Sensorimotor Processes for Learning Object Representations

Damir Omrčen [#1], Aleš Ude [#2], Kai Welke [*3], Tamim Asfour [*4], Rüdiger Dillmann [*5]

[#] *Jozef Stefan Institute*
Ljubljana, Slovenia
[1] damir.omrcen@ijs.si
[2] ales.ude@ijs.si

[*] *University of Karlsruhe*
Karlsruhe, Germany
[3] welke@ira.uka.de
[4] asfour@ira.uka.de
[5] dillmann@ira.uka.de

***Abstract*— Learning object representations by exploration is of great importance for cognitive robots that need to learn about their environment without external help. In this paper we present sensorimotor processes that enable the robot to observe grasped objects from all relevant viewpoints, which makes it possible to learn viewpoint independent object representations. Taking control of the object allows the robot to focus on relevant parts of the images, thus bypassing potential pitfalls of pure bottom-up attention and segmentation. We propose a systematic method to control a robot in order to achieve a maximum range of motion across the 3-D view sphere. This is done by exploiting the task redundancies typically found on a humanoid arm and by avoiding joint limits of the robot. The proposed method brings the robot into configurations that are appropriate for observing objects. It enables us to acquire a wider range of snapshots without regrasping the object.**

I. Introduction

Learning about new objects without any prior information about them is a difficult problem, which is not easy to solve by passive observers. It has been suggested that an active vision paradigm can resolve many of the ill-posed problems arising in static vision [1], [2]. A humanoid robot has the potential to explore its world using causality, by performing probing actions and learning from the response [3]. It has been shown that poking an object can be used to extract visual evidence for the boundary of the object, which is well suited for segmentation [4]. These approaches demonstrated that by actively exploring the environment, the robot can gain some knowledge about the objects in its world.

When learning about new objects, the robot needs to first find interesting areas in the scene and generate the initial grasp hypotheses, which is followed by attempts to grasp the object. While this paper does not solve the no doubt difficult problems of generating initial grasp hypotheses and grasping itself, we consider here the also difficult problem of acquiring snapshots of objects across a continuous portion of the view sphere without having prior information about the object appearance. The main idea is that by having physical control of the object, the robot can bring enough knowledge into the system to ensure that it can segment the object from the background, thus solving the *figure-ground discrimination* problem, and capture snapshots suitable for learning. We show that the proposed approach can be applied to learn representations suitable for tasks such as object recognition.

The focus of the paper is on sensorimotor processes necessary to realize the observation of objects from all relevant viewpoints of the view sphere:

- An explorative movement primitive that can be used to determine an optimal placement of the object with respect to the robot's eyes so that the object will be in the image center and have appropriate size for learning, i. e. it will cover significant portion of the image while being away from the image boundary.
- A primitive motion that can be used to observe the grasped object from various viewpoints while keeping it centered in the image. Due to the limited manipulation capabilities of humanoid robots and arms, it is unavoidable to regrasp the observed object to ensure that the robot looks at it from all relevant viewpoints. However, the number of necessary regrasps can be reduced by performing the exploratory movements in an optimal way so that the redundancy of the humanoid is exploited and the joint limits are avoided.
- A Bayesian visual process that enables the robot to segment the object from its surroundings and acquire snapshots of the object without having prior information about its appearance.

We utilize visual servoing techniques to realize the observation of objects. Although visual servoing was studied extensively in the past [5], these techniques can be quite difficult to apply in practice because of the limited workspace of the robots. These limits occur because of the presence of kinematic singularities throughout the workspace and the possibility of exceeding physical joint limits during manipulation. In some cases it is possible to specify the trajectories in advance considering robot's restrictions. However, more

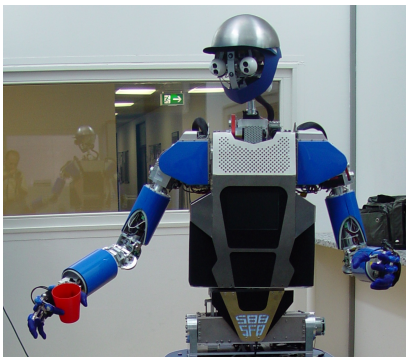


Fig. 1. The target system: humanoid robot ARMAR-III

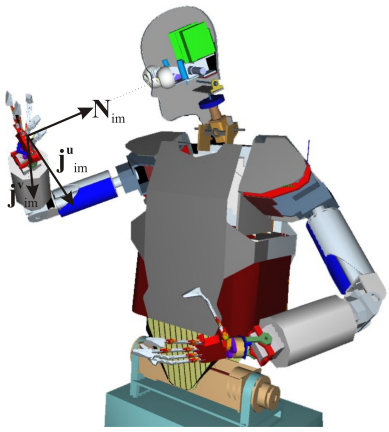


Fig. 2. Vectors in the world coordinate system $\mathbf{j}_{im}^{u}$ and $\mathbf{j}_{im}^{v}$ correspond to the u and v axis in the image coordinate system and $\mathbf{N}_{im}$ is the image null space vector, which is directed along the camera ray and is orthogonal to both $\mathbf{j}_{im}$ vectors

intelligent way is to adapt the trajectories on-line considering acquired information of the object and considering current state of the robot. To ensure that the robot can observe the object from all sides, the robot should re-grasp the object using one or both of its hands or even move the head and eyes in order to achieve better direction of view. We are currently working on this behavior.

II. Observation Primitives

We developed control algorithms that achieve an optimal scanning behavior by actively controlling the arm in the null space. The goal of the manipulation process is to first bring the object into the view of the robot cameras at an optimal size for observation, which is followed by rotating the object so that it can be observed over a continuous portion of a 3-D view sphere.

