Integration of static and self-motion-based depth cues for efficient reaching and locomotor actions

Beata J. Grzyb, Vicente Castelló, Marco Antonelli, and Angel P. del Pobil

Robotic Intelligence Lab, Jaume I University, 12071 Castellon, Spain
{grzyb,castellv,antonell,pobil}@icc.uji.es

Abstract. The common approach to estimate the distance of an object in computer vision and robotics is to use stereo vision. Stereopsis, however, provides good estimates only within near space and thus is more suitable for reaching actions. In order to successfully plan and execute an action in far space, other depth cues must be taken into account. Self-body movements, such as head and eye movements or locomotion can provide rich information of depth. This paper proposes a model for integration of static and self-motion-based depth cues for a humanoid robot. Our results show that self-motion-based visual cues improve the accuracy of distance perception and combined with other depth cues provide the robot with a robust distance estimator suitable for both reaching and walking actions.

Keywords: distance perception, depth cue integration, embodied perception, reward-mediated learning

1 Introduction

In the classical approach, a robot uses its sensors to create perception of the environment and then uses these percepts to select an appropriate action. Thus, the connection between perception and action is considered as one direction only, that is only perception has an effect on a subsequent action and not vice versa. Nethertheless, the evidences suggest that many actions are also taken for their perceptual consequences, and the boundary between perception and action sometimes fades. It has been argued that motor actions, such as locomotion, head and eye movements, and object manipulation affect perception and representation of three dimensional objects and space [6].

Space perception, and distance perception in particular, is influenced by the body in many ways [2, 3]. The distance to an object, in many situations, can be directly specified by visual angles inherent in optical information. For example, when the object and the observer are both located on level ground, the distance to the object is a function of its angular elevation scaled to the observer's eye-height. Rich knowledge of the third dimension can also be retrieved from body movements, namely by coordinating the gaze direction of the two eyes, by moving the head to produce parallax, by walking to get a different view of a scene, or by manipulating an object to better see its shape [6].

In computer vision and robotics, a classic solution to reconstruct the depth of the scene is to use stereopsis. The estimates provided by stereopsis are reliable only within near space, and thus they are suitable only for planning and execution of actions constrained to that area, such as reaching and grasping. The effectiveness of other commonly used methods of distance estimation, such as familiar size or range sonars, is also restrained to certain limits. Therefore, in order to successfully plan and execute an action in far space, other depth cues must be taken into account.

In this paper, we propose to use motion-based depth cues, such as motion parallax and motion perspective to improve distance perception in far space. Moreover, we suggest that motion-based depth cues and static depth cues should be combined in an action-specific manner, that is depending on the action that is to be performed. In our approach, we make use of reward-mediated learning, where different visual cues are combined with regard to reaching and walking action. Our results show that self-motion-based visual cues improve the accuracy of distance perception and combined with other depth cues provide a humanoid robot with a coherent representation of near and far space.

2 Depth estimation methods

Accurate estimation of depth is a challenging issue in the field of computer vision. Since three dimensional real objects present in the environment are projected into the two dimensional surfece of the camera sensor the depth information is lost, and additional constraints are required for its extraction. This section briefly introduces some methods of extracting absolute depth information from static and motion-based depth cues.

2.1 Static depth cues

Familiar Size When a physical size of an object is known, its absolute depth can be calculated by using the following equation:

$$z_{fs} = f \cdot \frac{S_{physical}}{S_{observed}} \tag{1}$$

where f is the focal length (in pixel term), $S_{physical}$ and $S_{observed}$ are a physical (in meters) and an observed size (in pixel term) of a known feature, respectively. Equation (1) assumes that the observed feature is presented orthogonally to the camera sensor. In case such a condition is not fulfilled and no information about inclination of the object is provided, the depth estimation can be largly overestimated.

Stereopsis Let θ_1 and θ_2 be the angular positions of the left and of the right cameras which allows for gazing a target object, the depth of such an object can

be computed using equation (2):

$$z_{st} = \frac{b}{tan(\theta_1) - tan(\theta_2)} \tag{2}$$

where b is the distance (baseline) between two cameras. Equation (2) works under the assumption that the nodal point lies on the center of rotation which is not always true. However, the error due to this assumption is usually negligible in many applications.

