A Benchmark on Stereo Disparity Estimation for Humanoid Robots

Jurgen Leitner, Alexandre Bernardino and José Santos-Victor

***Abstract*—This paper presents the experimental evaluation of a dense disparity estimation algorithm, focusing on the most relevant aspects for humanoid robots: real-time functionality and ability to deal with calibration errors. The method and its real time implementation are briefly described and several tests show its performance and the quality of its output as a function of the design parameters. Benchmark tests illustrate the computational cost of the method, implemented in C++ on a standard desktop computer, with open source libraries.**

I. Introduction

HUMANOID robotics is becoming a field of large interest for the development of personal robotic assistants. These robots are intended to be used in general social environments and interact with non-specialized humans. Having both an anthropomorphic aspect and human-like behavior are essential factors for acceptance and friendly interaction in human environments. Of major importance in the perceptual system of these robots is the ability to understand the depth at which objects are present in the environment, i.e. separating between objects that are close to the robot and therefore should be the focal point of its attention, from objects that are farther away from the robot (in the background of the robot's perception).

Stereo depth perception is based on the existence of disparities between the projections of 3D points in both retinas. Several methods have been proposed in the last decades to measure disparity in stereo images. For a recent review see [1]. Despite the multitude of existing disparity estimation methods, there are two strong requirements in humanoid robots that limit the application of most of the methods. One of the requirements is the need to perform in real-time. The other requirement is the ability to deal with calibration errors, since some humanoid robots have active stereo heads that may move during execution, thus being difficult to maintain a precise calibration. The system employed in this work is the stereo active head of robot Baltazar (Figure 1). The head is weak calibrated, i.e. from the encoder angles of the cameras' joints it is possible to have a coarse estimate of the extrinsic parameters (calibration information). However, the measurement of the encoder angles is not synchronized with the image acquisition times, and there are several misalignments in the mechanical setup, so the images are never perfectly calibrated.

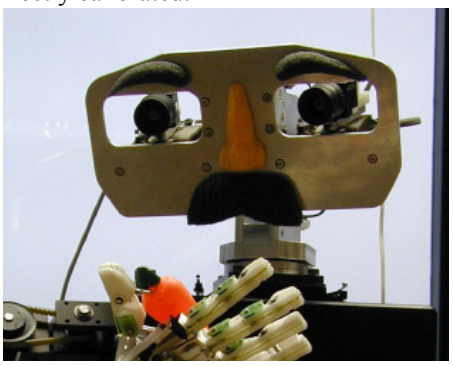

Figure 1: The active stereo head of robot Baltazar.

Most existing methods running in real-time do so by assuming a perfectly calibrated stereo camera setup, where disparity computation turns to a 1D search problem, along the epipolar lines. However, for non calibrated (or weakly calibrated) systems, some search must be done off the epipolar lines. A method that addresses this issue is the presented in [2]. It computes dense disparity maps by testing multiple disparity hypotheses in a Bayesian framework, including vertical disparities to take into account imperfect calibration.

An interesting aspect of that work is that disparity hypotheses can be added or removed in run time, thus adapting the disparity resolution and computational time required, as a function of the task. In that work it was used a log-polar representation of images [3] that reduces the number of pixels to process, and thus achieved real-time performance at the expense of loosing resolution in the periphery of the visual field. Also, commercial optimized software libraries were used. In the present work we aim at evaluating the applicability and real-time performance of the algorithm in [2] with conventional Cartesian images and standard open-source software libraries. We have performed tests for evaluating the quality of the computed disparity maps as a function of the algorithm parameters, and tests for measuring the computational cost of the method as a function of the number of disparity hypotheses and image sizes.

The paper is organized as follows. In Section II we briefly describe the method employed in the paper. In Section III we present some details on the method's implementation. Then in Section IV we show a qualitative evaluation of the disparity estimation method as a function of the design parameters. In Section V we perform tests for the computational benchmarking of our implementation. Finally in Section VI we present the conclusion of this work and future research topics.

