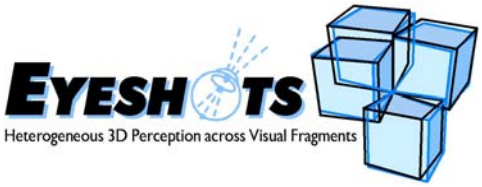

Project no.: FP7-ICT-217077
Project full title: Heterogeneous 3-D Perception across Visual Fragments
Project Acronym: EYESHOTS
Deliverable no: D2.2b
Title of the deliverable: Algorithm for 3D scene description through interactive visual stereopsis adaptation using the mechatronic system.

Date of Delivery:	28 February 2011	
Organization name of lead contractor for this deliverable:	K.U.Leuven	
Author(s):	K. Pauwels, M. Chessa, N. Chumerin, F. Solari, S.P. Sabatini, M. Van Hulle	
Participant(s):	K.U.Leuven, UG	
Workpackage contributing to the deliverable:	WP2	
Nature:	Technical report and demonstrator	
Version:	1.0	
Total number of pages:	15	
Responsible person:	Marc Van Hulle	
Revised by:	S.P. Sabatini	
Start date of project:	1 March 2008	**Duration:** 36 months

Project Co-funded by the European Commission within the Seventh Framework Programme		
Dissemination Level		
PU	Public	**X**
PP	Restricted to other program participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract:

An algorithm is presented for the transformation of retinal disparity into a 3D scene description based on head-centric disparity. This transformation accounts for general eye movements (including vergence and version movements supported by the mechatronic system), and since it operates directly on the response of a population of binocular energy neurons, it also solves the 2D correspondence problem by decoding the population response. Due to the complexity of the transformation, a learning approach has been used to determine the weights of a feedforward neural network. This same approach also enables learning a transformation for (limited) gaze estimation directly from the population response, which allows for dealing with inaccuracies of the motor system. In a final section, the feasibility of such an autocalibration procedure is demonstrated using real-world image pairs obtained with the robotic iCub-platform.

Contents

1 Executive Summary

This deliverable describes the methods and algorithms developed by K.U.Leuven and University of Genoa regarding Task 2.2 (Interactive Depth Perception) of Work Package 2 (Active Stereopsis).

Task 2.2 is concerned with the extraction of depth (3D structure) by integrating retinal disparity information across different eye movements. Transforming disparity from eye- to head-centric coordinates, but also estimating disparity (and controlling vergence) relies on accurate calibration information (in terms of the relative orientation of the eyes). In the active systems considered in EYESHOTS, the motor feedback can not provide this information (due to the limited precision) and therefore vision is used to improve upon this. The procedures developed for this constitute the main part of both this deliverable and deliverable 2.2a.

In deliverable 2.2a, we described, and provided software, for autocalibration methods that can operate in the retinal as well as in the cortical domain. Certain aspects of the previous methods are difficult to align with the experimental evidences reported in various neurophysiological studies. Therefore, we have now applied these same principles to develop a biologically plausible architecture. We no longer explicitly calculate retinal disparity, but operate directly on the response of a population of binocular energy neurons. Image warping operations have been omitted as well.

Using a learning approach a feedforward neural network has been developed that can

directly transform this population response together with the gaze angles into a 3D scene description based on head-centric disparity. Furthermore, the same architecture enables the extraction of (a limited set of) gaze angles directly from these responses.

Since the mechatronic system is not available to us for demonstration purposes, we have applied the autocalibration algorithm to real-world image streams obtained from the iCub-platform [5]. This platform is also used to demonstrate the vergence mechanisms from Task 2.1 and can operate co-jointly with these.

In particular, the methods proposed here can operate together with the vergence mechanisms presented in Task 2.1 in various ways. Improved calibration estimates can feed directly in the convolutional network for vergence control presented in Deliverable 2.1, but can also modulate the weights of the mechanism (also reported there) that integrates the population responses into the vergence control.

2 Introduction

The binocular information obtained through the responses of populations of binocular energy neurons in primary visual cortex, can be used to determine the relative locations of corresponding points in the two eyes. This *retinal* binocular disparity is related to distance but also depends on the orientation of the eyes (the gaze components). Complex disparity patterns typically occur with both horizontal and vertical disparities. To integrate information across different fixations, it is necessary to transform the retinal disparity from an eye-centric into a head-centric frame of reference that does not depend on eye position. A computational model has been proposed for this transformation [1] where noisy oculomotor signals are improved on the basis of vertical disparity. In this model, retinal and oculomotor signals of each eye are integrated *before* computing the disparity. This approach is similar to a warping approach typically used in computer vision methods and enables the system to deal with large eye movements that introduce large disparities. It is however not supported by experimental evidence, since many cells (in V1) are known to be sensitive to retinal disparity [6]. In this work we also rely on such a biologically plausible computational sequence, where first retinal binocular correspondences are found and only then transformed into a head-centric coordinate system.

The brain needs to perform a variety of coordinate transformations between eye-, head-, and body-centered reference frames. A large number of computational and neurophysiological studies have investigated this aspect. An early study points towards the central role of a gain modulation mechanism, by which populations of neuron responses are multiplicatively modulated by another signal, *e.g.* an oculomotor signal [11]. The response peak of individual neurons remains unchanged through such modulation, but arbitrary transformations can be performed downstream, where the population responses are combined. In that study, a black-box feedforward neural network approach, trained with backpropagation, was used to learn the gain modulation weights. The obtained weights were in close correspondence to those recorded experimentally. The transformation considered in [11] was much simpler than the problem considered here, since only monocular retinal signals were used.