2.2 Self-motion-based depth cues

Parallax The effectiveness of motion parallax lies in the sensorimotor relationship between the observer movement and consequent retinal image motion, which is dependent on observer movement, scene layout, and point of fixation [7]. Extracting the depth information from the parallax is straightforward once the displacement of the camera Δp and the displacement of the object in the image Δx are known using equation (3):

$$z_{px} = f \cdot \frac{\Delta p}{\Delta x} \tag{3}$$

where Δp is the displacement of the camera. The displacement of the camera can be provided by an inertial sensor. In the case we can access to the instantaneous acceleration a_k with a constant sample period ΔT and by assuming that the acceleration is constant during such a period, it is possible to calculate Δp using the following discrete system:

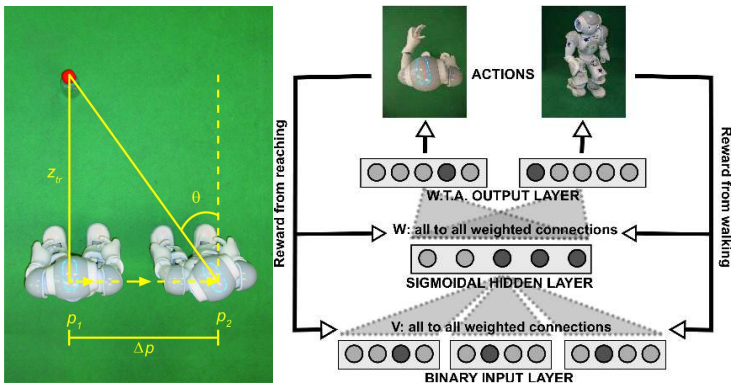
$$\begin{bmatrix} \Delta p_{k+1} \\ v_{k+1} \\ a_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta T & \Delta T^2/2 \\ 0 & 1 & \Delta T \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta p_k \\ v_k \\ a_k \end{bmatrix} \tag{4}$$

where v_k is the velocity and the only observable variable at the time k is the acceleration a_k. The problem of this system is that is a double integrator and a bias in the measure, for example due to the change of the temperature, can lead to an unbounded estimation error.

Motion perspective The motion perspective allows for calculating the depth of a feature using a technique that is a mix between the stereopsis and the parallax. As for the parallax the robot moves perpendicularly to the optical axis, while the yaw motor of the neck is rotated to maintain the fixation as for the stereopsis. Assuming that at the beginning the object is in front of the robot, so the neck is in the position zero, the depth can be estimated as follows:

$$z_{tr} = \frac{\Delta p}{\tan(\theta)} \tag{5}$$

where θ is the angular position of the neck that allow for gazing the target point after the displacement Δp. The displacement can be computed as for the parallax (see equation 4). Figure 1a shows an example of depth estimation using the motion perspective.



(a) Motion perspective (b) Neural network model

Fig. 1: General scheme of the reward-based learning model.

3 Depth cue integration

A reward-based learning approach has been proposed as an alternative for the near-optimal cue integration of auditory and visual depth cues [1][5]. Following this approach, a three-layer neural network (see Fig. 1b) is used to approximate the state-action mapping function. The input layer consists of $i = c * n$ binary neurons that encode the estimates of the c different depth cues covering the n discritized distance units. The activity of the neurons x_i is one at depth estimated by the corresponding cue, otherwise zero.

The input neurons are all-to-all connected with weights $v_{i,j}$ to j neurons in the hidden layer. A sigmoidal transfer function on the sum of the weighted inputs gives the outputs y_j of the hidden neurons:

$$y_j = \frac{1}{1 + e^{-\sum_i v_{i,j} x_i}} \tag{6}$$

The hidden neurons are fully connected to output neurons k with weights $w_{j,k}$. All weights are drawn from uniform distributions, $v_{i,j}$ between -0.1 and 0.1, and $w_{j,k}$ between -1 and 1.

Each output units represents an action. Two types of actions, that is reaching k_r and walking k_w, are possible. The *binning size*, which is the parameter responsible for discretization of the action space is 1 cm for reaching and 3 cm for walking action. The activation of the output neurons z_k is given by the weighted sum of the hidden layer activity, representing an approximation of the appropriate Q-value.

Based on the network's outputs, one action is chosen according to the *softmax* action selection rule [4]:

$$P_t(k) = \frac{e^{Q_t(k)/\tau}}{\sum_{b=1}^{n} e^{Q_t(b)/\tau}} \tag{7}$$

where $P_t(k)$ is the probability of selecting an action k, $Q_t(k)$ is a value function for an action k, and τ is a positive parameter called *temperature* that controls the stochasticity of a decision. A high value of τ allows fo more explorative behavior, whereas low value of τ favors more exploitative behavior. We start with a high temperature parameter $\tau = \tau_0$, so that the selection of action is only weakly influenced by the initial reward expectations. In our experiments, τ decreases exponentially with time $\tau(t) = \tau_0^{(\frac{v_\tau - t}{v_\tau})}$, where $\tau_0 = 10$ and $v_\tau = 50000$.

