///VSP///

Advanced Robotics 23 (2009) 305–328

ADVANCED ROBOTICS
www.brill.nl/ar

Full paper

Robot Head Pose Detection and Gaze Direction Determination Using Local Invariant Features

Javier Ruiz-Del-Solar * **and Patricio Loncomilla**

Department of Electrical Engineering, Universidad de Chile, Av. Tupper 2007,
837-0451 Santiago, Chile

Received 22 May 2008; accepted 5 October 2008

Abstract

Gaze direction determination can be a powerful anticipatory perceptual mechanism for determining the next action of other individuals, humans or robots. It can allow cooperation, synchronization or competition between robots. This is of particular importance in the case of anthropomorphic robots, which in addition of having a human-like body, should behave as humans and have similar attention mechanisms for tracking and gazing other individuals and objects. We address this problem by proposing a gaze direction determination system for robots. This system is based primarily on a robot head pose detection system that consists of two processing stages: computation of scale-invariant local descriptors of the scene and matching of these descriptors against descriptors of robot head prototypes already stored in a database. These prototypes correspond to images of robot heads taken under different view angles. After the robot head pose is detected, the robot gaze direction is determined by a composed coordinate transformation that considers the three-dimensional pose of the observing robot's camera, the detected robot head pose with respect to the observing camera, and the head model of the observed robot. Results of the successful application of the proposed system in real robots are presented.

Keywords

Gaze direction determination, head pose detection, multi-robot systems, local invariant features

1. Introduction

Multi-robot systems are becoming relevant as a result of the increasing number of industrial, service and exploration robots. Thus, cooperative robotics is an important problem in many scientific and industrial application areas such as collaborative manipulation, ground, space and underwater exploration, entertainment, surveillance, and autonomous rescue operations. However, cooperation between teams of

[*] To whom correspondence should be addressed. E-mail: jruizd@ing.uchile.cl

DOI:10.1163/156855308X397497

robots is not the only way of robot interaction. It will be more and more frequent that several robots, solving different tasks in a common environment, will need to interact and to communicate with each other. Depending on the specific situation, robots will develop different kinds of relationships just as humans do. In many cases robots will cooperate, but in other cases they will just observe each other, ignore each other or even compete. We can think, for example, of the following futuristic scenario. Several service robots meet each other in a supermarket. Some robots, will be cleaning the floor, other robots will be ordering objects and items, a third group will be buying, and a fourth group will be answering questions and helping humans. We can image the complex variety of interactions, that can exist among robots as well as between robots and humans.

In such complex scenarios, interaction based only on data communication will not be sufficient. Other interaction modalities, particularly visual communication, will be highly relevant. Consequently, gaze direction determination, i.e., the determination of the place where the other is looking at, can be a powerful anticipatory perceptual mechanism for determining the next action of other individuals. It can allow cooperation, synchronization or competition between robots. We postulate that as in the case of the human–human interactions, gaze direction determination can be relevant in many robot–robot interaction situations. In addition, anthropomorphic robots are expected to have human-like behaviors to be accepted as friendly and comfortable by humans. Looking at the object/human that is capturing the attention is a very important human-like behavior that must be present in anthropomorphic robots. For instance, many researchers have mentioned the importance that when interacting with humans, the robot tracks or gazes the face of the speaker [1–4]. Thus, independently of the multiple sensors that these robots could use, they will be designed to behave like humans, and they will incorporate foveation and gaze mechanisms.

As an example, we will analyze the robot soccer scenario (our system was originally proposed for the robot soccer context [5]), where sophisticated robot players should incorporate complex human abilities. Among many other capabilities, good soccer players should have the ability of anticipating the actions of opponents, and sometimes of teammates, by just observing the other players attitude and pose. As in other similar situations, the most employed human mechanism for solving this task is gaze direction determination. For instance, by using this mechanism an attacker player can determine if an opponent is observing them and then plan their next actions for avoiding the opponent approaching them or obstructing their trajectory. In another typical situation, a soccer player can know where the ball is by looking at the same position where an opponent is looking at (in case the opponent knows the ball position). In a third situation, a soccer player can send the ball, i.e., perform a pass, to a position where a teammate is looking at. Furthermore, when kicking the ball, first-class soccer players can mislead opponents by looking at a different place than the place where they are sending the ball. Related to the described situations, it can be affirmed that gaze direction determination of opponents and teammates is a very important ability of human soccer players that robot players should incorporate.

To the best of our knowledge, robot gaze direction determination is still an underdeveloped ability. We address this problem by proposing a gaze direction determination system for robots. This system is based primarily on a robot head pose detection system that consists of two processing stages: computation of scale-invariant local descriptors of the observed scene and matching of these descriptors against descriptors of robot head prototypes already stored in a model database. The prototypes correspond to images of robot heads taken under different view angles. After the robot head pose is detected, the robot gaze direction is determined by a composed coordinate transformation that considers the three-dimensional (3-D) pose of the observing robot's camera, the detected robot head pose with respect to the observing camera and the head model of the observed robot. This assumes that the robot camera (eye) is fixed to the robot head, as usually happens. If the camera is not fixed, then camera reference images need also to be computed. Local descriptors computation and matching is implemented using the scale-invariant feature transform (SIFT)-based Loncomilla and Ruiz-del-Solar (L&R) object recognition system, which has been designed to achieve robust operation in dynamic environments [6, 7].

The proposed robot head pose detection system is generic and can be employed for any kind of robot. The only requirement is to have reference images of the observed robot head. The robot gaze direction determination system is also generic. However, the determination of the 3-D pose of the observing robot camera depends on the robot geometry. Our system was designed originally to be used in the RoboCup Standard Platform (SP) league; therefore, observed and observing robots are Sony AIBO ERS7 robots.

This article is organized as follows. In Section 2, some work related to the topics of gaze direction determination and object recognition using local invariant features is presented. The L&R object recognition system is presented in Section 3. In Section 4, the proposed robot head pose detection system and gaze direction determination system are described. Results of the successful application of both systems in real robot scenes are presented in Section 5. Finally, some conclusions and projections of this work are given in Section 6.

2. Related Work

Human gaze direction (i.e., line of gaze) determination has been the subject of a large number of studies (e.g., Refs [8–12]), with applications in different fields such as medical research for oculography determination, car drivers, behavior characterization, and human–robot and human–computer interaction, including computer interfaces for handicapped people. There are two components of the human visual line of sight that need to be known to solve this problem: the pose of the human head and the orientation of the eyes within their sockets. There is a large variability of the methods and models that have been proposed for solving this problem. However, to the best of our knowledge there are no studies on determining the gaze direction in robots. Already developed methodologies employed for human gaze direction determination are not applicable for robots. They are based on anthropometric models of the human head and eyes (see, e.g., Ref. [12]), or they employ face or iris detection algorithms, or even special lighting (infrared lights). Therefore, new methodologies need to be developed for the robot case. Some alternatives could be the construction of explicit 3-D robot head models, the development of specific robot face detection algorithms or the use of local invariant features for performing the detection of the robot heads. In this last case the idea is to have different views of a robot head (references images), and to perform a matching between features computed in the input image and features computed in the different reference images for detecting the robot head pose. Taking into account the impressive development of object recognition algorithms based on local invariant features in recent years [13–17], and the fact that, for a given robot model, head and face variability is much smaller than in humans, we believe that matching against reference images using local features is, at present, the best methodology for solving the robot head pose detection and robot gaze direction determination problems.

Object recognition based on local invariant features works under the following principle: (i) invariant local interest points or keypoints are extracted independently from both a test image and a reference image (model), and characterized using invariant descriptors, and (ii) the invariant descriptors (features) are matched against each other. The most employed local interest point detectors are the single-scale Harris detector [18] and the multi-scale Lowe's sDoG+Hessian detector [14]. The best performing interest point detectors are the Harris-Affine and the Hessian-Affine [19], but they are too slow for real-time applications. On the other hand, the most popular and best performing descriptor [20] is the SIFT [14].