This work was supported by the European Commission, Project IST-004370 RobotCub, and by the Portuguese Government - Fundação para a Ciência e Tecnologia (ISR/IST plurianual funding) through the POS_Conhecimento Program that includes FEDER funds, and through project BIO-LOOK, PTDC/EEA-ACR/71032/2006."

J. Leitner is a Master Student at the course of "Space, Science & Technology" at Luleå University of Technology, Sweden. (e-mail: juxi.leitner@ gmail.com).

A. Bernardino and José Santos-Victor are with the Institute for Systems and Robotics, Instituto Superior Técnico, Lisboa, Portugal. (e-mail: {alex,jasv}@isr.ist.utl.pt).

II. Disparity Estimation Method

The basic principle for depth perception in stereo systems is the estimation of disparity from the world information projected in the cameras. Given the coordinates of P corresponding points in the left and right images, $\{(x_l^i, y_l^i)\}_{i=1\ldots P} \rightarrow \{(x_r^i, y_r^i)\}_{i=1\ldots P}$, we define horizontal disparity by:

$$d^i = x_l^i - x_r^i$$

Vertical disparity is defined analogously.

To estimate disparity from the visual information contained in the stereo pair of images, we employ the method in [2], that formulates the disparity estimation problem in a Bayesian framework, according to the following steps:

1. Define a finite discrete set of possible disparities and corresponding prior probabilities.
2. Given each disparity in the set, compute the likelihood of each pixel in the stereo pair, using a generative probabilistic model.
3. Use the Bayes rule to compute the posterior probability of each disparity value at every pixel, given the image data.
4. Reinforce the likelihood of similar disparities at close spatial locations to deal with multiple possible disparities in uniform image regions.
5. Identify, for each pixel, the disparity value with highest posterior probability.

In the following paragraphs we will describe in more detail each one of these steps.

A. The Prior Model

Taking the left image as the reference, disparity at point $\mathbf{x}_l$ is given by $\mathbf{d}(\mathbf{x}_l) = (x_l\text{-}x_r, y_l\text{-}y_r)$ where $\mathbf{x}_l = (x_l, y_l)$ and $\mathbf{x}_r = (x_r, y_r)$ are the locations of matching points in the left and right images, respectively. If a pixel at location $\mathbf{x}$ in the reference image is not visible in the right image, we say the pixel is occluded and disparity is undefined ($\mathbf{d}(\mathbf{x})=\varnothing$). Let us consider a discrete finite set of disparities D, representing the disparity values which are more likely to exist in a certain environment:

$$D = \{\mathbf{d}_n\},\ n = 1\ \ldots N$$

For each location $\mathbf{x}$ in the left eye, we define the following set of hypotheses:

$$H = \{h_n(\mathbf{x})\},\ n = 0\ \ldots N$$

where h_n represents a particular disparity value $\mathbf{d}(\mathbf{x}) = \mathbf{d}_n$. Hypothesis $h_0(\mathbf{x})$ represents the occlusion condition ($\mathbf{d}(\mathbf{x}_l)= \varnothing$). We make the following assumptions for the prior likelihood of each disparity hypothesis:

- The prior probability of occlusion is constant for all pixels:

$$\Pr(h_0)=q \qquad (1)$$

- Valid disparity values are uniformly distributed. A constant prior is considered and its values must satisfy $\Pr(h_n)\, N + q = 1$, which results in:

$$\Pr(h_n) = (1\text{-}q)/N \qquad (2)$$

B. The Likelihood Model

This phase of the algorithm consists in evaluating the likelihood of each possible disparity hypothesis, at each pixel in the acquired pair of images, $L_n(\mathbf{x})$.

We can imagine having, for each pixel, a set of computational units tuned to each of the disparities $\mathbf{d}_n$, that compute the degree of match between a pixel at location x in the left image and a pixel at location $\mathbf{x} - \mathbf{d}_n$ in the right image. This is illustrated in Fig. 4.3.