Gain modulation has also been investigated in the context of disparity transformation using basis function networks [3, 7, 8]. In that work, the responses of a population of binocular neurons (simulated, not computed from images) are transformed from a retinal to a head-centric frame of reference. The basis function approach simplifies learning, since arbitrary transformations can be approximated using a linear combination of the basis function responses. However, each signal that has to be included increases the dimensionality of the problem, which becomes unmanageable very rapidly [8].

In this deliverable, we consider a problem of much higher complexity. We need to combine a large number of signals, representing all six rotational degrees of freedom of a pair of cameras. This represents a problem for the basis function approach. We also obtain the neuron population responses directly from the images, by filtering with a biologically plausible filterbank. The correspondence problem thus has to be solved as well (albeit implicitly). In addition, we also investigate the possibility of directly estimating gaze angles from the population responses (without oculomotor signal). This enables dealing with the limited accuracy of the motor system and its feedback.

In the next section we first provide more details on the head-centric distance representation used in this work. Section 4 then details the choice of neural network and the training procedure. The results obtained on head-centric disparity and gaze estimation are shown in Section 5. Finally, in section 6, the autocalibration approach we developed is demonstrated on real-world image pairs obtained with the iCub robotic platform.

3 From Retinal to Head-centric Disparity

The disparity pattern in images obtained from cameras in a rectified configuration (with retinal planes coplanar and parallel to the baseline) is one dimensional, and the epipolar lines are horizontal. A simple relation exists between this rectified disparity, δ, and the distance, z:

$$\delta = -\frac{b}{z} \ , \tag{1}$$

with b the distance between the two cameras (baseline). We define the head-centric disparity as this rectified disparity. The gaze angles are expressed relative to this configuration. The retinal disparity pattern observed in general situations is highly complex, since the projection of a 3D rotation is applied to each eye's image [2]:

$$\mathbf{x}'_L = K_L \left(e^{[\boldsymbol{\omega}_L]_\times}\right) K_L^{-1} \mathbf{x}_L \ , \tag{2}$$

$$\mathbf{x}'_R = K_R \left(e^{[\boldsymbol{\omega}_R]_\times}\right) K_R^{-1} \mathbf{x}_R \ , \tag{3}$$

where K is the camera matrix representing the internal calibration, $\boldsymbol{\omega}$ is a vector containing the three rotation angles, and $\mathbf{x}$ and $\mathbf{x}'$ represent the pixel's location (in homogeneous coordinates) respectively before and after the transformation.

The transformation from retinal to head-centric disparity thus requires compensating for the transformations induced by (2) and (3). Since we operate on the responses of a population of binocular energy neurons, this transformation needs to be performed together with correspondence estimation. We have decided not to perform these steps

separately, but to rather directly modulate the responses so as to solve both problems at the same time. The complexity of this modulation warrants a learning approach, which is discussed in the next section.

4 Network Design and Training

As discussed above, gain modulation using basis function networks is not feasible here due to the large number of oculomotor signals that need to be combined with the population response. This leads to an explosion of dimensionality and a more efficient approach is required. We use a traditional black-box approach based on multi-layer perceptrons. Figure 1 provides an overview of the inputs and outputs and the network architecture used in the head-centric disparity estimation and gaze estimation scenarios. In the first scenario, the network combines the population response (which implicitly codes for retinal disparity) with the oculomotor signals (the gaze angles) into the head-centric disparity. In the second scenario, the network estimates the gaze angles directly from the population response. The different components of Fig. 1 are explained in more detail in the next sections.

4.1 Input Images and Target Data Generation

Due to the complexity of the transformation that needs to be learned, a large number of examples are required. It is therefore not feasible to use real-world images, and even the generation of rendered image pairs is prohibitive. We therefore generate textured images using normally distributed random numbers, and warp these in different ways to compose image pairs for our training dataset. A schematic overview of the image generation procedure is provided in Fig. 2.

The randomly generated image serves as cyclopean image. A random disparity field (corresponding to a smooth curved surface, see *e.g.* Fig. 3E) is applied to generate a stereo pair obtained with frontoparallel cameras (rectified). This disparity field corresponds to the head-centric disparity that is used as target data. Uniformly distributed randomly generated 3D rotations are then applied to both images to generate the final image pair corresponding to an arbitrary (but smooth) disparity field and random gaze angles. The input and target data are shown in bold blue in Fig. 2. Although not used in the training, the ground truth retinal vector disparity is also available. Figure 3 contains the image pairs and retinal and head-centric disparity for five randomly selected examples from the dataset.

