Autonomous View Selection and Gaze Stabilization for Humanoid Robots

Markus Grotz[1], Timothée Habra[2], Renaud Ronsse[2], and Tamim Asfour[1]

***Abstract*— To increase the autonomy of humanoid robots, the visual perception must support the efficient collection and interpretation of visual scene cues by providing task-dependent information. Active vision systems allow to extend the observable workspace by employing active gaze control, i.e. by shifting the gaze to relevant areas in the scene. When moving the eyes, stabilization of the camera images is crucial for successful task execution. In this paper, we present an active vision system for task-oriented selection of view directions and gaze stabilization to enable a humanoid robot to robustly perform vision-based tasks. We investigate the interaction between a gaze stabilization controller and view planning to select the next best view direction based on saliency maps which encode task-relevant information. We demonstrate the performance of the systems in a real world scenario, in which a humanoid robot is performing vision-based grasping while moving, a task that would not be possible without the combination of view selection and gaze stabilization.**

I. Introduction

Visual perception is a crucial source of information for many essential robotic tasks like grasping and manipulation, as well as for navigation and motion planning. To increase the robustness of vision-based tasks, a vision system should support the sufficient collection and efficient interpretation of visual scene cues by providing task-dependent information. Active vision and active head-eye systems of humanoid robots, like ARMAR-III [1], allow to extend the observable workspace by employing active gaze control, i.e. by shifting the gaze to relevant areas in the scene. The basic idea of active vision rests upon the integration of action and control strategies for a moving camera system, in order to improve perception [2], [3]. In contrast to passive camera systems where an increase in the field of view leads to a loss in the resolution of details, active systems allow to focus on the observable area and thereby to keep the details. This is even more beneficial for systems with foveal vision, i.e. with higher resolution in the center of the field of view. Such a behavior is desirable during locomotion and manipulation, especially if the task involves multiple spatially distributed objects. In cognitive science, eye movements performed to bring relevant parts of the environment into the field of view is coined *overt visual attention*. An extensive review on visual attention approaches is given in [4].

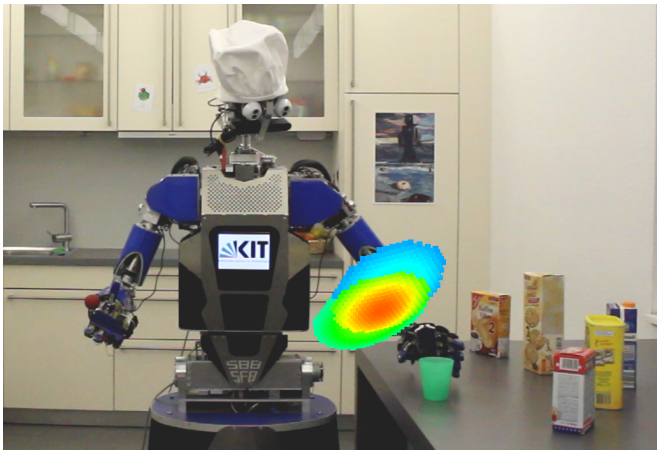

Fig. 1: ARMAR-III grasping a cup while the platform is moving at a constant speed. Parts of the sensory ego-sphere during the grasping while moving experiments are visually overlayed. The sensory ego-sphere consists of 40,000 nodes each representing a possible view direction. The color scheme is a heat map ranging from blue to red for nodes with low or high saliency. The computed view target is used by the gaze stabilization component to stabilize the camera image.

Humanoid vision systems such as in [1] and [5] realize foveation using two cameras in each eye, a wide angle camera for peripheral vision and a narrow angle one for foveal vision. Mimicking human vision, it allows to monitor the environment with the peripheral view and to deeper analyse objects of interest with foveal view. Active vision systems can simultaneously use peripheral and foveal vision to bring the object into the center of the fovea based on information from the peripheral cameras. This is necessary because the area of interest, e.g. an object that is tracked by the robot, can easily be lost from the fovea due to its narrow field of view. It is much less likely that the object would be lost from peripheral images that have a wider field of view.

The fact that robots (and humans) usually move while they perform tasks potentially affects their visual perception. When grasping or executing other manipulation tasks (see Fig. 1), the body movement affects the current camera position. Similar effects can be observed during locomotion, both in bipedal walking of a humanoid robot or when moving using a mobile platform. Furthermore, external disturbances

[1] Institute for Anthropomatics and Robotics (IAR), High Performance Humanoid Technologies Lab (H^2T), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany. markus.grotz@kit.edu

[2] Center for Research in Mechatronics, Institute of Mechanics, Materials, and Civil Engineering, Université catholique de Louvain (UCL), Louvain-la-Neuve, Belgium.

The research leading to these results has received funding from the European Union Seventh Framework Programme under grant agreement no 611832 (WALK-MAN).

may additionally influence the pose of the head and thereby the cameras. For instance, a sudden external push on a humanoid robot would lead to blurred camera images and changed field of view. Therefore, stable camera images are of utmost importance for object detection, self-localization and path planning. Hence, gaze stabilization mechanisms are required to allow executing such tasks in a robust way.