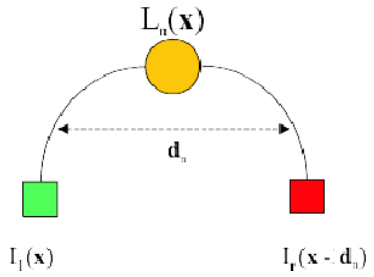

Figure 2. The likelihood function computes the similarity between pixels in the left and right images for each disparity hypothesis $\mathbf{d}_n$.

The disparity likelihood function $L_n(\mathbf{x})$ is defined according to the following assumptions:

- The appearance of object pixels do not change with view point transformations (Lambertian surfaces) and cameras have the same gain, bias and noise levels. This corresponds to the well known Brightness Constancy Assumption [5]. Considering the existence of additive noise, η, we get the following stereo correspondence model:

$$I_l(\mathbf{x}) = I_r(\mathbf{x} - \mathbf{d}(\mathbf{x})) + \eta(\mathbf{x})$$

- In the non-occluded case, the probability of a certain gray value $I_l(\mathbf{x})$ is conditioned by the value of the true disparity $\mathbf{d}(\mathbf{x})$ and the value of I_r at position $\mathbf{x} - \mathbf{d}(\mathbf{x})$. Restricting disparity values to the set D, we write:

$$\Pr(I_l|h_n, I_r) = \Pr(I_l(\mathbf{x})|\, \mathbf{d}_n, I_r(\mathbf{x} - \mathbf{d}_n))$$

- Noise is modeled as being independent and identically distributed with a certain probability density function, f. Thus, the above likelihood is given by:

$$\Pr(I_l|h_n, I_r) = f(I_l(\mathbf{x}) - I_r(\mathbf{x} - \mathbf{d}(\mathbf{x})))$$

Generally, zero-mean Gaussian white noise is accepted as a reasonable model, thus having

$$f(t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{t^2}{2\sigma^2}}$$

where σ^2 is the noise variance.

- If a pixel at location $\mathbf{x}$ is occluded in the right image, its gray level is unconstrained and can have any value in the set of M admissible gray values,

$$\Pr(I_l|h_0, I_r) = 1/M \qquad (3)$$

Given the previous assumptions, the disparity likelihood function is given by:

$$L_n(x) = \Pr\big(I_l(x) \mid h_n(x), I_r(x)\big) = \\ = \begin{cases} f\big(I_l(x) - I_r(x - d_n)\big) & \Leftarrow \quad n \neq 0 \\ M & \Leftarrow \quad n = 0 \end{cases} \tag{4}$$

C. The Posterior Model

With the previous formulation, the disparity estimation problem fits well in a Bayesian inference framework. The probability of a certain hypothesis given the image gray levels (posterior probability) is given by the Bayes' rule:

$$\Pr(h_n \mid I_l, I_r) = \frac{\Pr(I_l \mid h_n, I_r)\Pr(h_n, I_r)}{\sum_{i=0}^{N} \Pr(I_l \mid h_n, I_r)\Pr(h_n, I_r)}$$

where we have dropped the argument **x** because all functions are computed at the same point. Since the unconditioned random variables h_n and I_r are independent, we have $\Pr(h_n, I_r) = \Pr(h_n)\Pr(I_r)$. Using this and Eq. (4), the above equation simplifies to:

$$\Pr(h_n \mid I_l, I_r) = \frac{L_n \Pr(h_n)}{\sum_{i=0}^{N} L_i \Pr(h_i)} \tag{5}$$

