x_0}^{x_1} \int_{y_0}^{y_1} (I_r(x, y) - \bar{I}_r) (I_l(x, y) - \bar{I}_l) dx dy}{\sqrt{\int_{x_0}^{x_1} \int_{y_0}^{y_1} (I_r(x, y) - \bar{I}_r)^2 dx dy \int_{x_0}^{x_1} \int_{y_0}^{y_1} (I_l(x, y) - \bar{I}_l)^2 dx dy}} \quad [1.3]$$

where I_r and I_l are the right and left image and \bar{I}_r , \bar{I}_l represent their mean values respectively. $C(t)$ is almost invariant to changes of illumination. It is normalized in the range [0,1]. The normalized correlation measure can be employed in a standard proportional control law, as follows:

$$\dot{\theta} = K(C - C_0) \quad [1.4]$$

where K is a constant gain and $\dot{\theta}$, C are the first derivatives of the vergence angle and the fusion index respectively [24, 25].

However, this approach has several drawbacks:

- 1) $C(t)$ is not a linear estimation of the angular error.
- 2) $C(t)$ is constant, if the eyes are still and the object is not moving ($\dot{\theta} = 0$).
- 3) $C(t)$ has minima whose values are variable with image characteristics in a non-linear and unpredictable fashion.

For these reasons, though feasible, the use of $C(t)$ in vergence control is limited.

The third group concerns the use of direct estimates of the binocular disparity. It is, in our view, the most promising one, because it uses disparity directly. The latter can be easily related to the vergence control error. In fact, it can be shown that binocular disparity is related to depth, which is in turn related to the vergence angle. Consider again the situation depicted in figure 1; using simple geometry, it is easy to derive the following equation:

$$r = b \frac{\cos(\theta_r)\cos(\theta_l)}{\cos(\theta_v)\cos(\theta_p)} \quad 1.5$$

where r is the distance of the fixation point from the baseline, b the baseline length, θ_r, θ_l the eye angles, $\theta_v = \theta_l + \theta_r$ represents the vergence angle and $\theta_p = (\theta_l - \theta_r)/2$ the version (or gaze) angle. In principle, equation 1.5 can be used to find out the vergence angle required to move the fixation point from one location to another.

On the other hand disparity is related to depth z (relative to the fixation point reference frame) by the following approximate equation:

$$z = \frac{K(f, b)}{f \tan(\theta)} (x_r - x_l) \quad 1.6$$

where $K = b/\sin(\theta)$, f is the focal length of the cameras and $x_r - x_l$ the binocular disparity. Note that equation 1.6 is valid only in a neighborhood of the fixation point. Using equation 1.5 and 1.6 together, it is thus possible to link binocular disparity to vergence error directly. Moreover, we have actually converted the problem of estimating the vergence error in that of estimating disparity. (i.e. starting from disparity it is possible to compute depth by using equation 1.6, and from depth, which in this case represents the required motion, recover the vergence angle movement by employing equation 1.5. The goal of the controller is indeed that of zeroing depth).

Unfortunately, the direct disparity estimation approach has been undermined by some practical difficulties. In fact, in order to reduce the computational burden, an attention region should be selected in advance. This means that the object has to be segmented from the background; this has been proved to be a hard task in itself. On the other hand, this is not really an issue if we utilize log-polar images and we constrain ourselves to a global estimate of the object disparity. Of course there is nothing magic about the use of retina-like sensors. However, assuming the object of interest is close to the center of the image, its relevance (e.g. number of "pixels") becomes higher than that of more peripheral objects (its projection in the log-polar plane is effectively magnified) [28].

4. Disparity

In order to compute disparity it is necessary to solve a correspondence problem. That is, we have to establish which pixels on the left and right image planes map to the same point in space.

Formally, considering a standard Cartesian images case, a possible solution can be written as:

$$d(x, y) = \arg \max_d \{I_l(x, y), I_r(x + d, y)\} \quad [1.7]$$

Roughly speaking, the equation above can be explained as follow:

- i) For each pixel (x, y) on I_l shift the corresponding pixel on I_r by a quantity d .
- ii) Compute a similarity measure ρ for each possible shift $d \in [-d_{\max}, d_{\max}]$.
- iii) The estimated disparity is the value of d , which gives the best similarity measure.

The similarity measure ρ can be either a sum of squared difference (SSD) or some other criteria such as the normalized correlation. The usual approach is to consider a neighborhood of the pixel (x, y) . As regards the disparity measurement coordinates system, it must be noted that disparity is usually expressed with respect to the image coordinates system. One might argue that in order to employ it for vergence control it has to be converted into the appropriate motor coordinates system. Assuming symmetric vergence and a simple pinhole camera model, the two reference frames are related by the following equation:

$$d = 2f \sin\left(\frac{\Delta \theta_v}{2}\right) \quad [1.8]$$

where $\Delta \theta_v$ is the difference in the vergence angle, f the camera focal length, and d the measured disparity. It is worth noting that because equation 1.8 is monotonic in the required domain (i.e. $\Delta \theta_v \in [-\theta_v, \theta_v]$). Roughly speaking because the measure is well formed in Lyapunov sense ([29]) the closed loop system will be stable even if a simple PD controller is used.

Furthermore, the shift of image pixels can be represented by a disparity operator $disp_{cart} : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ defined as follow:

$$disp_{cart}(x, y, d) = [x + d, y] \quad [1.9]$$

where d represents the disparity and (x, y) a point in the image plane.

By applying equation 1.9 to an image point (x, y) , we can generate the corresponding matching point at disparity d . The rationale of defining $disp_{cart}$ will be clear when we will deal with non-uniform mapping such as the log-polar one. Combining equation 1.7 and 1.9 yields:

$$d(x, y) = \arg \max_d \{I_l(x, y), I_r(disp_{cart}(x, y, d))\} \quad [1.10]$$

In a general case, in order to apply equation 1.10 and recover the disparity, we need to provide:

- i) The disparity function $disp$, which embeds the description of the image geometry.

ii) The similarity function δ , which defines a distance measure between pixel blocks.