However, if the vision-based task does not constraint all robot's degrees of freedom (DOFs) then the robot is redundant with respect to the task. A redundant manipulator is more dexterous than a nonredundant [6]. Namely, it has the ability to move in the joint space without affecting the motion in the task space. Therefore, a redundant manipulator can execute given task (called primary task) together with an additional less important subtask (called secondary task). For example, a redundant manipulator can track a trajectory while avoiding obstacles [7] or singularities, optimizing joint torques [8] or optimizing various performance criteria (e.g. manipulability) [9]. By exploiting the redundancy the robot can for example achieve wider range of motion by avoiding singularities or joint limits [10], [11]. In this paper we propose a method for exploiting a robot's redundancy in such a way that the robot achieves wider range of motion with respect to the orientation of the object in depth.[1]

We developed our approach having in mind a humanoid robot ARMAR-III [12] that manipulates the object and observes it with its own eyes (see Fig. 1).

A. Image Jacobian and Null Space

The vision system uses one camera which is placed in the robot's right eye. Its position and orientation (extrinsic parameters) with respect to the robot's arm are known, i.e., they depend on the robot's kinematics. The intrinsic parameters of the camera are acquired using a chess board.

The relationship between a point in the 3D world (or arm) coordinate system and 2D image coordinate system is given by:

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \mathbf{A} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \tag{1}$$

here u and v are the horizontal and vertical position of a point in the camera image c.s., respectively. x, y and z are the coordinates of a same point expressed in the world c.s. Matrix $\mathbf{A}$ is the transformation matrix that determines the relationship between the world and the image c.s. and represents the calibration of the camera. The matrix $\mathbf{A}$ incorporates extrinsic (position and orientation of the camera) and intrinsic (focal lengths, pixel size, image center) camera parameters.

With a calibrated camera we can place the object grasped by the robot in the center of the camera image. We realized this behavior using an analytical expression for the image Jacobian that defines the relationship between velocities of the point in the 3-D world ($[\dot{x}, \dot{y}, \dot{z}]^T$) and in the 2-D image coordinate system ($[\dot{u}, \dot{v}]^T$):

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \mathbf{J}_{im} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} j_{11} & j_{12} & j_{13} \\ j_{21} & j_{22} & j_{23} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}. \tag{2}$$

Since Eq. (2) is underdetermined, one redundant DOF exist, i.e. we can find a vector $\mathbf{N}_{im}$ in the space of world velocities, which does not produce any movement of the point in the image (see Fig. 2). This vector is directed along the ray from the projection center to the observed 3-D point.

Fig. 2 also shows the two vectors ($\mathbf{j}_{im}^{u}$ and $\mathbf{j}_{im}^{v}$), which represent the vectors in the world coordinate system that

[1]Depth rotations are those rotations that cause a different part of the object to be visible in the image

produce only the movement along u and v direction in the image, respectively, and do not produce any motion along the camera ray. We can show that these two vectors are given by the rows of the Jacobian. We can compute $\mathbf{j}_{im}^{u}$ and $\mathbf{j}_{im}^{v}$ by normalizing the two rows of the Jacobian:

$$\mathbf{j}_{im}^{u} = \frac{\begin{bmatrix} j_{11} & j_{12} & j_{13} \end{bmatrix}^T}{\left\| \begin{bmatrix} j_{11} & j_{12} & j_{13} \end{bmatrix} \right\|}, \quad \mathbf{j}_{im}^{v} = \frac{\begin{bmatrix} j_{21} & j_{22} & j_{23} \end{bmatrix}^T}{\left\| \begin{bmatrix} j_{21} & j_{22} & j_{23} \end{bmatrix} \right\|}.$$

Vector $\mathbf{N}_{im}$, which does not produce any movement in the image, is in the null space of the image Jacobian. When computing the vector $\mathbf{N}_{im}$ using classical approach: $\mathbf{N}_{im} = \mathbf{I} - \mathbf{J}_{im}{}^{\#}\mathbf{J}_{im}$, discontinuities can appear because $\mathbf{N}_{im}$ is directed along the camera ray in any direction (to or away from the camera). Due to the changing sign of $\mathbf{N}_{im}$ it is hard to control the robot smoothly and without discontinuities in the control signal. One way to resolve this problem is to use Givens rotations for computing $\mathbf{N}_{im}$ [13]. Here a simpler solution is possible due to the low dimensionality of the problem. $\mathbf{N}_{im}$ can be calculated using vector product of $\mathbf{j}_{im}^{u}$ and $\mathbf{j}_{im}^{v}$:

$$\mathbf{N}_{im} = \frac{\mathbf{j}_{im}^{u} \times \mathbf{j}_{im}^{v}}{||\mathbf{j}_{im}^{u} \times \mathbf{j}_{im}^{u}||}. \tag{3}$$

The resulting vector $\mathbf{N}_{im}$ is orthogonal to both vectors $\mathbf{j}_{im}^{u}$ and $\mathbf{j}_{im}^{v}$ and is therefore in the null space of the $\mathbf{J}_{im}$.

B. Centering the Object at Optimal Distance

The controller is composed of two parts. The first part corresponds to the position control of the object and the second part corresponds to the size (or, equivalently, distance) control of the object. Our method thus belongs to the class of image-based control algorithms [5]. The task of the position controller is to bring the object to the center of the image. The size controller should only control the object size and should not affect the position control; hence it should act in the null space of the image Jacobian.

