

# Analysis of Two Methods for the Estimation of Partial 3D Velocity

Nuno Gonçalves and Helder Araújo\*

Institute of System and Robotics  
Dept. of Electrical Engineering–Polo II  
University of Coimbra  
3030 Coimbra  
Portugal

**Abstract.** In this paper we analyse and compare two methods for the computation of the total 3D velocity along the optical axis of a stereo system, using the rigid body motion model. One of the methods uses the temporal and spatial derivatives of the depth information and the image flows and the other one uses the depth information as well as the differential flow between left and right images. Both methods are compared in terms of uncertainty propagation. The problems of generating ground truth data as well as the problems related to depth resolution are also discussed. Finally the results of our experiments to compute the 3D velocity in the optical axis using synthetic and real images are reported.

## 1 Introduction

Vision is a sensor modality with several advantages for autonomous robots despite the difficulties in employing it. In particular vision can be used to estimate the egomotion of a robot, which is extremely important for the navigation system.

For egomotion estimation several methods have been proposed in the last decades, some using monocular sequences of images and others using stereo sequences. We consider calibrated cameras [2]. We are specially interested in those methods that can be easily used in real-time applications. We are also interested in methods that use sequences of stereo images and depth fields.

One of the methods, that was proposed by Harville et al. [4], uses a linear depth change constraint equation (DCCE), that is, assumes a model for the change of the depth fields. If depth measurements are available this method can be applied to sequences of monocular images, using the temporal and spatial derivatives of the depth. Otherwise sequences of stereo images can be used to estimate both depth and egomotion.

The second method was proposed by Waxman and Duncan [7], and uses stereo sequences to recover the 3D motion parameters. This method, as we shall see, uses the differential image flow between left and right images to compute the motion parameters.

Our goal is to compare those methods to recover the total tridimensional velocity in the optical axis using a stereo system. This velocity estimate is relevant for the computation of time to collision [3], which is useful for the robot navigation.

We developed the noise propagation equations of both methods considering independent Gaussian white noise for the inputs.

To test and compare those methods we used synthetic images and also real images. To estimate the image flow fields we used both ground truth velocities and the Lucas-Kanade algorithm[1].

In the next section both methods will be described. In section 3 the uncertainty propagation model is derived. Section 4 will discuss problems generating ground truth data as well as problems associated to depth resolution. In section 5 we present some results of tests made with synthetic images and with real sequences of images, including translational and rotational movements. In section 6 we discuss the results and the conclusions and future enhancements are presented.

---

\* (nunogon,helder)@isr.uc.pt

## 2 Motion estimation

The recovery of the total 3D velocity along the optical axis is our goal. Before the description of the methods used to compute the velocity, we shall first introduce the notations and geometry used throughout this paper.

In this paper we will designate a 3D point in space by its coordinate vector  $\mathbf{P} = [X \ Y \ Z]^T$  and the world coordinate system will be coincident with the cyclopean coordinate system, that is, centered in the middle point between the optical centers of both cameras. The origins of the local camera coordinate systems are the optical centers at a distance  $f$  (focal length) of the image plane. Both cameras are parallel to each other separated by the baseline  $b$ . The flow induced in the image planes is represented by  $v^l = (v_x^l, v_y^l)$  for the left image plane and by  $v^r = (v_x^r, v_y^r)$  for the right image plane.

The figure 1 shows the geometry of the stereo vision system and the world coordinate system.

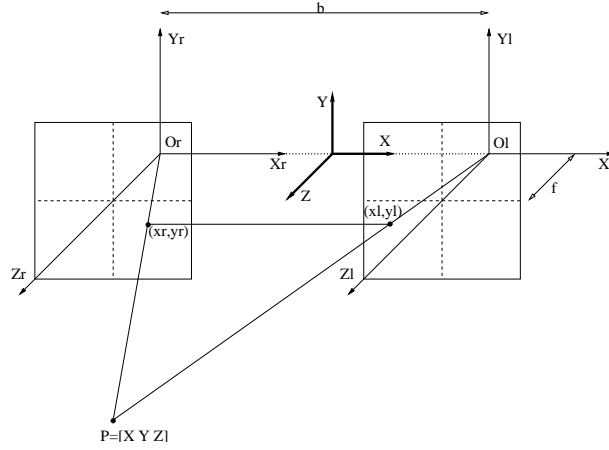


Fig. 1. World and stereo coordinate system

The model used for the 3D total velocity of a point  $\mathbf{P}$  in space is the rigid body motion. Let  $\vec{V}$  be the total 3D velocity of the point  $\mathbf{P}$ . As any rigid body motion can be expressed by a translation component given by  $\vec{t} = [t_X \ t_Y \ t_Z]^T$  and a rotational component given by  $\vec{\Omega} = [\Omega_X \ \Omega_Y \ \Omega_Z]^T$  we have the 3D velocity given by  $\vec{V} = \vec{t} + \vec{\Omega} \times \mathbf{P}$ .

