2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Basic Behavior Acquisition Based on Multisensor Integration of a Robot Head

Chenggang Liu and Jianbo Su

Abstract— **This paper addresses the basic behavior acquisition of the robot head based on the multisensor integration. A robot head generally has several degrees of freedom(D.O.F.) of motions as well as different kinds of sensors. The head motion is planned based on all sensors' feedback. We take advantage of the Jacobian matrix to describe the differential relations between the sensor feedback and the motor motions. Hence, the relation between two sensors could be formulated by the two respective Jacobian matrices of both sensors to motors. Consequently, multisensor integration can be employed for better performance of the robot head. Experiments of basic behavior acquisition like gazing and head posture control are conducted. Performances of both basic behaviors of the robot head before and after multisensor integration are compared, which demonstrate that the proposed multisensor integration way improves the performance of control, robustness against some sensor's failure, and reduces the overall computation.**

I. INTRODUCTION

Cognitive processes at high-level abstractions rely on a hierarchy of lower-level behaviors. Low-level autonomous behaviors can be constructed from basic sensorimotor behaviors [1], [2]. A basic sensorimotor behavior is a reflex, which is a direct motor response to sensory feedback. Basic behavior designs are essential to a behavior based robot. In order to acquire basic behaviors, it is most often to assume the way that an autonomous robot perceives the world and take advantage of the prior knowledge of the sensorimotor maps. These assumptions generally reduce the chances of generating new behaviors from the dynamic interaction of the robot with the environment. If basic behaviors are determined a priori, it is most probable that the behavior control might be inefficient at best, or completely wrong at worst. An alternative to overcome these limitations is to acquire the basic behaviors by the robot itself.

The control mechanisms for the stabilizing gaze have long been studied in biological systems [3], [4]. Stabilizing gaze is to maintain fixation on a possibly moving visual target from a possibly moving gaze platform. It is important because it has advantages of high-resolution, mathematical simplification, active visual sensing and so on [5]. For a humanoid robot, the posture control of its head is also very important. Biped locomotion is inherently unstable, so the posture of its head must be under control in order to keep the whole body stable. On the other hand, the posture control of its head is also necessary during human-robot interaction. Humans use many kinds of body language, such as nodding, shaking one's head and so on. All of these need the posture control of the head. For a human being, his vestibular system senses the position of the head and the body in space, which is critical to the control of his posture. The information provided by the vestibular system is fused with vision at a very early stage. It also plays a key role in the stabilizing gaze. On the other hand, visual cues aid the spatial orientation and body equilibrium. So the visual system also plays an important role on posture control [6]. Multisensor integration is to integrate different kinds of sensors in order to complement to each other [7], [8]. The stabilizing gaze and the posture control skills are so reactive and pervasive that require a significant amount of multisensor integration [9].

Some attempts have been made to acquire sensorimotor maps in order to generate basic sensorimotor behaviors of a robot head. Marjanović et al. [10] proposed a system using self-supervised learning method to get the map between eye, head, and arm end-point. The system can then perform fundamental visuomotor coordination task, such as gazing and pointing. In [11], a sensorimotor map was learned by a Growing Neural Gas (GNG) network. The result of the learning is the construction of a motor map which codes adaptively compensatory stabilization reflexes. In [12], instead of learning a direct mapping from the image sensing to a desired action, the system first learned a forward model, then performed gaze control. When these methods are used in a real robot, there seems to be a limitation for computational complexity and learning time. In robotics, the stabilizing gaze control techniques dealing with multisensor integration have received little attention so far. Shibata et al. [13] deal with the problem of visuo-inertial integration by reproducing an accurate computational model of the biological reflexes system. In [14], the proposed binocular system is controlled by integrating visual and inertial information. The integration of these informations is implemented by considering the geometry of the binocular system and the knowledge of actual gaze configuration (the gaze distance and direction). In [11], motion cues are integrated with visual cues by means of a neural controller. There is work on posture control of articulated mobile robots, but little on the integration of the posture control with visual information. In this paper, we propose a new approach to acquire basic behaviors based on multisensor integration. It is realized that the differential relation between the sensor feedback and the motor motions can be linearly described for such a servo control system. Hence two sensors can be related by their differential relations to motors. In the field of image-based

This work was supported by National Natural Science Foundation of China under Grant (60675041).