In this paper, we investigate the interaction between a gaze stabilization controller and a view planning strategy to select the next best view point for executing visual and grasping tasks on a humanoid robot. Both components support independently the robot's visual perception system and are crucial for a robust operation in real scenarios. In order to avoid a conflict of interest when controlling the robot, the two components need to be orchestrated carefully. For example, an active vision system itself generates movements of the head and thereby inducing noise to the camera position, that needs to be compensated by gaze stabilization.

We propose an architecture that includes both gaze stabilization and a task-specific gaze selection and considers the interaction between these two components. Overall, the main contributions are twofold. 1) We integrate two standalone components, i.e. a gaze stabilization controller and an active vision method for view planning, to interact and work autonomously in real-time. 2) We create a complex real world scenario with the humanoid robot ARMAR-III that demonstrates the integration of active vision and gaze stabilization. We show that gaze stabilization enables to fully exploit foveal cameras during locomotion.

The remainder of this paper is structured as follows. An overview of related work is given in section II, followed by a description of the proposed gaze stabilization system and active vision method (section III). The evaluation of the approach, based on several experiments performed with the ARMAR-III humanoid, is described in section V. In section VI, we conclude with a summary, discussion of results and future work.

II. Related Work

The state of the art in the most relevant areas can be divided into two categories: i) active vision and ii) gaze stabilization. The first category deals with the question of *how*, *when*, and *where* to perceive things visually, while the second area focuses on stabilizing the camera position.

A. Active Vision

The term *active vision* was coined by Aloimonos et al. [2] in the late 1980s. Active vision means that the camera view point is modified actively and purposefully with the goal to enhance the current perception. A similar definition for *active perception* was given by Bajcsy et al. [3]. A recent overview on the topic is given in [6]. For a survey on active vision methods, the reader is referred to [7]. In this work, however, we mainly consider active vision systems used on humanoid robots that utilize both foveal and wide cameras.

Ude et al. integrated foveal and peripheral vision to track moving objects [8]. Once a new area of interest is selected the robot directs its gaze towards it and the object is subjected to a more detailed analysis. Similar work was studied by Omrčen et al. [9]. The authors realized an object tracking controller for a Karlsruhe Humanoid Head [5] using a virtual joint. In [10], the authors exploit the properties of an active humanoid vision system to construct an effective object recognition system, where wide angle views were used to search for objects, direct the gaze towards them and keep them in the center of narrow-angle views.

Rasolzadeh et al. [11] present a visual attention system that is able to interact with the environment. The perceptual component is designed for the Karlsruhe Humanoid Head [5] and is processing information from both wide angle and foveal cameras in order to determine the next focus point. Regions of interest for the visual attention component are computed using a bottom-up and top-down saliency map. To learn the optimal bias of the top-down saliency map, an artificial neural network approach is used. After the combination of the saliency maps the final view point is selected by using a stochastic *winner takes all* approach. Furthermore, their pipeline also supports segmentation, detection, and grasping/manipulation.

In our earlier work, we proposed a view selection system tailored to manipulation tasks that considers incomplete or inaccurate world knowledge. It decides for optimal view directions leading to an overall reduction of localization uncertainty [12]. The saliency is based on the uncertainty of the pose of localized objects that are relevant for the current task, and the respectively required acuity. However, the previous mentioned work did not explicitly take gaze stabilization into account. Therefore, the proposed view selection could not be used during locomotion. A common approach is to discard data during locomotion or during the execution of saccade eye movements Besides that, it is also acknowledged in the literature that images on the cameras need to be stabilized [13].

B. Gaze Stabilization

Gaze stabilization methods are often inspired by human eye stabilization strategies replicating human eye reflexes. This includes, for example, an integrated eye and head stabilization framework for the iCub platform inspired by cerebellar theories [14]. Other methods, compensating for self-induced perturbations only, rely purely on kinematics information [15], [16]. The idea behind these methods is to intercept the motor commands which are then applied to a simulated robot model in order to predict and correct the next head position. New motor commands are then generated for the head to keep the visual frame stable.

Habra et al. propose a feed-forward gaze stabilization controller [16] based on copies of motor commands. The inverse Jacobian defined by the gaze stabilization controller is relaxed by minimizing the optical flow. Furthermore, a fast method to approximate the optical flow using the robot's kinematics is derived. The approach was recently evaluated in simulation using the active head of the humanoid robot ARMAR-4. Roncone et al. designed a gaze control

architecture allowing head stabilization and object tracking by executing saccadic eye movments on the iCub robot [17]. Nonetheless, the system only allows for a single object to be tracked and does not support attention shifts based on the task acuity, which are required for a more complex scenario like grasping.

III. Methods

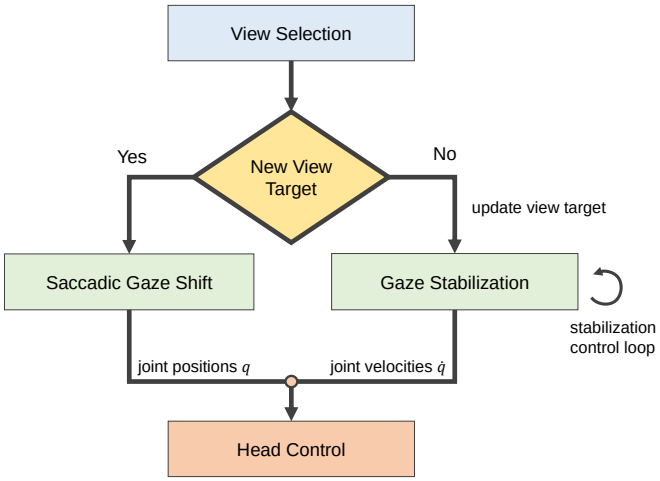

Fig. 2: The workflow of the active vision method and gaze stabilization system. The gaze stabilization constantly fixates a given view target provided by the view selection.