Computing the components of the total 3D velocity  $\vec{V}$ , we obtain the following expression:

$$\vec{V} = \begin{bmatrix} t_X + \Omega_Y Z - \Omega_Z Y \\ t_Y + \Omega_Z X - \Omega_X Z \\ t_Z + \Omega_X Y - \Omega_Y X \end{bmatrix} = \begin{bmatrix} V_X \\ V_Y \\ V_Z \end{bmatrix} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (1)$$

We are interested in estimating the third component of  $\vec{V}$ , that is  $V_Z$  which is the total 3D velocity along the optical axis.

### 2.1 Depth Constraint

The depth change of a point or rigid body over time is directly related to its velocity in the 3D space. We can use this principle to relate the velocity in the optical axis with depth.

For simplicity we will derive the linear depth change constraint equation relative to the left image coordinate system.

Consider a point  $\mathbf{P} = [X \ Y \ Z]^T$ , which projects in a point with coordinates  $(x, y)$  in the image plane at a time  $t$  and in point  $(x + v_x, y + v_y)$  at a time  $t + 1$ . So the depth at instant  $t + 1$  should be the depth at the instant  $t$  plus the amount of space that the point moved along the optical axis -  $V_Z$ . This relationship is given by the following expression, the linear Depth Change Constraint Equation - DCCE:

$$Z(x, y, t) + V_Z(x, y, t) = Z(x + v_x(x, y, t), y + v_y(x, y, t), t + 1) \quad (2)$$

where  $Z(x, y, t)$  is the depth of the point  $\mathbf{P}$  at a given time  $t$  and  $V_z(x, y, t)$  is the total 3D velocity in the optical axis.  $v_x(x, y, t)$  and  $v_y(x, y, t)$  are the components of the optical flow.

The first step is to approximate the equation 2 by a first-order Taylor series expansion. Then we obtain:

$$Z(x, y, t) + V_Z(x, y, t) = Z(x, y, t) + Z_x(x, y, t)v_x(x, y, t) + Z_y(x, y, t)v_y(x, y, t) + Z_t(x, y, t) \quad (3)$$

with  $Z_x(x, y, t)$  and  $Z_y(x, y, t)$  the depth gradients with respect to the bidimensional spatial coordinates  $x$  and  $y$  and  $Z_t(x, y, t)$  its temporal gradient.

Our DCCE equation then reduces to

$$V_Z(x, y, t) = Z_x(x, y, t)v_x(x, y, t) + Z_y(x, y, t)v_y(x, y, t) + Z_t(x, y, t) \quad (4)$$

In the last equation we relate the depth gradients over time and space with the depth assuming that the image flow is given.

As mentioned by Harville et al.[4], often motion recovered with depth information is more accurate than that recovered from the intensity images because it is less sensitive to illumination and shading problems.

## 2.2 Using Binocular Image Flows

In this subsection we will explain the second method to compute the  $V_Z$ . It is based on the differences between the flows induced by the movement of a point in a stereo pair of images [7]. The parallel stereo system is again used and is considered to move rigidly with the scene.

Consider again a point  $\mathbf{P} = [X \ Y \ T]^T$  that projects in both image planes as shown in the figure 1. The optical centers of both cameras dist  $b$  (baseline) from each other and the  $x$ - coordinate of  $\mathbf{P}$  is  $x^l$  in the left image and  $x^r$  in the right image.

Point  $\mathbf{P}$  in figure 1, its projection in each image plane ( $(x_l, y_l)$  and  $(x_r, y_r)$ ) and the optical centers ( $O_l$  and  $O_r$ ) define two similar triangles, so that we can write the relationship:

$$\frac{Z}{b} = \frac{Z - f}{b - (x^l - x^r)} \quad (5)$$

which can be simplified to

$$Z(x, y, t) = \frac{bf}{(x^l - x^r)} \quad (6)$$

Now, if we compute the temporal derivative of the equation 6, we obtain:

$$V_Z = -\frac{bf}{(x^l - x^r)^2} \times (v_x^l - v_x^r) = -\frac{Z^2}{bf} \Delta v_x \quad (7)$$

We now have the relation between the total 3D velocity in the optical axis ( $V_Z$ ) and the binocular image flow ( $\Delta V_x$ ).

In the next section we will derive the uncertainty propagation model for both the expressions obtained for  $V_Z$ .

## 3 Uncertainty Propagation

It is very important to analyse the properties of both expressions obtained for the total 3D velocity along the optical axis in terms of uncertainty propagation. As we shall see it is possible to determine the critical independent variables that in presence of uncertainties affect the recovery of motion parameters.

The first step is to define the independent variables for each expression:

$$\begin{cases} V_Z = Z_t + v_x Z_x + v_y Z_y = f(Z_x, Z_y, Z_t, v_x, v_y) \\ V_Z = -\frac{Z^2}{bf} \Delta v_x = f(Z, v_x^l, v_x^r) \end{cases} \quad (8)$$

where we assumed that the geometry parameters were known, that is, the baseline and the focal length.

