A robotic model of the development of gaze following

Hyundo Kim*§, Hector Jasso†, Gedeon Deák‡ and Jochen Triesch§
*ECE dept., University of California, San Diego, La Jolla, CA92093, USA, hyundo@ucsd.edu
†CSE dept., University of California, San Diego, La Jolla, CA92093, USA
‡Cognitive Science dept., University of California, San Diego, La Jolla, CA92093, USA
§Frankfurt Institute for Advanced Studies, Johann Wolfgang Goethe University, 60438 Frankfurt am Main, Germany

***Abstract*—For humanoid robots, the skill of gaze following is a foundational component in social interaction and imitation learning. We present a robotic system capable of learning the gaze following behavior in a real-world environment. First, the system learns to detect salient objects and to distinguish a caregiver's head poses in a semi-autonomous mannner. Then we present multiple scenes containing different combinations of objects and head poses to the robot head. The system learns to associate the detected head pose with correct spatial location of where potentially "rewarding" objects would be using a biologically plausible reinforcement learning mechanism.**

***Index Terms*—gaze following, joint attention, actor-critic reinforcement learning, habituation, human-robot interaction**

I. INTRODUCTION

Gaze following is the ability to shift one's gaze to the location where another agent is looking. Developmental studies show that infants are not born with it but the skill emerges during the first two years of life [1], [2]. Researchers in cognitive science and developmental psychology consider gaze following to be one of the foundational components of social interaction and learning in humans. An experimental study also showed strong positive correlation between gaze following at early age with the subsequent language scores at later ages [3].

The skill is also crucial for humanoid robots to learn and obtain skills through the social interaction with a caregiver as opposed to the laborious programming of individual behaviors. This paper is our continued effort to build an intelligent robotic system capable of learning by interacting with a human caregiver [4]. In our previous work, we proposed a computational model of how infants may acquire gaze following skills through a biologically plausible reinforcement learning scheme [5], [6], [7]. According to the model, the infant learns to locate salient, i.e. rewarding, targets by utilizing the head pose and gaze direction of the caregiver via an actor-critic reinforcement learning scheme. The model replicated well the development of gaze following in human infants observed by Butterworth and colleagues [1], [2].

In this paper we present an implementation of our model on a robot head. The robot learns gaze following through the interaction with a caregiver in a real-world environment as shown in Fig. 1. Learning is divided into two stages. First the robot learns to detect certain objects and distinguish different head poses. Second, we expose the robot head to various scenes containing different combinations of objects and head poses. The robot learns to associate the detected head pose with the necessary motor actions to look at salient object and thereby maximize its reward. After the training, we test the performance of the system. An overview of the complete system is shown in Fig. 3.

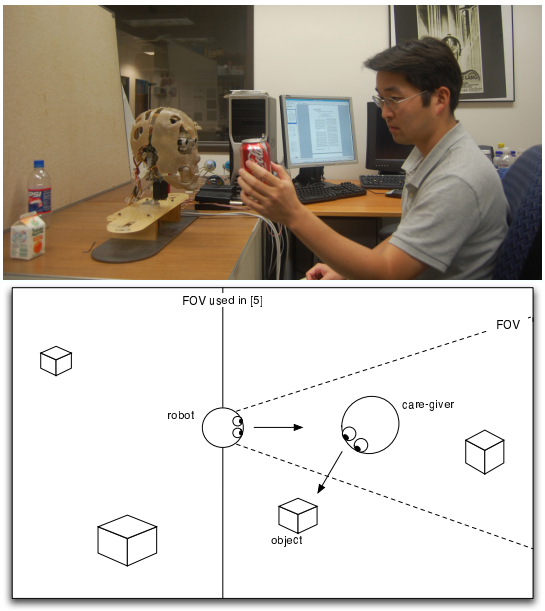

Fig. 1. A typical experiment scene setup. The caregiver is holding an object and attending to it. The robot is looking at the caregiver. The solid line indicates the boundary of the field of view (FOV) of the infant in [5], the dotted lines of the robot head. Our robot head has much narrower FOV (± 30) compared to that of the infant (± 90).

Several other groups have also proposed models of the development of gaze following behavior. We will briefly review some of their work.

