2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Redundant control of a humanoid robot head with foveated vision for object tracking

Damir Omrčen and Aleš Ude

Abstract— **This paper presents a novel approach to control a humanoid head for object tracking. The proposed approach is based on the concept of virtual mechanism, where the real head is enhanced with a virtual link that connects the eye with a point in 3-D space. We tested our implementation on a humanoid head with seven degrees of freedom and two rigidly connected cameras in each eye (wide-angle and telescopic). The experimental results show that the proposed control algorithm can be used to maintain the view of an observed object in the foveal (telescopic) image using information from the peripheral view. Unlike other methods proposed in the literature, our approach shows how to exploit the redundancy of the robot head. The proposed technique is systematic and can be easily implemented on different types of active humanoid heads. The results show good tracking performance regardless of the distance between the object and the head. Moreover, the uncertainties in the kinematic model of the head do not affect the performance of the system.**

I. Introduction

Robots with humanoid heads are not only more social, but they also posses an active vision system. Already in 1991, Ballard [1] described the implications of having a visual system that could actively position the camera coordinates in response to physical stimuli. Animate vision is much more challenging, however, it is also much more adaptable and flexible (e.g. using active vision wider field of view can be achieved).

Many visual tasks require both high resolution and a wide field of view. High acuity is needed for recognition tasks, while a wide field of view is needed for object detection, for tracking multiple objects, etc. A common trade-off found in biological systems is to sample part of the visual field at a high enough resolution to support the first set of tasks and to sample the rest of the field at an adequate level to support the second set. This is seen in animals with foveated vision, where the density of photoreceptors is highest at the center of the retina and falls off rapidly towards the periphery. Designers of a number of humanoid robots (Cog [2], [3], Kismet [4], Armar III [5], DB [6], etc.) attempted to mimic the foveated structure of the human eye by using two rigidly connected cameras in each eye. Here, the optical axes of the narrow- and wide-angle cameras are not aligned which makes the control problem more complex. In this paper we discuss the control of such humanoid visual systems and the associated 3-D vision processing.

D. Omrčen is with Jozef Stefan Institute, Slovenia damir.omrcen@ijs.si

A. Ude is with Jozef Stefan Institute, Slovenia and ATR Computational Neuroscience Laboratories, Japan ales.ude@ijs.si, aude@atr.jp

In [7] Laurutis et al. investigated human vestibulo-ocular reflex, which compensate for head movements during eye fixation. The vestibulo-ocular reflex has been later implemented in robotics by many authors [4], [8], [9].

Metta et al. use log-polar images for recognition and tracking of objects [10], [8]. To control the gaze direction they use learning methods, which define movements of the eye joints. Bernardino et al. [11] proposed a kinematic and dynamic controller, which is very simple since it decouples the kinematic relations of a robotic head. By decomposition in separate movements they achieved simplification of the sensorimotor process. The simplification results in common movement of both eyes pan, which is in most cases appropriate human-like movement of the eyes. However, that restricts the diversity of all possible movements.

Breazeal et al. [4] implemented object tracking on a Kismet humanoid robot. Here, narrow-angle cameras are in the robot eyes, while wide-angle cameras are fixed with respect to the head. Wide-angle cameras define position set-points for the eye motors. This transformation in general requires distance to the object, which is very noisy. Here, the crucial factor to achieve good transformation is also the distance (the transformation) between wide- and narrow-angle cameras. A similar approach was proposed in [12], where they used a well calibrated head to assure good tracking of object in space using wide- and narrow-angle cameras. Wide-angle cameras define 3-D position of the object to be tracked.

Ude et al. [13] proposed a simpler, decomposed controller, realized as a network of PD controllers. Since they use two cameras per eye, they additionally implemented a transformation, which assures that the object, which has to be tracked, is in the middle of the narrow-angle cameras, even though it is tracked by the wide-angle cameras. They proved that the transformation is valid only when the object is not too close to the cameras.

Vijayakumar et al. [14] proposed a resolved motion rate controller. The mapping from the 2-D image space to the 5-D joint space has been learned and has been done only for the right eye, while the left one only copies the same movement. The distance to the camera was neglected in the experiments.