Considering the log-polar case, we just need to replace the $disp$ operator with a suitable one for that particular topology. However, the mapping is not simple using log-polar images. A simple horizontal shift in Cartesian coordinates is mapped to a complex curve in log-polar coordinates. Given a (x, y) pair the disparity operator $disp_{log}: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is defined as:

$$disp_{log}(x, y, d) = (x + d, y) \quad [1.11]$$

The transformation is now given by:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r \cos \theta \\ r \sin \theta \end{pmatrix} = \begin{pmatrix} \sqrt{a^2 + d^2} \cos \left(\arctan \frac{d}{a} \right) \cos \theta \\ \sqrt{a^2 + d^2} \cos \left(\arctan \frac{d}{a} \right) \sin \theta \end{pmatrix} \quad [1.12]$$

By applying the $disp$ operator we can roughly simulate a horizontal shift of (I_b, I_r) without actually moving the cameras. In practice this is only approximately true because of two main reasons:

- 1) Space variant images introduce a corresponding space variant distortion. In fact, we cannot recover the missing information belonging to the periphery, where the resolution is coarser.
- 2) A real camera motion also distorts the image points along the vertical axis. However, the $disp$ operator, as it has been defined in equation 1.11, does not take this into account.

Concerning the space-variant resolution we are limited by the fact that the information loss on the periphery cannot be avoided. With regard to the contribution of the vertical disparity, it is a rather small effect, and we can neglect it as a first approximation. Figure 3 shows how a regular grid in the left image would be mapped in the right one for increasing values of disparity (in the range zero to nine pixels). These are exactly the graphical representations of equations 1.11 and 1.12.

Figure 3 about here.

Furthermore, as our objective was to measure a global disparity index, Equation 1.10 can be further simplified by including in the aforementioned “neighborhood” of the current pixel all the image pixels. In this case the disparity index d is no longer dependent on the pixel position itself. A final comment regards the computational load associated with equation 1.12. We can note that the equation depends only on the log-polar geometry. In fact it is not dependent on the

actual images. Therefore, all calculations can be performed in advance for all possible disparities and stored into a fixed Connection Map (CM). A CM is basically a look-up table (LUT), which implements a mapping according to equation 1.12. From a more general point of view, we can see the CM as a network, where the CM values represent network nodes, and the connections the log-polar geometry itself. This suits very well our conceptual biological bias. It is easy to imagine how these correspondence maps might be implemented in parallel using several layers of neurons.

4.1 Disparity computation

As it is stated above, in order to compute the disparity, we need to store the geometrical transforms. The corresponding set of CMs can be defined as follows:

$$CM_SET = \{CM_d, d = 0 \dots N\} \quad [1.13]$$

$$CM_d = \{(x_n, y_n)_{x,y} : x = 0 \dots x_{max}, y = 0 \dots y_{max}\} \quad [1.14]$$

where x_{max}, y_{max} is the size of the log-polar images in cortical coordinates, N the total number of CMs and:

$$(x_n, y_n)_{x,y,d} = disp_{\log}(x, y, (\frac{d}{N})^{x_{max}/2}) \quad [1.15]$$

Using this notation, an image transform can be rewritten as:

$$I_r = disp_{\log}(I_l, d) \quad [1.16]$$

Implicitly, in this notation is the fact that we are actually mapping all the pixels of the left image I_l into those of the right one I_r . That is:

$$I_r(x, y) = I_l(x_n, y_n)_{x,y,d} \quad [1.17]$$

It is worth noting that the set of CMs has a finite size. Consequently, in order to evaluate the arg max function over d we need to evaluate only a finite number of possible disparities. In other words, at each time instant t the result is the following discrete function (which is function of the disparity d):

$$C(d, I_l, I_r) = C(I_r, disp_{\log} I_l) \quad [1.18]$$

where $C(I_r, I_l)$ is the normalized correlation function of equation 1.3, i.e. we employed the normalized correlation as similarity criterion.

Figure 4 shows an exemplar plot obtained using the procedure described above. Without loosing the generality we can define the disparity function, for a given stereo pair (I_l, I_r) , as:

$$\rho(d) = \rho(I_l, I_r, d) \quad [1.19]$$

The global disparity index is simply:

$$d_t = \arg \max_x \rho(x) \quad [1.20]$$

It is worth stating that:

- 1) Given the fact that the disparity function $\rho(d)$ is an integral correlation measure over the whole image it is extremely robust and reliable.
- 2) $\rho(d)$ represents the global image disparity because its values are the result of the correlation of the entire image with its corresponding image, shifted by the geometrical transformation due to d .
- 3) The significant element in $\rho(d)$ is its maximum, which corresponds to the most common disparity value among the image points.
- 4) most of the noise is rejected, simply because it is located out of the global maximum.

As a matter of fact, a precise measurement of disparity requires a great resolution. In fact, even small disparity (in the sub-pixel range) might correspond to a large variation in the object distance (more than 10 cm). To prevent such a loss of accuracy it is possible to increase the image resolution. However, this is, in general, a resource consuming approach. Consequently we devised a few techniques to increase the estimated accuracy without varying the image resolution itself. They are presented in detail in the next paragraphs.

4.1.1 Disparity Non Uniform Sampling

One of the main drawbacks of the previously described correlation technique is the lacking of accuracy. The fact that we have used a set of precomputed CMs restricts disparity to one of these CMs. Therefore the accuracy is limited by the disparity difference, which corresponds to the minimum distance between CMs. In other words the maximum accuracy is:

$$d_{\min}^1 = \frac{\rho_{\max}}{N} \quad [1.21]$$

where N is the number of CMs, and ρ_{\max} the log-polar image radius.

A possible solution would be to increase the numbers of CMs (increasing N) but this would increase the computational cost accordingly. A second solution would be to reduce the radius of the log-polar image ρ_{\max} , reducing, consequently, the field of view. An alternative approach exploits the space variant techniques used by using the same number of CMs distributed in a non-linear fashion. In this way resolution for small disparities is improved without increasing the computational load. This solution does not affect the overall system performance, since for high disparity values there is no need for precise disparity estimation because the higher the value of

disparity the higher is the distance of the world point from fixation. On the other hand, when disparity is small, high precision is required, since the control should minimize even small errors.

We can therefore replace equation 1.13 with:

$$CM_SET_{var\ iant} = \{CM_d^{var\ iant}, d \in 0 \dots N\} \quad [1.22]$$

$$CM_d^{var\ iant} = \{(\theta_n, \phi_n)_{\theta, \phi} : \theta \in 0 \dots \theta_{max}, \phi \in 0 \dots \phi_{max}\} \quad [1.23]$$

where $(\theta_{max}, \phi_{max})$ is the log-polar image size in cortical coordinates, N is the total number of CMs and:

$$(\theta_n, \phi_n)_{\theta, \phi, d} = disp_{log}(\theta, \phi, \arctan^i(\frac{N}{2}))^{\frac{\theta_{max}}{2}} \quad [1.24]$$

where i is an odd natural number, in order to be sign preserving. i acts as a steep enhancer in the non linear transformation and it is typically $i=3$.