In this section, we describe our methods for the active vision system and the gaze stabilization controller. Based on different saliency maps the view selection computes a view target x, which is then fixated by the gaze stabilization controller. The workflow between the two components is outlined in Fig. 2.

A. View Selection

Similar to the work in [18], the view selection is realized using a *sensory ego-sphere* (SES) representation, which serves as a short-term memory system of the robot and allows the fusion of sensory data. In this work, the SES is used as an egocentric saliency map in the form of a sphere centered at the robot's head, which represents the possible view directions. The rotation around the view axis is ignored in this representation, as it does not significantly influence what is visible in the camera images. The sensory ego-sphere is discretized to 40,000 equally distributed points, each of which is annotated with a value that indicates the overall saliency for this particular view point. The intuition is that this saliency should correspond to the importance of the information that can be gained if the robot were to look into that direction.

In our previous work [12], we focused on gaze selection during manipulation tasks. For completeness, we describe the modeling of the object saliency map briefly. Given an object o_i that the robot needs to localize, the respective uncertainty about the object pose is modeled as a Gaussian distribution with the covariance matrix Σ_i. This uncertainty is translated

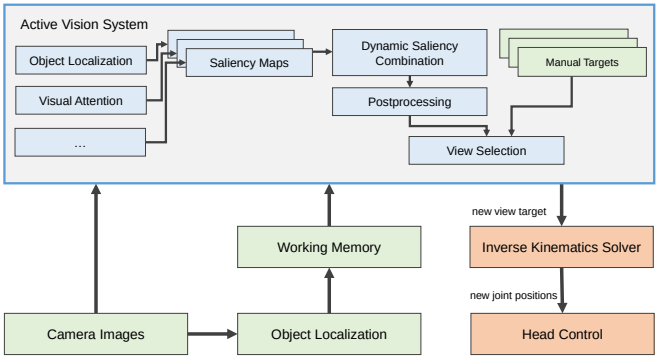

Fig. 3: The underlying attention model. Multiple saliency maps are computed and aggregated into a single sensory ego-sphere. View directions are post-processed and unreachable positions are discarded. The region with the highest saliency defines the next best view direction.

to the scalar value $\sigma_i = \|\Sigma_i\|^{\frac{1}{6}}$, which equals the radius of a sphere with the same volume as the ellipsoid spanned by Σ_i . The object pose and the uncertainty are stored in the *working memory*, which is part of the ArmarX memory system. New object localization results are fused with a Kalman filter. The saliency of object o_i indicates the potential information gain given its pose uncertainty, and is thus set to equal to the differential entropy

$$u_i = \frac{1}{2} \ln \left(\left(2\pi e {\sigma_i}^2 \right)^3 \right), \tag{1}$$

where σ_i is a scalar value to represent the uncertainty of the object localization result. In order to include this task-specific guidance in the gaze selection strategy, we introduce the task acuity in the calculation of object saliencies. To incorporate the localization acuity α_i required by the task, the differential entropy arising from this tolerated level of uncertainty can be calculated as follows:

$$b_i = \frac{1}{2} \ln \left(\left(2\pi e {\alpha_i}^2 \right)^3 \right). \tag{2}$$

The resulting saliency for an object o_i is then $s_i = u_i - b_i$. To avoid the accumulation of negative saliency values s_i is enforced to be ≥ 0. For each view direction with an object visible in the field of view s_i is added to the corresponding point p_j on the *sensory ego-sphere*. A point p_j on the sphere can contain multiple saliency values. An example is illustrated in Fig. 1. Here, the hand of the robot and the green cup need to be localized simultaneously. In the next step, a very small random noise is added to the saliency map in order to get variance in the view directions, which increases the robustness of the localization results that are integrated using a Kalman filter. It also makes the view selection more robust when objects move slightly while the robot is looking in another direction.

Other cues that should attract the robot's attention can be added to the *sensory ego-sphere* in the form of a saliency map. In this context, we use two different saliency maps. The first one models the object localization uncertainty as described in this section, while the other draws the attention

to single colored blobs in the scene. However, there is no limitation on adding different saliency cues with weights that depend on their importance for the current task. To accumulate different saliency maps a weighted sum is used to take different saliency measures into account. This yields to the saliency value of point p_j of the aggregated sensory ego-sphere

$$p_j = \frac{1}{\sum w_k} \sum_{k=1}^{N} w_k p_j^k, \tag{3}$$

where w_k is the weight of the saliency map k and p_j^k represents the saliency value corresponding to the j-point of the k-th map. Furthermore, a timestamp is added to each individual saliency map to discard outdated values.