Humanoid head is redundant with respect to the task of fixating on an object. A redundant manipulator has more DOFs than what is required to solve the task and is therefore more dexterous than a nonredundant. It has the ability to move in the joint space without affecting the motion in the task space. It is therefore beneficial to exploit the redundancy and not only to solve it by applying additional constraints. For example, when a robot neck is close to the joint limit the robot can use other joints (e.g. eyes) to accomplish the task (see joint limit avoidance in [15]). Similarly, we might optimize joint torques [16] and achieve lower energy consumption (or low fatigue in case of a human). Or we can simply apply different voluntary head motions such as nodding and still assure a stable gaze on the object.

The approach proposed in this paper demonstrates how to exploit robot's redundancy in such a way that the robot can perform additional tasks which are typical for humans. The other approaches proposed in the literature [11], [8], [10] only limit the variety of human movements by applying additional constraints.

II. METHODS

The initial goal of our system is to obtain high resolution images of an object in the robot's environment. Because the peripheral cameras have a very wide field of view, we cannot extract detailed object features from these images. Just like humans must fixate on an object to discriminate fine detail, our foveal cameras must be pointed in the direction of a given object in order to provide sufficient resolution.

To accurately track an object, we need to solve both perceptual and control problems:

- The perceptual problem deals with the estimation of location of a target object. We estimate 3-D position of the object based on kinematics of the robot head and stereo **wide**-angle camera image information.
- The control problem addresses the control of the head. It assures that an object is kept in the center of the **narrow**- or wide-angle cameras using the estimated target motion. To solve a control problem, we have introduced a virtual mechanism. Virtual mechanism is an auxiliary mechanism that points from both robot eyes to a 3-D point in space. It allows us to properly define a task (the gaze direction) and solve the control problem more systematically.

In this work we have used a humanoid head (Fig. 1) similar to the head used in Armar III humanoid robot [5]. The head has seven degrees of freedom (DOFs) and two eyes. The eyes have a common tilt and can pan independently. The visual system is mounted on a four DOFs neck mechanism, which is realized as a Pitch-Roll-Yaw-Pitch mechanism. Each eye is equipped with two digital color cameras (wide- and narrow-angle) to allow visuo-motor behaviors such as tracking and saccadic motions. The head features human-like characteristics in motion and response, that is, the neck and the eyes have a human-like speed and range of motion.

A. The perceptual problem - acquiring 3-D position of object

The perception problem deals with the problem of acquiring 3-D position of an object in space. The problem can be solved using two images from the wide-angle stereo cameras. To define the 3-D object position a good intrinsic and extrinsic camera models are crucial. Extrinsic camera parameters describe the position and orientation of the cameras in space.

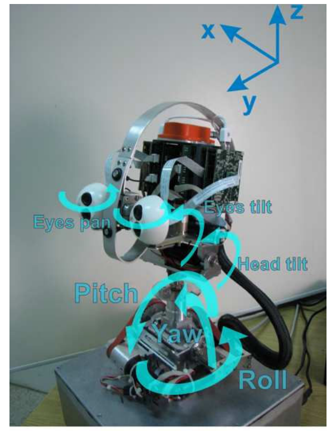

Fig. 1. Humanoid robot head

Having kinematic model of the head, we can determine the extrinsic parameters of the cameras. Here, we consider the poses of the wide-angle cameras since those are used for object detection. The accuracy of the head kinematics can be quite low on light weight humanoid robot head. The intrinsic camera parameters such as focal length, resolution, optical center etc., can be estimated using a chess board with a rather good accuracy.

Based on the assumption that the intrinsic and extrinsic camera parameters are known and that we are able to detect the position of an object in both camera images than the 3-D position of an object can be easily calculated using standard vision techniques [17].

B. The control problem - virtual mechanism approach

The main purpose of the virtual mechanism is the systematization of the task definition. That allows us to use the well-known control approaches to control the robot head.

Let us explain the virtual mechanism on a simple planar example. The task of the robot head is to keep the object in the center of the camera image. Fig. 2 shows an example where the eyes are turned toward the object. When the head moves the eyes have to change its orientation (φ) in order to keep the gaze on the object. So the robot task can be defined as the angle of the eye φ and is a function of the object position and also camera position. The position of the camera depends on the head configuration - which means that **the head configuration is involved in the task definition**:

$$\mathbf{task_{1\ DOF}} = \varphi = \mathbf{f}(\text{head conf., object pos.}).$$

The above specification of the problem is not the most common way to describe a task - in general, a task is not a function of the robot configuration. Therefore, it is very complex to implement well-known control approaches to control the robot head. To solve this problem in a more systematic way, we propose the use of a virtual mechanism. The purpose of the virtual mechanism approach is to define the task in a more systematic way, such that the task is not configuration dependent.