Figure 5 is indeed the plot of equation 1.24 for $i=3$. The minimum resolution is now equal to:

$$d_{min}^1 = \frac{\theta_{max}}{2} \arctan^i(1/N) = d_{min}^1 \quad [1.25]$$

By observing equation 1.24 we might think that the accuracy can be increased just by modifying the non-linear function, i.e. increasing i . However, by reducing the disparity step, we reduce also the offset applied to the original log-polar mesh. For small disparity values, the offset would be less than one foveal pixel. This means that the log-polar mesh would remain roughly the same. In other words, if d_{min} is too small:

$$I_r = disp_{log}(I_l, d_{min}) = I_r \quad [1.26]$$

Hence, as it is intuitively obvious, it is not possible to compute directly a value of disparity smaller than one foveal pixel.

4.1.2 Quadratic Interpolation

It is clear that $\theta(d)$ is actually a sampled version of an underlying continuous function (i.e. we might imagine to define an infinite disparity set instead of the finite one). It is thus possible to apply an interpolation technique on the samples $\theta(d)$. A straightforward solution is to apply a quadratic interpolation technique as sketched in figure 6. In equation form:

$$d_n = \frac{\theta(d_{n-1}) \theta(d_{n+1})}{2 \theta(d_{n-1}) \theta(d_n) \theta(d_{n+1})} \quad [1.27]$$

where d is the disparity value computed without interpolation and θ is the minimum disparity value ($\theta = d_{min}$) or a suitable multiple.

Given the heuristic of this last technique there are no particular reasons to prefer an interpolation technique from another. Note also that the interpolation is not performed on the final data (the estimated disparity) but on the raw data of the disparity function. Of course, this means that it is not just a mathematical smoothing but it represents a real improvement over the non-interpolated counterpart.

5. Vergence Control from Disparity Computation

Vergence control has been implemented by applying the following control law:

$$\dot{\theta} = K \theta d_t \quad [1.28]$$

where K is a constant gain, $\dot{\theta}$ is the first derivative of the vergence angle and d_t is the disparity index (see equation 1.20). The advantages of this approach can be summarized as follows:

- 1) d_t is a linear estimation of the angular error, as shown in figure 7.
- 2) d_t provides the same information irrespective of the state of the system, because it does not matter if the cameras are moving or if they are still.
- 3) d_t is robust to noise, environmental modifications, changes in lighting conditions, and object properties.

A few words are needed to explain the previous points. The properties of d_t are graphically illustrated on figure 7 and figure 9 in which it is possible to see that d_t is linear in a range of approximately 0.2 radians. This range is compatible with the goal of keeping the right angle of vergence, a task that is usually achieved for small values of disparity. Regarding the second point it is important to stress that one of the major drawbacks of other implementations of vergence control was the inability of detecting the correct angle without moving the cameras (e.g. [30]). In other words, for each pair of images we have a meaningful value for the controller without having to acquire a new pair in order to compute a gradient. Lastly, the robustness of d_t is partially derived from the robustness of the correlation function $C()$ and partially from the property of the integration of several correlation functions as described in paragraph 3.

6. Experiments

In order to demonstrate the feasibility of the approach we tested the algorithm on a binocular robotic set-up. This consists of five degrees of freedom robot head (figure 8a). The head kinematics allows independent vergence (both cameras can move independently around a vertical axis) and a common tilt motion. Furthermore the neck is capable of a tilt and pan independent motion. However, in the following experiments, only two d.o.f. will be used to test the algorithm. The robotic set-up is equipped with two space-variant color cameras [18]. Experiments were carried out on a standard Windows NT based Pentium II 400MHz machine.

The log-polar images were 32×64 pixels and they were processed at a video rate (25 frames/sec). Actually, the overall computation time for control cycle was only 10ms. The limiting factor in this case was the image acquisition process. The proposed technique was tested under two different experimental conditions both under controlled and uncontrolled stimulation. An object (the clown head of figure 8b) was fixed on a programmable moving slider. We then generated different back and forth motion profiles. In this situation only vergence control was required (no version) to actually track the moving stimulus. The object was able to move along either a triangular or a sinusoidal trajectory. Motion amplitude was 1.20 m. The nearest point was at 20cm in front of the cameras (figure 8c). This corresponded to an angular difference (in term of vergence angle) of 45 degrees.

Concerning the closed-loop case, figure 9 depicts the effect of the quadratic interpolation. Figure 9a is the disparity estimation without interpolation; figure 9b is the same trace using the described quadratic interpolation. It is possible to notice the difference in the smoothness of the estimate (the upper plot is step-wise). The data were obtained using the set-up described by figure 8. In this case the cameras were still, the object was moving along a sinusoidal trajectory (at a fixed frequency).

Figure 10 presents the closed loop condition where the estimated global disparity is used to drive vergence control. In this case the control was activated and the trajectory had a triangular profile. The four plots represents: the estimated disparity (figure 10a), the fusion index $C(t)$ (figure 10b), the motor command (speed) (figure 10c) and the vergence angle (figure 10d). An interesting effect is visible by comparing the disparity plot with the vergence angle: the biggest error (in module) corresponds to the minimum distance of the slider, whilst the minimum estimated error corresponds to the maximum distance. Besides, when the change in direction occurs at the minimum distance there is a relatively big overshoot in the disparity estimate. On the other hand, when it corresponds to the maximum distance there is almost no overshoot at all. Nevertheless, by observing the absolute scale of the computed error we can see that the maximum error was only 0.05 pixels.

As far as the performance in an unconstrained environment is concerned the behavior of the robotic head was tested for several prolonged experimental sessions. Its performances have been reliable, robust and consistent. The frequency response of the system was practically flat over the tested range (10^{-1} Hz-2Hz). However, these last experiments were conducted in a qualitative fashion and, therefore, their significance is limited. What we can say is that the head proved to be robust to noise in the periphery of its visual field as well as to noise derived by abrupt modification in light intensity or target object unexpected movement. While the robustness with respect to peripheral noise was mainly due to log-polar images, the capability of responding quickly to object modifications both in its dynamic (speed, directions, trajectory) and in its appearance (shape, position, color, reflectance, shape, and shades) factors is due to the robustness of the processing itself.