A. Review on previous work

Matsuda and Omori [8] used a temporal-difference (TD) reinforcement learning scheme for learning joint visual attention. In their simulation, the caregiver first attends to the infant's face. When the infant makes eye contact with the caregiver, the caregiver shifts its gaze toward some toy. If the infant follows the caregiver's gaze correctly, the observer operates the attended toy to make a movement which serves as reward

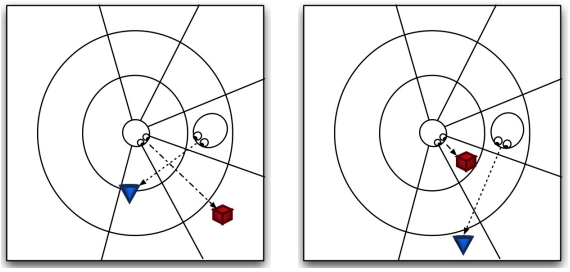

Fig. 2. Illustration of two ambiguous situations. Distractor object (red box) appears either behind (left) or in front of (right) the caregiver's gaze direction. Blue cone is the target object that the caregiver is looking at.

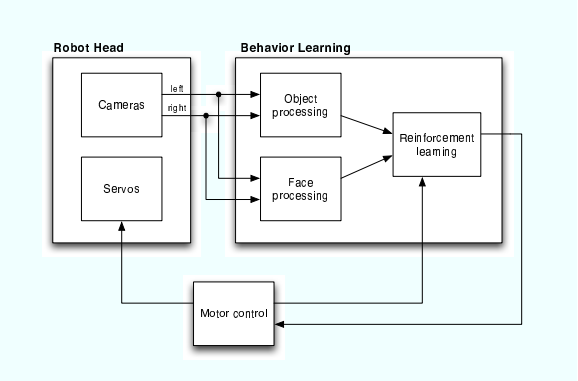

Fig. 3. An overview of the complete system.

for the infant. Their model is limited in the sense that the infant only gets reward from the moving object operated by the observer. Also the caregiver's face is treated separately from the objects and does not lead to any reward.

Nagai et al. [9], [10] used face edge features and motion information (optical flow) to estimate the sensorimotor coordination between these two inputs and the motor output using two separate neural networks. A coordinator module computes the average motor output from the outputs of the two neural networks. In other words, the system associates the pan and tilt angles of a caregiver to the target object within the robot's FOV. Their model does not utilize the depth information (i.e. distance from the robot to the object) and thus can not handle ambiguous situations where an object appears in robot's gaze direction that may not be located within the caregiver's gaze direction as shown in Fig. 2 whereas our system does use this depth information and can handle the situations mentioned above correctly.

Shon et al. [11], [12] presented a probabilistic model of gaze imitation where estimated gaze vectors are used in conjunction with the saliency maps of the visual scenes to produce maximum a posteriori (MAP) estimates of objects looked at by the caregiver. The fused saliency map is based on the bottom-up saliency model proposed by Itti et al. [13] combined with the top-down saliency map representing the preferences for objects learned from repeated experiments with a specific caregiver. To our knowledge, their work is the first to integrate the saliency learning mechanism into the gaze following framework. Still, a biologically plausible account of the development of gaze estimation skill is missing from their work.

Doniec et al. [14] used reaching and pointing gestures to change the problem of development of gaze following behavior into a supervised learning problem. Their proposed model is a two-stage learning process. First the robot learns to reach and point to some target. Then the robot uses its pointing skills to guide the caregiver's attention to a specific object in the scene. This idea of infant guiding the attention of the caregiver would work well in the cases when the object is close to the infant so that the pointing does not become ambiguous. Also it would be useful when the infant wants to transfer his intention to the caregiver. However their model does not address the issues such as following the caregiver's gaze to an object outside the infant's FOV or caregiver not following the infant's gestures.

The issues mentioned above are handled by our model. The rest of the paper is organized as follows. Section II first explains the visual processing, and then the reinforcement learning mechanism, and lastly how they are integrated. Section III describes the experiment setup and the results. The last section discusses the results and some possible directions for future work.

II. SYSTEM DESCRIPTION

In our previous works [5], [6], [7] we proposed a computational model of the development of the gaze following behavior and demonstrated simulation results that validated the plausibility of our model. Here we have implemented our model on our lab's robot head platform and we show the feasibility of our model in a real-world environment.