To select the local detector and invariant descriptor to be used in a given application one should take into account the algorithm's accuracy, robustness and processing speed. Lowe's system [14, 21] using the SDoG+Hessian detector, SIFT descriptors and a probabilistic hypothesis rejection stage is a popular choice, given its recognition capabilities and near real-time operation. However, the main drawback of Lowe's system is the large number of false-positive detections when the objects to be detected are not present in the image. This is a serious problem when using it in real-world applications where video sequences are analyzed, e.g., robot self-localization [22].

One of the main weaknesses of Lowe's algorithm is the use of just a simple probabilistic hypothesis rejection stage, which cannot successfully reduce the number of false positives. Loncomilla and Ruiz-del-Solar have proposed a system (L&R) that largely reduces the number of false positives by using several hypothesis rejection stages [5–7]. This includes a fast probabilistic hypothesis rejection stage, a linear correlation verification stage, a geometrical distortion verification stage, a pixel correlation verification stage, and the use of the RANSAC algorithm and

Table 1.
Comparative evaluation of the different algorithms (see main text for description)

Algorithm	TPR (%)	FPR (%)	FPR/FPR baseline (%)
Baseline	64.0	81.9	100.0
LinearCorr	66.0	61.8	75.5
GeoDistortion	73.0	26.2	32.0
PixelCorr	66.0	5.6	6.9
FastProb	66.0	4.1	5.0
RANSAC	65.0	3.2	3.8

TPR: true-positive rate; FPR: false-positive rate.

a semi-local constraints test. In Ref. [6], we compared Lowe's and L&R systems using 100 reference–test pairs of real-world highly-textured images with variations in position, view angle, image covering, partial occlusions, and in-plane and out-of the-plane rotations. In these experiments, the 100 test images are matched against each of the 100 reference images, although only 100 correct matches are possible. The results show that in this dataset, the L&R system reduces the false-positive rate from around 80 to around 3%, while increasing slightly the detection rate and the processing speed, when compared with Lowe's system (see more detailed results in Table 1). For this reason we choose to use this system in this work.

3. L&R Object Recognition System

This system considers four main stages: (i) generation of local interest points, (ii) computation of the SIFT descriptors, (iii) SIFT matching using nearest descriptors, and (iv) transformation computation and hypothesis rejection tests. The first three stages are the standard ones proposed by Lowe [14], while the fourth stage is employed for reducing the number of false matches, giving robustness to the whole system. This last stage is implemented by the following procedure:

(1) *Similarity transformation determination*. Similarity transformations are determined using the Hough transform (see description in Ref. [21]). Bins sizes are 30° for the orientation axis, a factor of 2 for the scale axis, and 0.25 times the width and height of the projected training image for each position axis. After the Hough transform is computed, a set of bins, each one corresponding to a similarity transformation, is determined. Then:
   (a) Invalid bins (those that have less than four votes) are eliminated.
   (b) Q is defined as the set of all valid candidate bins, i.e., the ones not eliminated in (1a).
   (c) R is defined as the set of all accepted bins. This set is initialized as a void set.

(2) *Transformation verification*. For each bin B in Q the following tests are applied (the procedure is optimized for obtaining high processing speed by applying first tests consuming less time):

  (a) Bins filtering. If the bin B has a direct neighbor in the Hough space with more votes, then delete bin B from Q and go to (2).

  (b) Linear correlation test. Calculate r_{REF} and r_{TEST}, which are the linear correlation coefficients of the interest points corresponding to the matches in B, that belong to the reference and test image, respectively. If the absolute value of any of these two coefficients is high, it means that the corresponding points lie, or nearly lie, in a straight line and that the affine transform to be obtained can be numerically unstable. If this condition is fulfilled delete bin B from Q and go to (2). This test is described in the Appendix.

  (c) Fast probability computation. Calculate the probability associated to bin B. If this probability is lower than a threshold P_{TH1}, delete bin B from Q and go to (2). The main advantage of this test is that it can be computed before calculating the affine transformation, which speeds up the whole procedure. This test is described in the Appendix.

  (d) Affine transformation determination. Calculate an initial affine transformation T_B using the matches in B.

  (e) Geometrical distortion test. Compute the affine distortion degree of T_B using a geometrical distortion verification test (see Appendix). A certain affine transformation should not deform very much an object when mapping it. Therefore, if T_B has a strong affine distortion, delete bin B from Q and go to (2).

  (f) Top-down matching. Matches from all the bins in Q that are compatible with the affine transformation T_B computed in (2d) are summarized and added to bin B. Duplication of matches inside B is avoided.

  (g) Lowe's probability computation. Compute Lowe's probability P_{LOWE} of bin B (see description in Ref. [21]). If P_{LOWE} is lower than a threshold P_{TH2}, delete bin B from Q and go to (2).

  (h) RANSAC test. To find a more precise transformation apply RANSAC inside bin B. In case of RANSAC success, a new transformation T_B is calculated and B is labeled as a RANSAC-approved bin.

  (i) Bin acceptance. Accept the candidates B and T_B, i.e., delete B from Q and include it in R (the T_B transformation is accepted).

(3) *Transformation fusion*. For all pairs (B_i, B_j) in R, check if they may be fused into a new bin B_k. If the bins may be fused and one of them is RANSAC-

approved, do not fuse them and delete the other in order to preserve accuracy. If the two bins are RANSAC-approved, delete the least probable (using P_{LOWE} value). Repeat this until all possible pairs (including the new created bins) have been checked. The fusion procedure is described in the Appendix.

(4) *Semi-local constraints test*. For any bin B in R, apply the semi-local constraints procedure to all matches in B. The matches from B that are incompatible with the constraints are deleted. If some matches are deleted from B, T_B is recalculated. This procedure is described in Ref. [23].

(5) *Pixel-correlation test*. For any bin B in R, calculate the pixel correlation r_{pixel} using T_B. Pixel correlation is a measure of how similar the image regions being mapped by T_B are. If r_{pixel} is below a given threshold t_{corr}, delete B from R. This test is described in the Appendix.

(6) *Priority assignation*. Assign a priority to all bins (transformations) in R. The initial priority value of a given bin will correspond to its associated P_{LOWE} probability value. In case that the bin is a RANSAC-approved one, the priority is increased by one. Thus, RANSAC-approved bins have a larger priority than non-RANSAC-approved ones.

To quantify the contribution of the different steps of the proposed verification procedure, we use the same testing procedure and 100 pairs of real-world images described in Ref. [6]. The different algorithm flavors to be compared are:

- *Baseline*. Lowe's algorithm with the steps 1, (2a), (2d), (2f), (2g) and (2i) from the transformation computation and hypothesis rejection tests stage.
- *LinearCorr*. Baseline algorithm plus step (2b).
- *GeoDistortion*. Baseline algorithm plus steps (2b) and (2e).
- *PixelCorr*. Baseline algorithm plus steps (2b), (2e) and (5).
- *FastProb*. Baseline algorithm plus steps (2b), (2c), (2e) and (5).
- *Ransac*. Baseline algorithm plus steps (2b), (2c), (2e), (2h), (5) and (6).

As can be observed in Table 1, the proposed verification tests are able to reduce largely the number of false positives. The combined use of different test allows obtaining different improvements in the true-positive rate and the false-positive rate.