So any noise in the values of the depth data, their temporal and spatial derivatives and in the binocular image flows will affect the computation of  $V_Z$ .

Let  $\vec{F}$  be the vector to be estimated and the vector with the independent variables  $\vec{S}$ .

For both calculations we assume that all variables are affected by Gaussian random white noise, denoted by  $\sigma_i^2$ , where  $i$  denotes the variable.

### 3.1 Depth constraint

For the first expression we have  $\vec{F} = [V_Z]$  and  $\vec{S} = [Z_x \ Z_y \ Z_t \ v_x \ v_y]^T$ .

We also assume that the noise in the variables is independent so the covariance matrix for this input signal  $\vec{S}$  is given by:

$$A_1 = \begin{bmatrix} \sigma_{Z_x Z_x}^2 & \dots & 0 \\ \sigma_{Z_y Z_y}^2 & & \\ \vdots & \sigma_{Z_t Z_t}^2 & \vdots \\ 0 & \dots & \sigma_{v_x v_x}^2 & \sigma_{v_y v_y}^2 \end{bmatrix} \quad (9)$$

We can define the Jacobian matrix that maps vector  $\vec{S}$  to the total 3D velocity in the optical axis  $\vec{F}$ , which is given by  $\frac{\partial \vec{F}}{\partial \vec{S}}$ .

To compute the covariance matrix of the function vector, a first order approximation can be used:

$$\Gamma_1 = \frac{\partial \vec{F}}{\partial \vec{S}} A_1 \frac{\partial \vec{F}}{\partial \vec{S}} = [v_x \ v_y \ 1 \ Z_x \ Z_y] A_1 \begin{bmatrix} v_x \\ v_y \\ 1 \\ Z_x \\ Z_y \end{bmatrix} \quad (10)$$

The resulting covariance matrix is a  $1 \times 1$  matrix given by the expression:

$$\Gamma_1 = \sigma_{Z_x Z_x}^2 v_x^2 + \sigma_{Z_y Z_y}^2 v_y^2 + \sigma_{Z_t Z_t}^2 + \sigma_{v_x v_x}^2 Z_x^2 + \sigma_{v_y v_y}^2 Z_y^2 \quad (11)$$

As we can see the perturbation in the computation of  $V_Z$  depends on the square of all the variables except the temporal derivative of depth.

### 3.2 Binocular flow

Using a similar reasoning we have for the second method the following:

$$\vec{S} = \begin{bmatrix} Z \\ v_x^l \\ v_x^r \end{bmatrix} \quad (12)$$

$$A_2 = \begin{bmatrix} \sigma_{ZZ}^2 & 0 & 0 \\ 0 & \sigma_{v_x^l v_x^l}^2 & 0 \\ 0 & 0 & \sigma_{v_x^r v_x^r}^2 \end{bmatrix} \quad (13)$$

And the Jacobian matrix is then:

$$\frac{\partial \vec{F}}{\partial \vec{S}} = \left[ \frac{\partial V_Z}{\partial Z} \ \frac{\partial V_Z}{\partial v_x^l} \ \frac{\partial V_Z}{\partial v_x^r} \right] = \left[ \frac{-2Z}{bf} \Delta v_x - \frac{Z^2}{bf} \ \frac{Z^2}{bf} \right] \quad (14)$$

The covariance matrix of the function vector is then:

$$\Gamma_2 = \frac{\partial \vec{F}}{\partial \vec{S}} A_2 \frac{\partial \vec{F}}{\partial \vec{S}} = \frac{4Z^2}{(bf)^2} (\Delta v_x)^2 \sigma_{ZZ}^2 + \frac{Z^4}{(bf)^2} \sigma_{v_x^l v_x^l}^2 + \frac{Z^4}{(bf)^2} \sigma_{v_x^r v_x^r}^2 \quad (15)$$

which reduces to the simpler form:

$$\Gamma_2 = 4V_Z \sigma_{ZZ}^2 + \frac{Z^4}{(bf)^2} \left( \sigma_{v_x^l v_x^l}^2 + \sigma_{v_x^r v_x^r}^2 \right) \quad (16)$$

Both expressions obtained for the covariance of the total 3D velocity along the optical axis can be used to analyse the uncertainty propagation properties of both methods. Both of them are sensitive to noise in the depth data but we can expect that the second method is highly sensitive to the uncertainty in the image flows. These results are supported by the experiments, as we shall see.

## 4 Experimental conditions

Before describing the experimental results it is important to address some practical questions. In the following subsections we will discuss the problem of obtaining ground truth data, and the problem of depth resolution.

### 4.1 Ground truth

To test any method ground truth information to compare with is essential. So it is really important to have the ground truth values of what is to be computed and those ground truth values should be as accurate as possible.

In this study we started to test the  $V_Z$  computed using synthetic data in a world where the left and right images were known as well as the ground truth velocities and depth fields. No problems existed with the computation of the ground truth values to compare with.