Now, substituting the priors (1), (2) and (3) in (5), we get the disparity posterior probability:

$$\Pr(h_n \mid I_l, I_r) = \begin{cases} L_n / R & \Leftarrow \quad n \neq 0 \\ \dfrac{qN}{M - qM} \Big/ R & \Leftarrow \quad n = 0 \end{cases}$$

$$R = \sum_{i=1}^{N} L_i + \frac{qN}{M - qM}$$

D. MAP Estimation

The choice of the hypothesis that maximizes the above equations leads to the MAP (maximum a posteriori) estimate of disparity. Since R is just a normalization factor, this leads to the solution:

$$\hat{\mathbf{d}} = \arg\max_n P_n$$

$$P_n = \begin{cases} L_n & \Leftarrow \quad n \neq 0 \\ \dfrac{qN}{M - qM} & \Leftarrow \quad n = 0 \end{cases} \tag{6}$$

However, without any further assumptions, there may be many ambiguous solutions. It is known that in the general case, the stereo matching problem is under-constrained and ill-posed [1], especially in image regions with uniform brightness. On a pixel by pixel basis, in low-textured image areas, the disparity posterior probability may have similar values for many disparity hypotheses. One way to overcome this problem is to assume that neighbor pixels tend to have similar disparities. In this work we will assume that the scene is composed by piecewise smooth surfaces, and will allow spatial interactions between neighboring pixels.

E. Dealing with Ambiguity

There is evidence on the existence of neurons, in the visual cortex area V2 in primates, that are tuned to similar disparities and are organized in clusters [6]. Thus, in a biological perspective, the value of the likelihood images L_n at each image location **x** can be interpreted as the response of disparity selective binocular neurons in the visual cortex, expressing the degree of match between corresponding locations in the right and left retinas. When many disparity hypotheses are likely to occur (e.g. in textureless areas) several neurons tuned to different disparities may be simultaneously active. In a computational framework, this "aperture" problem is usually addressed by allowing neighborhood interactions between units, in order to spread information from non-ambiguous regions to ambiguous regions.

A formal representation of these interactions leads to Markov Random Field techniques, whose existing solutions (annealing, graph optimization) are still very computationally expensive. Neighborhood interactions are also very commonly found in biological literature and several cooperative schemes have been proposed, with different facilitation/inhibition strategies along the spatial and disparity coordinates [4].

For the sake of computational complexity we propose a spatial-only facilitation scheme whose principle is to reinforce the output of units at locations whose coherent neighbors (tuned for the same disparity) are active. This scheme can be implemented very efficiently by convolving each of the likelihood images with a low-pass type of filter, resulting on N + 1 *Filtered Likelihood Images*, F_n. We use a fast IIR isotropic separable first order filter, that only requires two multiplications and two additions per pixel. The filter is implemented in two cascaded passes (forward-backward), in both the horizontal and vertical directions:

$$T_1(x,y) = (1-\alpha)L(x,y) + \alpha T_1(x-1,y)$$
$$T_2(x,y) = (1-\alpha)T_1(x,y) + \alpha T_2(x+1,y)$$
$$T_3(x,y) = (1-\alpha)T_2(x,y) + \alpha T_3(x,y-1)$$
$$F(x,y) = (1-\alpha)T_3(x,y) + \alpha F(x-1,y)$$
$$\text{with} \quad 0 < \alpha < 1$$

In the equation above, $L(x,y)$ represents each of the original likelihood images, $F(x,y)$ are the filtered likelihood images, and T_i are auxiliary images for the intermediate steps of the filtering stage.

Filters of large impulse response (large α) are preferred because information is spread to larger neighborhoods and favor larger objects, at the cost of missing small or thin structures in the image.

To compute the final solution, in Eq. (6) we replace L_n by F_n, and the disparity estimate for each pixel becomes:

$$\hat{\mathbf{d}}(\mathbf{x}) = \arg\max_n P_n(\mathbf{x})$$

$$P_n(\mathbf{x}) = \begin{cases} F_n(\mathbf{x}) & \Leftarrow \quad n \neq 0 \\ \dfrac{qN}{M - qM} & \Leftarrow \quad n = 0 \end{cases}$$

III. IMPLEMENTATION

The algorithm presented in the previous section was implemented in C++ following an object oriented paradigm, and interfacing with the OpenCV, Open Source Computer Vision library [7].