C. Liu and J. Su are with the Department of Automation, Shanghai Jiaotong University, Shanghai, 200240, China {frankliu, jbsu}@sjtu.edu.cn.

visual servoing, image Jacobian model is used to linearly describe the differential relation between the visual sensory feedback and the robot motion. It is widely used and has been the mostly investigated approaches. We take advantage of the Jacobian matrix to describe the differential relations between the sensor feedback and the motor motions. The Jacobian matrix is then used to generate basic behaviors, such as stabilizing gaze and posture control, from the corresponding sensor. Furthermore, the relation between two sensors could be described by the two Jacobian matrices of both sensors to motors. Consequently, multisensor integration can be realized for better performance of the robot head. In order to realize the flexibility of the proposed model itself, a system identification method is used to make the system free from calibration and a multi-agent-based implementation is proposed.

The rest of the paper is organized as follows. In section II, we propose our model and the multi-agent-based implementation of the control system. Section III gives a detailed description of our robot head. Experiments are provided in section IV to demonstrate the validity and performance of the proposed model. Conclusions and future work are provided in Section V.

II. MODEL

A. Acquisition of Basic Behaviors

Suppose the system has n motors, the current motor states, namely their rotation angles, are denoted by $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$. The control input $\mathbf{u}_c(t)$ to the system at time t is the desired increment of $\mathbf{x}$. Suppose sensor M has l output, their current states and desired states are denoted by $\mathbf{y}^M = [y_1^M, y_2^M, \cdots, y_l^M]^T$ and $\mathbf{r}^M = [r_1^M, r_2^M, \cdots, r_l^M]^T$, respectively. The control output based on sensor M's feedback is denoted by $\mathbf{u}^M$. The problem here is to determine the control input $\mathbf{u}_c$ based on the current states and the desired states of all sensors.

We suppose that the mapping between sensors and motors is continuous, differentiable, and time-invariant. To simplify the notation in the following manipulations, the superscript or subscript of $\mathbf{y}^M$, $\mathrm{d}\mathbf{y}^M$,f^M, a^M, and J_M will be suppressed. So the correspondence between the current states of sensor M and the current motor states is

$$y_i = f_i(\mathbf{x}) \qquad i \in [1, l]. \tag{1}$$

Thus the relation of differential is

$$dy_i = \mathbf{a}_i^T \mathrm{d}\mathbf{x} \quad i \in [1, l], \tag{2}$$

where $\mathbf{a}_i = [\frac{\partial y_i}{\partial x_1}, \frac{\partial y_i}{\partial x_2}, \cdots, \frac{\partial y_i}{\partial x_n}]^T$. The simultaneous equations are

$$\mathbf{y} = f(\mathbf{x}), \tag{3}$$

and

$$\mathrm{d}\mathbf{y} = J\mathrm{d}\mathbf{x}, \tag{4}$$

where $\mathbf{y} = [y_1, y_2, \cdots, y_l]^T$ and $J = [\frac{\partial \mathbf{y}}{\partial \mathbf{x}}] = [\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_l]^T$. J is a Jacobian matrix which relates the sensor feedback with the motor motions. We will propose to estimate the Jacobian matrix based on an on-line system identification method.

In order to generate basic behaviors, resolved-rate motion control [15] is used. Like the earliest approaches to image-based visual servoing control, a simple proportional control law is given by

$$\mathbf{u} = KJ^*(\mathbf{r} - \mathbf{y}), \tag{5}$$

where K is the constant gain matrix and J^* is the pseudo-inverse matrix of J. There is one control output based on one sensor feedback. If there are multiple sensors, there will be multiple control outputs and it will be necessary to integrate all these control outputs into one control input to the system. We use the most recently computed control output based on any sensor feedback as the control input to the system. Hence we get the control input to the system,

$$\mathbf{u}_c(t) = Merge(\mathbf{u}^1, \mathbf{u}^2, \ldots), \tag{6}$$

where $\mathbf{u}^i$ denotes the control output based on sensor i and function $Merge(\cdot)$ returns the most recently computed $\mathbf{u}^i$.