4.2 Binocular Energy Neuron Population Response

The input images are first processed by a population of simple and complex cells. Each simple cell has a binocular receptive field $g_L(x, y) + g_R(x, y)$ defined by a pair of Gabor functions:

$$g(x, y, \psi, \theta) = e^{-(x_\theta^2 + y_\theta^2)/2\sigma^2} \cos(2\pi k_o x_\theta + \psi) \tag{4}$$

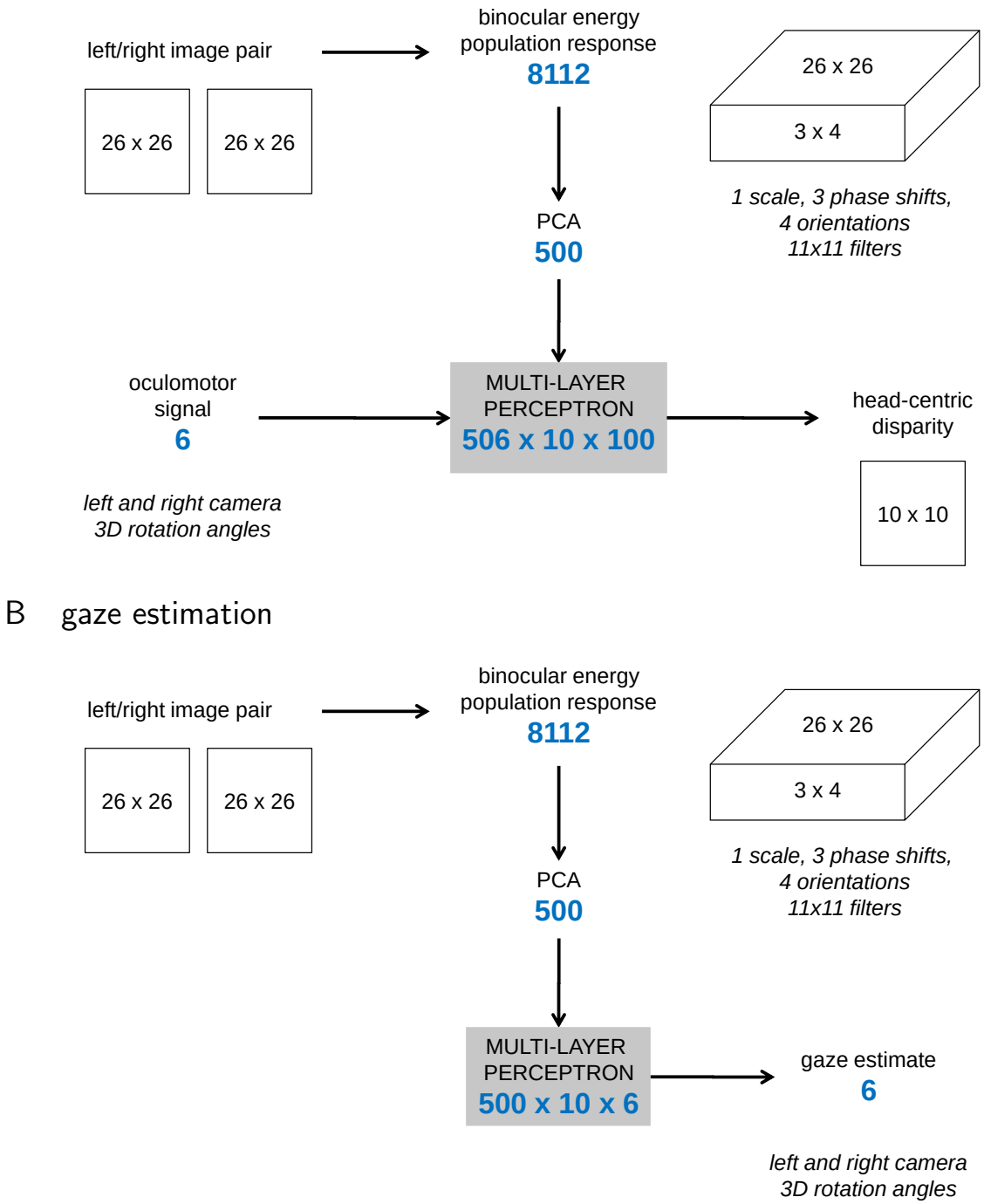

Figure 1: Training procedure and network architecture employed for head-centric disparity estimation (A) and gaze estimation (B).

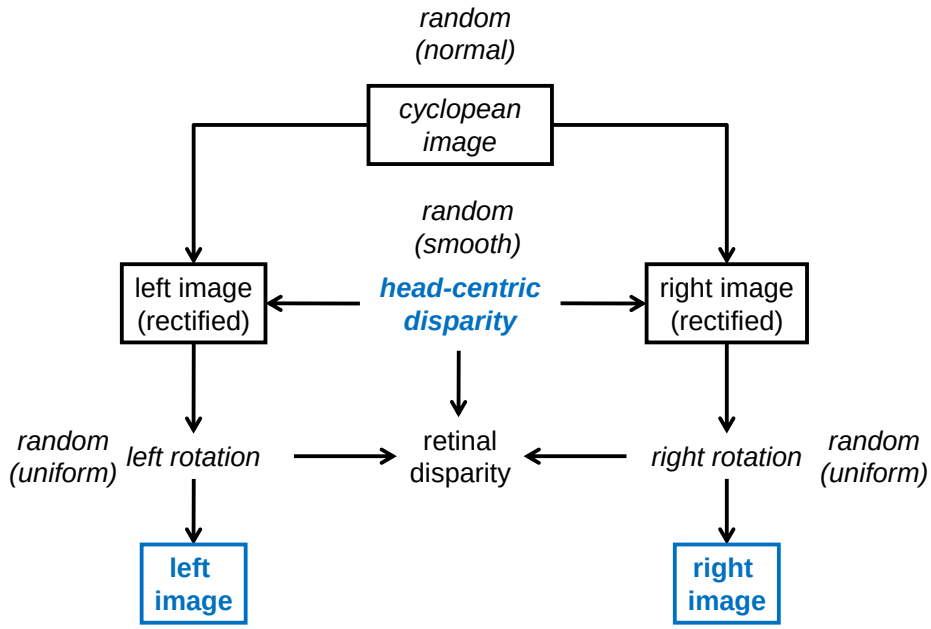

Figure 2: Training data generation. The randomly generated components are in italic and the final input images and target data used for training the network are shown in bold blue.