Finally, the saliencies are post-processed; directions that are not reachable due to kinematic limitations are set to zero. Those directions, that are close to the current view point, are slightly preferred to discourage larger head motions unless necessary. The direction with maximal saliency is selected, a solution of the inverse kinematics problem is computed and the head and the eyes are moved accordingly. For this step we utilize the inverse kinematics solver available in the Simox library [19]. Fig. 3 visualizes the saliency map combination and view target selection as described in this section.

B. Gaze Stabilization

The gaze stabilization strategy used in this work is based on the work in [16] and was adapted for use on the ARMAR-III humanoid robot. Since the focus of this work is not the gaze stabilization *per se* but rather the higher level architecture combining active vision and gaze control, we give only a brief overview of the gaze stabilization and highlight the main adaptations for the current work.

The method is based on a task space formulation of the stabilization problem. Here, the task state is defined as the position of the fixation point x, i.e. a point in space where the robot is gazing at. Inspired from the work of Omerčen and Ude, it is build on top of a virtual model [9]. The fixation point can then be seen as a virtual end-effector. This offers to elegantly reformulate the gaze stabilization as the classical control of a serial robot manipulator. The stabilization can then be resolved using the so-called closed loop inverse kinematics method [20]. In [16], the virtual model from [9] is extended with two revolute joints offering to resolve the redundant kinematic problem based on an optical flow minimization criterion. Typically, the desired state velocity $\dot{x}_{des}$ is computed as the sum of 1) feedback of the position error and 2) feed-forward velocity $\dot{x}_{FF}$ as represented in Fig. 4. For implementation details, the reader is referred to [16].

In [16], the term $\dot{x}_{FF}$ is computed as a compensation for the self-induced velocity, which is estimated from the velocity commands sent to the whole-body joints. In this contribution, instead of the velocity commands, we use direct velocity measurements (from the encoder) to compute the feed-forward term. Also, the feed-forward compensation was

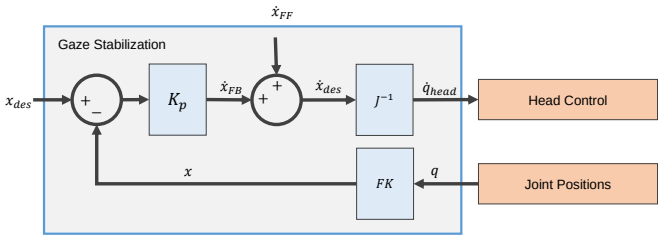

Fig. 4: Inverse kinematics methods for gaze stabilization. Based on the current joint velocities compensatory head movements are computed to stabilize a given fixation point. Figure adapted from [16].

extended with the pose estimation module of the omnidirectional platform of the robot, which uses 2D laser scanner data for self-localization. This permits to compensate all the robot motion in space. Finally, the kinematic redundancy resolution was adapted in order to consider the difference between the eye and the neck joints for active vision. Indeed, the stereo-vision algorithm relies on a calibration of the left and right cameras pose. Thus, moving the eyes affects the active vision much more than moving the neck joints. Therefore, velocity minimization of the eye joints were given a weight factor eight times larger than for the neck joints.

IV. System Architecture

The proposed solution is fully integrated into the robot development environment ArmarX [21]. The framework consists of different layers. It abstracts a robot's hardware and functionalities and provides support for distributed applications. Additionally, ArmarX features a sophisticated memory structure, including a *working memory* and a *prior memory*. The working memory fuses the object localization results

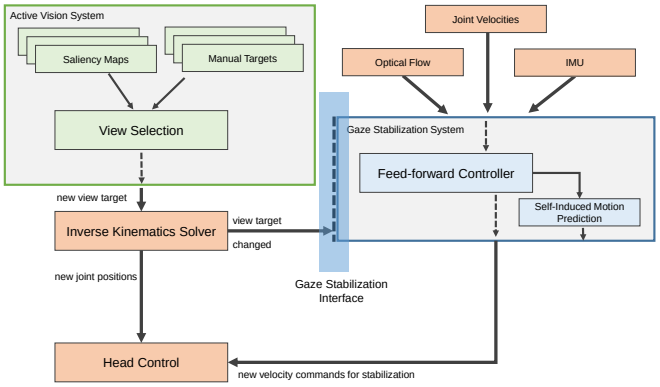

Fig. 5: The system architecture combining gaze stabilization and active vision. Different saliency maps are aggregated by the view selection component to generate view targets while the gaze stabilization component computes motor commands to fixate and stabilize the current view target. The active vision system passes the new view target to the gaze stabilization controller via a common interface. The gaze stabilization controller supports the active vision system by providing stable images.