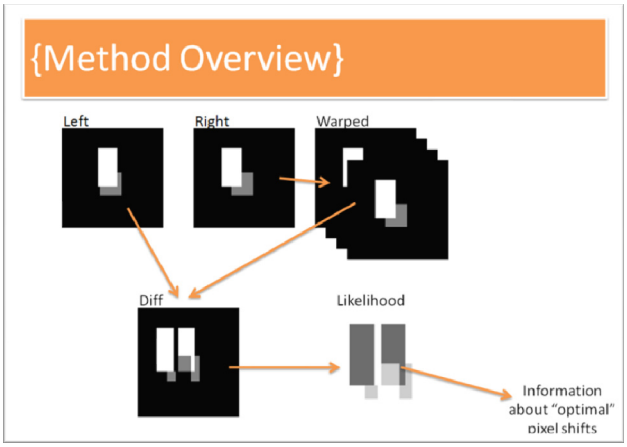

Figure 3: Overview of the method used to estimate depth

Figure 2 shows the basic methodology of the algorithm used. It stores the two images from the cameras (which provide stereoscopic images) and starts by taking one of the images (usually the right) and generating "warped" images out of it, i.e. shifting the image according to the several disparity hypotheses. The shifting is usually done in horizontal and vertical ranges.

A common configuration used in the tests is a horizontal range from 30px to the left (-30px) to 30px to the right, in steps of 1px and a vertical range from 6px up (-6px) to 6px down in steps of either 1px. These 427 (61 horizontal times 7 vertical) images were stored and used in the following step. From these "warped" images we generate "DiffImages" (difference images) which are easily computed by doing a subtraction of the warped image from the other camera image (the left one).

To generate the likelihood images, L_n, out of "DiffImages", they are processed according to Eq. (4), which involves an exponentiation operation using either a Gaussian or Laplace model. Here some parameters must be appropriately selected and tuned as a function of the cameras characteristics.

The next step is not shown in the above figure, for the sake of simplicity, but constitutes an important step in the method: it applies a low-pass filter to the likelihood images to spread information from neighboring regions in the map, which is especially useful in uniform areas in the images. Here we use a low pass fast Infinite Impulse Response (IIR) of first order filter.

From the filtered likelihood images a maximum activation map and a horizontal as well as a vertical disparity map is generated. This is done by simple comparing, for every pixel, which of the filtered likelihood images has the highest activation (likelihood). The disparity maps store the corresponding pixel shifts in the vertical and horizontal axis respectively. The horizontal disparity map is representative of stereoscopic data because from the horizontal disparity alone we can infer the relative depth of objects in the scene.

Figure 4 shows a typical disparity map obtained from a stereo pair of images. The images were acquired from the active stereo head from robot Baltazar (Figure 1).

Figure 4: A sample disparity map corresponding to the stereo pair in the top.

IV. QUALITATIVE EVALUATION

In this section we evaluate qualitatively the performance of the method in terms of the resulting disparity maps. We also evaluate the influence of some design parameters such as the inclusion of vertical disparity channels. And the filter bandwidth in the filtering steps.

The first comparison, shown in Figure 5, illustrates the influence of using vertical disparity channels in the disparity hypotheses set. We show results using 1, 3, 5, and 7 vertical disparity hypotheses, with steps of two pixels. We can notice some improvements when going from 1 to 3 disparity hypotheses, but beyond 3 channels the improvements are not very significant.