For a robotic implementation one also needs to handle the visual processing of the salient objects and head poses. The object processing module detects and estimates the depth or distance of the detected object. The face processing module detects faces, distinguishes different head poses and estimates the depth of the caregiver's face. Outputs from the two visual processing modules along with the information obtained from the motor control module are used to compute a state vector $\mathbf{u}$ which comprises a body-centered saliency map $\mathbf{s}$ and the caregiver's head pose $\mathbf{h}$, shown in Fig. 4. This state vector is fed to the reinforcement learning module and the module computes which action a to take to maximize its reward, which is just the saliency of the object itself. Once an action is complete, the reward r is computed and the internal parameters are updated accordingly.

A. Object processing module

The goal of the object processing module is to detect salient objects in the scene and estimate objects' depth. We picked a couple of household objects and trained the module to detect them as salient versus any other objects that may also be present in the scene. Objects are learned in a semi-autonomous fashion as in [4] where we simply show the objects one at a time to the robot head. The robot head detects the presence

of the object in its visual field and starts to actively track the object while the caregiver shows the object in varying poses and scales. Vergence control movements put the object at approximately the same location in both left and right images and stereo information is used to segment the object from the background. A collection of these segmented images is used for learning the object representations which is a collection of 2-dimensional views of an object.

Harris corner detectors are used for interest point/region detection and 40-dimensional Gabor wavelets are used as local features or texture descriptors. We cluster the segemented features to create a fixed size dictionary using the K-means algorithm. The Hough transform is used for encoding the feature positions in each view of the object. The approximate kd-tree algorithm is used for fast nearest neighbor search of features in the dictionary. Object detection is carried out using the left image. Once an object is detected, a set of points belonging to the detected object in the left image is compared to all corner points found in the right image to compute the stereo disparity in image pixels. We fixed the two cameras in the parallel axis position. So the depth of the object is computed simply using the equation:

$$d = -f\frac{b}{z} \tag{1}$$

where d is the horizontal pixel disparity in a single left-right image pair, f is the focal length of the camera, b is the base distance between two cameras and z is the estimated depth of the object.

Note that the module only computes the relative location of the object from the center of the visual field. Later, at the integration stage, we will combine this relative location information with the output from the motor system to compute the body-centered spatial position of the detected object.

This pre-training of salient objects can be omitted if we assume that the objects are placed far apart from each other such that the robot can easily detect an object based on its spatial location. However in our model, multiple objects can appear in the close proximity making it almost impossible to distinguish an object from the other just using spatial information.

B. Face processing module

The goal of the face processing module is to detect a face, distinguish different head poses and estimate depth. Here we do not need to estimate the gaze direction. We only need to distinguish one pose from the other. Again we use the semi-autonomous approach [4] to train the face processing module.

The processing is divided into three stages, detection, pose discrimination and depth estimation. In the first stage, the module detects the presence of a face. We used the face detection function provided in the OpenCV library. Face poses are learned in a similar manner to the object processing module described above. We use scale-covariant Maximally-Stable Extremal Regions (MSER) operator as the interest point/region detector instead of the Harris detector because faces do not contain enough corner-like points which can be reliably detected through varying poses and scales [15].

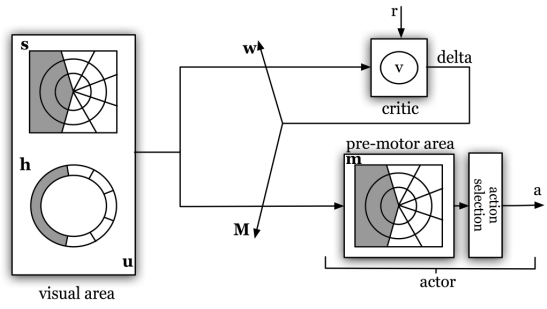

Fig. 4. Actor-critic reinforcement learning system. Gray areas in $\mathbf{s}$ indicates that the robot is unable to see areas beyond ± 240 degrees. In $\mathbf{h}$, it shows that the head pose discrimination is done approximately within ± 90 degrees. Again gray areas in $\mathbf{m}$ indicate the spatial bin locations that the robot is unable to reach.

The reason that we use our own face processing system instead of using an off-the-shelf head pose estimation system is first we do not need the full gaze direction estimation system which would also make the model biologically unplausible. After all, the gaze direction information is what the robot learns by interacting with the caregiver! Second, depending on the experimental parameters, it is very easy to increase or decrease the number of head poses that the system can distinguish since we are using the semi-autonomous approach [4]. Note that our face processing module has not been tested with multiple experimenters since that is not the main point of this paper.