After performing the selected action $\hat{k}$ the true reward $r(\hat{k})$ is provided. The reward is maximal when $\hat{k}$ equals the true object position k_t, decaying quadratically with increasing distance within a surrounding area with radius ρ (in our case $\rho = 4$) and zero otherwise.

$$r(\hat{k}|X) = max(0, (\rho - |\hat{k} - k_t|))^2 \tag{8}$$

To minimize the error between the actual and expected reward, we make use of gradient descent method which is widely used for function approximation, and is particularly well suited for reinforcement learning [4]:

$$v_{i,j}(t+1) = v_{i,j}(t) - \epsilon(r_{\hat{k}} - z_{\hat{k}})(-w_{j,\hat{k}})y_j(1-y_j)x_i \tag{9}$$

$$w_{j,\hat{k}}(t+1) = w_{j,\hat{k}}(t) - \epsilon(r_{\hat{k}} - z_{\hat{k}})(-y_j) \tag{10}$$

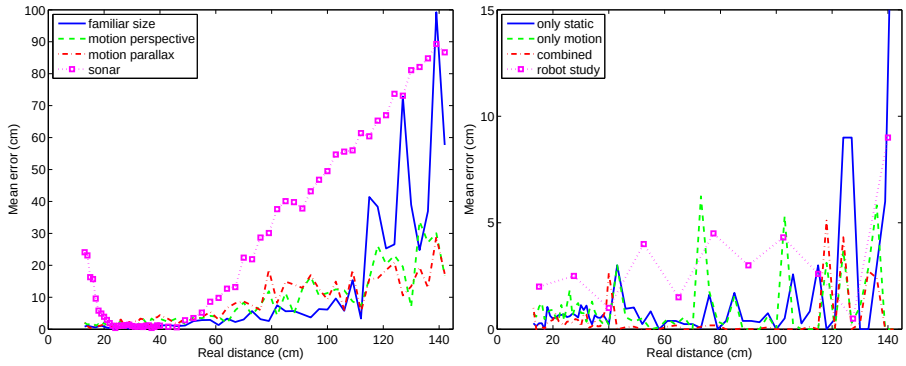
It is worth noting, that in case of the update of weights $w_{j,k}$ only the output weights connected to the winning output unit $\hat{k}$ are updated. The learning rate ϵ, decreases exponentially, according to the formula $\epsilon(t) = \frac{\epsilon_0}{ceil(\frac{t}{v_\epsilon})}$, where $\epsilon_0 = 0.05$, and $v_\epsilon = 100000$.

4 Experimental framework

4.1 Real data collection

Aldebaran's comercially available humanoid robot NAO with 25 DoF is used as a platform for the examined depth estimation methods. Although, the robot is provided with two identical video cameras placed in the forehead, their location does not allow the use of stereo vision methods for depth calculation. The NAO robot is also equipped with a 3-axis linear accelerometers that can be used to measure accelerations and four ultrasonic sensors located in the torso that provides rough distance estimation to obstacles in its surroundings.

The following visual depth estimation methods are tested: familiar size, motion parallax and motion perspective. As sonars are widely used in mobile robots for obstacle detection and their effectiveness is limited to a certain range, we include them for comparison reasons.



(a) Single estimators (b) Merged estimators

Fig. 2: Mean distance estimation errors. Please note that these figures uses different scales for the sake of clarity.

The procedure for collecting depth estimation data is as follows. The object is placed manually in front of the robot approximatelly on its eye-height. The robot centers the objects in the image, and then it estimates the distance to the object using static depth estimation methods (i.e. familiar size and sonar). Afterwards, it executes a lateral movement to the right during 2.4 s. During the movement the robot accesses the acceleration data with a constant sample period ΔT, and calculates its displacement according to the state updating of eq. (4), where Δp and v at the starting time (Δp_0, v_0) were both set to zero. Once the final position is reached, the object distance is calculated again by using the motion parallax method. Then, the image is once again centered so as to calculate the distance by motion perspective. After all distance methods have been calculated, the object is replaced manually for the next trial. The measures are taken every 1 cm for near distances (13 cm to 40 cm) and every 3 cm for middle and far distances (40 cm to 140 cm). Mean estimation values are shown in Fig. 2a. As is clearly visible, all proposed methods give good results within near or middle space. Due to internal constraints, the estimates of the familiar size and sonar methods worsen significantly in far space.

4.2 Results

One of the shortcomings of the reward-based methods is the large number of training examples needed for the neural network to converge. In our simulations we take $t = 100000$ time steps for the learning process. Such a large number of repetitions would be extremely time-consuming and unfeasible for any robot platform. Thus, in this phase weights of the neural network are learned offline with the real data collected with the use of our robot.