positioned in the corresponding points $\mathbf{x} = (x, y)$ of the left and the right images, rotated by the same angle θ with respect to the horizontal axis, and characterized by the same peak frequency k_0 and spatial envelope σ, and by a proper binocular phase shift ($\Delta\psi = \psi_L - \psi_R$), along the rotated axis x_θ.

For a specific orientation and phase shift, the simple cell response is obtained as follows:

$$r_s(\mathbf{x}, \theta, \Delta\psi) = (I_L * g_L)(\mathbf{x}) + (I_R * g_R)(\mathbf{x}) \ , \tag{5}$$

where the $*$ operator depicts convolution. The response of a complex cell r_c is obtained by summing the squared response of a quadrature pair of simple cells [9]:

$$r_c(\mathbf{x}, \theta, \Delta\psi) = r_s^2(\mathbf{x}, \theta, \Delta\psi) + r_s^2(\mathbf{x}, \theta, \Delta\psi + \pi/2) \ . \tag{6}$$

The filterbank has been designed with efficiency in mind and relies on 11×11 separable spatial filter kernels [10]. We use a total of four orientations and three phase shifts (both evenly distributed). Since the images are of resolution 26×26, each input sample has a dimensionality equal to 26×26×4×3 = 8112. The training procedure can be made more efficient by first decorrelating the input data. Using principal components analysis, we reduce the dimensionality to 500, while retaining approximately 95% of the variance.

4.3 Network Architecture and Training Procedure

The neural network is designed and trained using Matlab's Neural Network Toolbox. A variety of learning algorithms are provided. Due to the size of the networks considered

