Neuromorphic Computing and Engineering

PAPER • OPEN ACCESS

Learning torsional eye movements through active efficient coding

NEUROMORPHIC
Computing and Engineering

OPEN ACCESS

RECEIVED
13 January 2022

REVISED
25 July 2022

ACCEPTED FOR PUBLICATION
28 July 2022

PUBLISHED
18 August 2022

PAPER

Learning torsional eye movements through active efficient coding

Qingpeng Zhu[1], Chong Zhang[2], Jochen Triesch[3] and Bertram E Shi[1,*]

[1] Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, People's Republic of China
[2] Tencent Robotics X, Shenzhen, People's Republic of China
[3] Frankfurt Institute for Advanced Studies, Frankfurt am Main, Germany
[*] Author to whom any correspondence should be addressed.

E-mail: eebert@ust.hk

Keywords: developmental robotics, Listing's law, bioinspired robots, eye movements, self-calibration

Abstract

The human eye has three rotational degrees of freedom: azimuthal, elevational, and torsional. Although torsional eye movements have the most limited excursion, Hering and Helmholtz have argued that they play an important role in optimizing visual information processing. In humans, the relationship between gaze direction and torsional eye angle is described by Listing's law. However, it is still not clear how this behavior initially develops and remains calibrated during growth. Here we present the first computational model that enables an autonomous agent to learn and maintain binocular torsional eye movement control. In our model, two neural networks connected in series: one for sensory encoding followed by one for torsion control, are learned simultaneously as the agent behaves in the environment. Learning is based on the active efficient coding (AEC) framework, a generalization of Barlow's efficient coding hypothesis to include action. Both networks adapt by minimizing the prediction error of the sensory representation, subject to a sparsity constraint on neural activity. The policies that emerge follow the predictions of Listing's law. Because learning is driven by the sensorimotor contingencies experienced by the agent as it interacts with the environment, our system can adapt to the physical configuration of the agent as it changes. We propose that AEC provides the most parsimonious expression to date of Hering's and Helmholtz's hypotheses. We also demonstrate that it has practical implications in autonomous artificial vision systems, by providing an automatic and adaptive mechanism to correct orientation misalignments between cameras in a robotic active binocular vision head. Our system's use of fairly low resolution (100 × 100 pixel) image windows and perceptual representations amenable to event-based input paves a pathway towards the implementation of adaptive self-calibrating robot control on neuromorphic hardware.

1. Introduction

Designing artificial intelligent agents capable of autonomously and dynamically adapting their control strategies during interaction with the environment is still an open problem. This requires interdisciplinary knowledge, including robotics, machine learning, biology, and neuroscience. However, many intelligent robotic systems have ignored biological inspiration. One advantage of biological systems is that they can provide more robust performance. Bio-inspired neural learning systems may also benefit neuroscientists, by providing a deeper understanding of why and how biological agents develop and adapt to changes in the environment.

One main challenge for intelligent agents is unifying the development of the sensory representation and behavioral control. To address this problem, active efficient coding (AEC) provides a framework for the co-development of perception and behavior [54]. AEC is an extension of Barlow's efficient coding hypothesis [1], which states that sensory representations adapt to encode the input stimuli efficiently. One of the important predictions of efficient coding is that the neural representations depend upon the statistics of the sensory input. Motivated by the dependence of the input statistics the agent's actions, AEC extends Barlow's hypothesis to include behavior. AEC hypothesizes that perception and action both evolve to optimize the same internal reward: fidelity of the neural perceptual representation subject to a sparsity constraint. We interpret sparsity as a bound on the metabolic cost [28]. AEC uses unsupervised learning to learn the perceptual representation and reinforcement learning to learn behavioral control. Although reinforcement learning is classically considered to lie somewhere between supervised and unsupervised learning, AEC's use of an internally generated reward makes the entire framework self-supervised. The motivation and potential significance of AEC are similar to those of foundation models [3]. It exhibits both emergence and homogenization: emergence because perception and behavior are induced by seeking a generic, rather than task specific, goal and homogenization because AEC can account for the development of a wide range of reflexive eye movement behaviors. Examples include vergence [22, 54], saccades [56], accommodation [42], smooth pursuit [40, 53], optokinetic nystagmus [52], and their combinations [9, 20, 47, 57].

Biological vision systems are ideal for the study of the development of the sensory representation and behavioral control. Their sensory representations develop without external supervision to support a wide range of visually guided behaviors. Among the first to emerge are reflexive automatic behaviors, such as binocular vergence, which aligns the two eyes on the same point, and the optokinetic reflex, which stabilizes the image over time, and torsional eye movements, which rotate the eye slightly around the line of sight.