7. Conclusions

In this paper we presented a global disparity estimation method and its use in controlling the vergence angle of a stereoscopic artificial vision system. We were able to extract a measure of

the vergence error by using space variant stereo images. This measurement was proved to be linear, fast, and robust to environmental changes. Consequently, it has been used to implement a closed loop vergence control that, through the use of both non-linear distribution of CMs and space variant images, achieves accuracy much smaller than one pixel (typically around 10^{-1} pixels relatively to image size). Last but not least, we were able to extract the disparity information at frame rate using a standard hardware platform.

Experimental results substantiate the above claim. The robotic head control proved to be robust to noise in the periphery of its visual field as well as from noise derived by abrupt modification in light intensity (as showed in § 3) or target object unexpected movement (as showed in § 5). Present investigation is aimed at the integration of the proposed method with tracking control and saccade like movements.

Acknowledgments

This work has been supported by the ESPRIT project NARVAL and ROBVISION, and by funds from the Italian MURST. Doctor Gasteratos is supported by a fellowship from the TMR project-VIRGO.

References

1. Panerai, F., G. Metta, and G. Sandini, Visuo-inertial Stabilization in Space-variant Binocular Systems. in *Robotics and Autonomous Systems*, 1999(Special Issue on Biomimetic Robotics): p. To Appear in Dec 1999.
2. Manzotti, R., *et al.*, Primary Ocular Movements Revisited, . 1994, LIRA-Lab.
3. Srinivasan, M.V. and S. Venkatesh, *From living eyes to seeing machines*. 1997, London: Oxford University Press.
4. Howard, J.P. and B.J. Rogers, *Binocular Vision and Stereopsis*. Oxford Psychology Series. 1995, Oxford: Clarendon Press.
5. Ballard, D.H. and C.M. Brown, Principles of Animate Vision. in *Computer Vision Graphics and Image Processing*,, 1992. **56**(1): p. 3--21.
6. Aloimonos, J., I. Weiss, and A. Bandyopadhyay, Active Vision. in *International Journal of Computer Vision*, 1988. **1**(4): p. 333--356.
7. Miles, F.A. and C. Busetini, Ocular compensation for self motion: visual mechanisms, . 1992, Ann. NY Acad. Sci.,. p. 220--232.
8. Pahlavan, K., T. Uhlin, and J.-O. Eklundh. Integrating Primary Ocular Processes. in *Proc. ECCV92 - European Conference of Computer Vision*. 1992. Santa Margherita Ligure, Italy: Springer-Verlag.

9. Taylor, J., T. Olson, and W.N. Martin. Accurate Vergence Control in Complex Scenes. in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1994. Seattle, WA, USA.
10. Kanade, T. and M. Okutomi, A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment. in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1994. **16**(9): p. 920--932.
11. Jenkin, M.R.M., A.D. Jepson, and J.K. Tsotsos, Techniques for Disparity Measurement. in *Computer Vision Graphics and Image Processing: Image Understanding*, 1991. **53**(1).
12. De Vleeschauwer, D., An Intensity-Based Coarse-to-Fine Approach to Reliably Measure Binocular Disparity. in *Computer Vision Graphics and Image Processing: Image Understanding*, 1993. **57**(2).
13. Tieh-Yuh, C. and A.C. Bovik. Stereo Disparity from Multiscale Processing of Local Image Phase. in *International Symposium on Computer Vision*. 1995.
14. Rougeaux, S., Real-Time Active Vision for Versatile Interaction, in *Centre d' Etude Mechanique d' Ile de France*. 1999, Universite d' Evry: Courcouronnes, France.
15. Matthies, L., Stereo Vision for Planetary Rovers: Stochastic modeling to near real-time implementation. in *International Journal of Computer Vision*, 1992. **8**: p. 71--91.
16. Stewart, C.V., R.Y. Flatland, and K. Bubna, Geometric Constraints and Stereo Disparity Computation. in *International Journal of Computer Vision*, 1996. **20**(3): p. 143--168.
17. Oshiro, N., *et al.* Binocular Tracking Using Log Polar Mapping. in *Proc. of IROS'96*. 1996. Osaka, Japan.
18. Sandini, G., *et al.* The Project SVAVISCA: a Space-Variant Color CMOS Sensor. in *Proc. AFPAEC'98*. 1998. Zurich: SPIE.
19. Daniel, M. and D. Whitteridge, The Representation of the Visual Field on the Cerebral Cortex in Monkeys. in *J. Physiol (London)*, 1961. **159**: p. 203--221.
20. Schwarz, U. and F.A. Miles, Ocular Responses to Translation and Their Dependence on Viewing Distance. I. Motion of the Observer. in *J. Neurophysiol*, 1991. **66**: p. 851--864.
21. Schwartz, E.L., A Quantitative Model of the Functional Architecture of Human Striate Cortex with Application to Visual Illusion and Cortical Texture Analysis. in *Biological Cybernetics*, 1980. **37**: p. 63--76.
22. Marefat, M.M., L. Wu, and C.C. Yang, Gaze Stabilization In Active Vision - I. Vergence Error Extraction. in *Pattern Recognition*, 1997. **30**(11): p. 1829--1842.
23. Marefat, M.M., L. Wu, and C.C. Yang, Gaze Stabilization In Active Vision - II. Multi-

- Rate Vergence Control. in *Pattern Recognition*, 1997. **30**(11): p. 1843--1853.
24. Grosso, E., *et al.* A Space-Variant Approach to Oculomotor Control. in *IEEE Int. Symposium on Computer Vision*. 1995. Coral Gables, Florida.
 25. Bernardino, A. and J. Santos-Victor. Vergence Control for Robotic Heads using Log-polar Images. in *Proc. of IROS'96*. 1996. Osaka, Japan.
 26. Horswill, I. and M. Yamamoto. A \$1000 Active Stereo Vision System. in *Proceedings of IEEE Symposium on Visual Languages*. 1994.
 27. Nielsen, J. and G. Sandini. Learning Mobile Robot Navigation : A Behavior-Based Approach. in *IEEE International Conf. on Systems, Man and Cybernetics*. 1994. San Antonio, Texas.
 28. Capurro, C., *et al.* A Binocular Active Vision System Using Space Variant Sensors: Exploiting Autonomous Behaviors for Space Application. in *Int. Conf. on Digital Signal Processing*,. 1993. Nicosia, Cyprus,.
 29. Casalino, G., *et al.* Closed loop steering for unicycle like vehicles: a simple Lyapunov like approach. 1994. Capri - Italy,,: Syroco.
 30. Capurro, C., F. Panerai, and G. Sandini, Dynamic Vergence using Log-polar Images. in *International Journal of Computer Vision*,, 1997. **24**(1): p. 79--94.