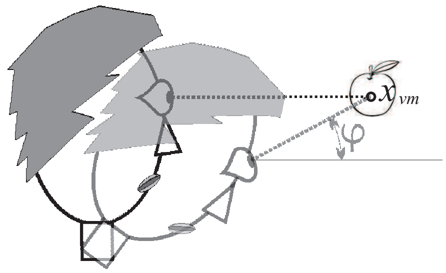

Fig. 2. Gaze direction changes during head movement. Virtual mechanism touches the object

Let us expand our humanoid head mechanism with an additional virtual link (mechanism). This virtual link can be treated as an additional prismatic joint which is fixed to the eye (see schematics in Fig. 3). When the eye moves the virtual link also moves. By adding the virtual link we add additional DOF to the system.

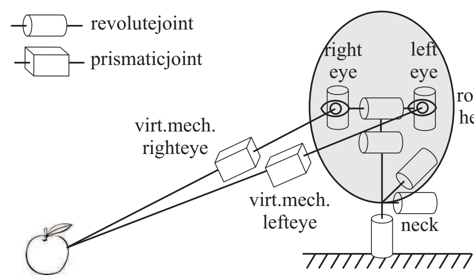

Fig. 3. Schematics of humanoid head with virtual mechanisms

The task of the system can now be reconfigured. Instead of controlling direction of each eye, we can now control the position x_{vm} of the end of the virtual link (see end of virtual link in Fig. 2). By adding the virtual prismatic link to the eye, we require that the end of the eye's virtual extension touches the object which is to be tracked. The task can now be defined as a simple position tracking problem and is not a function of the head position:

$$\mathbf{task_{2\ DOFs}} = \mathbf{x_{vm}} = \mathbf{f}(\text{object pos.}).$$

Note that the angle of the cameras still have to change during the head movement, the only advantage is the description of the task (x_{vm}) which does not change during the head motion.

In the original task definition we used 1 DOF (φ) to describe the task. By introducing the virtual mechanism we add one additional DOF to the system (the length of the virtual mechanism); however, we also add one DOF to the description of the task (2 DOFs for the $\boldsymbol{x}_{vm}$ position in plane). By introducing virtual mechanism the degree of redundancy remains the same.

In the spatial 3-D case circumstances are similar. In this case the task of the robot head is to control the orientation of both cameras and point them directly toward the object. Here, in the original task description the task had four DOFs (two camera angles per eye) and the robot had seven DOF. After the introduction of two virtual mechanisms (one per eye) the number of DOFs of the robot has been increased by two; however, the degree of the task has also been increased by two. The task of positioning requires three DOF per each eye. The introduction of the virtual mechanism increases the number of DOFs of the mechanism as well as of the task. The degree of redundancy is the same in both task definitions. The main purpose of the virtual mechanism is that the description of the task is simplified and systematized - instead of specifying the desired pointing direction, we can consider the problem as a classic inverse kinematics task, which can be solved by classic control approaches.

C. Controller

Having additional virtual mechanism in each eye the kinematics of the head is given by the following form:

$$\boldsymbol{x}_{vm} = \begin{bmatrix} \boldsymbol{x}_{vmL} \\ \boldsymbol{x}_{vmR} \end{bmatrix} = f(\boldsymbol{q}_{head}, l_{virt.m.}), \quad (1)$$

where $\boldsymbol{x}_{vm()}$ denotes the position of the end of the virtual link of the left and right eye, while $\boldsymbol{q}_{head}$ and $l_{virt.m.}$ denote the head joint angles and the lengths of the virtual mechanisms. To simplify the notation we treat the lengths of the virtual mechanisms as additional joint variables, such that $\boldsymbol{q} = [\boldsymbol{q}_{head}, l_{virt.m.}]$. The above kinematic model is defined for the narrow-angle cameras since those are used for object tracking.