B. Jacobian Matrix Online Estimation

The recursive least-squares (RLS) estimation is widely used for system identification. RLS can be interpreted as a Kalman filter for process

$$\theta(k+1) = \theta(k) \tag{7}$$

$$v(k) = \varphi(k)^T\theta(k) + \nu(k), \tag{8}$$

where $\theta(k)$ is the state to be estimated, $\varphi(k)^T$ is measurement matrix, $v(k)$ is measurement, and $\nu(k)$ is measurement noise. Let

$$\theta(k) = [\mathbf{a}_1^T, \mathbf{a}_2^T, \cdots]^T \tag{9}$$

$$v(k) = \mathbf{y}(k) - \mathbf{y}(k-1) \tag{10}$$

$$\Delta\mathbf{x}(k) = \mathbf{x}(k) - \mathbf{x}(k-1) \tag{11}$$

$$\varphi(k) = \begin{bmatrix} \Delta\mathbf{x}(k) & & 0 \\ & \ddots & \\ 0 & & \Delta\mathbf{x}(k) \end{bmatrix}, \tag{12}$$

where $\mathbf{x}$ is the current motor states, $\mathbf{y}$ is the sensor reading, and $\mathbf{a}_i^T$ is the *ith* row element of the Jacobian matrix J. Then we can use the RLS algorithm to estimate J. By the RLS, the estimation could be

$$K(k+1) = P(k)\varphi(k+1)(I + \varphi^T(k+1)P(k)\varphi(k+1))^{-1} \tag{13}$$

$$\hat{\theta}(k+1) = \hat{\theta}(k) + K(k)[v(k+1) - \varphi^T(k+1)\hat{\theta}(k)] \tag{14}$$

$$P(k+1) = (I - K(k+1)\varphi^T(k+1))P(k). \tag{15}$$

In [16], such an image Jacobian matrix estimator has been proved to be an effective method for online estimation of it.

C. Multisensor Integration

For sensor M and sensor N ($M \neq N$), according to (4), we have

$$\mathrm{d}\mathbf{y}^M = J_M \mathrm{d}\mathbf{x} \quad (16)$$

$$\mathrm{d}\mathbf{y}^N = J_N \mathrm{d}\mathbf{x}. \quad (17)$$

Hence

$$\mathrm{d}\mathbf{y}^M = J_M J_N^* \mathrm{d}\mathbf{y}^N, \quad (18)$$

where J_N^* is the pseudo-inverse matrix of J_N. Then we have

$$\begin{aligned}\hat{\mathbf{y}}^{M|N}(t) &= \mathbf{y}^M(kT^M) + \int_{kT^M}^{t} \mathrm{d}\mathbf{y}^M \\ &= \mathbf{y}^M(kT^M) + J_M J_N^*[\mathbf{y}^N(t) - \mathbf{y}^N(kT^M)],\end{aligned} \quad (19)$$

where $\hat{\mathbf{y}}^{M|N}(t)$ denotes the estimation of sensor M's current states with the help of sensor N's and T^M is the sample period of $\mathbf{y}^M$. Equation (19) illustrates that one sensor's current states can be estimated with the help of another's. Combining all estimations and sensor M's reading, we can get the estimation of sensor M's current states,

$$\hat{\mathbf{y}}^M(t) = Merge(\mathbf{y}^M(kT^M), \hat{\mathbf{y}}^{M|1}(t), \hat{\mathbf{y}}^{M|2}(t), \ldots), \quad (20)$$

where $Merge(\cdot)$ returns the most recent sensor reading $\mathbf{y}^M(kT^M)$ or the most recent estimation $\hat{\mathbf{y}}^{M|i}$ with the help of sensor i. This kind of integration has many advantages. First, it can reduce the overall computational complexity for a given task. Suppose sensor M is computationally intensive, it can save many computational resources to estimate the current states of sensor M with the help of computationally less intensive sensors. Second, the system could be more robust against failure. When some sensor fails, its current states can still be estimated from the others'. Third, $\hat{\mathbf{y}}^M$ has a higher sample rate than $\mathbf{y}^M$. This can improve the control performance of the system in most cases.