Figure Captions

Figure 1. Stereoscopic vision system: the version (θ_p) and vergence (θ_v) angles are shown.

Figure 2. The log-polar transformation: (a) the log-polar layout mapped into the Cartesian space and (b) the corresponding cortical image.

Figure 3. The geometrical transformation of the cortical mesh; the disparity values increase from zero and nine pixels (from the left to the right and from the top to the bottom, respectively). In each graph the horizontal axis represents the eccentricity, while the vertical axis the angular position. The small circle and the box show the effects of the transformation.

Figure 4. A measured disparity function obtained with an object almost in vergence. It is possible to notice the maximum of the function near the center of the curve.

Figure 5. The non-linear distribution of CMs: the upper graph is the transformation function. The middle one is the density of the distribution. The bottom plot represents the maximum detectable precision of disparity.

Figure 6. An example of interpolation by using the quadratic approach (equation 1.10): the accuracy improved to about 0.3 pixels.

Figure 7. The disparity function: these two plots were obtained by changing the vergence angle at a constant speed (total span was 0.16 radians). For each angular position we measured the disparity (upper plot) and the fusion index $C(t)$ (lower plot), respectively.

Figure 8. The experimental set-up: (a) the five d.o.f. robot head, (b) a log-polar image of the subject, acquired during the experiment (here presented both in image and in cortical planes). (c) A clown head is moving back and forth in front of the robot head at different speeds. The amplitude of the movement was 120 cm; the nearest point was at 20 cm from the cameras.

Figure 9. Effect of the quadratic interpolation: the target, in this case, was moving along a sinusoidal trajectory with fixed frequency. The cameras were still, i.e. the vergence control was not applied. (a) This graph illustrates the estimated disparity without interpolation and (b) this one depicts the improved final result, after the interpolation.

Figure 10. Vergence control: the object moved along a triangular trajectory at a constant frequency. The vergence control was activated. (a) The estimated disparity, (b) the fusion index $C(t)$, (c) the motor command (speed) and (d) the vergence angle.

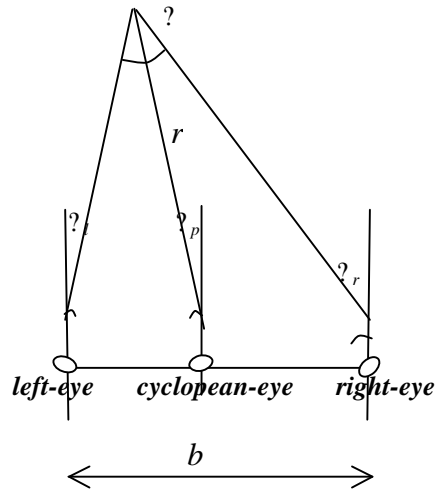


Figure 1.

Title:
logpolar.ps
Creator:
MATLAB, The Mathworks, Inc.
Preview:
This EPS picture was not saved
with a preview included in it.
Comment:
This EPS picture will print to a
PostScript printer, but not to
other types of printers.

(a)

(b)

Figure 2.

Title:
logbn1.ps
Creator:
MATLAB, The Mathworks, Inc.
Preview:
This EPS picture was not saved
with a preview included in it.
Comment:
This EPS picture will print to a
PostScript printer, but not to
other types of printers.