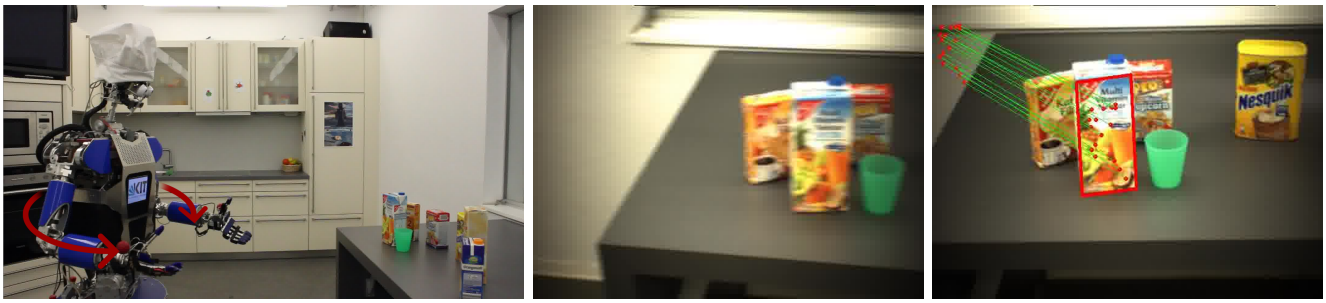

Fig. 6: Setup for the object localization experiment (left). The torso of the humanoid robot ARMAR-III is subjected to a sinusoidal perturbation while the objects are constantly localized. The image of the left foveal camera is shown for both the unstabilized case (center) and the stabilized case (right). In addition, the successful object localization result is visible in the stabilized case.

and further enriches them with additional information from the prior memory. In the context of this work, the object localization is provided by methods described in [22] and [23].

From a software engineering point of view, both systems, i.e. gaze stabilization and view selection, have been developed as standalone components with a well-defined interface to allow for interaction between these components. The gaze stabilization indirectly feedbacks the view selection by providing more stable input images. The system architecture is shown in Fig. 5.

First, saliency maps for object localization are computed and used as the input of the view selection component. Saliency maps are then aggregated into a single sensory ego-sphere, as described in section III-A. View directions are computed in regular intervals (currently every $2.0\,\mathrm{s}$) or when interrupted by an external event, e.g. when a new view target is added based on input from a higher level component. Such an event can be triggered with the statechart system of ArmarX [24]. Once a new view point x is computed, the gaze stabilization system is triggered to allow the execution of saccadic eye movements and support visual perception by stabilizing the new view point. Saliency maps for object localization are computed in an external application and passed over to the view selection component by using a shared interface.

In order to control the eye-head system of the robot, the gaze stabilization component is sending velocity commands to the hardware abstraction layer of the robot. As input the gaze stabilization controller uses the joint velocities, optical flow and the IMU gyroscope values. Several reflexes for gaze stabilization can be activated. In this work, however, we neglect external perturbations and only consider self-induced perturbations. Therefore, we designed the controller to work only with a reflex based on kinematics as described in section III-B. Besides stabilizing, the gaze stabilization component further predicts the self-induced optical flow and the expected angular velocity of the head. Details regarding the gaze stabilization are described in [25].

V. Experimental Results

All experiments were performed using the humanoid robot ARMAR-III [1]. The head of ARMAR-III features seven degrees of freedom (DoF) and is also available as a stand-alone version, known as the Karlsruhe Humanoid Head [5]. The head has four DoF in the neck and three DoF in the eyes for common tilt and independent pan eye movements. An Inertial Measurement Unit (IMU) is mounted at the center of the head providing linear and angular velocity measurements. Each eye is equipped with a separate wide angle and a foveal *Point Grey Dragonfly 2* camera. The cameras offer a resolution of 640×480 and are able to adapt to the environmental conditions by automatically tuning the parameters for exposure, shutter time and gain. All these parameters play a crucial role since lighting conditions may change at any time.

A. Object Localization Evaluation

The accuracy of the object localization methods were previously evaluated in [23], [22]. In all experiments, no external camera perturbations have been considered since such perturbations can lead to very blurred images making object detection impossible. To assess the effectiveness of the object localization methods with foveal cameras during self-induced perturbations we perform the following experiment. A single colored cup was placed in front of the robot, at distance of $1.50\,\mathrm{m}$. An overview of the setup is depicted in the first image of Fig. 6. The view direction was manually specified to ensure that the object is visible in the center of the foveal cameras. The object was constantly localized while the torso joint (*hip yaw*) of the robot was periodically rotating. Consequently, the head and eyes were moved according to a sinusoidal motion in lateral direction. The frequency of the motion was set to $0.25\,\mathrm{Hz}$ with an amplitude of $20°$. The same experiment was repeated using a textured object. In both experiments, the gaze stabilization greatly improves the localization results using the foveal cameras during motion. Fig. 7 shows the benefit of the gaze stabilization for a successful localization of textured and single colored (segmentable) objects while the torso is moving. During the motion, the object localization methods was called every

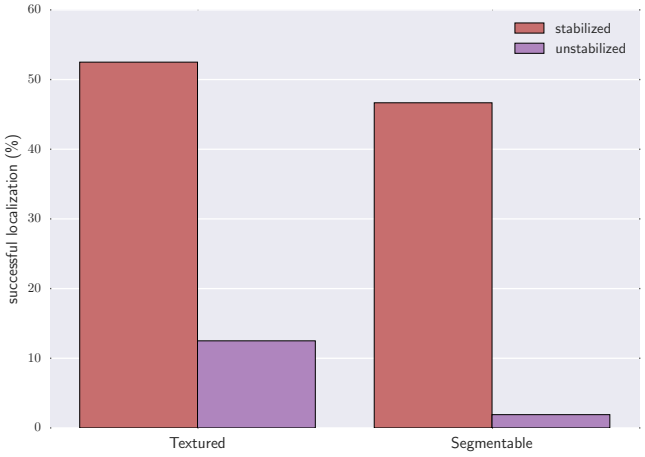

Fig. 7: Successful localization of textured and single colored (segmentable) objects with and without stabilization.