During gaze shifts, torsion is actively controlled and follows Listing's law [19, 30]. Listing's law states that eye torsional angles are a function of gaze direction, i.e., pan, tilt and vergence. Unlike visually-driven eye movements like smooth pursuit and saccades, torsion described by Listing's law is driven by the motor efference copy. The advantages of following Listing's law have been well documented. Unless torsion is actively controlled, orientation misalignments are inevitable, since changes in the vergence angle change the epipolar geometry [37]. Proper torsional control can ensure that the retinal search zones required for binocular matching are unchanged.

To our knowledge, no models have been proposed to explain how torsional eye control develops, nor how it can adapt dynamically to perturbations introduced by, e.g., prisms or by muscle palsy due to stroke. This study makes several contributions to address this gap:

- We propose the first developmental model of torsional eye movements. The model does not require any external rewards or supervision. The learning of both the sensory representation and behavior are driven by the same intrinsic motivation: efficient coding.
- We show that the torsional control behaviors develop so that they are consistent with the binocular extension of Listing's law (L2). Thus, AEC provides a parsimonious account for the emergence of Listing's law.
- We show that this model can be applied to an active vision system on a humanoid robot, the iCub, enabling it to overcome an orientation misalignment between the left and right cameras without any manual calibration.

2. Background

There are two types of torsional eye movements: cycloversion and cyclovergence. Cycloversion is a conjugate movement, which rotates the eyes around their lines of sight in the same direction. Cyclovergence is a disconjugate movement, which rotates the eyes in opposite directions.

Torsional eye control has two components: a gaze direction dependent component and a visually induced component. This paper considers only the gaze direction dependent component, which is virtually independent of the visual environment [16]. Donders measured eye torsion by comparing afterimages and found that the eye assumes a unique torsional angle for each gaze direction. Listing's law specifies that angle. Its binocular extension (L2) takes into account the relative positions (vergence) between the two eyes [36]. Listing's law and L2 are followed during many eye movements, including fixation, saccades, smooth pursuit, and vergence [24, 43, 44].

In L2, the cycloversion angle C_s and cyclovergence angle C_g are specified according to the pan P, vergence V and tilt T angles by:

$$\begin{aligned} C_s &= -\frac{1}{2}PT \\ C_g &= \frac{1}{2}VT. \end{aligned} \tag{1}$$

Angles are specified in Helmholtz coordinates. As the cycloversion angle C_s depends only on the pan and tilt angles, it applies equally when fixating on objects in both the near and far fields. Its magnitude increases with increasing eccentricity. On the other hand, since the cyclovergence angle C_g scales with vergence, it is most pronounced for objects in the near field.

Biological findings suggest that cycloversion and cyclovergence are controlled by different neural mechanisms. Porrill *et al* [32] recorded the variation of the cycloversion and the cyclovergence in human subjects. While cycloversion exhibits dramatic intrasubject and intersubject variation, cyclovergence is more stable. Van Rijn *et al* [45] measured the torsional eye movements in response to visual pattern. They found out that cycloversion and cyclovergence have different response latencies.

There have been two primary hypotheses about why the human eye rotation follows Listing's law: (1) optimization of motor efficiency and (2) simplification of visual processing. Fick and Wundt argued that by following Listing's law, the eyes achieve the desired gaze direction with minimal muscle effort [11, 49]. They proved that Listing's law can be derived based on the principle of minimizing the rotational eccentricity. Helmholtz and Hering both proposed that Listing's law makes the eyes best suited for the purpose of vision [14, 15]. Following Listing's law maintains constancy of the orientation of the object projected on the retina, simplifying subsequent visual processing.

Most likely, Listing's law serves a combination of both goals [48]. Violations of Listing's law during sleep and during the vestibular ocular reflex argue against a pure motor explanation. Torsional behavior during vestibulo-ocular reflex seems to be a compromise between motor efficiency and image stabilization. Schreiber *et al* [37] argue that L2 is a trade-off between minimizing rotational eccentricity and minimizing search zones for binocular correspondence.

Prior modelling work focused primarily on how Listing's law might be implemented [5, 13, 34], but not on how and why it develops, the focus of this paper. The model we present builds upon a preliminary model reported in [59]. That model only accounted for the emergence of cyclovergence. The model presented here accounts for the simultaneous emergence of both cyclovergence and cycloversion. It extends past work on AEC to a new visual behavior and to a new input control modality: the motor efference copy. Past visual behaviours modelled by AEC were driven by visual input.