Figure 3

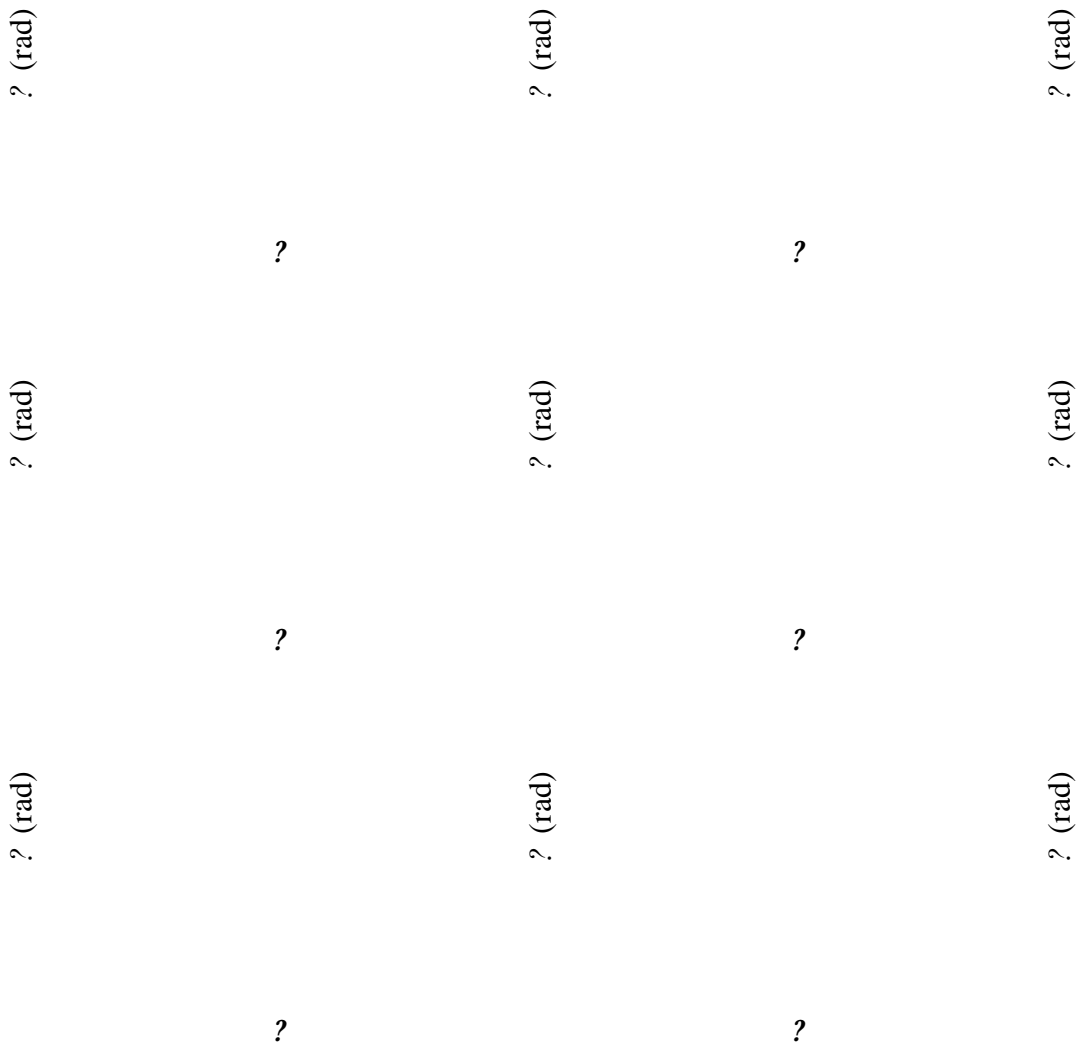


Figure 3.

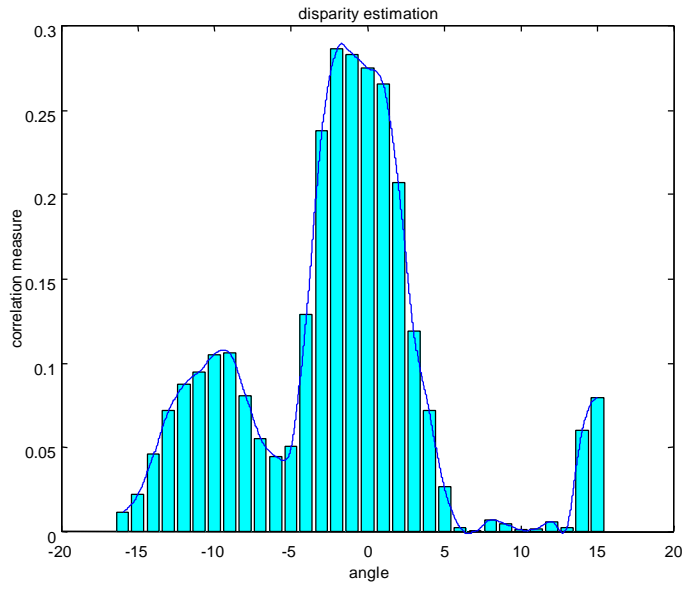


Figure 4.

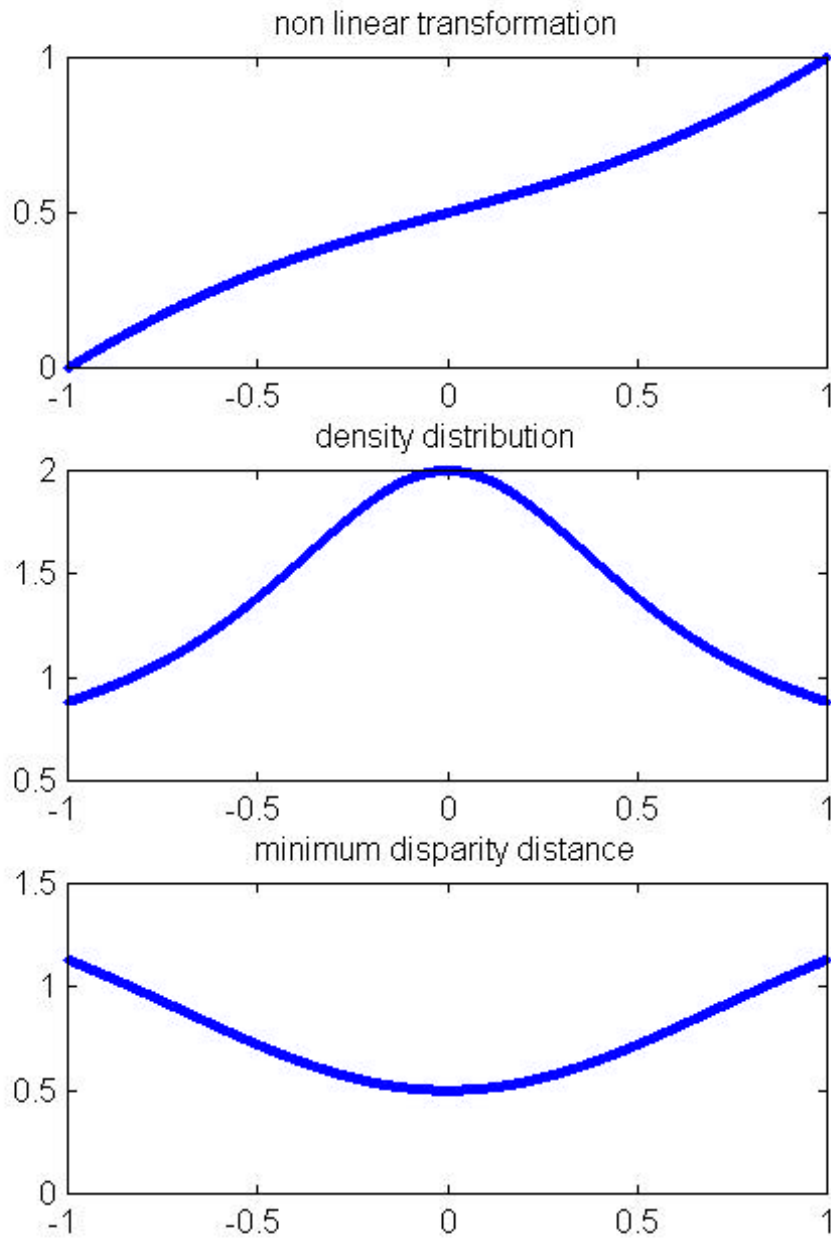


Figure 5.