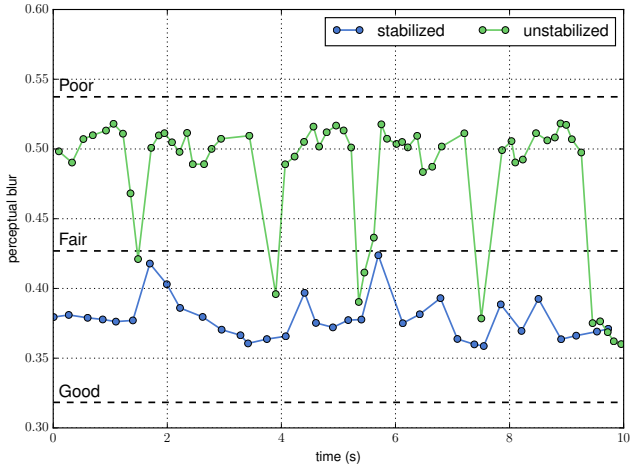

Fig. 8: Perceptual blur for the right foveal camera images during the object localization experiment. Higher values indicate a higher perceptual blur. Three subjective image quality levels are shown with a dashed line using the correlation provided by [26].

50 ms for single colored objects and 70 ms for textured objects respectively. The different localization frequency is due to the fact that the localization method for textured objects is more computationally intensive. Successful localization could be obtained in 13 % of all localizations attempts in the case of textured objects and 2 % of all localizations attempts in the case of single colored objects without stabilization. The difference in the numbers between textured and single colored objects can be attributed to the fact that a combination of model-based and appearance-based approaches is used in the underlying localization methods ([23], [22]), making single colored objects more prone to blurred images. Furthermore, the perceived color information heavily depends on the current illumination. A camera image for both the stabilized and unstabilized case is shown in the last two images of Fig. 6. The images clearly depict a difference regarding the blurriness between the unstabilized (center) and the stabilized case (right). In this work, we quantify the blur in both cases objectively by resorting to a no-reference image quality metric. We choose the metric proposed by Crete et al. [26] since the authors also provide a correlation between subjective tests and their perceptual blur metric. This metric quantifies the perceptual blur of an image by blurring it artificially and then comparing the variations between neighboring pixels between the images. The idea is that neighboring pixels will change with a major variation if the input image has a low perceptual blur. A lower value of the metric corresponds with a low perceptual blur, whereas a higher value corresponds with a high perceptual blur. Fig. 8 shows the perceptual blur for the recorded camera images. With gaze stabilization the perceptual blur is significantly lower than for the unstabilized case. The measured impairment for the stabilized case can be attributed to the fact that the recorded images were compressed using the H264 codec during the experiments. Without gaze stabilization the image quality is affected by the velocity of the sinusoidal perturbation applied to the torso joint. Overall, our experiment shows the benefits of using gaze stabilization to support foveated vision.

B. Experiment - Grasping While Moving

So far we have shown a successful integration of the gaze stabilization controller and its benefits for object localization. In order to evaluate our system we conducted the following experiment. The humanoid robot ARMAR-III is located in a kitchen environment and searches for a green cup while moving. A saliency map based on the color information of the peripheral camera images is computed. Thus, the view selection shifts automatically the gaze to single colored blobs in the scene. This step allows to leverage the foveated cameras to localize the cup. Using the foveal cameras for object localization is necessary since the object is far away. The goal is to grasp the object during locomotion. The idea for this experiment is inspired by Mansard et al. [27].

We are utilizing a position-based visual servoing controller to reach the final grasping pose of the object [28]. A grasp is executed by positioning the end-effector relatively to the object position. Once the position is reached the robot closes its hand to grasp the object. However, due to the locomotion the robot is unable to position the TCP fast enough w.r.t. the final grasping pose. Therefore, we implement a *pre-visual servoing* strategy to align the pose of the end-effector and reducing the difference to the final grasping pose. Knowing the speed of the robot, the object localization result is projected to a reachable pose within robot's base coordinate system. This allows the robot to already start to position the hand in advance using the visual servoing controller. Intuitively, the projected position corresponds to the expected position at the time when the robot is able to reach the object. Consequently, this step is repeated to allow the robot to continuously update the end-effector pose until the object is in a reachable area. We were unable to utilize the foveal