Since the positioning task is orthogonal to the sizing task it is possible to join both tasks together. We can define the following relationship between the velocities in the image space together with the velocity along the camera ray and the velocities in the world space:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{d} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{im} \\ \mathbf{N}_{im} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}, \tag{4}$$

where $\dot{d}$ is the velocity along the camera ray.

To control a robot, we have to define the control velocities in the joint space $\dot{\boldsymbol{q}}$. The relationship between the task and the joint velocities is given by the positional part of the robot's Jacobian $\mathbf{J}_r{}^{pos}$ as:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \mathbf{J}_r{}^{pos}\dot{\boldsymbol{q}}. \tag{5}$$

From the equations (4) and (5) we can derive the relationship between the velocities in the image space together with the velocity along the camera ray and between the joint velocities:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{d} \end{bmatrix} = \left(\begin{bmatrix} \mathbf{J}_{im} \\ \mathbf{N}_{im} \end{bmatrix} \mathbf{J}_r{}^{pos} \right) \dot{\boldsymbol{q}}. \tag{6}$$

Since the task has three DOFs (position of all three coordinates in space) and the robot arm used in the experiments has seven DOFs, the degree of redundancy is four. The following controller can be used:

$$\dot{\boldsymbol{q}}_c = \left(\begin{bmatrix} \mathbf{J}_{im} \\ \mathbf{N}_{im} \end{bmatrix} \mathbf{J}_r{}^{pos} \right)^{\#} \begin{bmatrix} \dot{u}_c \\ \dot{v}_c \\ \dot{d}_c \end{bmatrix} + \mathbf{N}_r{}^{impos}\dot{\boldsymbol{q}}_n. \tag{7}$$

Here $()^{\#}$ denotes the generalized inverse and $\mathbf{N}_r{}^{impos}$ is the corresponding null space projection matrix. Control velocities $[\dot{u}_c, \dot{v}_c, \dot{d}_c]^T$, which correspond to the position and size controller can be defined by simple P controllers as:

$$\begin{bmatrix} \dot{u}_c \\ \dot{v}_c \\ \dot{d}_c \end{bmatrix} = \begin{bmatrix} K_p^u(u_d - u) \\ K_p^v(v_d - v) \\ K_p^d(size_d - size) \end{bmatrix}, \tag{8}$$

where u_d, v_d and $size_d$ are the desired and the u, v and $size$ are the actual position of the point (or object) in the image coordinate system and size of the object, respectively. K_p^u, K_p^v and K_p^d are the controller gains.

Note that using the object size estimated by the vision system can sometimes lead to inaccuracy because the size does not depend only on the distance of the object from the camera. In such cases we propose to use a controller that controls the distance from the object to the camera instead of size control. The distance can be calculated using kinematics of the robot and extrinsic parameters of the camera, which can be extracted from the calibration matrix as shown in [14].

Due to the robot's redundancy we can define an additional subtask in order to achieve better performance of the robot. The null space term $\dot{\boldsymbol{q}}_n$ in Eq. (7) defines the secondary task. There are various possibility to select the secondary task. Based on the experiments we have selected joint limits avoidance as a secondary task to achieve largest range of motion. The function that defines the null space motion for each joint is shown in Fig. 3 and is defined as:

$$q_n = K_{ns} tan\left(\frac{3\left(q - \frac{q_{max}+q_{min}}{2}\right)}{q_{max} - q_{min}} \right), \tag{9}$$

where q_{min} and q_{max} are minimal and maximal joint limit and K_{ns} is the gain.

C. Showing the Object from Different Viewpoints

To acquire data about the object from different viewpoints, the robot needs to rotate it in depth with respect to the image coordinate system. Rotation in depth is defined as any rotation with the rotation axis not parallel to the camera ray. Largest

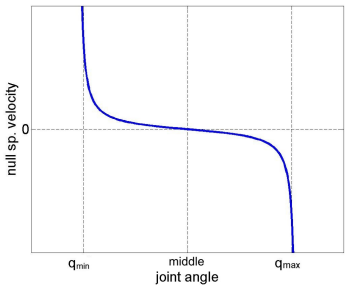


Fig. 3. Function that defines null space motion for joint limits avoidance

rotations in depth will therefore be caused by rotations about $\mathbf{j}_{im}^u$ and $\mathbf{j}_{im}^v$, since these two vectors are orthogonal to the camera ray (see Fig. 2). Note that the rotation about the vector in the direction of the camera ray ($\mathbf{N}_{im}$) does not produce any depth rotation; therefore, it is not important and can be considered as redundant. Due to the additional two DOFs for rotation, the task now has five DOFs and the degree of redundancy is two. Let us define the robot Jacobian that corresponds to the both largest depth rotations about $\mathbf{j}_{im}^u$ and $\mathbf{j}_{im}^v$ axes:

$$\mathbf{J}_r{}^{dr} = \begin{bmatrix} \mathbf{j}_{im}^u & \mathbf{j}_{im}^v \end{bmatrix}^{\#} \mathbf{J}_r{}^{rot}. \tag{10}$$

Here, $\mathbf{J}_r{}^{dr}$ is the robot Jacobian, where first row corresponds to the rotation about vector $\mathbf{j}_{im}^u$ and the second row to the rotation about $\mathbf{j}_{im}^v$. $\mathbf{J}_r{}^{rot}$ is the rotational part of the robot Jacobian with dimension 3 $\times$ number of DOF.