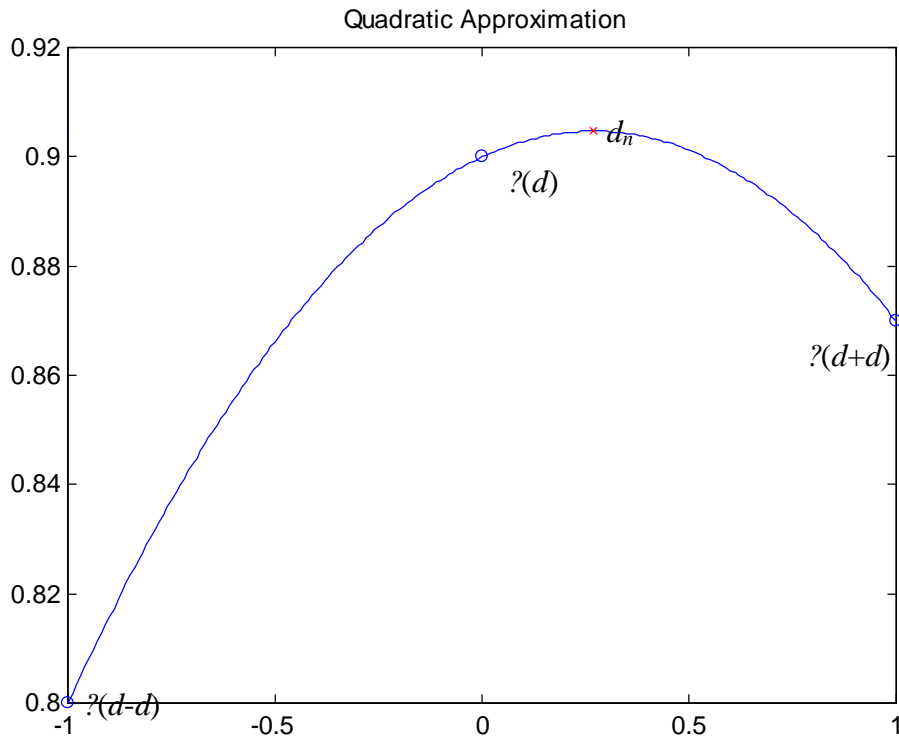


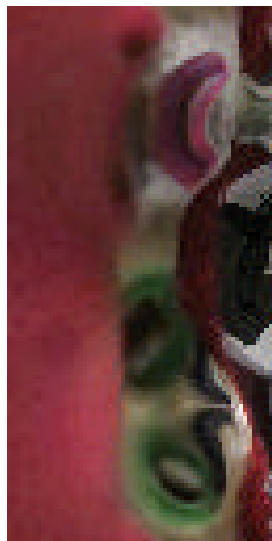
Figure 6.

Title:
lp2disp.ps
Creator:
MATLAB, The Mathworks, Inc.
Preview:
This EPS picture was not saved
with a preview included in it.
Comment:
This EPS picture will print to a
PostScript printer, but not to
other types of printers.

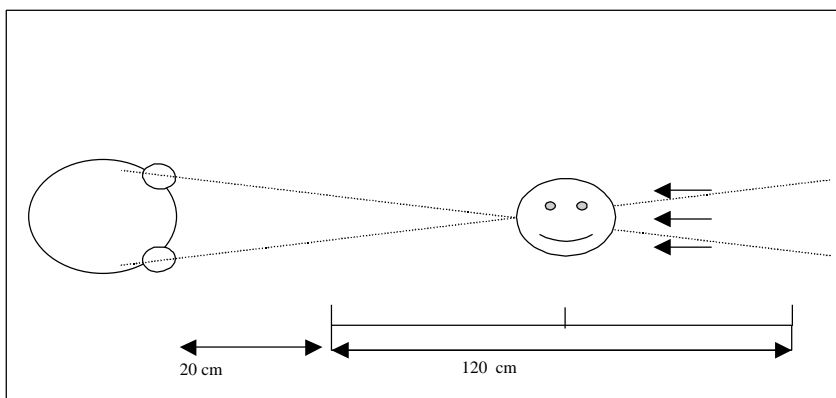
Figure 7.



(a)



(b)



(c)

Figure 8

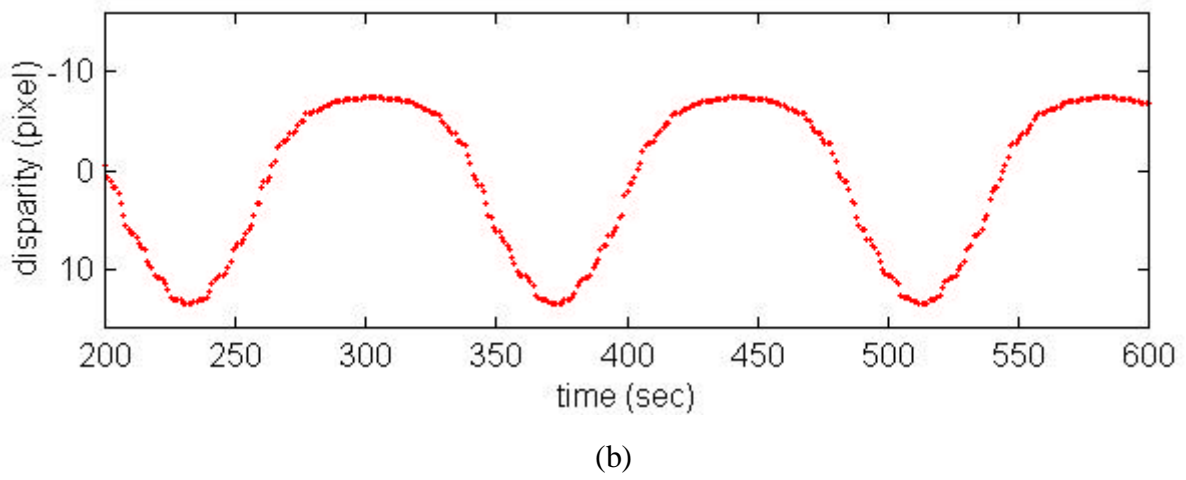
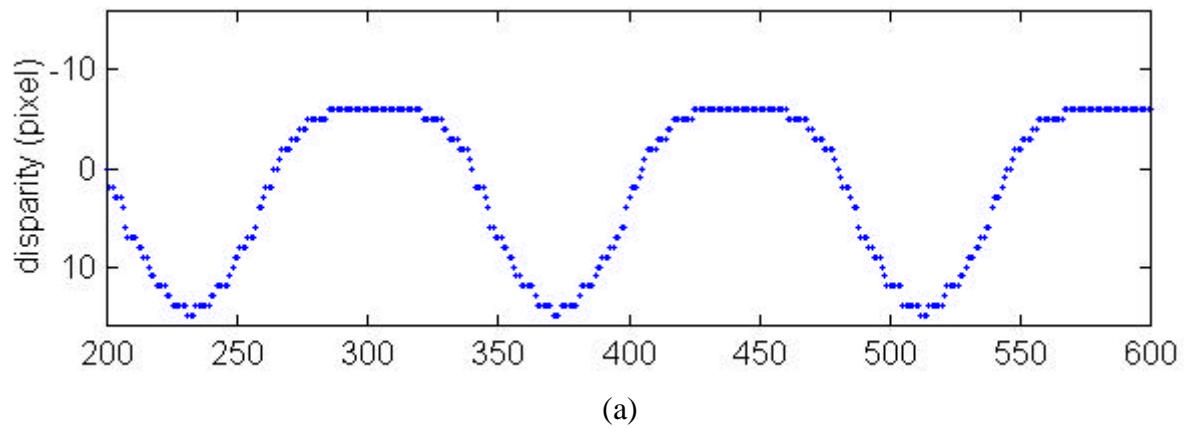


Figure 9.