C. Reinforcement learning module

Gaze following behavior is learned using a simplified version of the biologically plausible actor-critic reinforcement learning scheme described in [16], [6]. The state vector $\mathbf{u}$ from the visual system serves as an input to an actor-critic reinforcement learning module that controls the action as shown in Fig. 4. The state vector has size N_s which is sum of the size of the spatial bins N_a plus the number of head poses. Action a is a gaze shift to one of N_a spatial regions represented in a body-centered coordinate system. N_a is the total number of possible actions that the system can take and is also identical to the number of spatial bins present. Reward $r_a(t)$ is obtained as the saliency of the position to which attention is directed after a gaze shift has been made and the saliency map $\mathbf{s}$ has been updated.

The critic's role is to estimate the value of the current state $v(t)$ and compute the temporal difference error $\delta(t)$ using the estimated value of the current state $v(t)$ and the reward $r_a(t)$ of taking an action a. The actor decides which action to take in a probabilistic fashion using the softmax decision rule:

$$P(a) = \frac{\exp(\beta m_a)}{\sum_{a'=1}^{N_a} \exp(\beta m_{a'})} \tag{2}$$

Each action a, $a = 1, ..., N_a$ gets assigned a probability

Fig. 5. Five different head poses and five objects used in the experiment.

value computed by the equation above where m_a is the action value corresponding to action a computed using $\mathbf{m} = \mathbf{Mu}$ and β is the inverse temperature parameter that controls the amount of exploration vs. exploitation.

Weight vector $\mathbf{w}$ is used by the critic to estimate the value of the current state $v(t)$. It is updated according to:

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \eta\delta(t)\mathbf{u}(t) \tag{3}$$

where η is the learning rate.

The weight matrix $\mathbf{M}$ used by the actor to compute action values $\mathbf{m}$ is updated according to:

$$M_{a'b}(t+1) = M_{a'b}(t) + \eta(\delta_{aa'} - P[a'; u(t)])\delta(t)u_b(t) \tag{4}$$

where $P[a'; u(t)]$ is the probability of taking action a' in state $u(t)$ and $\delta_{aa'}$ is the Kronecker delta function defined as 1 if $a = a'$ and 0 otherwise.

Note that increasing the number of states in reinforcement learning may cause an explosion in the total number of iterations it takes to succesfully complete the training. Accelerating the reinforcement learning is an interesting research topic in its own right but was not addressed here since it was not required at this stage.

D. Integration

The integrated system consists of the three modules described in previous subsections plus the motor control subsystem as shown in Fig. 3. The integrated system also internally keeps a memory of object labels, locations and their saliencies that it has seen. Memory decay and habituation play an important role in the learning of gaze following. Outputs from two visual processing modules along with motor position are used to compute saliency map $\mathbf{s} = (s_1, ..., s_{N_a})^T$ which indicates the presence of visual saliency in a body-centered coordinate system around the robot head. The saliency or activation s_i at location i is computed as:

$$s_i(t) = \begin{cases} \sum_j f_j(t)\phi_{o_j}(t) : \text{location i visible} \\ ds_i(t-1) : \text{location i not visible} \end{cases} \tag{5}$$

where d is a memory decay speed parameter. The sum runs over objects j present in location i including the caregiver. $\phi_{o_j}(t)$ is the habituated saliency of object j. $f_j(t)$ is a foveation factor defined by:

$$f_j(t) = \exp(-\theta_{o_j}^2/\sigma_F^2) \tag{6}$$

where θ_{o_j} is the angle between the robot's line of sight and the object j and σ_F determines the range of attenuation. Habituation further decreases the perceived saliency according to:

$$\frac{d\phi_{o_j}(t)}{dt} = \frac{\alpha_H}{\tau_H}(\Phi_{o_j} - \phi_{o_j}(t)) - \frac{1}{\tau_H}S_{o_j}(t) \tag{7}$$

where Φ_{o_j} is the unhabituated original saliency of object j and τ_H is a time constant for specifying the rate of habituation. The saliencies of the objects Φ_{o_j} are drawn from an exponential probability distribution with mean equal to 1 as in [5]. $S_{o_j}(t)$ is equal to Φ_{o_j} if the robot is looking at object j at time t and 0 otherwise.