When controlling the position of the object together with two orientations of the object the following controller can be used:

$$\dot{\boldsymbol{q}}_{c_2} = \begin{bmatrix} \begin{bmatrix} \mathbf{J}_{im} \\ \mathbf{N}_{im} \end{bmatrix} \mathbf{J}_r{}^{pos} \\ \mathbf{J}_r{}^{dr} \end{bmatrix}^{\#} \begin{bmatrix} \dot{u}_c \\ \dot{v}_c \\ \dot{d}_c \\ \dot{s}_u \\ \dot{s}_v \end{bmatrix} + \mathbf{N}_r{}^{impos,dr} \dot{\boldsymbol{q}}_n, \tag{11}$$

where $\dot{\boldsymbol{s}}_u$ and $\dot{\boldsymbol{s}}_v$ are the rotation velocities about $\mathbf{j}_{im}^u$ and $\mathbf{j}_{im}^v$ vectors. $\mathbf{N}_r{}^{impos,dr}$ is the corresponding projection in the null space. $\dot{\boldsymbol{q}}_n$ is same as in previous case (9), and is used to avoid the joint limits and to achieve larger range of rotations of the robot.

III. Snapshot Acquisition

Now we turn to the problem of how to acquire snapshots of objects manipulated by the robot. We are not interested in using standard turntables to acquire snapshots of objects because an autonomous humanoid should be able to learn new objects by itself.[2] Consequently it is necessary to solve problems that do not need to be considered in carefully prepared turntable setups. Most importantly, to solve the figure-ground discrimination problem, we cannot use techniques like chroma keying because we do not want to assume a uniform background. However, having physical control over the object allows the robot to acquire some information about the rest of the scene without making hard assumptions like uniformly colored background.

[2]Besides autonomy, this also has the advantage that we can take snapshots of objects from all relevant viewing directions, whereas classic turntables only allow for rotations around one degree of freedom.

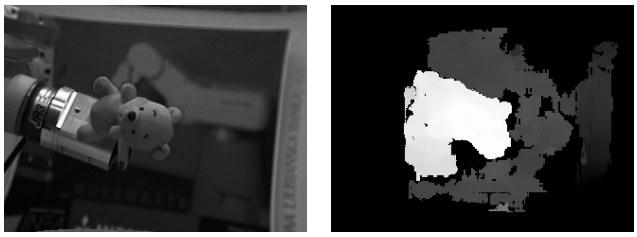


Fig. 4. One of the collected images after rectification and the corresponding disparity map

In general, background models are subject to frequent changes. However, this is less of a problem here because background models are needed only for short periods of time and can be learned anew if necessary. We therefore developed our own Bayesian snapshot acquisition algorithm that is tailored to our problem instead of using more general algorithms, like for instance [15], which can deal to some extent with varying backgrounds, but do not take into account specific characteristics of our system. The approach assumes the existence of the following probabilistic processes to model the projection of the external world onto the robot images:

- the unknown object (denoted by process Θ_{o}),
- the background (Θ_{b}),
- the hand (Θ_{h}), and
- the outlier process (Θ_{t}), modeling any unexpected events in the scene.

Stationary background can be modelled using various features such as for example color distribution, disparity, and motion. Currently, we use the first two of them. The color distribution at each pixel in the stationary background is modelled by a Gaussian process $\Theta_{\mathrm{b1}} = \{\overline{\boldsymbol{I}}_{\boldsymbol{u}}, \overline{\boldsymbol{\Sigma}}_{\boldsymbol{u}}\}_{\boldsymbol{u}}$, which is characterized by mean $\overline{\boldsymbol{I}}_{\boldsymbol{u}}$ and covariance matrix $\overline{\boldsymbol{\Sigma}}_{\boldsymbol{u}}$ at each pixel $\boldsymbol{u}$ with the associated probability distribution

$$\mathrm{p}(\boldsymbol{I}_{\boldsymbol{u}}, \boldsymbol{u}|\Theta_{\mathrm{b1}}) = \frac{1}{2\pi\sqrt{\det(\overline{\boldsymbol{\Sigma}}_{\boldsymbol{u}})}} \cdot \exp\left(-\frac{1}{2}(\boldsymbol{I}_{\boldsymbol{u}} - \overline{\boldsymbol{I}}_{\boldsymbol{u}})^T \overline{\boldsymbol{\Sigma}}_{\boldsymbol{u}}^{-1} (\boldsymbol{I}_{\boldsymbol{u}} - \overline{\boldsymbol{I}}_{\boldsymbol{u}})\right). \tag{12}$$

The means and the covariances are learned by gathering statistics of the background images $\boldsymbol{I}$ for about 10 seconds just before the robot brings the object into the fovea. We did not observe big differences when using either of the two color spaces, but more experiments are needed to confirm this point. It is essential to smooth the images significantly before applying this calculation; otherwise even small disturbances can cause failure.

Disparity as shown in Fig. 4 is another strong cue with a good property of being robust against changes in lighting conditions. Let D be the estimated disparity image. We

model the disparity distribution as a Gaussian process $\Theta_{\mathrm{b}2} = \{\overline{D}\boldsymbol{u}, \sigma^2_{\mathrm{D}}\}\boldsymbol{u}$. In the same way as for color we estimate the means $\overline{D}\boldsymbol{u}$ at each pixel by collecting disparity images of a stationary background for 10 seconds. The standard deviation σ_{D} is not estimated but is set to a constant value. Finally, we calculate the following estimate for the background distribution

$$\mathrm{p}(\boldsymbol{I}\boldsymbol{u}, D\boldsymbol{u}, \boldsymbol{u}|\Theta_{\mathrm{b}}) = \mathrm{p}(\boldsymbol{I}\boldsymbol{u}, \boldsymbol{u}|\Theta_{\mathrm{b}1})\mathrm{p}(D\boldsymbol{u}, \boldsymbol{u}|\Theta_{\mathrm{b}2}). \quad (13)$$