Regarding the experiments with real images, what was available were the intensity images (left and right) and the rigid body motion parameters of the stereo system:  $\vec{t}$  and  $\vec{\Omega}$ . To recover the depth fields we used the SVS (*Small Vision System* [5]) software that gives us the disparity for each point in the intensity images with a resolution of 1/4 of pixel. With the disparity images we calculate the depth using the simple expression  $Z = bf/d$  where  $d$  is the disparity. For the image flows we used a very-well known algorithm - the Lucas-Kanade algorithm. There are obviously several sources of noise.

For that reason we want to calculate the ground truth values of the  $V_Z$  as well as of the image flow true values.

Let us start with the image flow. Regarding equation 1 we have the total 3D velocity of a point  $\mathbf{P}$  in space in cyclopean coordinates.

Using now the perspective projection model ( $x = fX/Z$ ,  $y = fY/Z$ ) to project the total 3D velocity in the image plane, we have:

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = f \begin{bmatrix} \left(\frac{\dot{X}}{Z}\right) \\ \left(\frac{\dot{Y}}{Z}\right) \end{bmatrix} = \begin{bmatrix} f \left(\frac{\dot{X}}{Z} - X \frac{\dot{Z}}{Z^2}\right) \\ f \left(\frac{\dot{Y}}{Z} - Y \frac{\dot{Z}}{Z^2}\right) \end{bmatrix} \quad (17)$$

Replacing equation 1 in equation 17 we obtain the image flow for the cyclopean coordinate system:

$$\begin{cases} v_x = \left\{ f \frac{t_X}{Z} - x \frac{t_Z}{Z} \right\} + \left\{ -\frac{xy}{f} \Omega_X + \left( f + \frac{x^2}{f} \right) \Omega_Y - y \Omega_Z \right\} \\ v_y = \left\{ f \frac{t_Y}{Z} - y \frac{t_Z}{Z} \right\} + \left\{ -\left( f + \frac{y^2}{f} \right) \Omega_X + \frac{xy}{f} \Omega_Y - x \Omega_Z \right\} \end{cases} \quad (18)$$

From the expression of the image flow one concludes that it is possible to separate the translational and rotational terms of the flows. As it is well known the rotational terms of the flow do not depend directly on depth but only on the image coordinates.

But the translational part depends on the depth and if ground truth image flows are to be computed the available depth information has to be considered exact which results in introducing error in the ground truth values.

To compute the image flow equations for left and right cameras we use  $(x_l, y_l)$  and  $(x_r, y_r)$  instead of  $(x, y)$  and the motion parameters for each camera that are related with the cyclopean system motion parameters by the following equation:

$$\begin{cases} \vec{\Omega}_l = \vec{\Omega}_r = \vec{\Omega} \\ \vec{t}_l = \vec{t} + \vec{\Omega} \times \frac{b}{2} \hat{i} \\ \vec{t}_r = \vec{t} - \vec{\Omega} \times \frac{b}{2} \hat{i} \end{cases} \quad (19)$$

For the right image flow, the image coordinates  $(x_r, y_r)$  are given by  $y_r = y_l$  (because the cameras are parallel to each other) and  $x_r = x_l - d$  so we use again the depth (disparity) field to calculate the right image flow.

Since the depth/disparity images are affected by noise, the ground truth velocities computed will be affected by that noise. For the  $V_Z$  values the conditions are similar. The expression is  $V_Z = t_Z + \Omega_X Y - \Omega_Y X$ . If the motion has only a translational component the value for  $V_Z$  is only  $t_Z$  and since that value is known there is no uncertainty in the ground truth value. But when there is rotational component the computation of the ground truth is more difficult. To overcome this difficulty  $X$  and  $Y$  can be computed indirectly by

using the inverse perspective projection model (together with the depth values, therefore introducing again some error in the ground truth values).

The ground truth values for  $V_Z$  can also be computed by using the image flows and the exact equation of the second method  $V_Z = -Z^2 \Delta v_x / bf$  whose ground truth values are also corrupted by noise and in that situation what is indeed being compared are the velocities and not  $V_Z$ .

Therefore for rotational motion the exact structure of the world has to be known to obtain the true values of  $V_Z$ , which is difficult. For the moment we considered the depth data computed by SVS as being unaffected by error.

## 4.2 Resolution of the Depth Data

The software used to obtain the disparity fields has a resolution of 1/4 of pixel. So, some changes in the real depth of a point do not produce any change in the disparity and since depth is inversely proportional to the disparity its value is calculated with decreasing resolution as the value of the depth itself increases.

Let  $\Delta d$  be the minimum change in disparity. Then for the minimum change in depth to produce change in disparity we have:

$$Z = \frac{bf}{d} \rightarrow \Delta Z = -\frac{\Delta d}{d + \Delta d} \cdot \frac{bf}{d} \quad (20)$$

So, let us consider  $bf = 468$ ,  $\Delta d = 0.25$  and the pixel width  $pw = 0.012$ . All values are in mm.