The saliencies of the caregiver and infant are set to 2 which means most of the objects will have saliencies lower than that of the infant or the caregiver. A real infant probably uses more sophisticated saliency computation such as assigning higher/lower values for food/non-food, mother/stranger, etc, that may also depend on the internal state (e.g. hungry/bored/sleepy) of the infant. Higher saliency values will make it more likely to attend to the object for longer duration of time whereas the robot will quickly loose interest in an object with low saliency value.

III. EXPERIMENTS

Our model was tested on the 9 degrees of freedom (DOF) robot head developed for studying development and learning in the social context [17]. The robot has 2 CCD cameras capable of capturing 640×480 resolution color images up to 30 frames per second. Each camera has approximately 60° horizontal and 40° vertical FOV. The robot's neck is capable of making ±90° horizontal movement. For this experiment, we fixed the two cameras with parallel axes in order to make depth computation easier. Since our robot head has limited visual and motor capabilities compared to that of the model infant in [5], we adjusted some of the experimental parameters.

A. Learning about objects and head poses

The experiment was carried out in two stages. First we trained the two visual processing modules to detect "salient" objects and distinguish different head poses in a semi-autonomous manner as described in [4]. Our semi-autonomous learning scheme makes it very simple to train the system.

For objects, the experimenter simply held the object in front of the robot head and rotated it arbitrarily while the robot detected, tracked, segmented and learned about the object. 500 left-right image pairs are captured for training of each object. The system runs at roughly 10 frames per second so capturing 500 image pairs takes about one minute. We chose 5 different household objects to be used as salient objects as shown in Fig. 5. An identical object can appear multiple times in our gaze following experiment setup as long as it appears in different spatial locations.

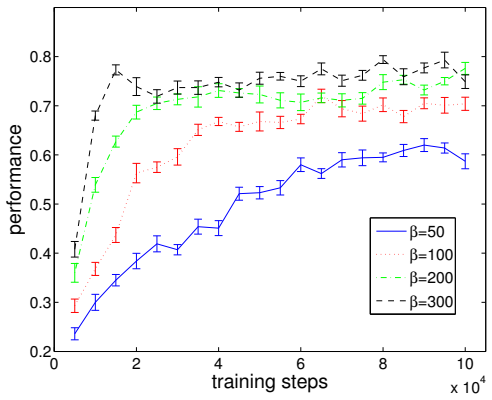

Fig. 6. Gaze following performance over training time. Error bars indicate the standard error of the mean (10 simulations).

Similarly the experimenter presented his face to the robot head at angles ±90, ±45, and 0. Starting from the angles mentioned above, the experimenter rotated his head arbitrarily but with care so the head poses do not cross the boundary. Again 500 left-right image pairs are captured for training. Each group of images are given labels such as 'Pose0', 'Pose1' and so on without any explicit head direction information.

After training the two visual processing modules with the captured 500 left-right image pairs per object or per pose, we tested the recognition performance on the 100 left-right image pairs captured separately for testing. For the face processing module, a 4K feature dictionary was built using K-means clustering. Recognition performance was 96.4%. For the object processing module, the pose-invariant recognition performance was 75.1% with a 64K feature dictionary. Low recognition performance for the objects is due to the fact that objects have several views such as back-side of a can or a box which contains texts with small fonts that are hard to distinguish in limited resolution images.

B. Learning the gaze following behavior

The second stage of the experiment the system learns the gaze following behavior. One problem is that reinforcement learning usually takes a large number of iterations to complete. In our simulations in [5], it took 900K iterations or 125 hours of non-stop interaction if one would assume that one iteration (gaze shift) takes half a second.

With the two visual processing modules combined, each iteration took about 5 seconds. So even if the simulation runs for 100K iterations, it would mean 139 hours of non-stop interaction excluding the time it takes to reorganize the objects around the room. Thus we found doing the experiment online, although possible, to be cumbersome and difficult. So instead we captured the scenes the robot sees when it is attending to one of the N_a spatial bins.

We divided ±120 degree horizontal range into 5 directions. Depth ranges are divided into 3 regions at approximately 0.25, 0.5 and 0.75 meters. Objects placed further than 0.75 meters appeared too small in the captured images and therefore were not detected reliably. We captured multiple images of the same spatial bin changing the object and head pose combinations. While capturing and testing the detection using two visual processing modules, we found out that the closer object often blocked the view of the object behind it making detection and depth estimation of the object behind difficult. To keep things simple in the offline training, we assumed that the robot only sees the front-most object when there were multiple objects in the same viewing direction. Also our captured scenes contained at most one object per spatial bin. The two restrictions mentioned above are for easier experiment setup and are not strictly necessary, however.