The relation between joint and task velocities is given by the robot Jacobian $\mathbf{J}$:

$$\dot{\boldsymbol{x}}_{vm} = \mathbf{J}\dot{\boldsymbol{q}}. \quad (2)$$

As already stated, the head has more DOFs than needed to accomplish the given task. To achieve a good tracking performance while exploiting the redundancy, the following velocity controller can be applied:

$$\dot{\boldsymbol{q}}_c = \mathbf{J}^{\#}\dot{\boldsymbol{x}}_{vm_c} + \mathbf{N}\dot{\boldsymbol{q}}_n, \quad (3)$$

where $\dot{\boldsymbol{q}}_c$ denotes the vector of joint velocities, $\mathbf{J}^{\#}$ is the weighted generalized inverse of the Jacobian matrix $\mathbf{J}$, $\dot{\boldsymbol{x}}_{vm_c}$ is the desired velocity in the task space, $\mathbf{N}$ is the projection onto the null space of $\mathbf{J}$, and $\dot{\boldsymbol{q}}_n$ is the desired joint velocity in the null space. The product $\mathbf{J}^{\#}\dot{\boldsymbol{x}}_{vm_c}$ represents the joint velocities due to the task space motion and $\mathbf{N}\dot{\boldsymbol{q}}_n$ represents the joint velocities of the null space motion. $\mathbf{J}^{\#}$ and $\mathbf{N}$ can be defined as follows:

$$\mathbf{J}^{\#} = \mathbf{W}^{-1}\mathbf{J}^T(\mathbf{J}\mathbf{W}^{-1}\mathbf{J}^T)^{-1}, \qquad \mathbf{N} = \mathbf{I} - \mathbf{J}^{\#}\mathbf{J}, \quad (4)$$

where $\mathbf{W}$ is a weighting matrix.

To control the position of the virtual link, the following $\dot{\boldsymbol{x}}_{vm_c}$ controller is proposed:

$$\dot{\boldsymbol{x}}_{vm_c} = \begin{bmatrix} \dot{\mathbf{r}} + K_p\boldsymbol{e}_L \\ \dot{\mathbf{r}} + K_p\boldsymbol{e}_R \end{bmatrix},$$

where $\boldsymbol{e}_{()}$, $\boldsymbol{e}_{()} = \mathbf{r} - \boldsymbol{x}_{vm()}$, is the task space tracking error for the corresponding eye and $\mathbf{r}$ is the desired task space position, i.e. the position of the object that has to be tracked and is acquired by the wide-angle cameras as indicated in Section II-A.

The topology of the robot head is a tree-like structure with one main branch and two subbranches (see Fig. 3). The main branch refers to the joints and links which are common to both parts of the system. Those are the neck joints and the eye tilt. On top of that, there are joints that are specific for each subbranch of the system. Those are the eye pans and the lengths of the virtual mechanisms. The most natural way to control a branching mechanism is to treat both branches equally so that the movement of the common DOFs is defined by the tasks of both sub-branches equally, while the motion of a particular branch only depends on a particular task. To control a branching mechanism we propose the branching mechanism controller presented in [18].

III. Results

The proposed approach has been verified on the real humanoid head described in section II and shown in Fig. 1. Each robot eye has two cameras where the wide-angle camera is placed above the narrow-angle one. The vertical distance between them is 18 mm and the resolution of all cameras is 640×480. The intrinsic camera parameters and transformations between cameras were calculated on a set of chess board snapshots. The robot head has been controlled with Matlab/Simulink via UDP connection. Sample time of the control loop has been set to 100 Hz, while frame-rate of the visual processing has been set to 60 Hz. For object localization and tracking in camera images we used color segmentation.

We performed different experiments to demonstrate the redundancy exploitation on the robot head. In the first experiment we applied a sinusoidal motion in the neck pitch joint, while the head was instructed to keep the contact with an object. In the second experiment we demonstrated joint limits avoidance.

A. Object detecting with wide-angle cameras and tracking with narrow-angle cameras

Many visual tasks require both high resolution and a wide field of view. High acuity is needed for recognition tasks and a wide field of view is needed for object detection. As explained in the introduction, our system models the foveated structure of biological vision systems by having two cameras in each eye, i. e. a narrow-angle foveal camera and a wide-angle camera for peripheral vision. Snapshots from both cameras are shown in Fig. 4.