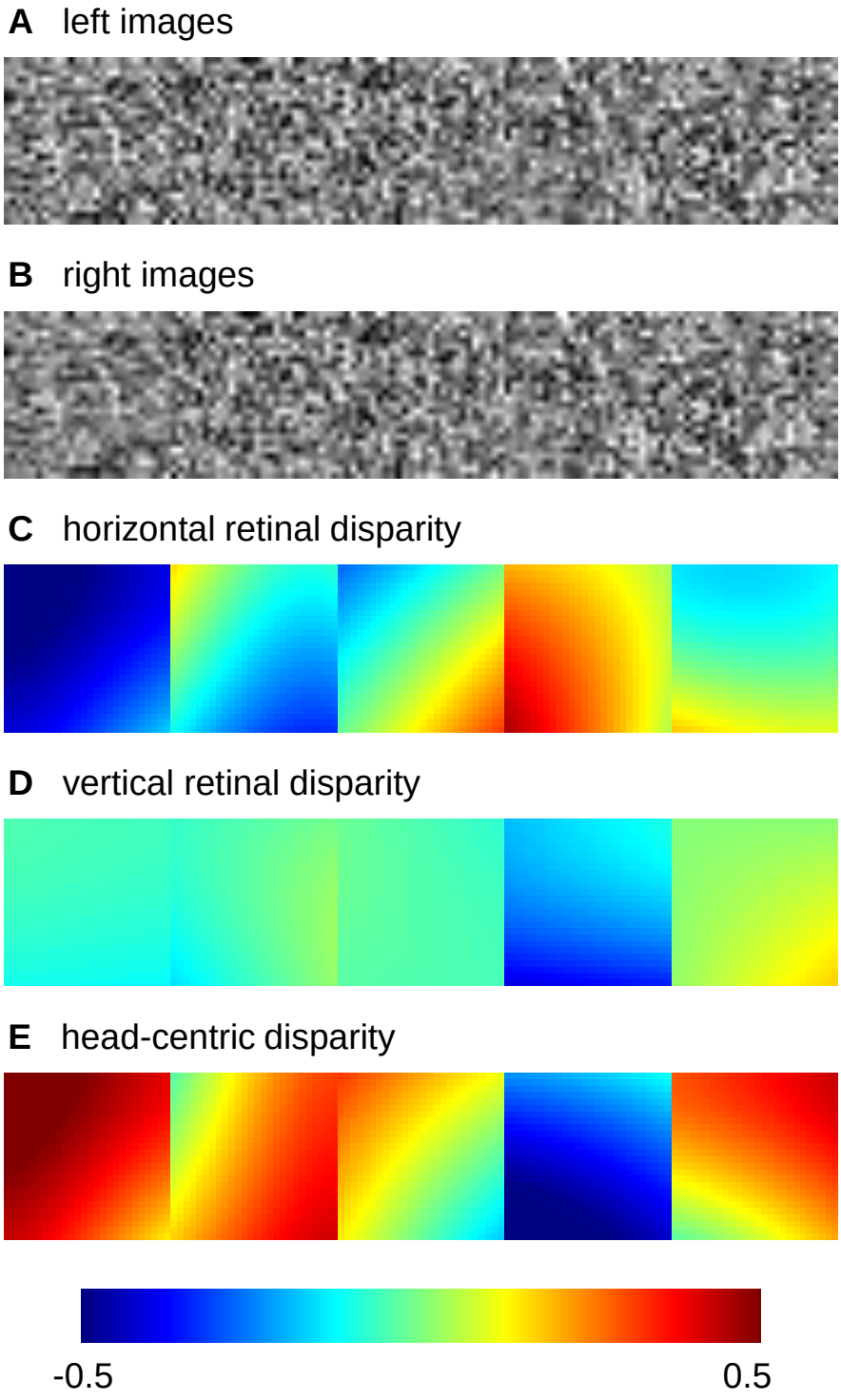

Figure 3: Image pairs (A,B), 2D retinal (C,D) and 1D head-centric (E) disparity for five samples (left to right) from the training set. The same scale is used for all the disparity values, ranging from -0.5 up to 0.5 pixels.

here, we use scaled conjugate gradient backpropagation for its modest memory requirements yet fast convergence [4].

For training, the dataset is randomly divided in three sets: a training, validation, and test set. The training set is used for computing the gradient and updating the network weights and biases. The error on the validation set is monitored during the training process. Training is stopped when this validation error starts to rise. This technique makes it less likely for the network to overfit the data. The test set error is not used during training, but is used to compare different models.

We have always evaluated a large number of different networks, and only report here on the performance obtained with the minimal size network. Larger networks always decreased the training error only.

5 Results

5.1 Head-centric Disparity Estimation

We use a total of 3750 samples for training the network, which is randomly divided in training (70%), validation (15%), and test set (15%). We then evaluate the performance on a completely independent test set consisting of 1250 samples.

To demonstrate that the network is able to correctly apply gaze information, and to show the importance of this information, we compare the performance of the network *with* gaze information shown in Fig. 1A, to a network that only has access to the population response (after PCA). The head-centric disparity map estimated with both networks is shown in Fig. 4(B,C) together with the ground truth head-centric disparity (Fig. 4A) for five typical samples from the test set. Note that the network without gaze input is able to predict the general magnitude of the disparity field, but cannot estimate its fine-structure. The network *with* gaze input performs the transformation with a much higher precision. On the complete test set, the correlation coefficient between the estimates and ground-truth is equal to 0.9192 without gaze input, and 0.9872 with gaze input.

5.2 Gaze Estimation

The flexibility of the neural network approach allows us to investigate the possibility of estimating the gaze itself, or certain components of it, directly from the population responses. This information can be used to correct the imprecise information available through proprioceptive feedback from the motor system, which enables better vergence/version control and more precise correspondence estimation.

We now use the network shown in Fig. 1B. This is very similar to the previous network, except that there is no longer gaze input, and the target has been reduced from the 10×10 head-centric disparity to (at most) six gaze angles.

Due to the complexity of the problem, we have now increased the size of the dataset to 15000 samples for training the network, which is again further divided in training (70%),

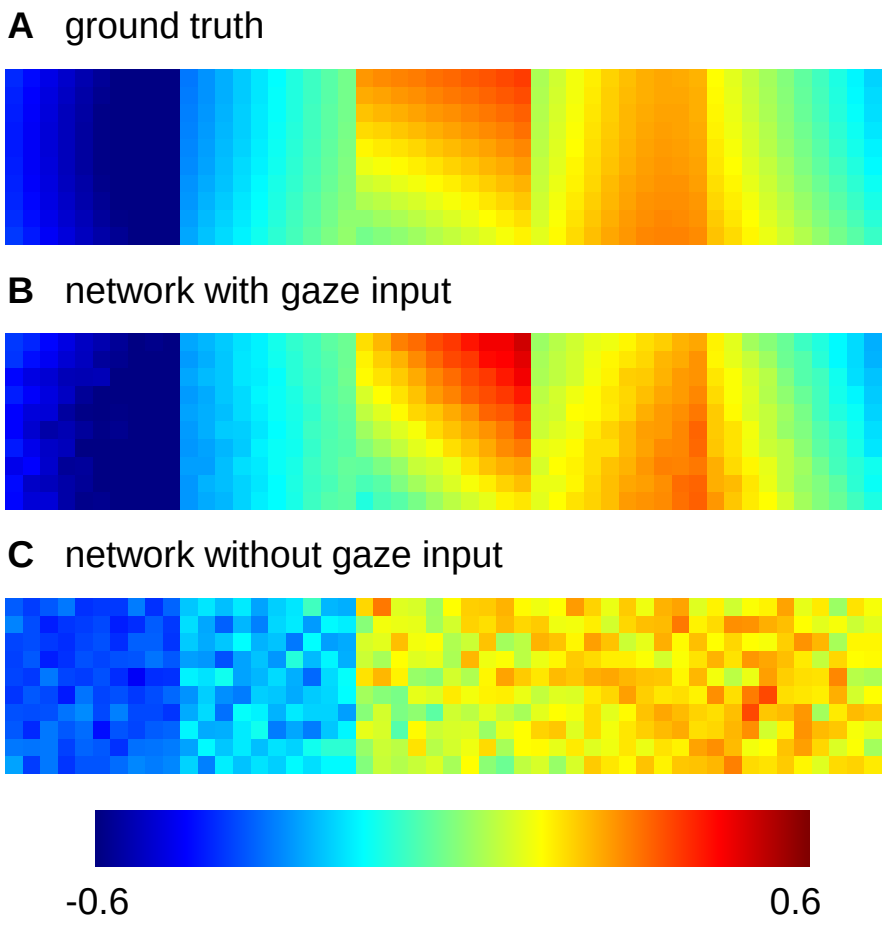

Figure 4: Ground-truth head-centric disparity (A) and head-centric disparity predicted by a network with (B) and without (C) gaze input.