3. Model

Figure 1(a) shows the architecture of the active vision system, which consists of three main parts: the perceptual representation mechanism, the cycloversion control mechanism, and the cyclovergence control mechanism.

During each fixation, the perceptual representation mechanism encodes the image inputs (left/right images at pre-/post-saccade) using a set of feature extractors learned by the generative adaptive subspace self-organizing map (GASSOM) algorithm [7]. The GASSOM algorithm can learn invariant features based on unlabeled input sequences generated by a realistic eye movement model that includes both saccades and fixational eye movements. For each fixation, the input to the perceptual representation mechanism consists of two stereo image pairs, one sampled before and one sampled after a saccade. These image windows are down-sampled to represent input at two different scales [22, 23].

The cycloversion and cyclovergence mechanisms are implemented using two separate neural networks, which learn the mapping from an efference copy of commanded eye position (vergence, pan, and tilt angles) to the cycloversion and cyclovergence angles.

The parameters of the GASSOM model and the cycloversion/cyclovergence control policies are initialized randomly and learned simultaneously. During training, the agent executes saliency-driven saccades, which direct the eyes towards salient image locations [35, 51]. Saliency-driven saccades speed up the learning of both the perceptual representation and the behavioral control, in comparison to random saccades [58].

We describe our environmental setup and the learning mechanisms in more detail below.

3.1. Experimental setup

The active vision system was trained and tested in the iCub simulation platform [41]. The iCub is a humanoid robot whose active binocular vision system includes three degrees of freedom: vergence, pan, and tilt. The iCub eye coordinate system is described by a Helmholtz gimbal system. In our simulations, the iCub's head remained stationary and upright. We only controlled the eye movements. Due to the constraints of the iCub simulator, we cannot rotate the cameras around the torsional axis. Cyclorotation was therefore implemented by rotating the left and right images around their centers in software.

In the simulation environment, a fronto-parallel plane was placed at varying depths in front of the agent. The plane was large enough that at all depths it covered both eyes' entire fields of view. The plane was textured with natural images taken from the McGill image database [27]. The depth of the fronto-parallel plane was sampled uniformly from the range between 0.3 m and 1.5 m. Every 100 iterations in the training, we created a new scene by choosing a new image and a new depth. For each scene, the agent saccaded to ten different locations on the plane and fixated on each for ten iterations.