4. Head Pose Detection and Gaze Direction Determination

4.1. Head Pose Detection Using Local Invariant Features

Basically, the robot head pose is determined by matching the input image descriptors with descriptors corresponding to robot head prototype images already stored in a database. The employed prototypes correspond to different views of a robot

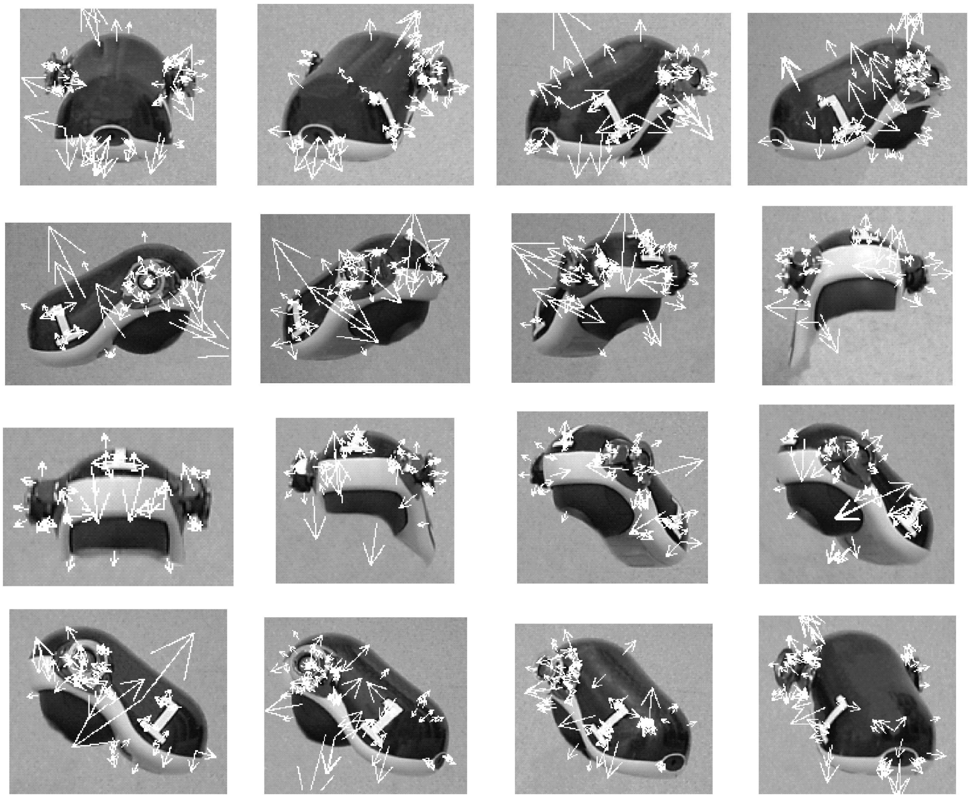


Figure 1. AIBO ERS7 robot head prototypes with their SIFTs. Pictures taken every 22.5° (yaw angle).

head. The system was originally developed in the context of the RoboCup soccer SP league. In this specific case we are interested in detecting the robot head pose of AIBO ERS7 robots, but also in recognizing the robot identity (robot number). For this reason prototypes for each of the four robot players are stored in a model database. In Fig. 1, the 16 prototype heads corresponding to one of the robots (number 1) are displayed. The pictures were taken every 22.5° (yaw angle).

As already explained in Section 3, the matching process with the database reference images is composed by several stages. After applying these stages, if a robot head was found, just one affine transformation remains — the one with the highest associated probability. The robot head pose is determined using this transformation together with the identity of the matched reference image, which has an associated view angle. The robot identity is determined by the identity of the matched reference image. This process is shown in Fig. 2.

4.2. Gaze Direction Determination

The line of gaze of the observed robot, in global coordinates, can be computed using the following information: (i) pose of the observing robot's camera in global coordinates, (ii) prototype view angle, and (iii) distance and rotation angles of the

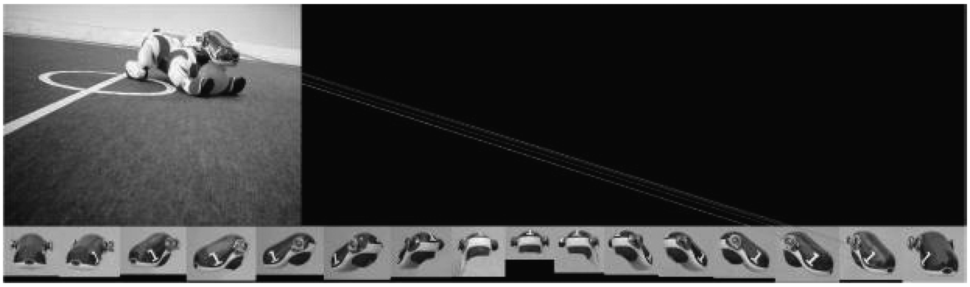


Figure 2. The robot head pose is determined using the obtained transformation together with the identity of the matched reference image. In this case the identity of the observed robot is ‘1’.

observed robot. The observing robot camera pose can either be known *a priori*, in the case that the camera is fixed in global coordinates, or estimated by the observing robot itself using self-localization and joints information. The prototype view angles are fixed and known *a priori*; they are defined when the model database is built. The distance and rotation angles of the observed robot head can be determined using information of the head detection process and *a priori* knowledge of the observing camera characteristics (resolution and angle of view).

To perform the computations we define the following reference systems: the global reference system (RF0) $\{\hat{i}_0\ \hat{j}_0\ \hat{k}_0\}$, a reference system fixed to the observing robot's camera (RF1) $\{\hat{i}_1\ \hat{j}_1\ \hat{k}_1\}$, a reference system fixed at the observed robot's head, which does not consider prototype rotations (RF2) $\{\hat{i}_2\ \hat{j}_2\ \hat{k}_2\}$, and a reference system fixed at the observed robot's head, which considers prototype rotations (RF3) $\{\hat{i}_3\ \hat{j}_3\ \hat{k}_3\}$. For the transformations between the different reference systems we will use standard homogeneous 3-D matrices: rotation matrices $\mathbf{R}_z(\theta_z)$, $\mathbf{R}_x(\theta_x)$ and $\mathbf{R}_y(\theta_y)$, and a translation matrix $\mathbf{T}_{xyz}(x_t, y_t, z_t)$.

The composed transformation from RF3 to RF0 is given by:

$$\mathbf{M}^* = \mathbf{M}_{10}\mathbf{M}_{21}\mathbf{M}_{32}, \tag{1}$$

with $\mathbf{M}_{ab}$ composed of 3-D matrices between reference systems a and b, given by (see angles and distance definitions in Table 2 and Fig. 3):

$$\begin{aligned} \mathbf{M}_{10} &= \mathbf{T}_{xyz}(\mathbf{C}_x, \mathbf{C}_y, \mathbf{C}_z)\mathbf{R}_z(\alpha)\mathbf{R}_y(\beta)\mathbf{R}_x(\gamma) \\ \mathbf{M}_{21} &= \mathbf{T}_{xyz}(\mathbf{P}_x, \mathbf{P}_y, \mathbf{P}_z)\mathbf{R}_z(\mu)\mathbf{R}_y(\nu)\mathbf{R}_x(-\phi) \\ \mathbf{M}_{32} &= \mathbf{R}_z(\varepsilon + \pi)\mathbf{R}_y(\delta)\mathbf{R}_x(\theta). \end{aligned} \tag{2}$$

In RF3, two points will define the line of gaze: the camera's position of the observed robot (at the origin of RF3) and the intersection of the gaze straight line with the floor at a position $(\lambda \quad 0 \quad 0)^{\mathrm{T}}$. This position can be translated to global coordinates using $\mathbf{M}^*$. The intersection with the floor will correspond, in global coordinates (RF0), to:

$$z_0(\lambda) = 0. \tag{3}$$

Table 2.
Angles and distances definitions

	Definition	Source
α	yaw (or pan) rotation angle of observing robot's camera, in RF0	prior knowledge or self-localization
β	pitch (or tilt) rotation angle of observing robot's camera, in RF0	prior knowledge or self-localization
γ	roll rotation angle of observing robot's camera, in RF0	prior knowledge or self-localization
C	3-D position of the observing robot's camera, in RF0	prior knowledge or self-localization
δ	prototype-head pitch (tilt) angle of the reference image	prior knowledge (prototype angle)
ε	prototype-head yaw (pan) angle of the reference image	prior knowledge (prototype angle)
θ	prototype-head roll angle of the reference image	prior knowledge (prototype angle)
ρ	prototype-head distance of the reference image	prior knowledge (prototype distance)
μ	yaw (pan) angle of the observed head, in RF1	robot-head pose determination system and camera geometry
ν	pitch (tilt) angle of the observed head, in RF1	robot-head pose determination system and camera geometry
ϕ	in plane rotation (roll) of the observed head, in RF1	robot head pose determination system
P	3-D position of the observed robot head, in RF1	robot head pose determination system and camera geometry