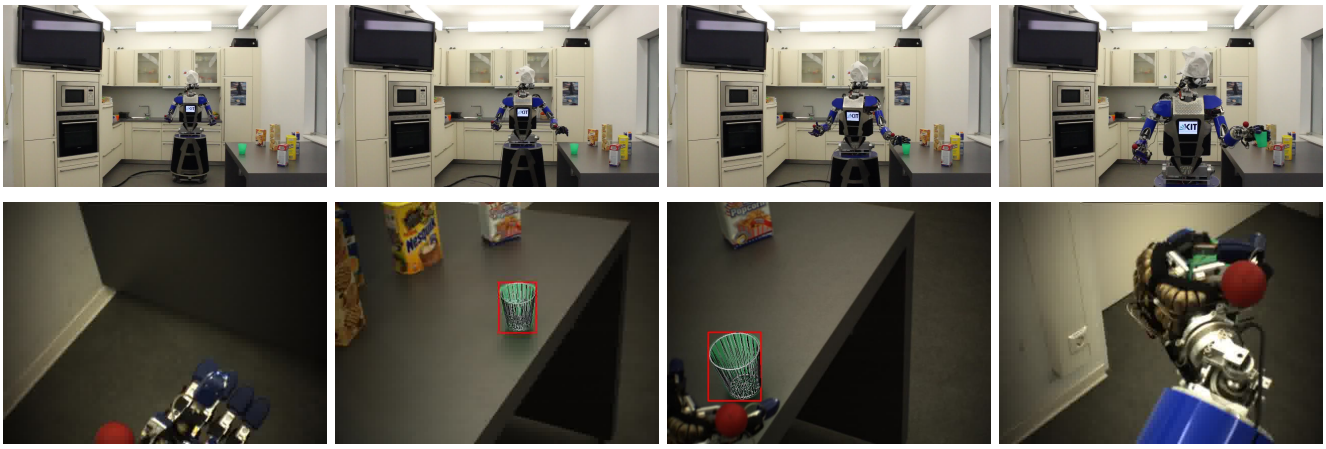

Fig. 9: Object grasping while moving. The top row shows an external view of the robot, while the bottom row shows the images of the right foveal camera.

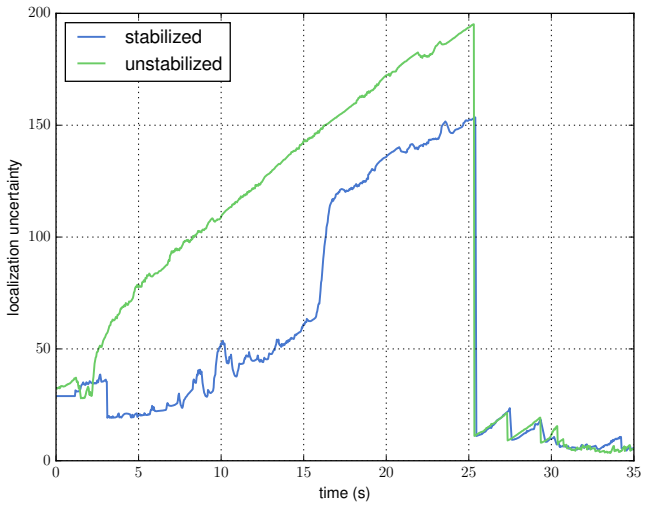

Fig. 10: Uncertainty of the object localization while using the foveal cameras with and without stabilization. After 25 s, the object is close enough and the visual servoing strategy increases the task acuity. Thus, the gaze is shifted from the hand towards the known localization of the cup. Once the object is in the field of view it is localized and thus the localization uncertainty drops.

cameras without the view selection component since the field of view is too narrow for object localization during locomotion. Snapshots from the experiment are illustrated in Fig. 9 and a video is attached to this contribution. The same experiment was run without gaze stabilization in order to show the benefits of the stabilizing controller. The object localization uncertainty is shown in Fig. 10. The gaze stabilization and view selection strategy reduces the object localization uncertainty significantly. Furthermore, the gaze stabilization reduced the average root mean squared error (RMSE) of the optical flow in the camera images by more than 50% as can be seen from the mean values in Table I.

Dense Optical Flow RMSE (deg/s)		
	Stabilized	Unstabilized
std	0.87	1.71
mean	1.01	2.06
max	3.41	10.49

TABLE I: *Standard deviation (std)*, *mean* and, *max* of the root mean square error (RMSE) the optical flow.

VI. Conclusion and Future Work

We have presented an integrated active vision system with gaze stabilization and view selection strategy. The system determines salient regions in the scene and computes view directions suitable for the current task. Given a desired view target, the gaze stabilization component fixates and stabilizes the view target during locomotion and facilities the robust execution on vision-based algorithms by ensuring stable camera images. Both our qualitative and quantitative evaluations show the benefits of using gaze stabilization for object localization. The gaze stabilization yields more stable images which are required for an accurate object localization. Furthermore, information provided by the foveated cameras can be leveraged during locomotion or whole-body manipulation tasks. Overall, a successful integration and interaction of view selection and gaze stabilization was demonstrated in a complex real world scenario, where a humanoid robot was able to grasp an object while moving. Future work will focus on the implementation of other whole-body manipulation tasks on ARMAR and on other humanoid robot platforms.

References

[1] T. Asfour, K. Regenstein, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, "Armar-iii: An integrated humanoid platform for sensory-motor control," in *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pp. 169–175, 2006.

[2] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active vision," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 333–356, 1988.

[3] R. Bajcsy, "Active perception," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 966–1005, 1988.

[4] S. Frintrop, E. Rome, and H. I. Christensen, "Computational visual attention systems and their cognitive foundations: A survey," *ACM Trans. Appl. Percept.*, vol. 7, no. 1, pp. 6:1–6:39, 2010.

[5] T. Asfour, K. Welke, P. Azad, A. Ude, and R. Dillmann, "The karlsruhe humanoid head," in *Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots*, pp. 447–453, 2008.