Since the optical axes of wide- and narrow-angle cameras are not aligned (the wide-angle camera is placed above the narrow-angle one), it is not trivial to assure that the object acquired by the wide-angle cameras is in the center of the narrow-angle cameras. Here, the distance to the object is crucial. In the proposed approach this is done systematically only by using the appropriate kinematic model in the control part. The next table shows that the object is in the center of the narrow-angle cameras regardless of the distance of the object from the cameras. Here the center of the image is at location (320, 240):

Obj.-cam. dist. (cm)	30	50	100	200	400
Horiz. obj. pos.	246	245	242	247	259
Vert. obj. pos.	317	316	327	311	318

a) Wide-angle image b) Narrow-angle image

Fig. 4. Snapshots of wide- and narrow-angle cameras

B. Applying neck motion while exploiting redundancy

It is beneficial to exploit the redundancy of the robot head instead of only solving it. The robot has to be able to move the head while keeping the eye contact with an object. Since the proposed virtual mechanism approach is very systematic we can use a great number of control approaches and optimization strategies proposed in the literature.

In order to demonstrate how to exploit the redundancy with the proposed control algorithm, we show an example where the head moves forward and backward (nodding) while keeping eye contact with an object. The null space motion is, therefore, defined in such a way that the neck pitch tracks a simple sinusoidal motion as follows:

$$\dot{q}_{n_{\text{pitch}}} = K_n(A * sin(\omega t) - q_{\text{pitch}_{\text{act}}}),$$

while the velocities for the other joints are equal to zero: $\dot{q}_{n_i} = 0, \forall i \neq pitch$.

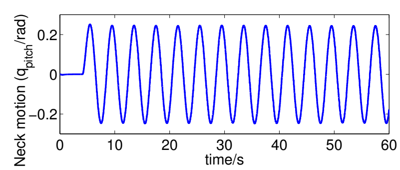

Fig. 5. The motion in the head neck pitch joint during object tracking

We compared two different controllers. In the first case the neck motion is a disturbance to the controller. The pitch motion is applied directly to the robot head, while this information is not provided to the controller. Controller can not consider the neck motion until it produces the task space error. In the second case the redundancy is exploited and the neck motion is applied as a secondary task. Here, the neck motion does not affect the primary task of object tracking - that means it does not produce any task space error. Here, the neck motion is compensated by the motion of other joints as defined in Eq. (3). Consequently, the task space error is significantly smaller. Fig. 5 shows the motion in the neck pitch for both cases. Figs. 6 and 7 shows the position of the object in both cameras during the sinusoidal neck movement. When exploiting redundancy the tracking results are significantly better. The neck pitch motion is compensated by the other head joints by exploiting redundancy of the head. Here, very basic redundant control algorithms have been used as shown above.

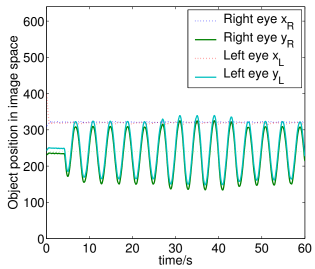

Fig. 6. Object position in camera images. Motion is considered as a disturbance.

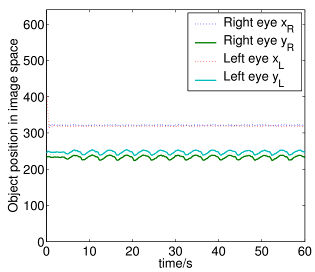

Fig. 7. Object position in camera images. The redundancy of the head is exploited and the object is kept in the image center despite the neck motion.

Fig. 8 shows the sequences of robot head during neck motion while observing a colored object. It can be seen that the head keeps the contact with the object despite large neck motion.

C. Joint limit avoidance by exploiting redundancy

In this test we have realized joint limit avoidance to demonstrate how simple is to implement different control strategies on humanoid head using the virtual mechanism.

We compared two controllers. The first controller did not consider any joint limits. During head movement the head got into the joint limit which resulted in significant tracking error. The second controller considered the joint limit. The null space term was set in such a way that the null space velocity retracts the head away from the limits. For the i-th joint the null space velocity has been defined in the following way:

$$\dot{q}_{n_i} = \begin{cases} 0, & |q_{\text{actual}_i}| < 0.9 q_{\text{jntlim}_i} \\ -K_a(q_{\text{actual}_i} - q_{\text{jntlim}_i}), & \text{otherwise,} \end{cases} ,$$

where q_{jntlim_i} is the i-th joint limit and K_a is a gain.