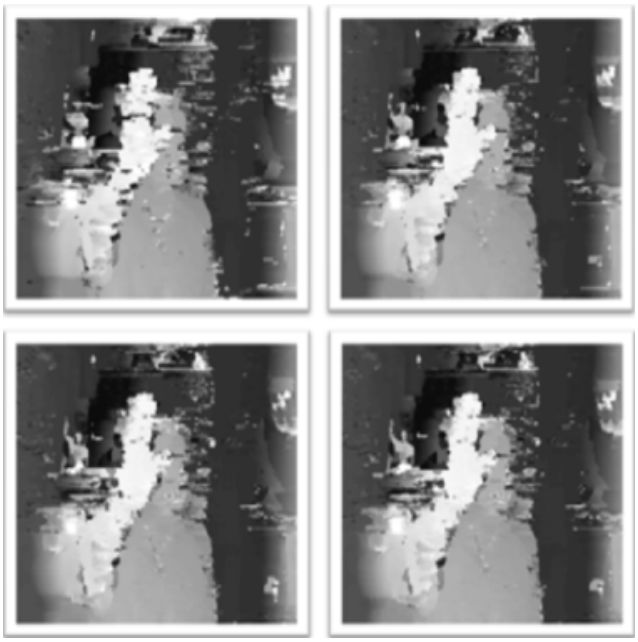

Figure 5. Comparison of disparity maps for several vertical disparity hypotheses. From top-left clockwise: 1 hypothesis (0); 3 hypotheses (-2,0,2); 7 hypotheses , (-6,-4,-2,0,2,4,6); and 5 hypotheses (-4,-2,0, 2,4).

We recall that these images were weakly calibrated, i.e. were rectified according to a coarse estimate of the stereo head joint angles. For completely non calibrated systems, the further increase in the number of vertical disparity channels may improve the results.

The second test shows the influence if increasing the resolution of the vertical disparity shifts. Whereas the previous test generated vertical disparity channels with steps of two pixels, in this test we compare steps of 1 pixel. It can be seen in Figure 6 that there is a very slight improvement in using 1 pixel shifts, but the improvement is not significant enough to make a definite conclusion about this issue.

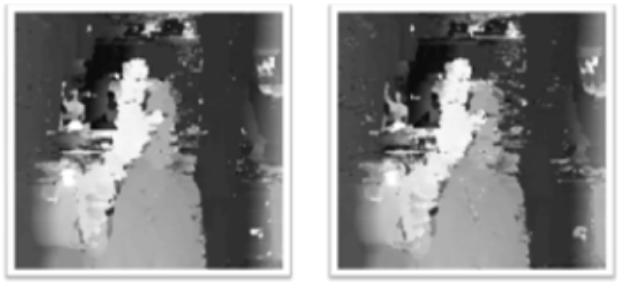

Figure 6. Comparison of disparity maps for different vertical disparity resolution. Left: 1 pixel. Right: 2 pixel.

Finally, in Figure 7 we show the influence of the filter parameter in the quality of the disparity maps. As can be observed, the best results are between 0.6 and 0.7.

V. BENCHMARKING RESULTS

In this section we perform some benchmarking tests on the presented disparity estimation method, evaluating the influence of the image size and number of disparity channels. All the benchmarks were done on a regular P4 3.00 Ghz DualCore PC with Windows XP as operating system and 1GB RAM.

Theoretically, the complexity of the algorithm is linear both with image size and with the number of channels. This is confirmed in all the experimental tests, meaning that there are no architectural bottlenecks in the computational implementation, up to the maximum image sizes and number of disparity channels employed. For larger ones, it is likely that cache and/or disk swap problems may slow down the processing.

The first test shows the performance of the algorithm with varying image sizes. Results are shown in Figure 8 and Table 1. For this test the original image was resized to different sizes and the average of 3 runs was taken for the plot. The number of disparity channels was fixed, horizontal from -30 to 30 with step width 1 (61 channels). No vertical shifts were introduced in this test. The parameter for the Gaussian likelihood is set to .67 and the filter parameter is set to .75. It is obvious the linear increase of the computation time with the number of pixels to process. Processing times range from about 200ms for small images (120x160) to 1.75s for large images (360x480).