In that particular case we have, for example:

- $d = 1 \rightarrow \Delta Z = -7800 \text{ mm}$
- $d = 5 \rightarrow \Delta Z = -371.43 \text{ mm}$
- $d = 10 \rightarrow \Delta Z = -95.12 \text{ mm}$
- $d = 20 \rightarrow \Delta Z = -24.07 \text{ mm}$

It can be seen that the low resolution in disparity/depth data can produce large errors with increasing distance to the optical center of the camera.

## 5 Experiments

Two groups of experimental tests were done. The first group of tests used synthetic images, therefore enabling the use of ground truth values for all the parameters. The second group of experiments was done with real images obtained with known motion parameters.

### 5.1 Synthetic images

To generate the synthetic images a virtual world was designed. This virtual world consisted on a ground, a front wall, left and right walls and two objects in the middle of the scenario. Using a stereo pair of virtual cameras every point was projected in the image planes. Their disparities and image flows were also computed. Figure 2 shows an example of the intensity and disparity images obtained.

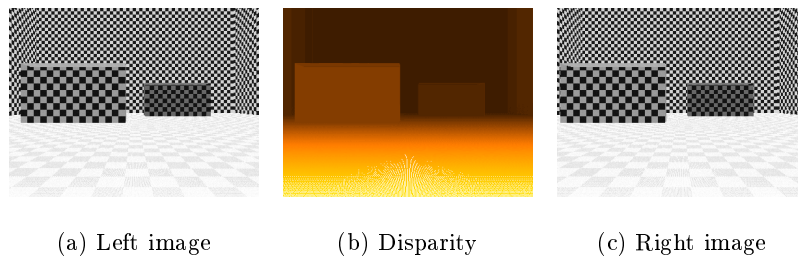


Fig. 2. Intensity images and disparity field for synthetic world

The stereo system motion is made up of sequences with only translational velocity along the optical axis and also combined with translational components along the other two axis and sections with rotational motion over the vertical axis and the horizontal axis (pan-and-tilt motion). There are also some sequences with combined translational and rotational motion.

Each sequence measurements were also computed considering that the stereo system moved not one image forward but 5 images (parameter STEP) in order to see the effect of having bigger motion parameters.

To approximate the experiments with synthetic images with those with real images we computed both methods when the depth fields are exactly the ground truth values and also when we round off the depth data to the closest resolution step in order to see the effect of the limited resolution in depth fields (discussed in subsection 4.2).

The results obtained were very close to the ground truth values and are shown below. Three statistical measurements were computed: the relative error of the mean value of  $V_Z$ , the mean value of the relative error (actually the mean of absolute value of the relative error) and the standard deviation of the relative error. All those measurements were computed in all points of the image where there were sufficient data to compute  $V_Z$ . The first measurement - the relative error of the mean value of  $V_Z$  will be plotted in a graph for each sequence showing the value calculated in each frame with STEP equal to 1 and 5 and also rounding off and not the depth data.

The results of three sequences are reported: (A) only translation over the optical axis; (B) only rotation over the vertical axis; (C) translation over all axis and rotation over vertical and horizontal axis. Other sequences were tested but for lack of space we only report those three. Figure 3 show the mean value of  $V_Z$  for those sequences in all the variations.