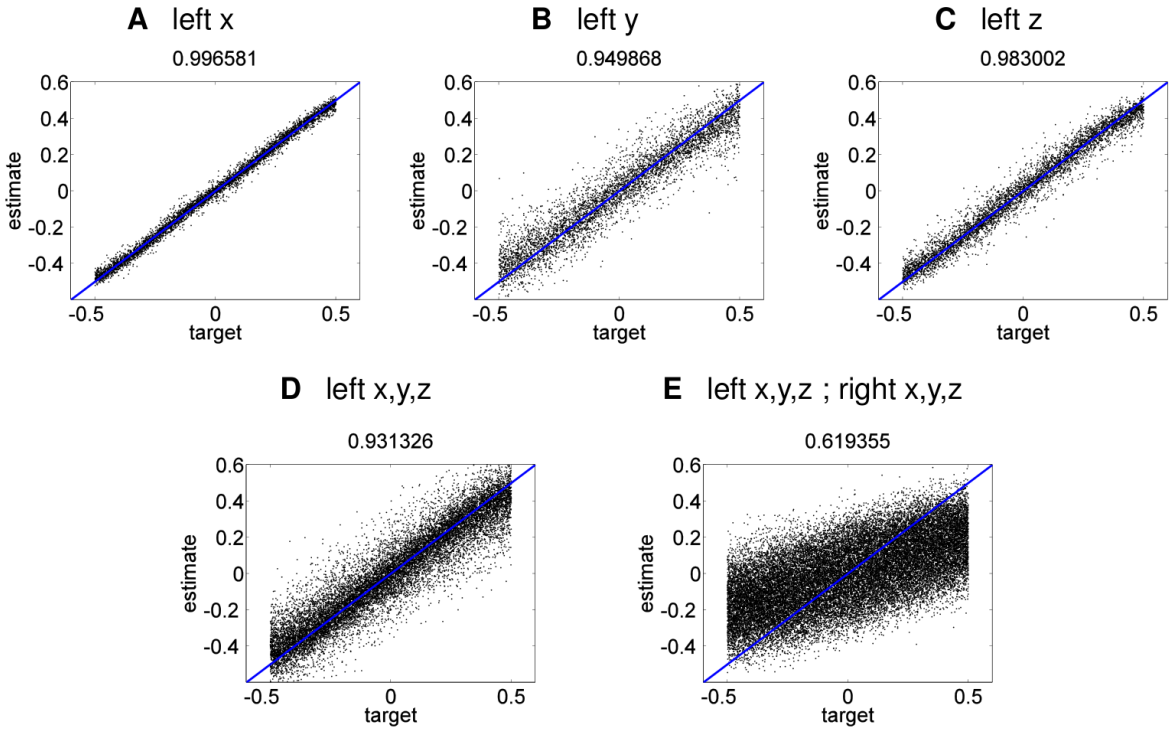

Figure 5: Ground truth versus estimated gaze angle scatter plots for training scenarios of different complexity. In the top row, only a single gaze parameter of the left eye is changed: tilt (A), pan (B), or torsion (C). In the bottom row, all the left eye parameters are changed in (D), and both left and right gaze parameters are changed in (E). Only test set data is shown, and the correlation coefficient obtained on this set is indicated above each figure.

validation (15%), and test set (15%). We then evaluate the performance on a completely independent test set now consisting of 5000 samples.

This problem is notoriously ambiguous, and only the essential matrix can be extracted from image data. Therefore, we have explored a set of problems with gradually increasing gaze complexity. The results are shown in Fig. 5. From this figure, we can see that the performance is quite good when only one eye is considered at a time. In the top row, a single gaze parameter is changed, while all the other remain zero. Both tilt (Fig. 5A) and torsional (Fig. 5C) rotations are easy to predict, because they introduce a strong vertical disparity pattern. Pan movements on the other hand are more difficult (Fig. 5B), because they are more easily confused with the disparity pattern. This however should not affect the precision of correspondence finding, and so is less relevant in this context.

The network is also able to simultaneously estimate all gaze parameters of a single eye (Fig. 5D) with a performance similar to the worst single parameter performance. We also examined the degree to which all gaze angles for both eyes can be predicted together, but (as expected) this did not yield very good performance (Fig. 5E).

The ability to estimate gaze in the presence of complex disparity patterns, appears to be

quite feasible following this approach for each eye, only. This means that also autocalibration is possible with this method, since in the computer vision approach we developed earlier, we used an alternating approach to correct the gaze estimate by considering one eye at a time. We next demonstrate this computer vision approach on real-world image pairs obtained with the iCub robotic platform available to partner UG.