If not mentioned specifically, the experimental setup and parameters are identical to that of [5]. The objects are distributed in a uniform distribution and not Gaussian. That is, it is equally likely that an object appears in any of the spatial bins. The total number of objects present in each scene is obtained from a geometric probability distribution with average set to 1. Due to the two restrictions mentioned above, the maximum number of objects that the robot can perceive in one scene is 5 which means objects are present in every viewing direction. Thus if the average number of objects parameter $\overline{N_O}$ was set to 4 like in [5], the system failed to learn the correct behavior or associations between the head pose and the corresponding motor actions.

Note that the robot only has 60° horizontal FOV compared to 180° for the model infant in [5]. This means that our robotic experiment is in some sense tougher than [5] since our robot does not get the chance to learn the easier associations between caregiver's head pose and objects inside the FOV. It must learn to associate to the objects outside its FOV from the start.

We ran the simulation for 100K iterations. Internal parameters for the reinforcement learning are stored every 5000 iterations. After the training we used these stored values to test the performance of the system. We stopped the adaptation of the parameters during the testing. Fig. 6 shows the gaze following performance over the training time. Each trial started with the caregiver and the robot looking at each other with no objects present in the scene. Then the caregiver turned his head to look at one randomly chosen bin location. If the robot's first head turn was consistent with the caregiver's gaze direction, it was counted as a correct movement. Performance is measured as the proportion of correct movements in 100 test trials. Note that the robot's movement or action is selected based on the soft-max decision rule. The robot does not always select the best action that produces the maximum expected reward, but often chooses other actions to continue exploring the environment. The test was repeated 10 times and the error bars indicate the standard error of the mean.

We changed the value of β during the test trials. Larger value of β resulted in more exploitation of trained knowledge which in turn improveed performance as can be seen in Fig. 6. However, large β during training can actually harm the proper learning of the skills due to limited exploration of the environment.

Fig. 7 shows the connection weights from a specific head

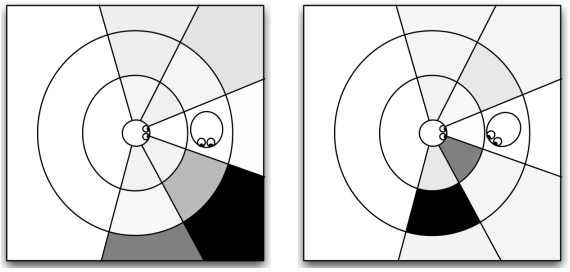

Fig. 7. Illustration of connection weights from the head pose to the action values $\mathbf{m}$. Caregiver is looking at 90° (left) and at 45° (right).

pose to corresponding action values $\mathbf{m}$ at iteration 50K. With each head pose, several spatial bins get activated which roughly correspond to the line of sight of the caregiver.

IV. Discussion

In this paper we have presented a robotic system capable of learning the gaze following behavior using a computational model developed earlier. Some of the limitations of the robot head made the problem more challenging. Nonetheless the system learned to make correct associations between the caregiver's head pose and the corresponding motor actions. Our goal is to provide a simple and parsimonious account of many of the experimental findings about the development of gaze following in human infants. Our discrete model shows a proof of concept. In the future we may use more sophisticated continuous space reinforcement learning schemes such as [18].

Since the reinforcement learning algorithm is slow, the learning was carried out offline. But nothing stops us from doing the training online as long as the experimenter can rigorously repeat the experiment, perhaps sitting in front of the robot head for days. We claim that the "long" training time of a couple of days is not a crucial shortcoming since a real infant would have ample time to interact and learn from the caregiver during the first two years of his life. In reality, additional cues such as motion of the caregiver's face, motion of the object (e.g. caregiver shaking the object) and auditory or tactile sensory inputs such as the caregiver talking to or touching the infant may be used to expedite the learning process. Nagai [10] and Doniec et al. [14] have used some of the additional cues mentioned above in their models. We did not use the eye direction cue since we did not have a gaze detection module readily available. But it it just an additional piece of visual information that can be easily integrated into our framework as shown in [5]. We hope to develop such a module and integrate it into our robotic system in the future.