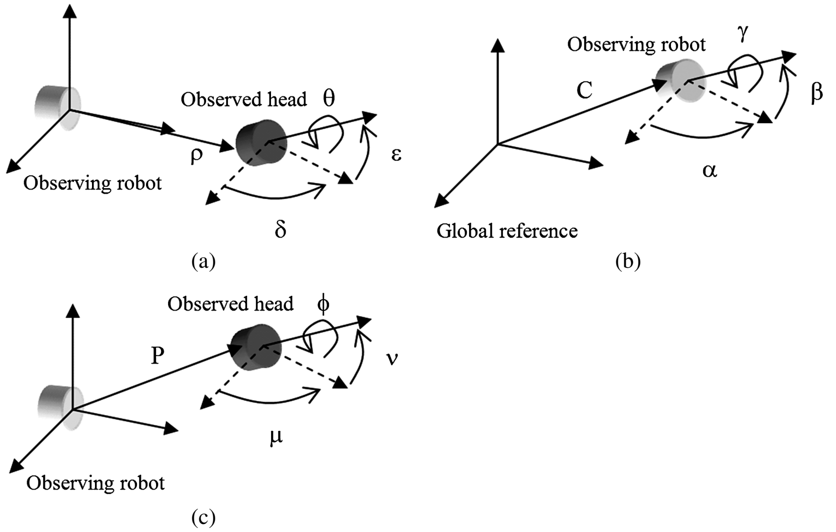


Figure 3. Angles and distances from Table 2, measured (a) when acquiring a prototype image, and (b) and (c) when analyzing a test image.

Then, λ as well as $x_0(\lambda)$ and $y_0(\lambda)$, the gaze coordinates at the floor, are calculated using (1)–(3). There remains the computation of the parameters of the observed robot's head pose (μ, ν, ϕ and **P**). A robot head is characterized by an affine transformation that maps the prototype image onto a certain portion of the image under analysis. First, ϕ is computed as the mean of the SIFT angles differences in all the keypoints matches used to compute the affine transformation. Then, the real distance between the observing camera and the observed robot's head is calculated as follows. The prototype head's image has four vertices A, B, C and D. The affine transformation maps this image onto a parallelogram with vertices A′, B′, C′ and D′. As the visual area decreases in a quadratic way with the distance, if the camera has no distortion, the $\mathbf{P}_x$ coordinate of the observed robot's head can be calculated as:

$$\mathbf{P}_x = \rho\sqrt{\frac{\text{prototype image area}}{\text{mapped area}}} = \rho\sqrt{\frac{\mathrm{d}(AB) \times \mathrm{d}(BC)}{\det(\overrightarrow{A'B'}\ \ \overrightarrow{B'C'})}}, \tag{4}$$

where ρ is the distance between the camera and the prototype head image at the acquisition time.

Finally, assuming the use of an ideal camera, the remaining robot's head pose components can be obtained as:

$$\begin{aligned} \mu &= \arctan\left(2\tan\left(\frac{W_u}{2}\right)\frac{M_u/2 - u}{M_u}\right) \\ \nu &= -\arctan\left(2\tan\left(\frac{W_v}{2}\right)\frac{M_v/2 - v}{M_v}\right) \\ \mathbf{P}_y &= \mathbf{P}_x \tan(\mu), \qquad \mathbf{P}_z = \mathbf{P}_x \tan(\nu), \end{aligned} \tag{5}$$

where W_u and W_v are the horizontal and vertical field of view of the camera, M_u and M_v are the horizontal and vertical resolution of the camera, and (u, v) is the head image's center coordinates in the image under analysis.

From all considered 3-D homogeneous matrices, $\mathbf{M}_{10}$ is the only one that depends on the geometry of the observing robot. This matrix is computed for the case of the AIBO ERS7 robot in Ref. [5].

5. Experimental Methodology and Results

5.1. Robot Head Detection Experiments

The robot head detection system was implemented in the AIBO ERS7. This robot has a 64-bit RISC processor (MIPS R7000) from 576 MHz, 64 MB RAM and a color camera of 208 × 160 pixels that delivers 30 f.p.s. The parameters of this camera are: horizontal field of view 56.9°, vertical field of view 45.2°, focal ratio F/2.8 and focal length 3.27 mm. The sub-sampled scale space is built from the original AIBO images. Under these conditions the determination of key points and SIFT descriptors takes about 300 ms, depending on the number of objects under observation. The matching voting and transformation calculation takes about 10 ms for each analyzed prototype head.

Table 3.
Robot head detection of an AIBO ERS7 robot

Full detections (head + identifier number)	68%
Partial detections (only the identifier number)	12%
Full + partial detections	80%
Number of false detections in 39 images	6

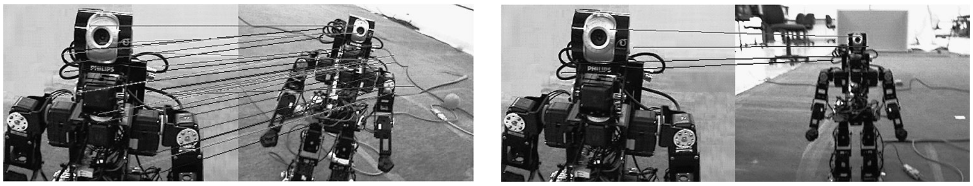

(a)
(b)

Figure 4. Two examples of humanoid head detection in a video sequence: (a) frame 36 and (b) frame 49. Each frame shows the reference image (left) — test image (right) pair.

Robot head detection experiments using real-world images were performed (see Table 3). In all of these experiments the 16 prototypes of robot player 1 were employed (see Fig. 1). These prototypes (around 100 × 100 pixels) are stored in the AIBO flash memory as BMP files. A database composed of 39 test images taken in a SP-league soccer field was built. In these images robot ‘1’ appears 25 times and the other robots appear 9 times. A detection is assumed correct when almost all the matches involved in the transformation are correct (incorrect detections rarely have any correct matches). If we consider full detections, in which both the robot head pose as well as the robot identity is detected, a detection rate of 68% is obtained. When we considered partial detections, i.e., only the robot identity is determined, a detection rate of 12% is obtained. The combined detection rate is 80%. At the same time, the number of false positives is very low (just six in 39 images). These figures are very good, because when processing video sequences, opponent and teammate robots are seen in several consecutive frames. Therefore, a detection rate of 80% in single images should be high enough for detecting the robot head in few frames.

To validate our system, we also carried out head detection experiments using our humanoid Hajime HR18 robot (see Fig. 4) and a video sequence containing 221 frame images, in which the robot was always seen. Our Hajime robot is powered with a Fujitsu Siemens n560 Pocket PC running Windows Mobile as the main processor and a Philips ToUCam III SPC900NC camera, mounted in the robot head (see description in Ref. [24]). The camera has 640 × 480 pixels, horizontal field of view 45°, vertical field of view 37°, focal ratio F/2.8 and focal distance 4.5 mm. In the experiments we used an external camera connected to a computer (notebook core-duo 1.66 GHz), where the detection system was running. The results are sum-

Table 4.
Detection of a humanoid Hajime HR18 robot, 221 frames (results were obtained with the system running in a notebook core-duo 1.66 GHz, 1 GB RAM, running Windows XP)

Flavor	Detection rate (%)	Number of false positives	Processing speed (f.p.s.)
Original image size: 320 × 240	80.1	14	4.4
Sub-sampled image: 240 × 170	75.1	7	4.7
Sub-sampled image: 160 × 120	64.3	3	11.5

marized in Table 4. As it can be observed, the obtained detection rates are similar to those obtained with the AIBO ERS robots.