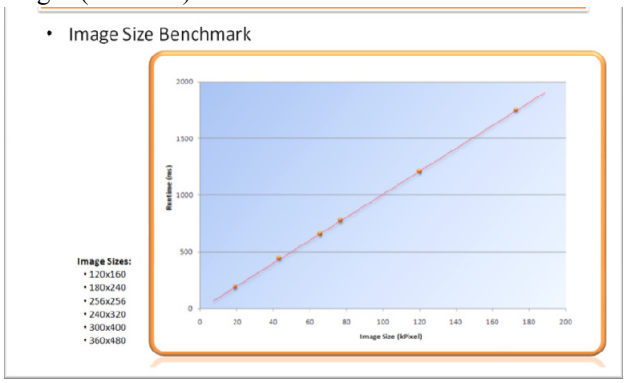

Figure 7. Computation time as a function of image size.

Table 1. Computation time for different image sizes

ticks	run1	run2	run3	avg	pixels
120x160	187	188	187	**187**	19200
180x240	469	437	421	**442**	43200
256x256	656	656	656	**656**	65536
240x320	750	765	812	**776**	76800
300x400	1203	1218	1203	**1208**	120000
360x480	1718	1750	1781	**1750**	172800

The second test shows the performance of the algorithm regarding to the number disparity channels (and therefore how many images are generated in each step of the algorithm). Results are shown in Figure 9 and Table 2. Tests were run with the same image size, 256x256 pixel.

Again, the average of 3 runs was taken for the plot. The shifting range varies only horizontally from (-10, 10) (21 channels) to (-70,70) (141 channels). No vertical shifts were considered. The parameter for the Gaussian likelihood is set to .67 and for spatial filtering is set to .75.

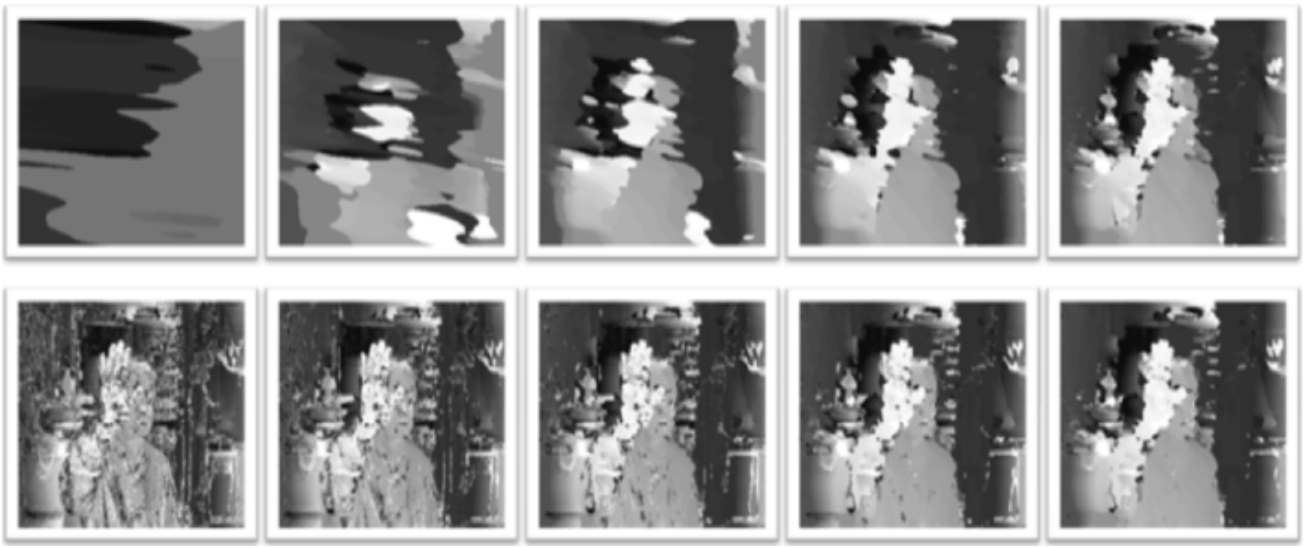

Figure 8. Comparison of disparity maps for different values of the filter parameter. From top-left clockwise: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.99