To compare both experiments an industrial robot was used to move the object in an reproducible way. It moved the object up and down 1000 mm in front of the robot head on a sinusoidal trajectory with the amplitude of 1000 mm. The joint limit in the head tilt joint was set to 0.2 rad.

Fig. 9 shows the position of the object in camera image for both experiments. It can be seen that the tracking error is significantly higher when the neck tilt joint gets into the joint limit. Next two figures 10 and 11 show the motion in the most significant joints. We can see in Fig. 11 that when the neck tilt joint is close to the limits the joint limits avoidance starts and the other joints start to move more in order to compensate for the lack in the neck tilt motion. Therefore, the joint does not get into the limit, while in the case of Fig. 10 it does.

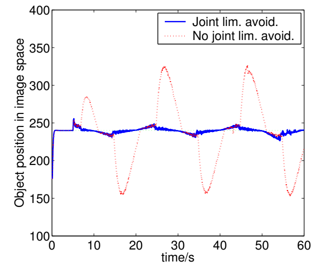

Fig. 9. Object position in camera images during object tracking.

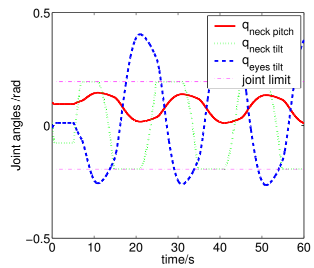

Fig. 10. Joint angle motion during object tracking - without considering joint limits.

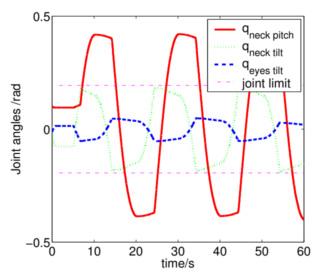

Fig. 11. Joint angle motion during object tracking - joint limits are avoided by exploiting redundancy.

D. Accompanying video description

The accompanied video contains two experiments. In the first experiment object tracking with the narrow-angle cameras is shown. Video shows the movement of the head and the images from the narrow- and wide-angle camera.

In the second experiment null space motion of the head is shown while keeping contact with an object. Here, we applied additional null space motion in the three neck joints. As the input device we have used Nintendo Wii remote

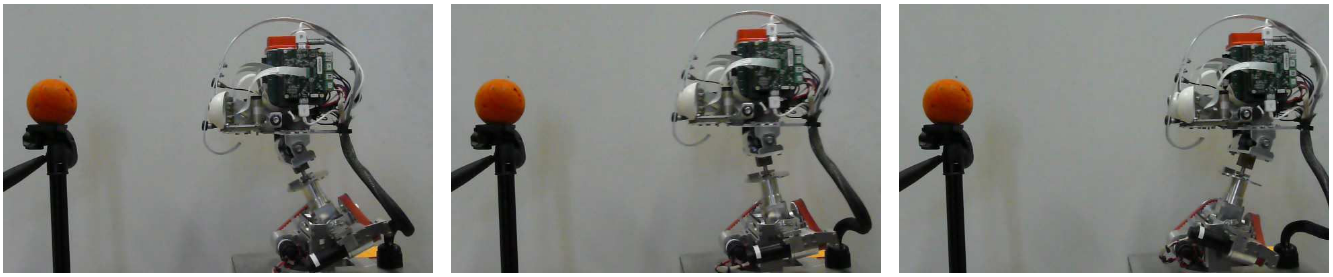

Fig. 8. Sequences of head motion during neck motion while observing a colored object

(Wiimote). The Wiimote has been interfaced with the control computer trough bluetooth [19].

IV. Conclusions

We presented an approach for controlling the gaze direction of a humanoid head. The humanoid robot head is equipped with two cameras per eye. The proposed approach uses wide-angle cameras to acquire 3-D position of an object in space. This information is later used in order to bring and keep an object in the center of the narrow-angle images regardless of the distance of the object from the eyes. To achieve that we introduced a virtual mechanism, which is the main contribution of the paper.

The introduction of virtual mechanism simplifies the description of the task. This allows us to use standard control approaches. In the paper we have shown how to exploit redundancy of the robot head and by that how to achieve more diverse head motion and better tracking while avoiding joint limits.