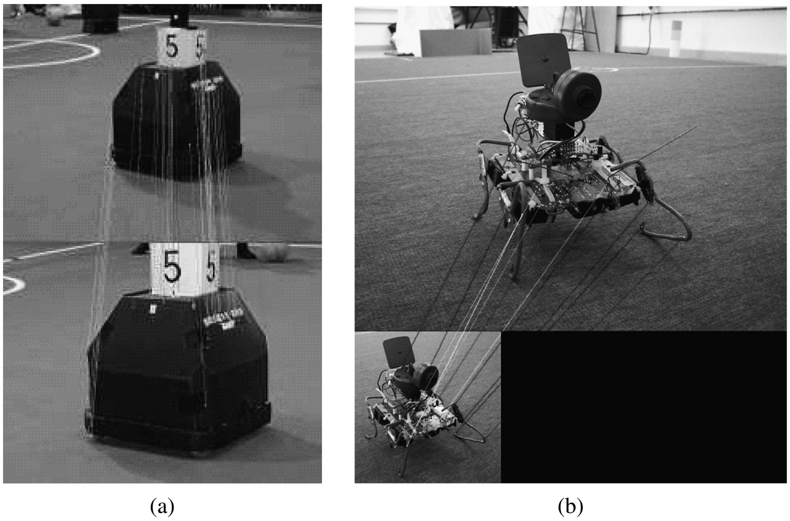
These preliminary experiments show the high potential of the proposed methodology as a way of achieving player recognition and gaze estimation. The SIFT descriptors are not based on color information; therefore they are complementary to existing vision systems employed in the RoboCup leagues. A mixed SIFT and color-based vision system could be employed in the SP league in the near future. A fast color-based vision system could be used for the general analysis of the images and for determining regions of interest where observed robots can be located. These regions of interest can be further processed by the SIFT-based vision system. This multi-method strategy could be also used in other contexts. Other possibilities for achieving a reduction in the processing time are the application of the head detector in non-consecutive frames (e.g., every 3 frames) and the use of local features that can be evaluated in less time (e.g., SURF features [17]).

One of the most interesting features of the proposed methodology is its robustness against environmental aspects such as occlusions, variable illumination and cluttered backgrounds. This is a main advantage over existing appearance-based object recognition approaches (e.g., eigenspace-based), whose performance usually depends on the mentioned aspects. For instance, in Ref. [25] it was demonstrated that in the task of recognizing faces in environments with occlusions, variable illumination and cluttered backgrounds, much higher recognition rates are obtained when using the proposed methodology than when applying an eigenspace-based method.

As mentioned above, the proposed robot head pose detection system is general purpose and, therefore, can be used for detecting the pose of other robot models. Figure 5 shows the body pose detection of two simple robots.

5.2. Gaze Direction Determination Experiments With AIBO ERS7 Robots

Exemplary experiments of the determination of the gaze direction were carried out using AIBO ERS7 robots. In these experiments both the observing and the observed robots are ERS7 robots. The experimental setup is shown in Fig. 6. In all cases prototype images in which only the yaw angle is exactly determined were employed (similar images to the ones shown in Fig. 1). Pitch and roll angle are about 0°. The



(a) (b)

Figure 5. Robot pose detection examples. (a) RoboCup middle-size robot. (b) Four-legged Walker robot.

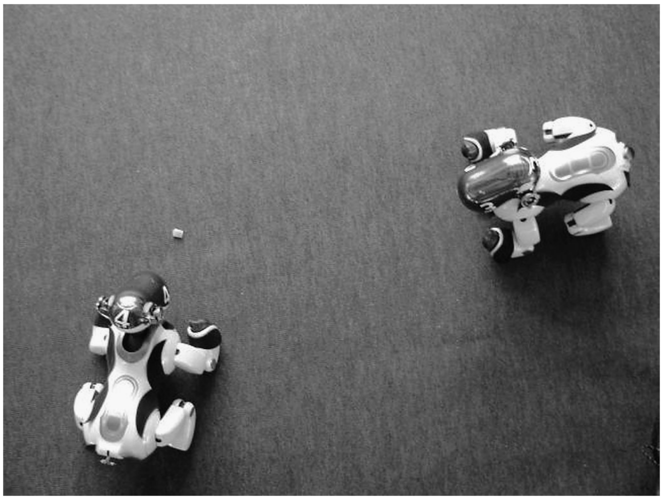


Figure 6. Typical setup for the gaze direction determination experiments. The observing robot (number 3) is placed at the right side, while the observed robot (number 4) is placed at the left side. The observed robot is looking at the small white square on the field.

observing camera position in the global reference system is $(0, 0, 10)$ (in centimeters), i.e., the observing robot's front feet are at the origin of the coordinate system. The obtained results are shown in Table 5. The Euclidian distance r_e between the real and estimated gaze coordinates on the floor is used as a measure of the system's error.

Table 5.
Gaze direction determination experiments (all (x, y, z) coordinates are measured in the global reference system (RF0); the observing camera is placed at $(0, 0, 10)$ and looking at the x-axis)

Experiment	3-D position of the observed camera						Gaze point at the floor				Error r_e (cm)
	Real			Estimated			Real		Estimated		
	x (cm)	y (cm)	z (cm)	$\hat{x}$ (cm)	$\hat{y}$ (cm)	$\hat{z}$ (cm)	x (cm)	y (cm)	$\hat{x}$ (cm)	$\hat{y}$ (cm)	
1	60	4	9	61.0	4.1	10.0	58	−4	57.4	0.4	4.4
2	15	−39	9	21.0	−32.4	14.0	20.5	−30	16.1	−8.4	22.0
3	24	22	10	22.5	23.5	13.3	58	−4	29.8	13.3	33.1
4	48	−19	10	42.5	−17.0	3.0	47	7.5	39.2	−13.0	21.9
5	36	−16	9	35.6	−15.1	8.8	37	−4	31.4	−3.9	5.6
6	39	6.5	10	38.6	7.5	27.3	48	2	46.6	−10.3	12.4
7	25	28	10	26.7	28.7	14.4	23	37	15.3	34.1	8.2
8	58	3	10	56.8	4.7	15.2	49	−6.5	39.4	−3.1	10.2
Mean error:											14.7
Standard deviation:											9.4

In the performed experiments the mean error is 14.7 cm and the standard deviation 9.04. We believe that these results are quite good, considering that the primary application of this gaze direction determination system is the RoboCup SP league, where the field dimension is 600 × 400 cm and where estimation errors of 30–40 cm are still acceptable. Thus, using the proposed system a robot soccer player can determine, with sufficient accuracy, the place at which opponents are looking at and, for example, know the position of the ball (in the case that the opponent is looking at the ball, which is very frequent). Errors in accuracy are produced mainly due to: (i) the limited accuracy of the accelerometer sensor of the AIBO ERS7 robot (β_R), (ii) the error in the pitch angle of the reference prototypes (δ), which was not exactly measured in the experiments, (iii) the employed model of the AIBO ERS7 head, which does not consider the round form of the nose, (v) the use of a limited number of prototype viewpoints, and (v) the employed method for computing the distance, which is based on the area size. However, we believe that these results show the high potential of the described approach. Knowing the place where another robot is looking at, with a limited error, is already significant and much better than not having this information at all.

5.3. Gaze Direction Determination Experiments With Humanoid Robots

To obtain a better characterization of the gaze direction determination system, additional experiments were carried using a humanoid Hajime HR18 robot. To isolate errors produced by the determination of the observing camera pose, the observing camera was placed out of the robot, at a fixed height zCam (see Table 6). The observed HR18 robot was placed at different relative positions with respect to the

Table 6.
Experiments using humanoid robots (all (x, y, z) coordinates are measured in a reference system centered in the observing camera position $(0, 0)$)