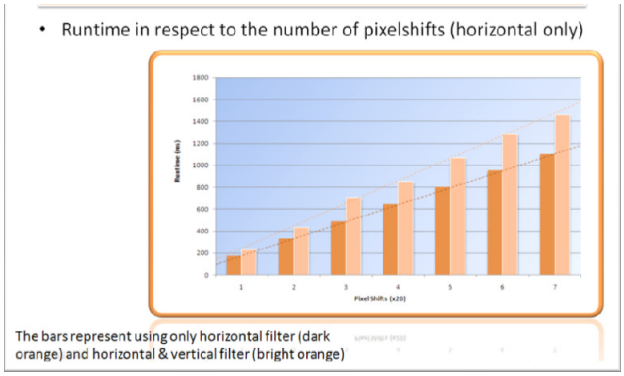

Figure 9. Computation time as a function of disparity channels.

In the plot, different bar colors represent different filtering strategies. The first uses only horizontal filter where the second bar applies horizontal and vertical filter.

It is visible the linear tendency of grouth in the computation time as a function of the number of channels. Computation times go from 200ms, for 21 channels, to 1.4s for 141 channels with both horizontal and vertical filtering steps.

Table 2. Different number of disparity channels.

ticks	run1	run2	run3	avg	with vertical filter enabled			
(-10,10,1)	171	171	172	**171**	**229**	218	234	234
(-20,20,1)	328	328	328	**328**	**432**	437	437	421
(-30,30,1)	500	484	484	**489**	**704**	640	671	802
(-40,40,1)	656	640	640	**645**	**848**	843	859	843
(-50,50,1)	812	796	796	**801**	**1067**	1093	1046	1062
(-60,60,1)	953	953	953	**953**	**1281**	1312	1250	1281
(-70,70,1)	1109	1093	1093	**1098**	**1463**	1453	1453	1484

I. CONCLUSIONS AND FUTURE WORK

In this paper we have presented an experimental evaluation of a dense disparity estimation method for humanoid robots. These robots usually have mobile camera setups that prevent a precise calibration of the images. Therefore, the proposed disparity estimation method is able to consider disparities off the epipolar lines by including hypotheses for vertical disparities in rectified images. We have evaluated the quality of the method with the number and resolution of vertical disparities hypotheses and have verified improvements in the resulting disparity maps.

The method is very flexible because it can add or delete disparity hypotheses in run time, which may be useful in dynamical scenarios and tasks. We have characterized the algorithm in computation time for several image sizes and disparity channels. These results are important for the customization of the algorithm as a function of the required precision and available processing time. We have show configurations whose computation times go from 200ms, for small images and a small number of disparity channels, and about 1.75 sec for large images and an extended number of disparity channels.

In future work we aim at further improvements by using not only information on the image gray level but also on its gradient. We believe that contour information will provide an important constraint for the disparity computation, to further reduce the ambiguities in the estimation process that produce spurious and irregular estimates in the disparity maps.

REFERENCES

[1] D. Scharstein and R. Szeliski. "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms". *International Journal of Computer Vision*,47(1):7–42, April-June 2002.

[2] A. Bernardino and J. Santos-Victor. "A Binocular Stereo Algorithm for Log-Polar Foveated Systems". *2nd Workshop on Biological Motivated Computer Vision*, Tübingen, Germany. 2002.

[3] E. Schwartz. "Spatial mapping in the primate sensory projection : Analytic structure and relevance to perception". *Biological Cybernetics*,25:181–194,1977.

[4] D. Marr and T. Poggio. "Cooperative computation of stereo disparity". *Science*, 194:283–287, 1976.

[5] B. Horn. *Robot Vision*. MIT Press, McGraw Hill, 1986.

[6] D. Hubel and T. Wiesel. "Stereoscopic vision in macaque monkey. cells sensitive to binocular depth in area 18 of the macaque monkey cortex". *Nature*, 225:41–42, 1970.

[7] Intel Research. Open Computer Vision Library. http://www.intel.com/research/mrl/research/opencv/