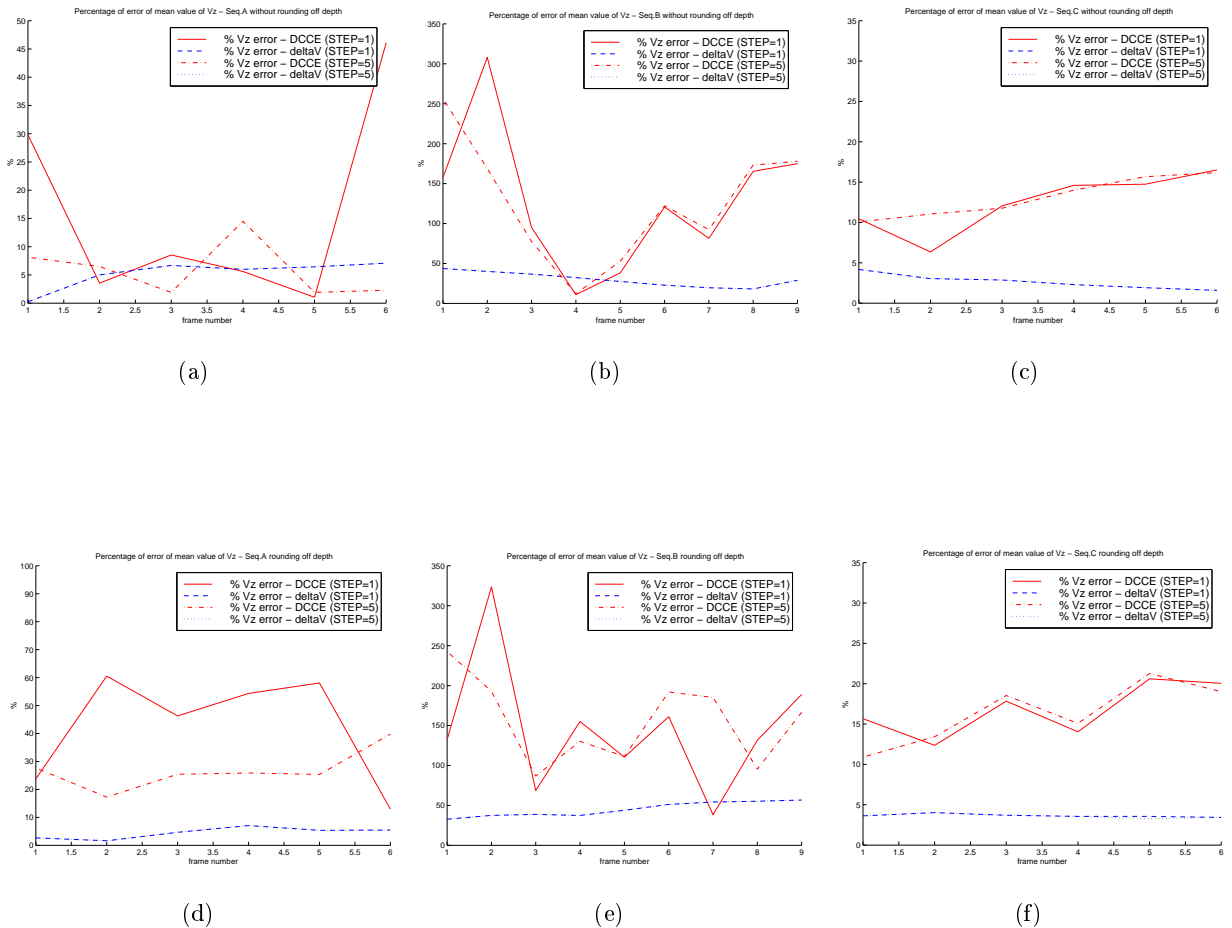


Fig. 3. Error of the mean value of  $V_Z$  in synthetic images (in percentage): (a) sequence A (b) sequence B (c) sequence C (d) sequence A with rounded off depths (e) sequence B with rounded off depths (f) sequence C with rounded off depths.

The other two measurements are shown in table 1. Note that all values are in percentage.

Seq.	Frame	DCCE		$\Delta v_x$		DCCE		$\Delta v_x$		DCCE		$\Delta v_x$	
		(a)	(a)	(a)	(a)	(b)	(b)	(b)	(b)	(c)	(c)	(d)	(d)
		ME	SD	ME	SD	ME	SD	ME	SD	ME	ME	ME	ME
A	20	79.0	65.0	12.2	5.5	150.3	93.2	12.3	7.0	28.2	12.2	85.7	12.3
A	30	51.7	38.9	8.4	1.6	122.5	63.5	8.1	3.0	26.6	8.4	85.2	8.1
A	40	58.1	45.0	6.8	0.0	130.5	70.5	5.3	0.2	29.0	6.8	168.0	5.3
A	50	43.2	31.0	6.2	0.0	112.2	37.4	7.6	0.2	28.5	6.2	82.3	7.6
B	20	546.4	856.6	23.9	19.0	877.0	1442.1	24.8	18.9	632.1	23.9	927.0	24.8
B	30	723.7	1195.2	24.1	19.4	899.1	1487.6	24.4	19.3	983.1	24.1	1193.3	24.4
B	40	630.9	1009.8	23.7	19.2	761.1	1217.9	23.8	19.2	748.2	23.7	892.9	23.8
B	50	419.6	597.1	24.0	19.3	501.4	731.9	24.1	19.3	475.7	24.0	557.2	24.1
C	20	123.9	61.3	18.1	16.9	145.3	74.5	19.3	18.2	121.9	18.1	138.5	19.3
C	30	118.3	53.4	17.8	16.4	140.1	66.0	19.7	18.8	121.1	17.8	139.1	19.7
C	40	121.1	57.9	17.2	15.6	142.3	70.8	19.0	17.7	119.2	17.2	138.5	19.0
C	50	120.1	56.7	17.4	15.8	137.0	70.1	19.3	18.2	118.5	17.4	135.7	19.3

Table 1. Mean relative error (ME) and its standard deviation (SD) in synthetic images: (a) STEP=1 without rounding off depth; (b) STEP=1 rounding off depth; (c) STEP=5 without rounding off depth; (d) STEP=5 rounding off depth. All values in percentage.

In the next subsection we report the results using real image sequences and in the next section we discuss the results.

## 5.2 Real images

To analyse the performance of both methods with real images a pair of cameras with 3.6 mm focal length and a baseline of 130 mm was used. The pixelwidth is 12.5  $\mu m$ . The stereo head was attached to a manipulator with high precision allowing complex paths to be performed. In this paper the results presented correspond to sequences of images with translational motion, rotational and both types of motion. Figure 4 shows a pair of intensity images and the corresponding disparity image.