There are many advantages of the proposed approach over the others described in the literature and are listed below:

- Comparing to the decomposed controllers that control each joint individually without considering the head kinematics (e.g. [13]), the proposed controller results in a more optimal head motion.
- Getting an object to the center of narrow-angle images even if it is acqured by the wide-angle cameras is simplified and does not depend on the precise placement of the cameras or on the distance of the object from the cameras.
- The proposed approach indicates how to exploit the redundancy of the head instead of only solving it.
- When using controllers that rely on image-based visual servoing it is more complex to achieve that the desired object is in the center of the narrow-angle images, since the object is sometimes visible only in wide angle images.
- It is a very systematical approach and can be easily implemented on different types of active stereo vision mechanisms.

Acknowledgment: The work described in this paper was partially conducted within the EU Cognitive Systems project PACO-PLUS (FP6-2004-IST-4-027657) funded by the European Commission.

References

[1] D. H. Ballard. Animate vision. *Artif. Intell.*, 48:57–86, 1991.

[2] B. Scassellati. Eye finding via face detection for a foveated, active vision system. In *15-th AAAI/IAAI conference*, pages 969 – 976, Madison, USA, 1998.

[3] Junji Yamato. *Tracking moving object by Stereo Vision Head with Vergence for Humanoid Robot, Master's thesis*. Dept. of Elect. Eng. and Comp. Sci., MIT, 1998.

[4] C. Breazeal, A. Edsinger, P. Fitzpatrick, and B. Scassellati. Active vision for sociable robots. *IEEE Tran. on Systems Man and Cybernetics-Part A: Systems and Humans*, 31(5):443–453, 2001.

[5] 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 Int. Conf. on Humanoid Robots (Humanoids 2006)*, Genoa, Italy, 2006.

[6] C. G. Atkeson, J. G. Hale, F. Pollick, M. Riley, S. Kotosaka, S. Schaal, T. Shibata, G. Tevatia, A. Ude, S. Vijayakumar, and M. Kawato. Using humanoid robots to study human behavior. *IEEE Intelligent Systems*, 15(4):4656, 2000.

[7] V.P. Laurutis and D.A. Robinson. The vestibulo-ocular reflex during human saccadic eye movements. *Journal of Physiology*, (373):209–233, 1986.

[8] F. Panerai, G. Metta, and G. Sandini. Visuo-inertial stabilization in space-variant binocular systems. *Robotics and Autonomous Systems*, 30(1-2):195214, 2000.

[9] A. Takanishi, T. Matsuno, and I. Kato. "development of an anthropomophic head-eye robot with two eyes - coodinated head-eye motion and pursuing motion in the depth direction-. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 799–804, Grenoble, France, 1997.

[10] G. Metta, A. Gasteratos, and G. Sandini. Learning to track colored objects with log-polar vision. *Mechatronics*, 14:9891006, 2004.

[11] A. Bernardino and J. Santos-Victor. Binocular visual tracking: Integration of perception and control. *IEEE Transactions on Robotics and Automation*, 15(6):1080–1094, 1999.

[12] C.-Y. Tang, Y.-P. Hung, and Z. Chen. *Automatic Detection and Tracking of Human Heads Using an Active Stereo Vision System*, volume 1351/1997, pages 632 – 639. Source Lecture Notes In Computer Science, Springer Berlin / Heidelberg, 1998.

[13] A. Ude, C. Gaskett, and G. Cheng. Foveated vision systems with two cameras per eye. In *Proc. IEEE Int. Conf. Robotics and Automation*, Orlando, USA, 2006.

[14] S. Vijayakumar, J. Conradt, T. Shibata, and S. Schaal. Overt visual attention for a humanoid robot. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, Hawaii, USA, 2001.

[15] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5(1):90–98, 1986.

[16] J. M. Hollerbach. Redundancy resolution of manipulators through torque optimization. *Journal of Robotics and Automation*, 3(4):308–316, 1987.

[17] Emanuele Trucco and Alessandro Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.

[18] K.-S. Chang and O. Khatib. Operational space dynamics: Efficient algorithms for modeling and control of branching mechanisms. In *International Conference on Robotics and Automation (ICRA 2000)*, pages 850–856, San Francisco, CA, 2000.

[19] A. Gams, A. J. Ijspeert, S. Schaal, and J. Lenarčič. On-line learning and modulation of periodic movements with nonlinear dynamical systems. *Autonomous Robots*, 27(1):3–23, 2009.