D. Implementation of the Model

In order to realize the flexibility of the model itself, we propose the multi-agent-based implementation of the model. Agents are independent function modules. Different kinds of sensors are put into separated agents. One agent's current states, desired states, and control output are its sensors' current output, desired output, and control output, respectively. The implementation of an agent is illustrated by Fig. 1. The inputs to an agent are its desired states, the current motor states, the current states and the Jacobian matrices of the other agents. The current states of an agent are read from its sensor output and the current motor states are read from the motion control board. The output of an agent is its current states, its Jacobian matrix, and its control output. The Jacobian matrix is estimated each time when the corresponding agent's current states change. Multisensor integration is performed when the states of any agent change.

The final multi-agent-based implementation of the control system is shown in Fig. 2. Each agent is fully connected with the others. Each agent sends and receives additional data if necessary. For a given task, it is first mapped to the desired states of some agents and then accomplished by all agents. If one agent's desired states are not explicitly defined, they are always assigned to its current states.

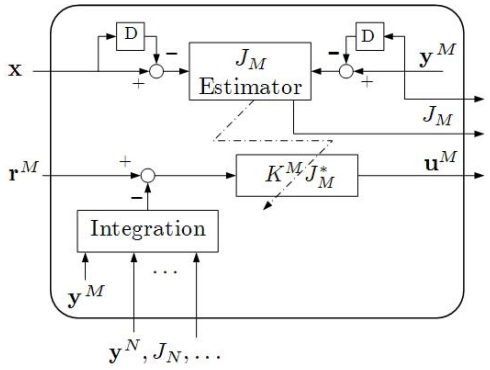


Fig. 1. Implementation of an agent

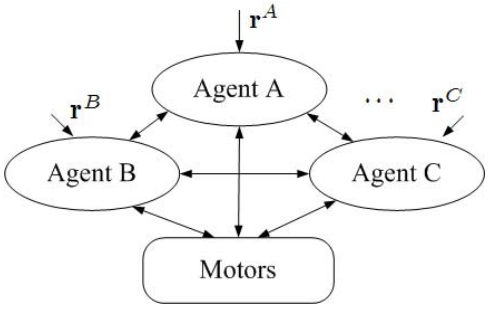


Fig. 2. The multi-agent-based implementation of the control system

III. ROBOT HEAD

This section gives a brief specification of our robot head shown in Fig. 3.

The artificial vestibular system of the robot head is simplified and is assembled with a low cost 2-axis tilt sensor provided by Zhichuan Tech Co.,Ltd. It is fixed on the top of the robot head and its two axes are tightly coupled with the head's motion (see Fig. 3(b)).

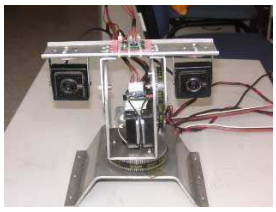
As for the visual system, there are two color CCD cameras with PAL TV output. Video signals are grabbed and then processed by a workstation with a Pentium D 2.8G CPU.

These two sensory systems are integrated within a binocular architecture (see Fig. 3(b)). Two stepper motors are used to drive the pan axis and the tilt axis. We have focused our efforts on the design and the construction of the head so as to minimize the nonlinearities by minimizing head structure parameters, h and a (see Fig. 3(d)).

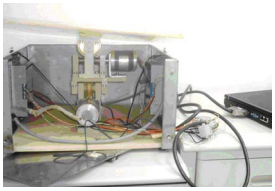
IV. EXPERIMENTS

Experiments are run to check the validity of the proposed model and the performance of the proposed control scheme. We use three kinds of sensors to implement three agents, namely the attention detecting agent, the retinal slip agent, and the vestibular agent. The current states and the desired states of the attention detecting agent are denoted by $\mathbf{y}_a$ and