Experiment	Camera height zCam (cm)	3-D position of the observed robot's camera						Gaze point at the floor				Error	
		Real			Estimated			Real		Estimated		r_e (cm)	$\alpha°$
		x (cm)	y (cm)	z (cm)	$\hat{x}$ (cm)	$\hat{y}$ (cm)	$\hat{z}$ (cm)	x (cm)	y (cm)	$\hat{x}$ (cm)	$\hat{y}$ (cm)		
1	79	100	0	50	126.5	0.1	42.3	14	0	18.1	2.2	4.7	3.3
2	79	100	0	50	132.2	−0.1	40.7	20.6	−32.9	29.4	−51.5	20.6	10.8
3	79	100	0	50	114.8	0.1	45.7	39.2	−60.8	6.0	−110.5	59.7	25.0
4	79	100	0	50	105.6	0.1	48.4	67.1	−79.6	52.2	−92.0	19.4	8.5
5	79	100	0	50	104.8	−0	48.6	100	−86.0	95.6	−88.1	4.9	1.7
6	79	100	0	50	83.6	−0.1	54.8	132.9	−79.5	114.6	−96.5	25.1	8.2
7	79	100	0	50	75.8	−0	57.0	160.8	−60.8	139.1	−75.1	25.9	7.1
8	79	100	0	50	72.2	0	58.1	179.5	−32.9	125.2	−24.4	54.9	8.4
9	79	100	0	50	92.6	0.1	52.2	186	0	149.0	0.2	37.0	4.9
10	79	100	0	50	122.9	0.1	43.4	179.5	32.9	164.3	18.8	20.7	4.1
11	79	100	0	50	96.3	0	51.1	160.8	60.8	139.0	54.0	22.9	3.3
12	79	100	0	50	98.0	0.2	50.7	132.9	79.5	133.9	88.1	8.7	2.3
13	79	100	0	50	115.5	0.3	45.7	100	86	120.8	113.9	34.8	5.9
14	79	100	0	50	117.6	0.3	45.0	67.1	79.5	79	97.2	21.4	5.1
15	79	100	0	50	106.1	0.2	48.3	39.2	60.8	19.2	81.8	29.0	14.4

Table 6.
(Continued)

Experiment	Camera height zCam (cm)	3-D position of the observed robot's camera						Gaze point at the floor				Error	
		Real			Estimated			Real		Estimated		r_e (cm)	α°
		x (cm)	y (cm)	z (cm)	$\hat{x}$ (cm)	$\hat{y}$ (cm)	$\hat{z}$ (cm)	x (cm)	y (cm)	$\hat{x}$ (cm)	$\hat{y}$ (cm)		
16	79	100	0	50	135.1	0.2	39.9	20.6	32.9	29.6	44.8	14.9	8.1
17	79	132.9	−79.50	50	176.3	−105.4	40.6	20.6	32.9	112.9	−16.9	104.9	49.1
18	79	186	0	50	203.0	−0	47.2	100	−86	126.8	−92.4	27.6	5.8
19	60	−41	−86.00	50	−50.1	−105.0	47.8	−127	0	−131	−37.8	38.0	14.7
20	60	45	0	50	46.4	−0.1	49.7	−101.8	−60.8	−67.7	−76.0	37.4	15.8
21	60	45	0	50	47.7	−0.2	49.7	−41	−86	−17.6	−118.4	40.0	15.8
22	60	45	0	50	43.9	−0.3	50.6	19.8	−60.8	54.8	−60.2	35.0	19.8
23	60	45	0	50	47.8	0.1	49.4	19.8	60.8	19.3	71.0	10.2	4.5
24	60	45	0	50	49.4	0.1	49.0	−41	86	−33.6	101.5	17.1	6.8
25	60	45	0	50	46.8	0	49.6	−101.8	60.8	−88.5	83.7	26.5	11.2
26	60	−41	86.00	50	−42.6	89.4	49.7	−127	0	−181.6	48.8	73.2	15.9
27	60	−41	86.00	50	−41.7	87.4	49.9	19.8	−60.8	−61.9	−17.8	92.3	63.4
28	60	−41	86.00	50	−44.1	92.5	49.2	45	0	18.1	40.9	49.0	38.1
Mean error:												34.1	13.7
Standard deviation:												24.2	14.5

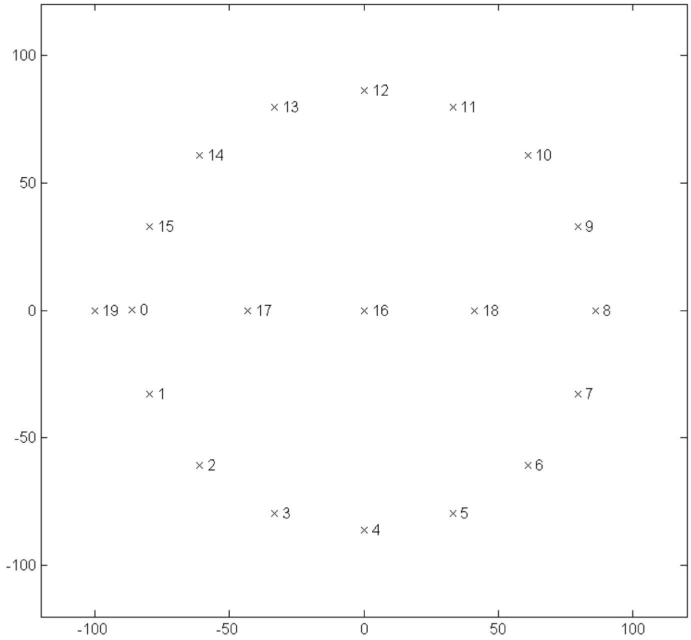


Figure 7. Diagram of the setup for the experiment, showing the positions marked on the floor. Position 16 is at the origin of the coordinates system. Positions 0–15 are places in a circle of radius 86. Positions 17, 18 and 19 have coordinates $(-43, 0)$, $(41, 0)$ and $(-100, 0)$, respectively. All distances are measured in centimeters.

observing camera. The observing camera was always facing the HR18 robot. Figure 7 shows a total of 19 points, which were marked on the floor for the purpose of placing the observing camera, the observed robot and the gaze point. The use of these 19 points simplifies the measurement of the ground-truth positions and improves the accuracy.

In the prototype image acquisition process, the observing camera is placed in position 17 and the observed robot is placed in position 16. The 16 prototypes correspond to the images captured by the observing camera, while the robot observes positions 0–15. In the testing stage, the camera is placed in position 19 or 16, while the observed robot is placed in different positions, looking in each case at a given position contained in the circle (e.g., observed robot place in position 9 looking at position 14). Altogether, 28 experiments were carried out (see Table 6). Figure 8 displays the real and estimated gaze positions for each experiment. In these experiments the mean Euclidian distance (r_e) between the real gaze point and the estimated gaze point is 34.1 cm, while the mean angular error (α) from the robot viewpoint (i.e., the angle defined by the real gaze point, the robot and the estimated gaze point) is just 13.7°. This last result tells us that the proposed system produces a very good estimation of the gaze direction. Errors in accuracy are produced by the same factors mentioned in Section 5.2, except for the case of the accelerometers, which in this case are not used for determining the pose of the observing camera.

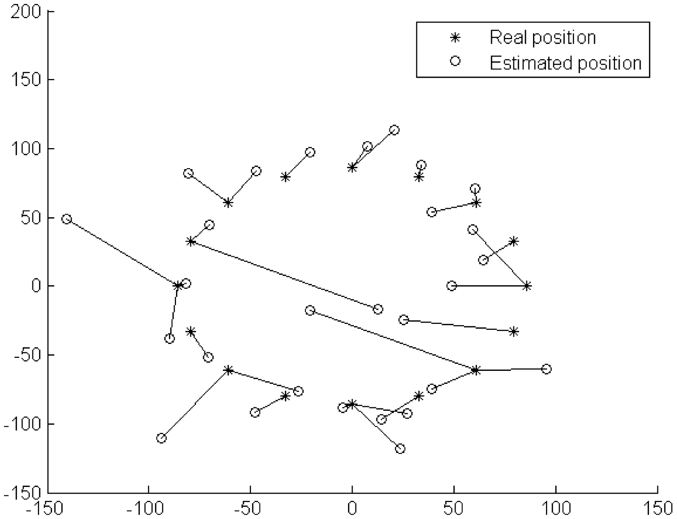


Figure 8. Real and estimated gaze position in the humanoid test. Real gaze positions are on positions 0–15 of the setup circle. The position of the observing camera is not displayed for visual clarity.