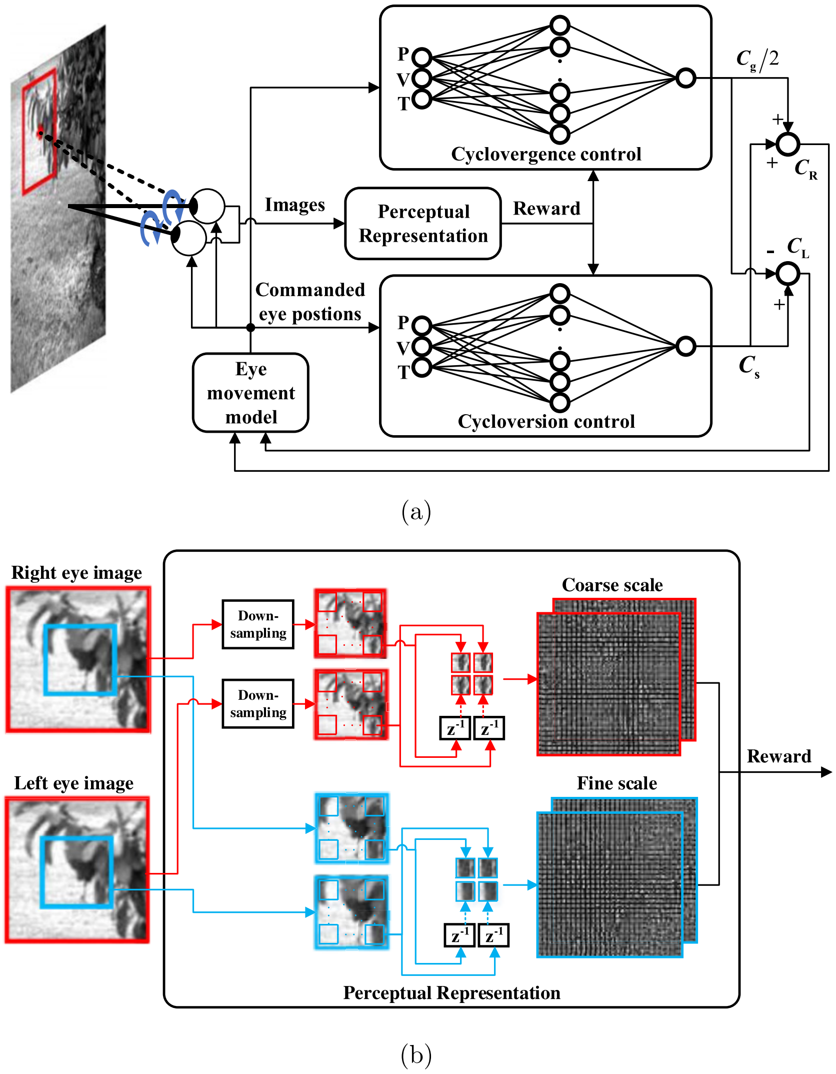

Figure 1. (a) The architecture of the active vision system. The plane on the left indicates one example of the input presented in the training environment of the iCub simulator. The solid/dashed lines represent the gaze of the iCub before/after saccading to the target point, which is denoted by a red cross on the image plane. The right-hand side shows the architecture of our proposed system, including the perceptual representation mechanism, the cyclovergence control mechanism, and the cycloversion control mechanism. *P*, *V*, *T* denote the pan, vergence and tilt angles, respectively. (b) The detailed architecture of the perceptual representation mechanism. The red and blue squares represent the sub-windows extracted at the coarse and fine scales, respectively. The solid and dashed lines represent the sub-windows obtained before and after the saccade to the target point, respectively.

Left and right eye images with size 320 × 240 pixels were generated by rendering the simulated environment based on the eye positions. The cameras' horizontal and vertical fields of view were 109° and 92°, respectively. We chose to work with fairly low resolution images in order to facilitate implementation on robotic platforms, where sensor resolution and compute power are typically limited. Images rendered by our setting of the iCub simulator are similar to those that would be acquired by neuromorphic event-based cameras, which are attracting increasing interest for robotic applications. For example, the DAVIS346 event camera[4] has a resolution of 346 × 260. With a 2.1 mm focal length lens, its horizontal and vertical FOVs are 113° and 97.7°.

The saccade targets were chosen based on a saliency map generated by the global binocular attention based on information maximization algorithm proposed in [58]. We computed the saliency map by first downsampling the original images by a factor of two and then calculating a saliency value for each 10 × 10 patch extracted from the down-sampled image with a stride of one (resulting in a 151 × 111 array of saliency values). We then obtained a saliency map at the original 320 × 240 image resolution by zero padding to 160 × 120 and up-sampling by a factor of two.

The visual input to the perceptual representation mechanism was obtained by extracting 110 × 110 pixel square windows centered at the saccade target points in the stereo image pairs before and after the saccade.

[4] https://inivation.com/wp-content/uploads/2019/08/DAVIS346.pdf

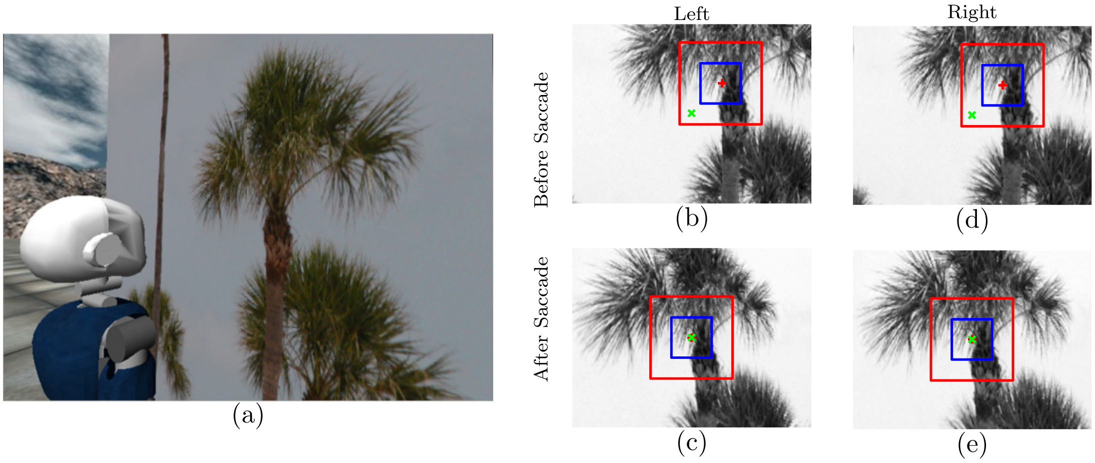

Figure 2. (a) A screenshot of the training environment in the iCub simulator. (b)–(e) Example images illustrating how the inputs of the perceptual representation mechanism were obtained. (b) and (d) Show left and right eye images before saccading to the target. (c) and (e) Show left and right eye images after saccading to the target. The red squares show the 110 × 110 pixel images sent to the perceptual representation mechanism. The blue squares show the area covered by the fine-scale input. The green 'X' at the center of the images shows the intersection with the optic axis, which we assume to coincide with the visual axis.