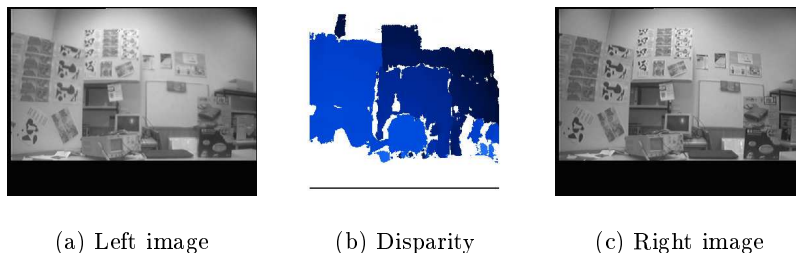


Fig. 4. Intensity images and disparity field

For the computation of the several input variables needed to the methods, we used well known techniques. The temporal and spatial gradients of depth were obtained by difference equations and the image flows were obtained by the Lucas-Kanade algorithm.

In figure 5 we show the error of the mean value of  $V_Z$  like we did in the previous subsection. Four sequences are presented: (B) with only rotational motion along vertical axis; (H) with translational motion over the optical axis; (L) with translation in both the optical and vertical axis and (O) with both rotation in the vertical axis and translation in the optical and vertical axis. For each sequence were computed the three measures explained in the previous subsection for Lucas-Kanade velocities and for ground truth velocities calculated by equation 18.

In table 2 we report the other two measurements for real image sequences. In the next section we interpret the results and make the conclusions of our work.



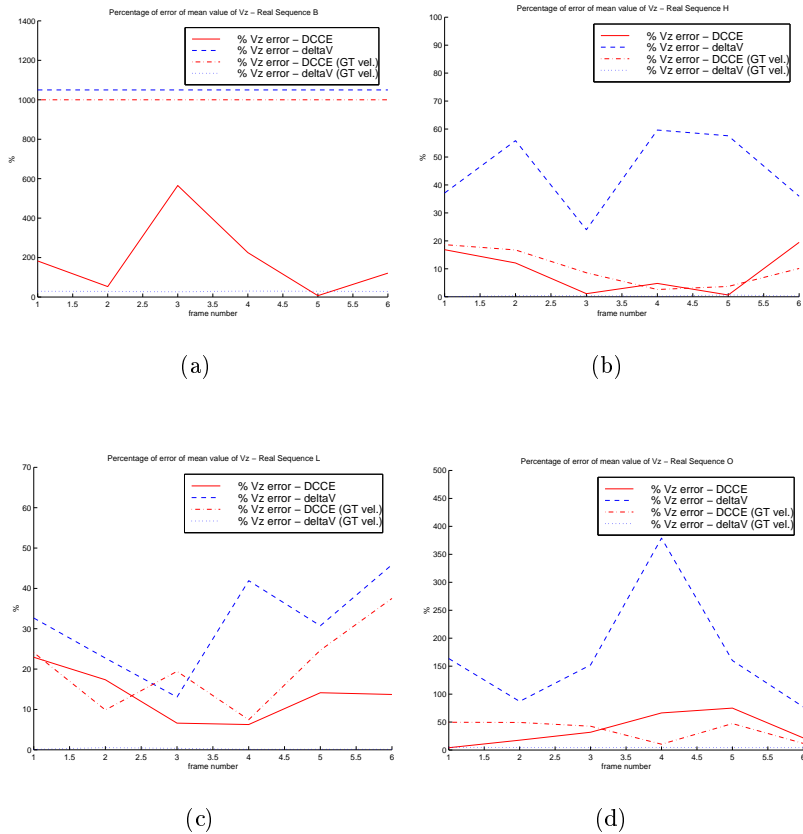


Fig. 5. Error of the mean value of  $V_Z$  in real images (in percentage): (a) sequence B ;(b) sequence H ;(c) sequence L ;(d) sequence O. In sequence B the values for binocular flow method with Lucas-Kanade velocities and DCCE with ground truth velocities were truncated because they were very high.

## 6 Discussion and Conclusions

In this paper we analysed and compared two methods that compute the total 3D velocity along the optical axis ( $V_Z$ ) using a stereo system. Both methods use image flows and depth fields. We also derived the uncertainty propagation models for both methods and conclude that the second method (based on the differential image flow) is much more unstable in the presence of noise than the first one.

We performed several tests with both synthetic images and real images for sequences with only translation, only rotation and both motions. For translation sequences the results were very good for synthetic images for both methods and good for real images. In that case the DCCE method seems to be better in real images and worst in synthetic images. We can explain this by noting that in synthetic images the velocities are very close to the real ones and so the binocular flow method has few sources of noise. When we use velocities affected by noise the binocular flow method seems to be less accurate.

When we analyse rotation sequences the results are completely different. Both in synthetic images and real images these methods give us bad results for the  $V_Z$ . The error is really very high for real images. The DCCE method seems to be worst in that case even when we use ground truth velocities. The binocular flow method yields very bad results for Lucas-Kanade velocities and reasonable values with ground truth flow.