6. Conclusions

Gaze direction determination can be a powerful anticipatory perceptual mechanism for determining the next action of other individuals. It can allow cooperation, synchronization or competition between robots. In this context, we proposed a gaze direction determination system for robots. The system is based primarily on a robot head pose detection system that uses local invariant features. After the robot head pose is detected, the robot gaze direction is determined by a composed coordinate transformation that considers the 3-D pose of the observing robot's camera, the detected robot head pose affine transformation with respect to the observing camera and the head model of the observed robot. For implementing the robot head pose detection system we used the L&R object recognition system, which has shown robust detection results in dynamic environments.

The robot head detection system was implemented in AIBO ERS7 robots and successful detection results were obtained. A limited processing speed of about 3 f.p.s. was obtained in these robots (576 MHz CPU). This processing speed can be improved by using faster processors, applying the head detector in non-consecutive frames, using local features that can be evaluated in less time or by processing not the whole images, but some regions of interest. The head detection experiments were validated using a HR18 humanoid robot.

Gaze direction determination experiments were also presented. In these experiments, estimations of the gaze coordinates on the floor with a limited accuracy of about 14.7 cm were obtained for the case of AIBO ERS7 robots and 34.1 cm for the case of HR18 humanoid robots. In this last case, the angular error of the gaze direction estimation was just 13.7°. The main sources of error are the limited number of prototype viewpoints and the method for computing the distance. In addition, the limited accuracy of the accelerometers can increase errors when they are used for determining the pose of the observing camera. However, we believe that these experiments show the great potential of this approach for the implementation of richer interaction modalities between robots. Knowing the place where another robot is looking at with a limited error is already significant, because it enables anthropomorphic robots to identify common centers of attention in space, which can be useful for collaborative/competitive tasks. Although the proposed robot head pose detection and gaze direction determination systems can be improved in terms of detection rate, gaze direction determination accuracy and processing speed, we believe that this is a first step in the direction of building more powerful robot–robot interfaces, which will be increasingly required in multi-robot systems.

Acknowledgement

This research was partially supported by FONDECYT (Chile) under project number 1061158.

References

1. H. Ishiguro, T. Ono, M. Imai and T. Kanda, Development of an interactive humanoid robot Robovie — an interdisciplinary approach, in: *Robotics Research*, R. A. Jarvis and A. Zelinsky (Eds), pp. 179–191. Springer, New York, NY (2003).
2. T. Kanda, H. Ishiguro, T. Ono, M. Imai and R. Nakatsu, Development and evaluation of an interactive humanoid robot Robovie, in: *Proc. IEEE Int. Conf. Robotics and Automation*, Washington, DC, USA, pp. 1848–1855 (2002).
3. Y. Matsusaka, S. Kubota, T. Tojo, K. Furukawa and T. Kobayashi, Multi-person conversation robot using multimodal interface, in: *Proc. World Multiconf. on Systems, Cybernetics and Informatics*, Orlando, FL, vol. 7, pp. 450–455 (1999).
4. K. Nakadai, K. Hidai, H. Mizoguchi, H. G. Okuno and H. Kitano, Real-time auditory and visual multiple-object tracking for robots, in: *Proc. 17th Int. Joint Conf. Artificial Intelligence*, Seatte, WA, pp. 1425–1432 (2001).
5. P. Loncomilla and J. Ruiz-del-Solar, Gaze direction determination of opponents and teammates in robot soccer, *Lecture Notes Comp. Sci.* **4020**, 230–242 (2006).
6. P. Loncomilla and J. Ruiz-del-Solar, A fast probabilistic model for hypothesis rejection in SIFT-based object recognition, *Lecture Notes Comp. Sci.* **4225**, 696–705 (2006).
7. P. Loncomilla and J. Ruiz-del-Solar, Robust object recognition using wide baseline matching for RoboCup applications, *Lecture Notes Comp. Sci.* **5001**, 441–448 (2008).
8. V. Bakic and G. Stockman, Real-time tracking of face features and gaze direction determination, in: *Proc. 4th IEEE Workshop on Applications of Computer Vision*, Princeton, NJ, pp. 256–257 (1998).
9. A. Perez, M. L. Cordoba, A. Garcia, R. Mendez, M. L. Munoz, J. L. Pedraza and F. Sanchez, A precise eye-gaze detection and tracking system, in: *Proc. 11th Int. Conf. in Central Europe on Computer Graphics, Visualization and Computer Vision*, Plzen-Bory (2003).
10. Q. Ji and X. Yang, Real-time eye, gaze, and face pose tracking for monitoring driver vigilance, *Real-Time Imaging* **8**, 357–377 (2002).

11. T. Ohno, N. Mukawa and A. Yoshikawa, FreeGaze: a gaze tracking system for everyday gaze interaction, in: *Proc. Symp. on Eye Tracking Research and Applications*, New York, NY, pp. 125–132 (2002).
12. J.-G. Wang, E. Sung and R. Venkateswarlu, Eye gaze estimation from a single image of one eye, in: *Proc. 9th IEEE Int. Conf. on Computer Vision*, Nice, pp. 136–143 (2003).
13. V. Ferrari, T. Tuytelaars and L. Van Gool, Simultaneous object recognition and segmentation by image exploration, *Lecture Notes Comp. Sci.* **3021**, 40–54 (2004).
14. D. Lowe, Distinctive image features from scale-invariant keypoints, *Int. J. Comp. Vis.* **60**, 91–110 (2004).
15. K. Mikolajczyk and C. Schmid, Scale & affine invariant interest point detectors, *Int. J. Comp. Vis.* **60**, 63–96 (2004).
16. F. Schaffalitzky and A. Zisserman, Automated location matching in movies, *Comp. Vis. Image Understand.* **92**, 236–264 (2003).
17. H. Bay, T. Tuytelaars and L. Van Gool, SURF: speeded up robust features, in: *Proc. 9th Eur. Conf. on Computer Vision*, Graz, pp. 404–417 (2006).
18. C. Harris and M. Stephens, A combined corner and edge detector, in: *Proc. 4th Alvey Vision Conf.*, Manchester, UK, pp. 147–151 (1998).
19. K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Van Gool, A comparison of affine region detectors, *Int. J. Comp. Vis.* **65**, 43–72 (2005).
20. K. Mikolajczyk and C. Schmid, A performance evaluation of local descriptors, *IEEE Trans. Pattern Anal. Machine Intell.* **27**, 1615–1630 (2005).
21. D. Lowe, Local feature view clustering for 3D object recognition, in: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Maui, HI, pp. 682–688 (2001).
22. S. Se, D. Lowe and J. Little, Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks, *Int. J. Robotics Res.* **21**, 735–758 (2002).
23. C. Schmid and R. Mohr, Local grayvalue invariants for image retrieval, *IEEE Trans. Pattern Anal. Machine Intell.* **19**, 530–534 (1997).
24. J. Ruiz-del-Solar, P. Vallejos, I. Parra, J. Testart, R. Briones, M. I. Avilés, J. Abuhadba and P. Ravest, UChile RoadRunners 2007 team description paper, in: *Proc. RoboCup Symp.* Atlanta, GA, CD-ROM (2007).
25. M. Correa, J. Ruiz-del-Solar and F. Bernuy, Face recognition for human-robot interaction applications: a comparative study, in: *Proc. RoboCup Symp.* Suzhou, CD-ROM (2008).

Appendix

Linear Correlation Test

An affine transformation can be calculated from a set of matches between interest points (x_i, y_i) in the reference image and interest points (u_i, v_i) in the test image. This affine transformation can be represented in the following two ways:

$$\begin{aligned}\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_X \\ t_Y \end{pmatrix} \Rightarrow \begin{pmatrix} u \\ v \end{pmatrix} \\ = \begin{pmatrix} x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \end{pmatrix}(m_{11} \quad m_{12} \quad t_X \quad m_{21} \quad m_{22} \quad t_Y)^{\mathrm{T}}.\end{aligned}$$

From the last expression and using least squares, the parameters of the transformation can be calculated from matches between points (x_i, y_i) and (u_i, v_i) as:

$$\begin{pmatrix} m_{11} \\ m_{12} \\ t_X \\ m_{21} \\ m_{22} \\ t_Y \end{pmatrix} = (X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}} \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \dots \end{pmatrix}; \qquad X = \begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 0 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \end{pmatrix}.$$

The parameters are calculable only if the 6×6 $X^{\mathrm{T}}X$ matrix is invertible and this is possible only if X has rank 6. However, if the points in the reference image lie on a straight line, $y_k = a \cdot x_k + b$ holds, and the first and second columns, in X, as well as the fourth and fifth columns, become linearly dependent, and X gets at most rank 4. Then, if the points in the reference image lie on a straight line, the parameters of a transformation from the reference to the test image cannot be successfully calculated. In the symmetric case, if the points in the test image lie on a straight line, a transformation from the test to the reference image cannot be calculated. Then, to get a numerically stable and invertible transformation, the points in the reference and the test image cannot lie, or approximately lie, on a straight line, i.e., the correlation coefficients of the points in both images must be low.