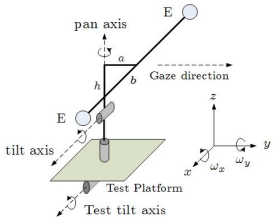
(a) Photo of the experiment enviroment



(b) Photo of the robot head

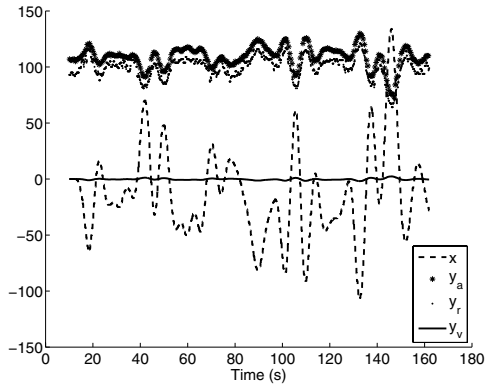


(c) Photo of the test platform

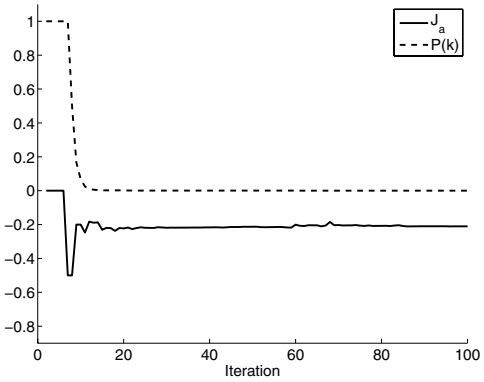


(d) The coordinate systems for the robot head and the tilt sensor

Fig. 3. The experiment environment, the robot head, and the coordinate systems



(a) Head voluntary movement and each agent's current states



(b) Jacobian matrix estimation in the attention detecting agent

Fig. 4. Head voluntary movement and Jacobian on-line estimation. In the above figure, y_a and y_r are both in pixel. y_v is in degree and x in step.

$\mathbf{r}_a$, respectively. They are the current and desired position of the attention point in current image. In experiments we choose the center of a cartoon human face as the attention point. The cartoon human face is detected by a boosted cascade of simple features method. This method is initially proposed by Paul Viola [17] and improved by Rainer Lienhart [18]. It is performed every 500 ms by the workstation. Generally, retinal slip can denote both a position error and the velocity of an image on the retina. We explicitly use it as a position error. In our experiments 10 feature points with big eigenvalues are selected in the 50×50 image whose center is the most recently detected attention point. Then they are tracked based on the method of sparse iterative version of Lucas-Kanade optical flow in pyramids [19]. The retinal slip is the mean position error of all feature points. The current states and the desired states of the retinal slip agent are denoted by $\mathbf{y}_r$ and $\mathbf{r}_r$, respectively. The calculation of the retinal slip is performed every 50 ms by the workstation. The vestibular agent's current states are the tilt sensor's output which is read every 33 ms. Its current states and desired states are denoted by $\mathbf{y}_v$ and $\mathbf{r}_v$, respectively.

Denote the head movement by $H(t)$, the involuntary ego-motion of the head by $I(t)$, the movement of the attention point by $O(t)$, then

$$\mathbf{y}_a = f_a(O(t), H(t)) \tag{21}$$

$$\mathbf{y}_r = f_r(O(t), H(t)) \tag{22}$$

$$\mathbf{y}_v = f_v(H(t)) \tag{23}$$

$$H(t) = \mathbf{x}(t) + I(t). \tag{24}$$

The Jacobian matrices of the attention detecting agent, the retinal slip agent and the vestibular agent can be expressed as $J_a = \partial f_a/\partial \mathbf{x}$, $J_r = \partial f_r/\partial \mathbf{x}$, and $J_v = \partial f_v/\partial \mathbf{x}$, respectively. Every Jacobian matrix is assigned to zero initially, estimated online in the background, and assigned to the estimated value until its priori estimate error covariance $P(k)$ is small enough. Involuntary ego-motion $I(t)$ will cause measurement noise in equation (8) and should be minimized or avoided during Jacobian Matrix estimation.

In the following experiments, the head is fixed on the top of the test platform (see Fig. 3(c)) which generates the involuntary ego-motion. The test platform generates tilt rotation along the test axis as shown in Fig. 3(d). The attention point is static. Only the left camera is used and the image acquired has 320×240 pixels. Every agent's constant gain matrix K in (5) is assigned to 2×2 diagonal matrix with all diagonal elements 0.1. Without loss of generality, only tilt rotation movement of the head is considered for simplicity hereafter.