6 Demonstration on the iCub platform

We have applied our multiscale autocalibration algorithm to high resolution real-world images obtained with the robotic iCub platform. This platform contains a robotic head with a pair of cameras that can pan individually and have a common tilt. The platform used here has a significant tilt offset between the cameras, and we demonstrate here how our autocalibration algorithm can correct this.

Figure 6A contains three example stereo pairs shown as anaglyphs with the left image in the red channel and the right image in the blue and green channels. The vertical offset is clearly visible and very large 2D disparities occur in the image. We compare the performance of the autocalibration algorithm to a standard two-frame optical flow algorithm [10] that operates on the same multiscale, multi-orientation filterbank responses. For each trial, the autocalibration algorithm was initialized with a (highly erroneous) rectified configuration.

The horizontal component of the estimated vector disparity is shown in row B for the autocalibration algorithm, and in row C for the optical flow algorithm. We determined invalid estimates (the white regions) using a standard left/right check by running the algorithms from the left to the right image and vice versa, and looking for inconsistent estimates. If the vector difference between both estimates exceeds one pixel, the estimate is considered unreliable and removed. Note that the autocalibration algorithm achieves a much higher density on each occasion. The estimates are also of a much higher precision, as can be seen by comparing rows D and E. Here the vertical component of the estimated disparity is shown, and a much more regular pattern is observed in the autocalibration estimates.

To further demonstrate the correctness of the proposed method, we also show the recovered epipolar geometry in Fig. 7. The red epipolar lines in the right image (B) correspond to the blue keypoints in the left image (A) and vice-versa. It's clear that the estimated geometry is very precise everywhere in the image, and also largely different from a rectified configuration.

Conclusion

We have presented an algorithm for the transformation from retinal to head-centric disparity that operates directly on the response of a population of binocular energy neurons. On the basis of an extensive data set consisting of stereo image pairs and oculomotor signals, a feedforward neural network was trained to both solve the correspondence problem,