Our model solved many of the issues present in other models such as gaze following to an object outside the FOV, etc (see Section I). Our work captures the interplay of perceptual skills, reward-driven behavior learning, habituation and a structured social environment in an integrated framework. Also extending the internal world model to 3D space will not be a problem. The face processing module can easily be enhanced to distinguish changes in tilt angles as well as pan angles of the caregiver's head pose. Therefore we plan to extend our model to 3D space in the future.

Our gaze following mechanism can also be used as a front-end for our semi-autonomous learning framework [4]. Instead of detecting the presence of an object based solely on its depth as in [4], the robot can attend to and learn about the object that the caregiver is looking at.

An interesting open question is whether the ability to distinguish different head poses is a prerequisite for the development of gaze following behavior or whether the two skills develop jointly and simultaneously. Ultimately, instead of carrying out the experiment in multiple stages, we want to build a system where skills for distinguishing different head poses, learning about salient objects, and gaze following are developing concurrently and where the progression of an individual skill influences the learning of the others.

References

[1] G. Butterworth and E. Cochran, "Towards a mechanism of joint visual attention in human infancy," *International Journal of Behavioral Development*, vol. 3, pp. 253–272, 1980.

[2] G. Butterworth and N. Jarrett, "What minds have in common is space: Spatial mechanisms serving joint visual attention in infancy," *British Journal of Development Psychology*, vol. 9, pp. 55–72, 1991.

[3] R. Brooks and A. Meltzoff, "The development of gaze following and its relation to language," *Dveleopmental Science*, vol. 8, no. 6, pp. 535–543, 2005.

[4] H. Kim, E. Murphy-Chutorian, and J. Triesch, "Semi-autonomous learning of objects," in *Proc. V4HCI workshop, CVPR*, 2006.

[5] J. Triesch, H. Jasso, and G. Deak, "Emergence of mirror neurons in a model of gaze following," *Adaptive Behavior*, vol. 15, no. 2, pp. 149–165, 2007.

[6] H. Jasso, J. Triesch, C. Teuscher, and G. Deak, "A reinforcement learning model explains the development of gaze following," in *Proc. Int. Conf. on Cognitive Modeling (ICCM)*, 2006.

[7] H. Jasso, "A reinforcement learning model of gaze following," Ph.D. dissertation, University of California, San Diego (UCSD), 2007.

[8] G. Matsuda and T. Omori, "Learning of joint visual attention by reinforcement learning," in *Proc. Int. Conf. on Cognitive Modeling (ICCM)*, 2001.

[9] Y. Nagai, K. Hosoda, A. Morita, and M. Asada, "A constructive model for the development of joint attention," *Connection Science*, vol. 15, no. 4, pp. 211–229, 2003.

[10] Y. Nagai, "The role of motion information in learning human-robot joint attention," in *Proc. Int. Conf. on Rob. and Aut. (ICRA)*, 2005.

[11] A. Shon, D. Grimes, C. Baker, M. Hoffman, S. Zhou, and R. Rao, "Probabilistic gaze imitation and saliency learning in a robot head," in *Proc. Int. Conf. on Rob. and Aut. (ICRA)*, 2005.

[12] M. Hoffman, D. Grimes, A. Shon, and R. Rao, "A probabilistic model of gaze imitation and shared attention," *Neural Networks*, vol. 19, pp. 299–310, 2006.

[13] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE PAMI*, vol. 20, no. 11, pp. 1254–1259, 1998.

[14] M. Doniec, G. Sun, and B. Scassellati, "Active learning of joint attention," in *Proc. Int. Conf. on Humanoid Robotics (Humanoids)*, 2006.

[15] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremum regions," in *Proc. British Machine Vision Conference (BMVC)*, 2002.

[16] J. Triesch, H. Jasso, and G. Deak, "Emergence of mirror neurons in a model of gaze following," in *Proc. Int. Conf. on Development and Learning (ICDL)*, 2006.

[17] H. Kim, G. York, G. Burton, E. Murphy-Chutorian, and J. Triesch, "Design of an anthropomorphic robot head for studying development and learning," in *Proc. Int. Conf. on Rob. and Aut. (ICRA)*, 2004.

[18] K. Doya, "Reinforcement learning in continuous time and space," *Neural Computation*, vol. 12, no. 1, pp. 219–245, 2000.