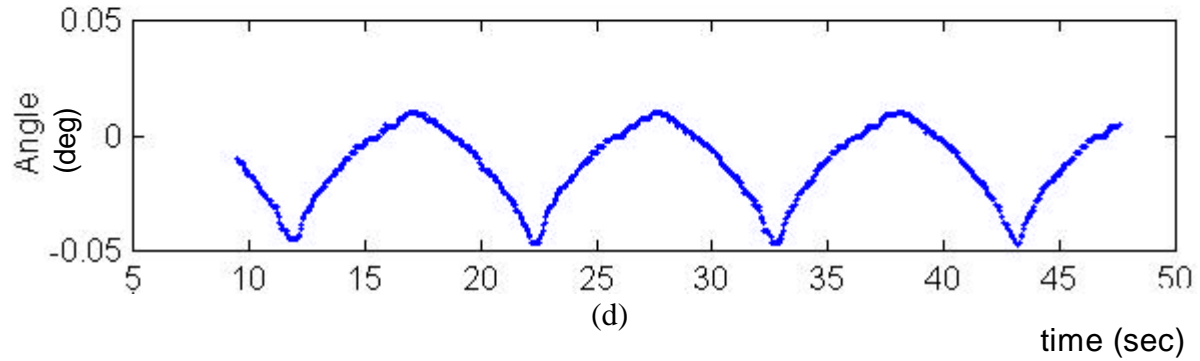
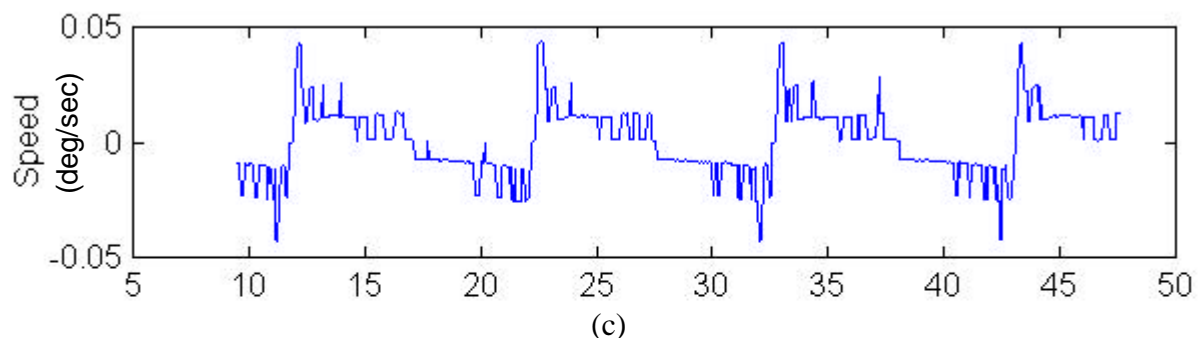
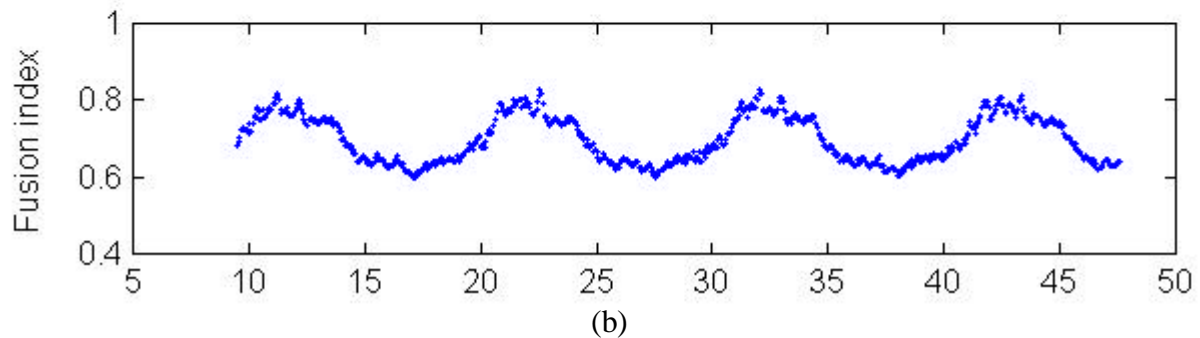
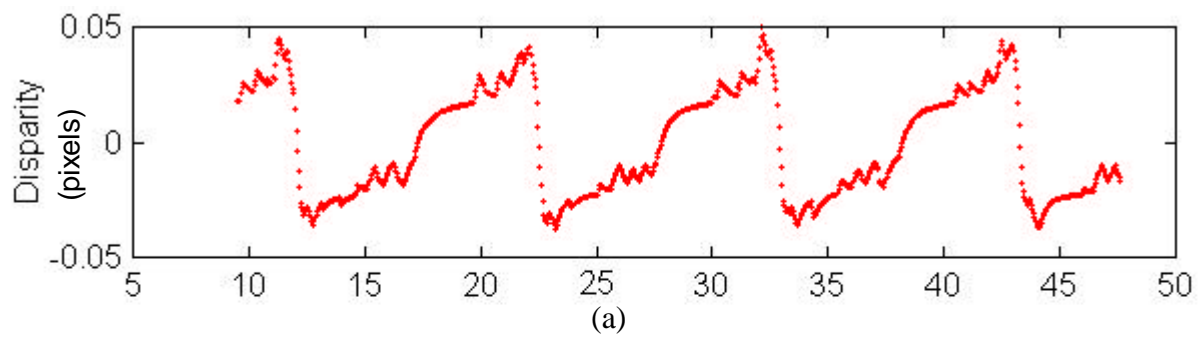


Figure 10.