A. Jacobian Matrix Estimation

In the first experiment, the test platform is static and the head moves voluntarily. The voluntary movement is generated randomly with the highest frequency less than 1.5 Hz and the amplitude less than 150 steps as shown in Fig. 4(a). During the voluntary movement, the Jacobian matrix in each agent is estimated online. The initial condition, $P(0)$, in each agent is an identity matrix. After less than 100 iterations, all priori estimate error covariances become very small. For example, in the attention detecting agent, $P(k)$ is less than 0.001 and J_a is nearly invariable after 50 iterations as shown in Fig. 4(b).

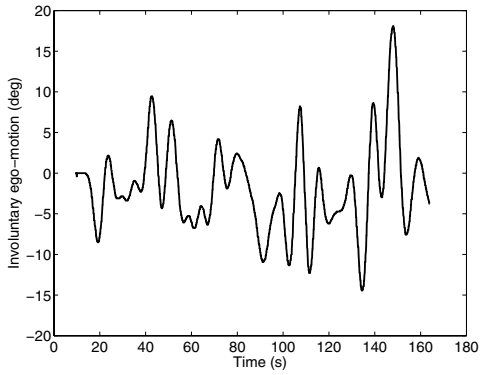


Fig. 5. Head involuntary ego-motion generated by the test platform

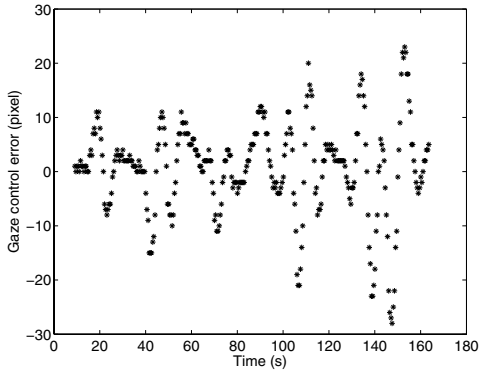


Fig. 6. Stabilizing gaze control before multisensor integration

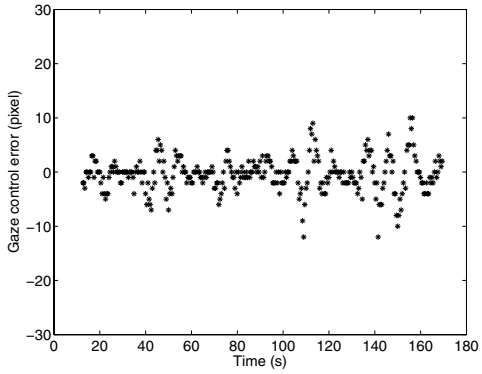


Fig. 7. Stabilizing gaze control after multisensor integration

B. Acquisition of Stabilizing Gaze and Head Posture Control

In the second experiment, the test platform generates the head involuntary ego-motion as shown in Fig. 5 and the gaze control errors' standard deviations before and after multisensor integration are compared. Let $r_a = 120$, $r_v = y_v$ and $r_r = y_r$. Before multisensor integration, the gaze control errors' standard deviation is about 8.526 pixels as shown in Fig. 6. After the attention detecting agent is integrated with the retinal slip agent and the vestibular agent, it drops to 3.147 pixels as shown in Fig. 7.

In the third experiment, the test platform generates the same head involuntary ego-motion as in the second experiment. The posture control errors' standard deviations before and after multisensor integration are compared. Let $r_a = y_a$, $r_v = 0$, and $r_r = y_r$. Before multisensor integration, the head posture control errors' standard deviation is about 0.318 degrees as shown in Fig.8. After the vestibular agent is integrated with the retinal slip agent and the attention detecting agent, it drops to 0.210 degrees as shown in Fig.9.