[6] R. Bajcsy, Y. Aloimonos, and J. K. Tsotsos, "Revisiting active perception," *Autonomous Robots*, pp. 1–20, 2017.

[7] S. Chen, Y. Li, and N. M. Kwok, "Active vision in robotic systems: A survey of recent developments," *The International Journal of Robotics Research*, vol. 30, no. 11, pp. 1343–1377, 2011.

[8] A. Ude, C. G. Atkeson, and G. Cheng, "Combining peripheral and foveal humanoid vision to detect, pursue, recognize and act," in *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, pp. 2173–2178, 2003.

[9] D. Omrčen and A. Ude, "Redundant control of a humanoid robot head with foveated vision for object tracking," in *2010 IEEE International Conference on Robotics and Automation (ICRA 2010)*, pp. 4151–4156, 2010.

[10] A. Ude and T. Asfour, "Control and recognition on a humanoid head with cameras having different field of view," in *2008 19th International Conference on Pattern Recognition (ICPR)*, pp. 1–4, 2008.

[11] B. Rasolzadeh, M. Bjorkman, K. Huebner, and D. Kragic, "An active vision system for detecting, fixating and manipulating objects in the real world," *The International Journal of Robotics Research*, vol. 29, no. 2-3, pp. 133–154, 2010.

[12] K. Welke, D. Schiebener, T. Asfour, and R. Dillmann, "Gaze selection during manipulation tasks," in *2013 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 652–659, 2013.

[13] S. Vijayakumar, J. Conradt, T. Shibata, and S. Schaal, "Overt visual attention for a humanoid robot," in *RSJ/IEEE International Conference on Intelligent Robots and Systems*, pp. 2332–2337, 2001.

[14] L. Vannucci, S. Tolu, E. Falotico, P. Dario, H. H. Lund, and C. Laschi, "Adaptive gaze stabilization through cerebellar internal models in a humanoid robot," in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pp. 25–30, IEEE, 2016.

[15] A. Roncone, U. Pattacini, G. Metta, and L. Natale, "Gaze stabilization for humanoid robots: A comprehensive framework," in *2014 IEEE-RAS 14th International Conference on Humanoid Robots (Humanoids 2014)*, pp. 259–264, 2014.

[16] T. Habra and R. Ronsse, "Gaze stabilization of a humanoid robot based on virtual linkage," in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pp. 163–169, IEEE, 2016.

[17] A. Roncone, U. Pattacini, G. Metta, and L. Natale, "A cartesian 6-dof gaze controller for humanoid robots," in *Robotics: Science and Systems 2016*, 2016.

[18] R. P. de Figueiredo, A. Bernardino, J. Santos-Victor, and H. Araújo, "On the advantages of foveal mechanisms for active stereo systems in visual search tasks," *Autonomous Robots*, pp. 1–18, 2017.

[19] N. Vahrenkamp, M. Kröhnert, S. Ulbrich, T. Asfour, G. Metta, R. Dillmann, and G. Sandini, "Simox: A robotics toolbox for simulation, motion and grasp planning," in *Intelligent Autonomous Systems 12*, vol. 193 of *Advances in Intelligent Systems and Computing*, pp. 585–594, Springer Berlin Heidelberg, 2013.

[20] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano, "Closed-loop inverse kinematics schemes for constrained redundant manipulators with task space augmentation and task priority strategy," *The International Journal of Robotics Research*, vol. 10, no. 4, pp. 410–425, 1991.

[21] N. Vahrenkamp, M. Wächter, M. Kröhnert, K. Welke, and T. Asfour, "The robot software framework armarx," *it - Information Technology*, vol. 57, no. 2, 2015.

[22] P. Azad, T. Asfour, and R. Dillmann, "Accurate shape-based 6-dof pose estimation of single-colored objects," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009)*, pp. 2690–2695, 2009.

[23] Pedram Azad, Tamim Asfour, and R. Dillmann, "Stereo-based 6d object localization for grasping with humanoid robot systems," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 919–924, 2007.

[24] M. Wächter, S. Ottenhaus, M. Kröhnert, N. Vahrenkamp, and T. Asfour, "The armarx statechart concept: Graphical programing of robot behavior," *Frontiers in Robotics and AI*, vol. 3, p. 87, 2016.

[25] T. Habra, M. Grotz, D. Sippel, T. Asfour, and R. Ronsse, "Multimodal gaze stabilization of a humanoid robot based on reafferences," 2017.

[26] F. Crete, T. Dolmiere, P. Ladret, and M. Nicolas, "The blur effect: Perception and estimation with a new no-reference perceptual blur metric," SPIE Proceedings, p. 64920I, SPIE, 2007.

[27] N. Mansard, O. Stasse, F. Chaumette, and K. Yokoi, "Visually-guided grasping while walking on a humanoid robot," in *2007 IEEE International Conference on Robotics and Automation*, pp. 3041–3047, 2007.

[28] N. Vahrenkamp, S. Wieland, P. Azad, D. Gonzalez, T. Asfour, and R. Dillmann, "Visual servoing for humanoid grasping and manipulation tasks," in *2008 8th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2008)*, pp. 406–412, 2008.