Even though the hand position in the image could be calculated using proprioceptive information, this information is not sufficient because we cannot know in advance which part of the hand is visible and which part is covered by the manipulated object. We thus need to model the appearance of the hand in the image. For the modelling of the hand appearance, we experimented with standard approaches from the object tracking theory such as color histograms [16] and Gaussian (mixture) models [17]. Unlike in tracking, we are not really interested in computing the hand position but only in estimating the probability that a particular pixel belongs to the hand. Both color histograms and Gaussian mixture models offer this ability. Gaussian mixture models are defined as follows

$$\mathrm{p}(\boldsymbol{I}\boldsymbol{u}|\Theta_{\mathrm{h}}) = \sum_{k=1}^{K} \frac{\omega_k}{2\pi\sqrt{\det(\boldsymbol{\Sigma}_k)}} \cdot \exp\left(-\frac{1}{2}(\boldsymbol{I}\boldsymbol{u} - \overline{\boldsymbol{I}}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{I}\boldsymbol{u} - \overline{\boldsymbol{I}}_k)\right) \quad (14)$$

and this is what we use to model the hand.

While motion cues could certainly help to extract the object from the hand and background, such cues alone are not sufficient for the extraction of the object appearance. When the robot holds the object, the object motion is the same as the motion of the robot hand. We can thus not distinguish between the object and the hand based on the motion cue only. In addition, motion estimates are normally calculated by differential methods which make them relatively noisy. Hence motion should be used only as support for other cues and not as the sole feature for segmentation.

Since we have no prior knowledge about the object, we obviously cannot model its appearance, which is actually what we want to learn. The motion that we use to manipulate the object is, however, well defined and we know approximately where the object is in the image. We can thus model the probability that an image pixel falls within the extent of the object by using the mean value $\overline{\boldsymbol{u}}$ and the covariance $\overline{\boldsymbol{\Sigma}}$ of pixels belonging to the object in the previous step. This results in the following distribution

$$\mathrm{p}(\boldsymbol{u}|\Theta_{\mathrm{o}}) = \frac{1}{2\pi\sqrt{\det(\overline{\boldsymbol{\Sigma}})}} \exp\left(-\frac{1}{2}(\boldsymbol{u} - \overline{\boldsymbol{u}})^T\overline{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{u} - \overline{\boldsymbol{u}})\right). \quad (15)$$

Since the robot attempts to bring the object to the center of the image and to keep it there, the object's position is normally close to the image center and we can initialize the appearance extraction by assuming that the object is centered in the image with an initially large extent. The calculation of object probabilities thus shows the integration between perception and motor control on our system.

Fig. 6 shows that images sometimes contain other parts of the arm besides the hand. Having no prior information about the appearance of the arm and other unexpected objects that might appear in the scene, we model such events in the image by an outlier process, which is assigned a small, constant probability $\mathrm{P}(\Theta_{\mathrm{t}})$ regardless of the position of the pixel in the image or color intensity value at this pixel. The interaction between this process and the object process Θ_{o} occurs in such a way that an area with texture different from the background and hand will be classified as an object of interest if it is close to the expected object position and outlier otherwise (see Eq. (19)).

As for the arm, the part of the image containing it can be excluded from calculations using proprioceptive information. On a dynamic humanoid robot like ARMAR-III, proprioceptive information provides only a rough estimate for the location of the arm in the image. It is nevertheless sufficient to exclude from the calculations most of the image containing the arm.

Assuming that every pixel in the image stems from one of the mutually independent processes $\boldsymbol{\Theta} = \{\Theta_{\mathrm{b}}, \Theta_{\mathrm{h}}, \Theta_{\mathrm{o}}, \Theta_{\mathrm{t}}\}$ (closed-world assumption), we can write the probability that color $\boldsymbol{I}\boldsymbol{u}$ was observed at location $\boldsymbol{u}$ using the total probability law

$$\mathrm{P}(\boldsymbol{I}\boldsymbol{u}, \boldsymbol{u}|\boldsymbol{\Theta}) = \omega_{\mathrm{b}}\mathrm{P}(\boldsymbol{I}\boldsymbol{u}, D\boldsymbol{u}, \boldsymbol{u}|\Theta_{\mathrm{b}}) + \omega_{\mathrm{h}}\mathrm{P}(\boldsymbol{I}\boldsymbol{u}|\Theta_{\mathrm{h}}) + \omega_{\mathrm{o}}\mathrm{P}(\boldsymbol{u}|\Theta_{\mathrm{o}}) + \omega_{\mathrm{t}}\mathrm{P}(\Theta_{\mathrm{t}}), \quad (16)$$

where ω_x are the prior (mixture) probabilities to observe the processes Θ_x and $\omega_{\mathrm{b}} + \omega_{\mathrm{h}} + \omega_{\mathrm{o}} + \omega_{\mathrm{t}} = 1$.