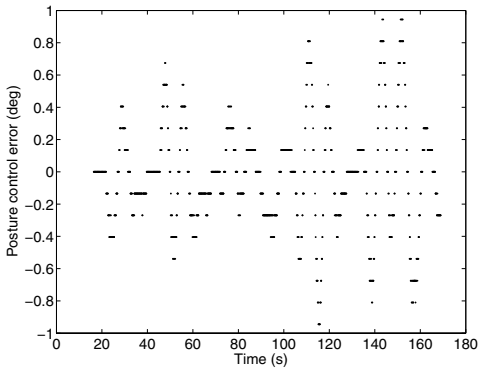


Fig. 8. Posture control before multisensor integration

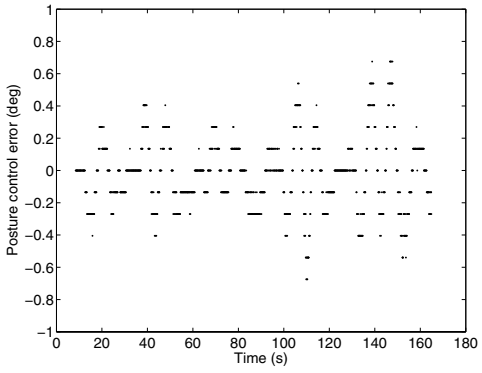


Fig. 9. Posture control after multisensor integration

The second and the third experiments show that the proposed model with the appropriate desired states assignments can generate basic behaviors like stabilizing gaze and head posture control. They also demonstrate that multisensor integration improves the performance of stabilizing gaze control and head posture control.

C. More Benefits of Multisensor Integration

In the last experiment, we will further explore the benefits of multisensor integration to robustness against some sensor's failure and computational simplification. The attention point detecting is fragile to luminance change and image distortion. The attention detecting agent may fail to detect the attention point occasionally as we have seen during our experiments. In this experiment, we use a piece of white paper to cover a small part of the cartoon face occasionally. It will make the attention detecting agent fail to detect the attention point more frequently. The gaze control errors are compared before and after integration. Fig. 10 and 11 show that the gaze control errors are reduced by the integration with the other two agents. When the attention detecting agent fails to detect

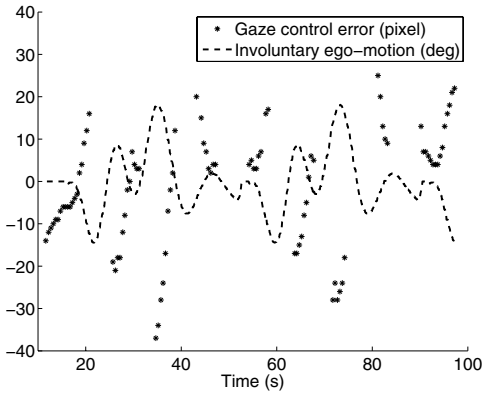


Fig. 10. Stabilizing gaze control before multisensor integration.

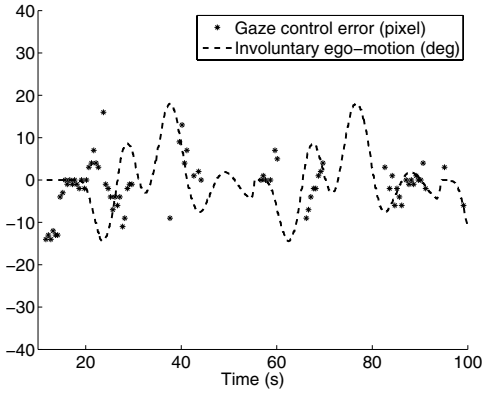


Fig. 11. Stabilizing gaze control after multisensor integration.

the attention point, the position of the attention point can still be estimated from the current states of the retinal slip agent and the vestibular agent. So in spite of the attention detecting agent's failure, the gaze control errors are still relatively small after multisensor integration. This means the proposed multisensor integration way can improve the robustness against some sensor's failure. Because the attention detecting agent is computationally more intensive than the others, it can simplify the computation by deliberately reducing the attention detecting frequency but still keep relatively small gaze control errors by the proposed multisensor integration method.

V. CONCLUSIONS AND FUTURE WORK

Our research goal is to investigate a mechanism to acquire basic behaviors and multisensor integration for a more complex humanoid robot. This paper takes a first step towards this goal by exploring stabilizing gaze control and head posture control. We propose the multi-agent-based implementation of the control system. We demonstrate that Jacobian matrices can be used to describe the sensorimotor correlation and generate basic sensorimotor behaviors. Jacobian matrix can also be used to relate different sensors' information and then realize multisensor integration. A system identification method, RLS, is used to estimate Jacobian matrix online. Online estimation makes the system free from advance calibration between the sensors and the motors and makes the system robust against some sensor's and motor's failures. Multisensor integration improves the control performance in stabilizing gaze control and head posture control. It can improve the robustness against some sensor's failure and reduce the overall computation for a given task.