Fast Probability Computation

The probability that a bin B represents a true mapping m_B can be calculated without knowing the associated affine transformation. We compute this probability to all bins, with four or more votes, as [6]:

$$P(m_B|\#B \geqslant 4) = \frac{P(m_B)}{P(m_B) + \sum_{\alpha=4}^{N} \binom{N}{\alpha} p_B^{\alpha}(1 - p_B)^{N-\alpha}},$$

with $\#B$ the number of votes in the bin B, $N = 16 \times n$ the total number of random votes generated by the n matches that exists in all the bin-space, and $p_B = p(z)p(k)p(i, j|k, z)$.

If it is assumed that the density of interest points along the sub-sampled scale space is constant, $p(z)$ is given by $p(z) = 3/5(1/4)^{|z|}$ [6]. $p(i, j|k, z)$ is estimated as the ratio between the space covered by the matches compatibles with the bin (i, j, k, z) and the space covered by all the possible matches that can be generated between the pair of images. An expression for this probability is calculated in Ref. [6]. Finally, $p(k)$ is calculated as $w/360°$, where w is the angular width of the bin.

Geometrical Distortion Test

A correct affine transformation should not deform an object very much when mapping it. Therefore, if the transformation produces a large geometrical distortion, it

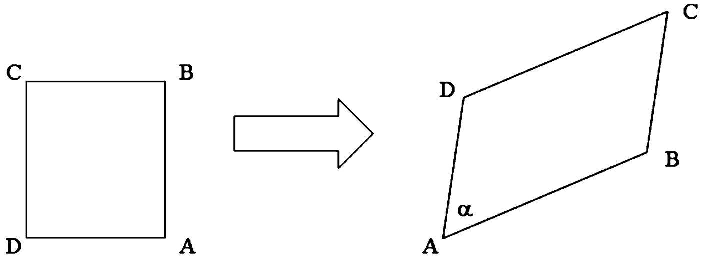


Figure A1. Affine mapping of a square into a parallelogram.

should be rejected. Given that we have just a hypothesis of the object pose, it is not easy to determine the object distortion. However, we do have the mapping function, i.e., the transformation. Therefore, we can verify if the mapping function produces distortion or not using a known, regular and simple object, such as a square. The transformation of a square should produce a rotated parallelogram. If the transformation does not produce a large distortion, the conditions that the transformed object should fulfill are (see notation in Fig. A1):

$$\max\left\{\frac{\mathrm{d}(AB)/\mathrm{d}(A'B')}{\mathrm{d}(BC)/\mathrm{d}(B'C')}, \frac{\mathrm{d}(BC)/\mathrm{d}(B'C')}{\mathrm{d}(AB)/\mathrm{d}(A'B')}\right\} < th_{\mathrm{prop}};$$

$$\alpha = \sin^{-1}\left|\frac{\det(\overrightarrow{A'B'}\ \ \overrightarrow{B'C'})}{\mathrm{d}(A'B') \times \mathrm{d}(B'C')}\right| > th_{\alpha},$$

$\overrightarrow{A'B'}$ is a vector from A′ to B′ and $\det(\overrightarrow{A'B'}\ \ \overrightarrow{B'C'})$ computes the parallelogram area.

Transformation Fusion

It can happen that more than one correct transformation corresponding to the same object is obtained after the matching of a reference–test pair of images. There are many reasons for that, e.g., small changes in the object view with respect to the prototypes views, transformations obtained when matching parts of the object as well as the whole object, etc. When these multiple, overlapping transformations are detected, they should be merged. Therefore, a procedure for verifying the similarity of a pair of transformations is required. We use a similar idea to that employed for computing the geometrical distortion test. That is, we map a certain square using the both affine transformation whose similarity is been verified. If the obtained parallelograms have a certain overlap ‘over’, then the transformations are similar and they can be fused:

$$\mathrm{over} = 1 - \frac{\mathrm{dist}(A'_1A'_2) + \mathrm{dist}(B'_1B'_2) + \mathrm{dist}(C'_1C'_2) + \mathrm{dist}(D'_1D'_2)}{\mathrm{perimeter}(A'_1B'_1C'_1D'_1) + \mathrm{perimeter}(A'_2B'_2C'_2D'_2)} > th_{\mathrm{over}},$$

with $\{A'_1, B'_1, C'_1, D'_1\}$ and $\{A'_2, B'_2, C'_2, D'_2\}$ the vertices defining each parallelogram.

In addition, it should be also verified if the difference between the rotations produced for each transform is not very large. Therefore, we ask also that:

$$|\mathrm{rot}_1 - \mathrm{rot}_2| < th_{\mathrm{diff\_rot}},$$

with rot_1 and rot_2 the rotations produced by each transformation, which are computed as the mean value of the differences between the orientation of each matched SIFTs keypoint in the prototype and test image.

Pixel Correlation Test

Pixel or graphical correlation is a measure of how similar the regions being mapped by the affine transformation are. Transformations producing low graphical correlation between the object prototype image and the candidate object sub-image should be discarded. The graphical correlation is given by:

$$r_{\mathrm{g}} = \frac{\sum_{u=0}^{U}\sum_{v=0}^{V}(I(u,v)-\bar{I})(I'(x_{\mathrm{TR}}(u,v),y_{\mathrm{TR}}(u,v))-\overline{I'})}{\sqrt{\sum_{u=0}^{U}\sum_{v=0}^{V}(I(u,v)-\bar{I})^2\sum_{u=0}^{U}\sum_{v=0}^{V}(I'(x_{\mathrm{TR}}(u,v),y_{\mathrm{TR}}(u,v))-\overline{I'})^2}},$$

where $(x = x_{\mathrm{TR}}(u,v), y = y_{\mathrm{TR}}(u,v))$ defines the transformation, and $I(u,v)$ and $I'(x,y)$ correspond to the prototype image and the candidate object sub-image, respectively.

About the Authors

Javier Ruiz-del-Solar received his diploma in Electrical Engineering and the MS degree in Electronic Engineering from the Technical University Federico Santa Maria, Chile, in 1991 and 1992, respectively, and the DE degree from the Technical University of Berlin, in 1997. In 1998, he joined the Electrical Engineering Department of the Universidad de Chile of as an Assistant Professor. In 2001, he became Director of the Robotics Laboratory and, in 2005, Associate Professor. His research interests include Mobile robotics, human–robot interaction and face analysis. He is the recipient of the IEEE RAB Achievement Award 2003, RoboCup Engineering Challenge Award 2004, RoboCup@Home Innovation Award 2007 and RoboCup@Home Innovation Award 2008. Since 2006, he is a Senior Member of the IEEE and, since 2008, Distinguished Lecturer of the IEEE Robotics and Automation Society.

Patricio Loncomilla received the BS degree and the Diploma in Electrical Engineering from University of Chile, Santiago, Chile, in 2004 and 2005, respectively. He was a member of the Robotics Laboratory at the University of Chile from 2005 to 2007. He worked in color-based computational vision applied to AIBO robots for RoboCup competitions and generic object recognition using local descriptors. Currently, he is a member of the Computational Vision Laboratory and a PhD candidate in the Department of Electrical Engineering at the University of Chile, where he is working on his Doctoral thesis about automatic generation and recognition of 3-D visual landmarks. His research interests include computer vision techniques for object recognition, 3-D scene analysis and mobile robot navigation.