We need to estimate the current position of the unknown object and its extent, which will provide us with an appearance image for learning. This can be achieved by maximizing the probability of observing image $\boldsymbol{I}$ given processes $\boldsymbol{\Theta} = \{\Theta_{\mathrm{b}}, \Theta_{\mathrm{h}}, \Theta_{\mathrm{o}}, \Theta_{\mathrm{t}}\}$. Neglecting the correlation of assigning neighboring pixels to processes, we can evaluate the overall probability of observing image $\boldsymbol{I}$ as follows

$$\mathrm{P}(\boldsymbol{I}) = \mathrm{P}(\boldsymbol{I}|\boldsymbol{\Theta}) = \prod_{\boldsymbol{u}} \mathrm{P}(\boldsymbol{I}\boldsymbol{u}, \boldsymbol{u}|\boldsymbol{\Theta}). \quad (17)$$

Since the background and the color distribution of the hand are assumed stationary, we can maximize (17) with respect to the position $\overline{\boldsymbol{u}}$ of the object, the covariance $\overline{\boldsymbol{\Sigma}}$ of pixels belonging to the object, and mixture probabilities ω_{b}, ω_{h}, ω_{o}, and ω_{t}. Instead of maximizing (17), it is easier to minimize the negative log likelihood

$$\mathrm{L}(\boldsymbol{\Theta}, \boldsymbol{\omega}) = -\log(\mathrm{P}(\boldsymbol{I}|\boldsymbol{\Theta})) = -\sum_{\boldsymbol{u}} \log\left(\mathrm{P}(\boldsymbol{I}\boldsymbol{u}, \boldsymbol{u}|\boldsymbol{\Theta})\right). \quad (18)$$

where $\boldsymbol{\omega} = (\omega_{\mathrm{b}}, \omega_{\mathrm{h}}, \omega_{\mathrm{o}}, \omega_{\mathrm{t}})$. Using the Lagrange multipliers theory, it is possible to show that the above log likelihood can

Fig. 5. Images of four objects used in the experiments after warping. Such images are used as input for training and classification.

Fig. 6. The robot holding an object to be learned. The object's position and extent are estimated using the knowledge about the robot's motion and short term background models.

be minimized by an EM algorithm. Writing

$$\mathrm{P}(\boldsymbol{I}_{\boldsymbol{u}}, \boldsymbol{u}|\Theta_x) = \frac{\omega_x \mathrm{p}(\boldsymbol{I}_{\boldsymbol{u}}, \boldsymbol{u}|\Theta_x)}{\sum\limits_{y\in\{\mathrm{o,h,b,t}\}} \omega_y \mathrm{p}(\boldsymbol{I}_{\boldsymbol{u}}, \boldsymbol{u}|\Theta_y)} \tag{19}$$

where $x = \mathrm{o},\ \mathrm{h},\ \mathrm{b},\ \mathrm{t}$, the EM-algorithm consists of the expectation step, in which pixel probabilities (19) are estimated, and the maximization step, in which the probabilities $\mathrm{P}(\boldsymbol{I}_{\boldsymbol{u}}, \boldsymbol{u}|\Theta_\mathrm{b}) = \mathrm{P}(\boldsymbol{u}|\Theta_\mathrm{b})$ are used to estimate the mean and the covariance of the object pixels

$$\overline{\boldsymbol{u}} = \frac{1}{\sum_{\boldsymbol{u}} \mathrm{P}(\boldsymbol{u}|\Theta_\mathrm{b})} \sum_{\boldsymbol{u}} \mathrm{P}(\boldsymbol{u}|\Theta_\mathrm{b})\boldsymbol{u}, \tag{20}$$

$$\overline{\boldsymbol{\Sigma}} = \frac{1}{\sum_{\boldsymbol{u}} \mathrm{P}(\boldsymbol{u}|\Theta_\mathrm{b})} \sum_{\boldsymbol{u}} \mathrm{P}(\boldsymbol{u}|\Theta_\mathrm{b}) (\boldsymbol{u} - \overline{\boldsymbol{u}}) (\boldsymbol{u} - \overline{\boldsymbol{u}})^T. \tag{21}$$

Note that probabilities $\mathrm{P}(\boldsymbol{I}_{\boldsymbol{u}}, \boldsymbol{u}|\Theta_\mathrm{b})$ and $\mathrm{P}(\boldsymbol{I}_{\boldsymbol{u}}|\Theta_\mathrm{h})$ remain constant throughout the EM process and thus need to be estimated only once for each image. This helped us to implement the extraction of the object appearance at video rate, i. e. at 30 Hz. The mixture probabilities can either be assumed to be constant or we can estimate them as part of the EM-process

$$\omega_x = \frac{1}{n} \sum_{\boldsymbol{u}} \mathrm{P}(\boldsymbol{I}_{\boldsymbol{u}}, \boldsymbol{u}|\Theta_x), \tag{22}$$

where n is the number of pixels and $x = \mathrm{o},\ \mathrm{h},\ \mathrm{b},\ \mathrm{t}$.

After estimating the enclosing ellipse, the image of each object is warped into a window of constant size. This ensures invariance against scaling and planar rotations and also provides images of standard size, which can be compared to each other. Fig. 5 shows the warped images of four objects used in our experiments.

IV. Experimental Results

Here we present the experiments carried out on humanoid robot ARMAR-III consisting of a humanoid head with seven DOFs, two arms (seven DOFs per arm) and five-finger hands (eight DOFs per hand), a torso with three DoFs, and a holonomic mobile platform [12] (see Fig. 6). The upper body of the robot has been designed to be modular and light-weight while retaining similar size and proportion as an average person. For the locomotion, we use a mobile platform which allows for holonomic movability in the application area. The head is equipped with two eyes which have a common tilt and can pan independently. Foveated vision is realized using two cameras per eye, one with wide-angle lens for peripheral vision and one with narrow-angle lens for foveal vision. Such visual system allow the implementation of simple visuo-motor behaviors such as tracking and saccadic motions towards salient regions, as well as more complex visual tasks such as hand-eye coordination.

In the experiments we have used only the right arm and the head and the eyes were placed in a predefined position and were static during the experiment. The object to be learned was placed in the robot hand.

The purpose of the proposed method is to achieve the widest range of directions of view. To show the efficiency of the proposed method we compared three different approaches. In the first approach we controlled the robot without joint limits avoidance and without exploiting the redundant DOFs about the axis along the camera ray $\mathbf{N}_{im}$. Here, the robot does not configure in the optimal configuration for performing the observation procedure. Additionally, the orientation of the object about the camera ray axis is fixed, which significantly influence the range of motion.