Three different setups are prepared, so as to test whether the combined static and self-motion-based methods have any predominance over the usage of only

Table 1: Mean (and STD) of estimation errors of the neural network (in CM)

Testing scenario	Near (13 - 40)	Middle (40 - 100)	Far (100 - 140)
Only static	0.65 (0.64)	0.53 (1.06)	2.21 (0.74)
Only motion	0.77 (0.76)	0.92 (1.58)	1.72 (2.21)
Combined (simulation)	0.37 (0.48)	0.07 (0.36)	1.15 (1.73)
Combined (robot study)	1.83 (0.48)	3.25 (0.36)	2.47 (1.72)

static or only self-motion-based depth estimation methods. As the dataset collected by the robot was quite limited (5 trials per each distance), we added a white noise with the standard error deviation of various depth estimators to the final training dataset. The mean estimation errors (over 10 trials) for all tested setups are shown in Fig. 2b. For the network trained only with static depth cues, the errors in the near and middle space are quite small. The increasing error in distance estimation provided by sonars is nicely compensated by the relatively good estimates given by familiar size method. However, in far space, these errors cannot be compensated any further as the errors given by familiar size also begin to increase, which in turn leads to quite large estimation errors. The network trained with only motion-based depth cues gives overall good results. For very few values, however, the errors are much bigger than the errors in their nearest neighbourhood. These "peaks" may result from a relatively small number of data collected per each distance and used for training, as well as high level of noise in these data. Further work should investigate in more detail the reasons for their occurance. The combination of static and motion-based depth cues, as predicted, give quite small errors throughout tested distances.

The effectiveness of the proposed depth cues integration model was tested with the NAO humanoid robot platform. Only the neural network trained with the combination of static and motion-based depth cues was examined. The procedure was the same as for collecting data with the difference of the number of tested positions. In this case, only 10 different equally distributed position were tested, and the procedure was repeated 5 times. The mean distance estimations are shown along with the results of simulated networks in Fig. 2b. As is clearly seen, the errors given by experiments with the robot are much bigger than the errors obtained in simulations. The real data are quite noisy, which results in bigger errors. The results of a combined cues, however, are much better than any single distance estimatior. The mean values of distance estimation errors in the near, middle and far space for both simulations and robot study are shown in Table 1. The overall estimation error obtained in robot study is less than 3.5 cm, which is good enough to approach the target and if necessary to be corrected by a small random movement of the hand. In case of a misalignment between the robot's predictions and real outcome of action, such an error signal should be used by the robot to automatically update the weights of the network.

5 Discussion and Future work

The results obtained by our depth-cue integration model are very promising. At this moment, the robot is not yet able to online improve its estimations. Thus the most straightforward future extension of the proposed work is the implementation of real-time learning. Aforehand, however, a few issues need to be addressed. The self-movement-based depth estimations strongly depend on the results of the movement itself, ie. whether it is a precise lateral movement, as well as a calculation of robot displacement. The data provided by accelerometers, especially when they are placed in a moving robotic platform are quite noisy and may sometimes lead to the large errors (the mean error was approx $0.6cm$). The calculation of the displacament could be based on visual information only, ie. optic flow or based on combined optic flow and accelerometer data.

Range sonars are commonly used in mobile robots in spite of their limitations. As presented in this work the estimates provided by sonars can be complemented by the visual depth information. It raises, however, an interesting problem of correspondance between these two different sensors. Sonars return information about distance to the closest detected object which may not always correspond to the visually determined object. Moreover, sonar sensors are subjected to several problems as reflections in the corners, their cone-shaped beam produces uncertainty in locating the target or crosstalk.

Acknowledgments This research was partly supported by Ministerio de Ciencia e Innovacin (FPU grant AP2007-02565, FPI grant BES-2009-027151, DPI2011-27846), by Generalitat Valenciana (PROMETEO/2009/052) and by Fundaci Caixa Castello-Bancaixa (P1-1B2011-54).

References

1. Karaoguz, C., Weisswange, T.H., Rodemann, T., Wrede, B., Rothkopf, C.A.: Reward-based learning of optimal cue integration in audio and visual depth estimation. In: The 15th International Conference on Advanced Robotics, Tallinn, Estonia (2011)
2. Proffitt, D.R.: Distance perception. Current Directions in Psychological Science 15(3), 131–135 (2006)
3. Proffitt, D.R.: Embodiment, Ego-Space, and Action, chap. An action-specific approach to spatial perception, pp. 177–200. Psychology Press (2008)
4. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT Press, Cambridge, MA (1998)
5. Weisswange, T.H., Rothkopf, C.A., Rodemann, T., Triesch, J.: Bayesian cue integration as a developmental outcome of reward mediated learning. PLoS ONE 6(7), 1–11 (2011)
6. Wexler, M., van Boxtel, J.J.A.: Depth perception by the active observer. TRENDS in Cognitive Sciences 9(9), 431–438 (2005)
7. Yoonessi, A., Jr., C.L.B.: Contribution of motion parallax to segmentation and depth perception. Journal of Vision 11(9): 13, 1–21 (2011)