Like in the conventional image Jacobian matrix approach, the convergence and stability of the proposed method is a problem and under investigation.

REFERENCES

[1] R. A. Brooks, A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation*, vol. 2(1), 1986, pp. 14-23.

[2] M.J. Matarić, Integration of Representation into Goal-driven Behavior-based Robots, *IEEE Transactions on Robotics and Automation*, vol.8, 1992, pp. 304-312.

[3] M. C. Schubert and L. B. Minor, Vestibulo-ocular Physiology Underlying Vestibular Hypofunction, *Physical Therapy*, vol.84, 2004, pp. 4.

[4] D. A. Robinson, "The oculomotor control system: A review", *in Proc. of the IEEE*, vol. 56(6), June 1968, pp. 1032-1049.

[5] D. Coombs and C. Brown, "Intelligent gaze control in binocular vision", *in Proc. of 5th IEEE International Symposium on Intelligent Control*, vol.1, Sept. 1990, pp. 239-245

[6] T. Nagata, Y. Fukuoka, A. Ishida, and H. Minamitani, "Analysis of role of vision in human upright posture control", *Engineering in Medicine and Biology Society, in Proc. of the 23rd Annual International Conference of the IEEE*, vol. 2, 2001, pp. 1155-1158.

[7] K. A. Fleming, R. A. Peters, and R. E. Bodenheimer, "Image mapping and visual attention on a sensory ego-sphere," *in Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 241-246.

[8] K. A. Hambuchen, "Multi-modal attention and event binding in humanoid robots using a sensory ego-sphere," Ph.D. Dissertation, Vanderbilt University, May 2004.

[9] R. A. Brooks, C. Breazeal, M. Marjanović, B. Scassellati, and M. M. Williamson, The cog project: Building a humanoid robot, *Lecture Notes in Computer Science*, vol. 1562, 1999, pp. 52-87.

[10] M. Marjanović, B. Scassellati, and M. Williamson, "Self-taught visually-guided pointing for a humanoid robot", *in Fourth International Conference on Simulation of Adaptive Behavior*. Cape Cod, MA,US, 1996, pp. 35-44.

[11] F. Panerai, G. Metta, and G.Sandini,"Learning vor-like stabilization reflexes in robots", *in 8th European Symposium on Artificial Neural Networks*, Bruges, Belgium, 2000.

[12] Per-Erik Forssén, "Learning saccadic gaze control via motion prediction", *in 4th Canadian Conference on Computer and Robot Vision*, Montréal, Québec, Canada, 2007, pp. 44-51.

[13] T. Shibata and S. Schaal, Biomimetic gaze stabilization based on feedback-error-learning with nonparametric regression networks, *Neural Network*, vol. 14(2), 2001, pp. 201-216.

[14] F. Panerai, G. Metta, and G. Sandini, "Visuo-inertial stabilization in spacevariant binocular systems robotics and autonomous systems," *Robotics and Autonomous Systems*, vol. 30, 2000.

[15] S. Hutchinson, G. Hager, and P. Corke, A tutorial on visual servo control, *IEEE Transactions on Robotics and Automation*, vol. 12(5), 1996, pp. 651-670.

[16] J. Qian and J. Su, "Online estimation of image jacobian matrix by kalman-bucy filter for uncalibrated stereo vision feedback", *in Proc. of the 2002 IEEE international Conference on Robotics and Automation*, Piscataway,NJ,USA, 2002.

[17] P. Viola and M. J. Jones, "Rapid object detection using a boosted cascade of simple features", *In Proc. IEEE Conference on CVPR* ,Kauai, Hawaii, USA, 2001, pp. 511-518

[18] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection", *in Proc. of International Conference on Image Processing*, vol. 1, 2002, pp. 900-903.

[19] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision", *in Proc. of the International Joint Conference on Artificial Intelligence*, 1981, pp. 674-679.