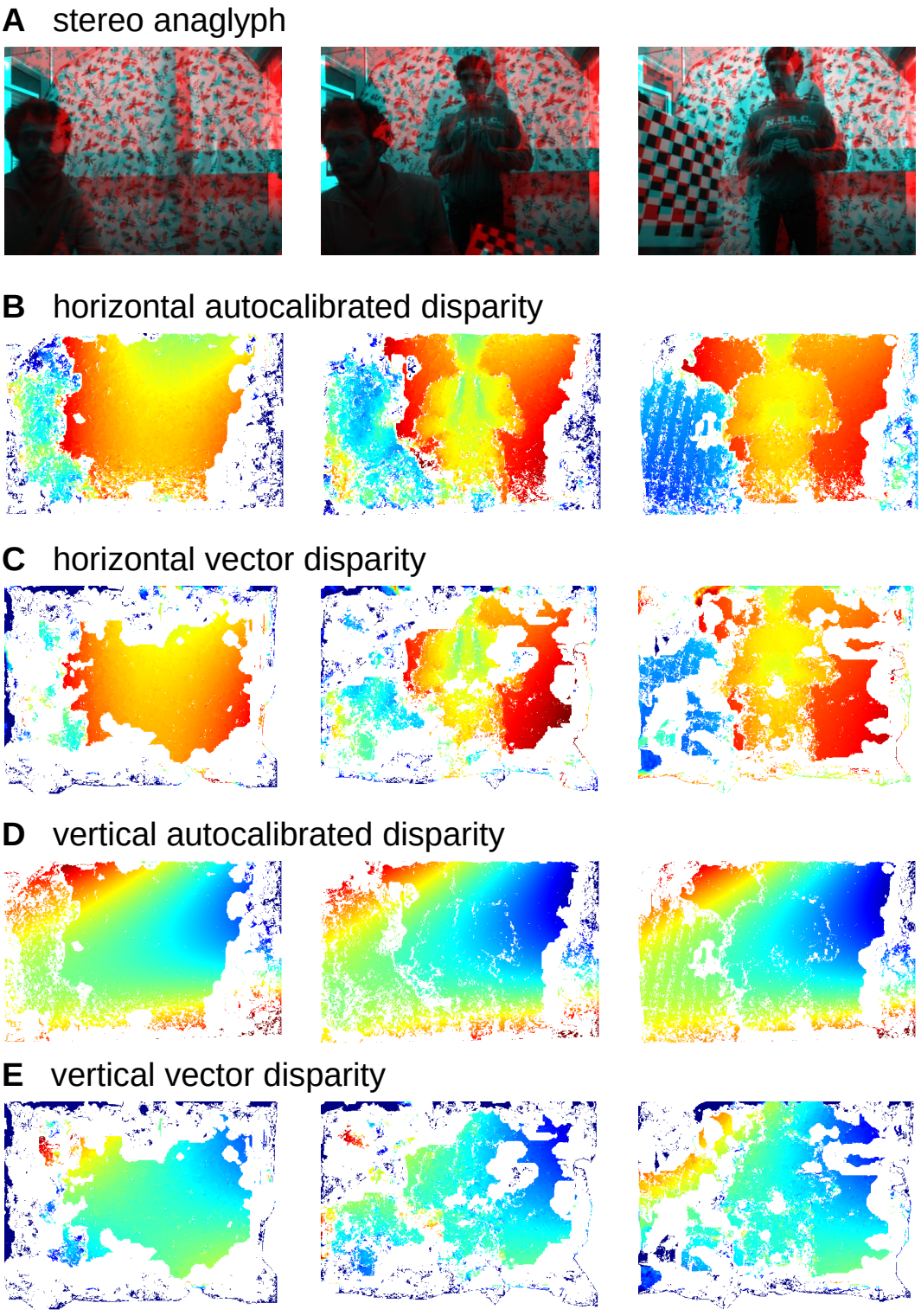

Figure 6: Disparity estimation results on stereo images obtained with the iCub platform. Rows B and D contain the estimates obtained with autocalibration, and rows C and E contain the estimates obtained with a standard vector disparity algorithm (*cf.* optical flow). The horizontal disparities range from -15 (blue) up to 40 pixels (red) and the vertical disparities range from 35 (blue) to 55 pixels (red).

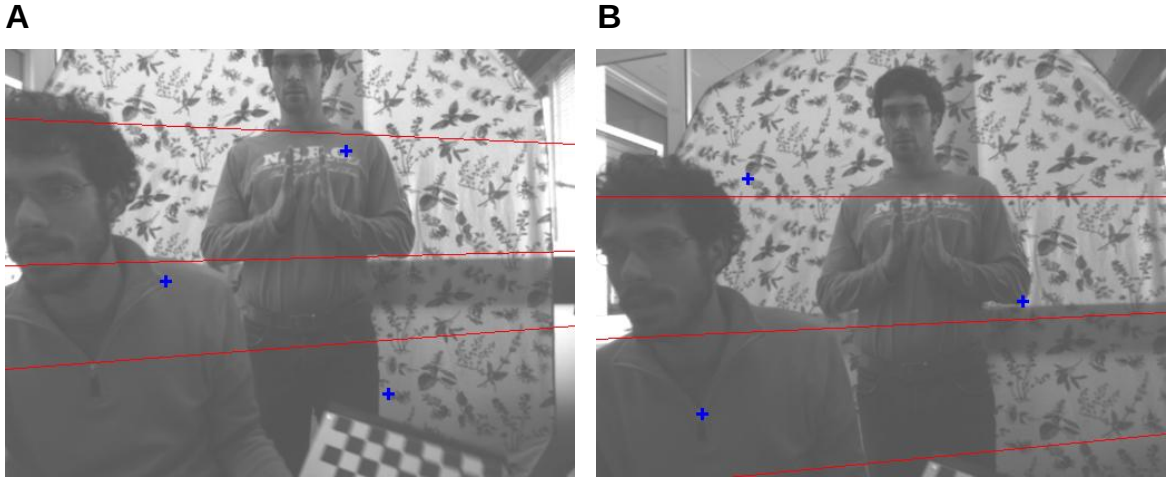

Figure 7: Recovered epipolar geometry for the scenario of Fig. 6 (center column). Red epipolar lines in the right image (B) correspond to the blue keypoints in the left image (A) and vice-versa.

and perform the complex coordinate transformation required to obtain head-centric disparity. Furthermore, the same network architecture was shown capable of extracting a limited set of gaze parameters directly from the population responses. In this way, we have demonstrated the suitability of this biologically-motivated vision-based approach for improving the limited accuracy of the motor system. Finally, we have applied our autocalibration approach to real-world image pairs obtained with the iCub-platform and obtained greatly improved vector disparity estimates as compared to a standard two-frame optical flow algorithm.

References

[1] C.J. Erkelens and R. van Ee. A computational model of depth perception based on headcentric disparity. *Vision Research*, 38(19):2999–3018, 1998.

[2] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[3] S.R. Lehky, A. Pouget, and T.J. Sejnowski. Neural Models of Binocular Depth Perception. *Cold Spring Harbor Symposia on Quantitative Biology*, 55:765–777, 1990.

[4] M.F. Møller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6(4):525–533, 1993.

[5] N. Nosengo. Robotics: The bot that plays ball. *Nature*, 460:1076–1078, 2009.

[6] I. Ohzawa, G.C. DeAngelis, and R.D. Freeman. Stereoscopic depth discrimination in the visual cortex: neurons ideally suited as disparity detectors. *Science*, 249(4972): 1037–1041, 1990.

[7] A. Pouget and T.J. Sejnowski. A Neural Model of the Cortical Representation of Egocentric Distance. *Cerebral Cortex*, 4(3):314–329, 1994.

[8] A. Pouget and T.J. Sejnowski. Spatial transformations in the parietal cortex using basis functions. *Journal of Cognitive Neuroscience*, 9(2):222–237, 1997.

[9] N. Qian. Computing stereo disparity and motion with known binocular cell properties. *Neural Computation*, 6:390–404, 1994.

[10] S.P. Sabatini, G. Gastaldi, F. Solari, K. Pauwels, M.M. Van Hulle, J. Díaz, E. Ros, N. Pugeault, and N. Krüger. A compact harmonic code for early vision based on anisotropic frequency channels. *Computer Vision and Image Understanding*, 114(6): 681–699, 2010.

[11] D. Zipser and R. A. Andersen. A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, 331(6158): 679–684, February 1988.