In the second approach the joint limits are avoided so that the robot moves in the appropriate configuration for performing depth rotations. However, we still do not exploit redundancy of the rotation about the camera ray axis.

In the third approach, which was proposed in this paper, we avoid the joint limits and exploit the redundancy about the camera ray axis using controller (11).

A. Collecting the Views

To objectively show the range of motion for all three compared methods, we defined a rotation matrix, which is fixed to the object. Since we are not interested in orientation of the object about the camera ray (such rotations do not change the part of the object visible to the camera system), we defined the $azimuth$, $elevation$ and $rotation$ as a new rotation representation. This representation is shown in Fig. 7, where $azimuth$ and $elevation$ correspond to a point on the sphere, i.e. direction of view, while $rotation$ corresponds to the rotation about the camera ray axis, which is insignificant. We are interested in the range of orientations with respect to the initial orientation of the object. In the initial configuration $azimuth$, $elevation$ and $rotation$ angles are zero. These three

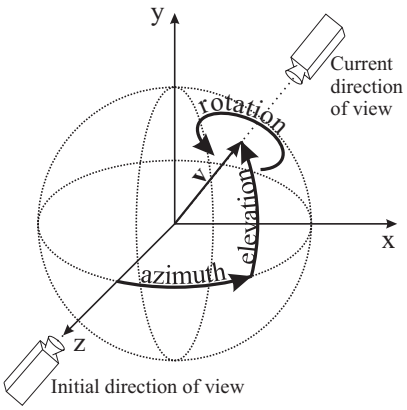


Fig. 7. The coordinate system used for validating the direction of view

angles can be converted into the rotation matrix as follows

$$R = rot(z, rotation)rot(y, azimuth)rot(x, elevation).$$

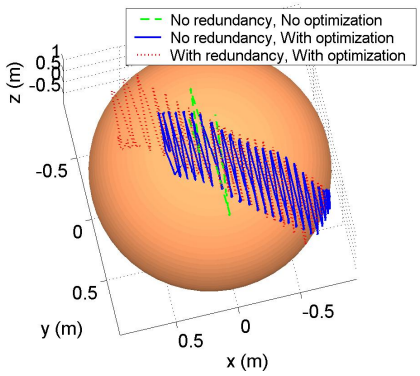
Using the above formula we can easily extract $azimuth$ and $elevation$ from the rotation matrix.

We compared the three methods mentioned above on ARMAR-III humanoid robot. We are interested only in the range of motion for the $azimuth$ and $elevation$ angles. The robot attempted to rotate the object about both axes to achieve maximal depth rotations with angular velocity of 0.2 and 0.02 rd/s, respectively. The robot moved the object about first axis until it came to the end of its workspace (i.e. until singularity or joint limit occurred). Then the robot changed the direction of rotation. With this procedure we acquired the range of rotatory motion for each method.

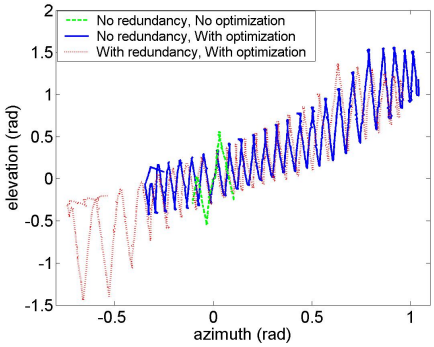
Fig. 9 shows the direction of view while rotating the object. Fig. 9 a) shows the direction of view shown on the sphere, while Fig. 9 b) shows the direction of view represented in $azimuth$, $elevation$ angles. It is clear from these figures that the largest range of motion is achieved when we make use of the redundancy of the rotation about the camera ray axis in addition to a suitable configuration control using joint limits avoidance. Without joint limit avoidance the robot comes very quickly tn the limit of one of the joints. With joint limit avoidance but without exploiting the redundancy, the robot successfully avoids the limits, but its range of motion is smaller than in the case where additional redundancy is exploited.

B. Object Learning

To prove that the proposed approach can indeed be used for learning object representations, we compared two approaches for the acquisition of snapshots for learning. In our older approach the user showed the object to the robot with its own hand and the position and extent of the object were estimated using a known color model. In the second approach the views were collected using the above manipulation procedure and the snapshots were acquired using short term background models and the knowledge about the robot motion. This data was used to train a classifier based on support vector machines [18].



a) represented on the sphere



b) represented in the $azimuth/elevation$ space

Fig. 9. Direction of view while rotating the object

TABLE I
CLASSIFICATION RESULTS

	Correct	Errors	Recognition rate
Full library	7307	421	94.6 %
Objects shown by the user	4897	303	94.2 %
Snapshots acquired by manipulation	2410	118	95.3 %

Altogether we collected 104 views of 14 different objects. The appearance images of four of them were acquired using the approach described in this paper, while the snapshots of the rest of the objects were collected by the old approach. For training of a fully rotationally and scale invariant classifier on a library of 14 objects, we thus employed 1456 feature vectors of dimension 16080.

For testing we collected other 7728 appearance images of objects from the library. Results in Tab. I prove that the views collected by the proposed approach are just as usable as the views that we collected using our previous data acquisition procedure. The recognition results with the proposed approach were even a bit better, although this was caused by a relatively bad classification rate for one the object for which we used color texture segmentation to extract the views. Excluding this object, the recognition rates were almost identical.