For complex motion with translation and rotation motion, the results are close to those obtained in only translation sequences, specially when the rotation parameters are not as significant as the translation ones. Both methods present similar results. In the absence of noise in the flow the binocular flow method seems to be more accurate and less sensitive to the depth field but when the noise is not negligible the DCCE methods presents better results.

In real images we observed an interesting fact. The mean value of the  $V_Z$  is reasonably closer to the ground truth values in several sequences but the mean relative error and the standard deviation are very high, specially in the second method (as was expected). This is relevant fact since it suggests that when

		DCCE	DCCE	$\Delta v_x$	$\Delta v_x$	DCCE	DCCE	$\Delta v_x$	$\Delta v_x$	Observations
		(a)	(a)	(a)	(a)	(b)	(b)	(b)	(b)	
	Frame	ME	SD	ME	SD	ME	SD	ME	SD	
B	16	714.3	685.9	3438.2	2415.5	2720.0	4363.3	32.8	32.2	$\Omega_Y = -\pi/360$ rad/frame
B	20	668.3	657.2	5339.8	3516.6	2591.8	4149.4	33.8	33.0	idem
H	7	66.9	54.0	220.1	184.3	82.5	65.2	2.7	2.9	$t_Z = -20$ mm/frame
H	12	68.0	57.6	228.5	187.8	88.1	75.6	2.8	3.1	idem
L	4	125.8	106.7	327.3	251.9	167.5	145.4	3.3	4.0	$t_Y = 10, t_Z = -20$ mm/frame
L	7	117.7	96.7	269.1	215.9	156.2	128.8	2.3	2.2	idem
O	11	498.2	632.2	3676.1	4482.9	1900.9	2992.1	23.0	32.4	$\Omega_Y = \pi/900, t_Y = 5, t_Z = -5$
O	14	461.0	571.0	4028.3	4614.8	1184.6	1721.8	28.1	41.0	idem

Table 2. Mean relative error (ME) and its standard deviation (SD) in real images: (a) Lucas-Kanade velocities; (b) ground truth velocities. All values in percentage.

computing  $V_Z$ , a high number of measurements has to be done, in order to allow the cancellation of the error when all points in the image are used.

Another measurements made are when we multiply by 5 the motion, that is, we analyse each 5 frames instead of the one and the next. We noted that the binocular flow method is relatively insensitive to this change but the DCCE method increases the accuracy in both translation and rotation sequences. This can be explained by the fact that the DCCE method depends on the gradients of depth fields and as the depth change increases we expect that the limited resolution in depth become less important (see subsection 4.2).

For seeing the impact of the depth resolution in both methods, we rounded off the depth value for the closest step in synthetic images so introducing quantization error in the depth data. We saw that the binocular flow method is almost insensitive to this change and that the DCCE method deteriorates its results because the depth gradients change radically.

Concluding, the binocular flow method seems to be better when synthetic images are used and so the noise in the image velocities is little or when real images are used and the image velocity is accurate and the DCCE method reveals to be better in real images with inaccurate image velocities. Both methods have bad results with rotation motion and the DCCE method is more sensitive to the resolution in depth data. Increasing the translation parameters tends to minimize this effect.

We expect in the future to use those methods to compute the complete motion parameters and study the critical ones and those that we can acquire with reasonable accuracy. By our experience with  $V_Z$  we expect that rotation parameters are less accurate.

## 7 Acknowledge

The authors gratefully acknowledge the support of project PRAXIS/P/EEI/10252/1998, funded by the Portuguese Foundation for Science and Tecnology.

## References

1. Barron, J.L.; Fleet, D.J. and Beauchemin, S.S. Performance of Optical Flow Techniques. *IEEE International Journal of Computer Vision*, 12(1):43–77, 1994.
2. Batista, Jorge; Araújo, Helder and Almeida, Aníbal T. Iterative Multistep Explicit Camera Calibration. *IEEE Transactions on Robotics and Automation*, 15(5):897–917, October 1999.
3. Colombo, C and Del Bimbo, A. Generalized Bounds for Time to Collision from First-Order Image Motion. In *7th IEEE International Conference on Computer Vision*, pages 220–226, Corfu, Greece, September 1999. IEEE.
4. Harville, M; Rahimi, A.; Darrel, T.; Gordon, G. and Woodfill, J. 3D Pose Tracking with Linear Depth and Brightness Constraints. In *IEEE International Conference on Computer Vision*, 1999.
5. Konolige, Kurt. Small Vision System - Development System. <http://www.ai.sri.com/~konolige/svs/svs.htm>.
6. Shieh, Jen-Yu; Zhuang Hanqi and Sudhakar, R. Motion Estimation from a Sequence of Stereo Images: A Direct Method. *IEEE Transactions on System, Man and Cybernetics*, 24(7):1044–1053, July 1994.
7. Waxman, Allen M. and Duncan, James H. Binocular Image Flows: Steps Toward Stereo-Motion Fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):715–729, November 1986.

This article was processed using  
the  $\text{\TeX}$  macro package with SIRS2001 style