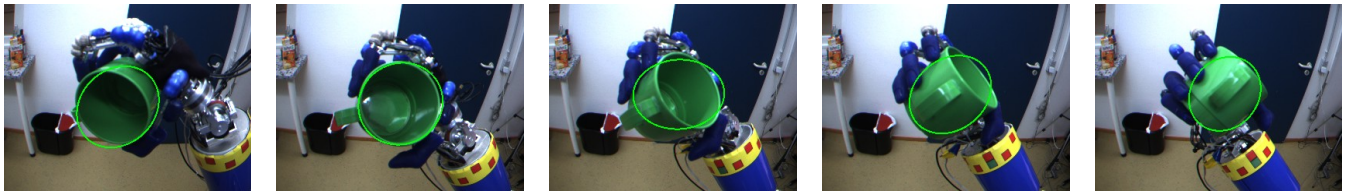


Fig. 8. Some of the collected snapshots of the robot's hand holding and rotating a green cup

V. Conclusions

The main result of this paper is the procedure for acquiring object views and for learning complete object representations for recognition by a humanoid robot without any prior knowledge about the objects and without manual tinkering with the images. Our experiments showed that the generated models are fully scale and rotationally invariant in 3-D and that we achieve comparable recognition rates on the proposed system as on the earlier system that used prior knowledge about the objects' color textures to discern their images from the rest of the scene.

We also proposed a systematic method to control a robot in order to observe an object across a continuous portion of the view sphere. The robot is actively controlled in the null space. Our experiments proved that by avoiding joint limits the scanned area on the view sphere was increased. Additionally, we have exploited the redundancy of a rotation about the camera ray axis. Using the proposed method, the configuration of the robot is much more appropriate for the observation task and we can achieve wider range of directions of view without regrasping the object.

The main goal of the future work is the integration of all primitive actions that are needed to achieve cognitive behavior of a robot in order to perform object discovering and learning. At the beginning the robot should find and pick an object in its surrounding using visual attention. Next, the robot should try to grasp the object. When the object is in the robot's hand, the proposed motion primitives for placing the object in the camera view and for rotation of the object should be performed. We shall explore how to acquire only the most views of the object instead of the statistical approach studied in this paper.

Acknowledgment

The work described in this paper was conducted within the EU Cognitive Systems project PACO-PLUS (FP6-2004-IST-4-027657).

A special thanks goes to the guys at the ITEC at the University of Karlsruhe for their help, time and advises.

References

[1] R. Bajcsy, "Active perception," *Proc. IEEE*, vol. 76, no. 8, pp. 996–1005, August 1988.

[2] D. H. Ballard, "Animate vision," *Artif. Intell.*, vol. 48, pp. 57–86, 1991.

[3] P. Fitzpatrick and G. Metta, "Grounding vision through experimental manipulation," *Philosophical Transactions of the Royal Society: Mathematical, Physical, and Engineering Sciences*, vol. 361, no. 1811, pp. 2165–2185, 2003.

[4] P. Fitzpatrick, "First contact: an active vision approach to segmentation," in *Proc. 2003 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, Nevada, 2003, pp. 2161–2166.

[5] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Trans. Robotics Automat.*, vol. 12, no. 5, pp. 651–670, 1996.

[6] D. N. Nenchev, "Redundancy resoulution through local optimization: A review," *The Journal of Robotic Systems*, vol. 6, no. 6, pp. 769–798, 1989.

[7] L. Žlajpah and B. Nemec, "Force strategies for on-line obstacle avoidance for redundant manipulators," *Robotica*, vol. 21, pp. 633–644, 2003.

[8] J. M. Hollerbach, "Redundancy resolution of manipulators through torque optimization," *Journal of Robotics and Automation*, vol. 3, no. 4, pp. 308–316, 1987.

[9] T. Yoshikawa, "Basic optimization methods of redundant manipulators," *Laboratory Robotics and Automation*, vol. 8, no. 1, pp. 49–60, 1996.

[10] E. Marchand, F. Chaumette, and A. Rizzo, "Using the task function approach to avoid robot joint limits and kinematic singularities in visual servoing," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Osaka, Japan, 1996, pp. 1083–1090.

[11] B. Nelson and P. Khosla, "Strategies for increasing the tracking region of an eye-in-hand system by singularity and joint limits avoidance," *Int. Journal of Robotics Research*, vol. 14, no. 3, pp. 255–269, 1995.

[12] T. Asfour, K. Regenstein, P. Azad, J. Schröder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, "ARMAR-III: An Integrated Humanoid Platform for Sensory-Motor Control," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids 2006)*, Genoa, Italy, December 2006.

[13] B. Nemec, L. Žlajpah, and D. Omrčen, "Comparison of null-space and minimal null-space control algorithms," *Robotica*, vol. 25, no. 5, pp. 511–520, 2007.

[14] A. Fusiello, E. Trucco, and A. Verri, "A compact algorithm for rectification of stereo pairs," *Machine Vision and Applications*, vol. 12, no. 1, pp. 16–22, 2000.

[15] E. Hayman and J.-O. Eklundh, "Statistical background subtraction for a mobile observer," in *Proc. IEEE. Int. Conf. Computer Vision*, Nice, France, 2003, pp. 67–74.

[16] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 5, pp. 564–577, 2003.

[17] S. J. McKenna, Y. Raja, and S. Gong, "Tracking colour objects using adaptive mixture models," *Image and Vision Computing*, vol. 17, pp. 225–231, 1999.

[18] A. Ude, C. Gaskett, and G. Cheng, "Support vector machines and Gabor kernels for object recognition on a humanoid with active foveated vision," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Sendai, Japan, 2004, pp. 668–673.