Figure 2 shows an example of the input to the perceptual representation. Before the saccade, the saccadic target (the center of the extracted windows) is off center. After the saccade, the saccadic target is at the image center.

In order to investigate the system's ability to self-calibrate, we ran simulations with different constant orientation offsets between the left and right eyes. We varied the orientation offsets from $-15°$ to $15°$ with a spacing of $5°$. We defined the orientation offset as the orientation of the right eye minus the orientation of the left eye, where positive orientation follows the right-hand rule around the optic axis.

3.2. Perceptual representation

Figure 1(b) shows the architecture of the perceptual representation mechanism. To simulate peripheral (low resolution/coarse scale) and foveal (high resolution/fine scale) vision, we extract two sub-windows from each of the four 110 × 110 pixel windows. The coarse-scale sub-window is obtained by down-sampling the original windows by a factor of 2. The fine-scale sub-window is the 55 × 55 pixel sub-window at the center of the original window. The coarse-scale sub-window covers 51 degrees of visual angle, extending to cover most of the near periphery. The fine-scale sub-window covers 27 degrees of visual angle, which includes the FOV of the macula, which is responsible for central vision [39]. The different roles of the central and peripheral windows have been investigated in a hierarchical AEC model [55]. The two regions play complementary roles. The high resolution fovea can drive precise short range movements. The lower resolution periphery supports coarser long range movements.

Each 55 × 55 pixel sub-window is further separated into a 10 × 10 array of 10 × 10 pixel patches with stride of five. Larger patch sizes increase the range of possible spatial frequencies represented, but also the computational complexity. We chose the patch size and stride for consistency with past work on AEC for vergence control [54], whose patch sizes were consistent with those used in seminal sparse coding models of visual cortical field development [28, 29]. Smaller strides improve shift-invariance, at the expense of computational complexity. The stride of 5 pixels ensures neighboring receptive fields overlap, as observed in the visual cortex [17]. The 50% overlap between receptive fields ensures that each point in space is covered by four receptive fields, as observed in retinal ganglion cells [4].

At each scale s and for the jth patch, we concatenate the image intensities from the left/right pre/post-saccade images into a 400-dimensional vector,

$$\mathbf{x}_{s,j} = \begin{bmatrix} \mathbf{x}_{\mathrm{L,Pre},s,j} \\ \mathbf{x}_{\mathrm{L,Post},s,j} \\ \mathbf{x}_{\mathrm{R,Pre},s,j} \\ \mathbf{x}_{\mathrm{R,Post},s,j} \end{bmatrix} \in \mathbb{R}^{400} \tag{2}$$

where $s \in \{\mathrm{C}, \mathrm{F}\}$ (C and F stand for the coarse and fine scale respectively) and the vectors, $\mathbf{x}_{\mathrm{L,Pre},s,j}$, $\mathbf{x}_{\mathrm{L,Post},s,j}$, $\mathbf{x}_{\mathrm{R,Pre},s,j}$, $\mathbf{x}_{\mathrm{R,Post},s,j}$ contain the pixel intensities from the pre/post-saccade left/right image patches. The four vectors are separately normalized to have zero mean and unit variance.

By concatenating the pre/post-saccade images into a single vector, our model adopts a non-uniform timescale, since the duration of a saccade varies according to its duration. However, we do not believe this to be a significant limitation. Although saccade durations vary, due to their fast speed, they are short, ranging from about 20 ms to 100 ms. It has been shown that the visual cortex can retain information about a preceding stimulus over several 100 ms [25]. Visual input during a saccade is ignored, in a phenomenon known as saccadic suppression. Thus, we believe that our assumption that the visual system can store information across a saccade with little disruption due to the rapidly changing visual input during the saccade is reasonable.

The perceptual representation mechanism encodes each input $\mathbf{x}_{s,j}$ using the GASSOM model, a probabilistic sparse generative model [7] of efficient coding, which can be extended to process event-based input [6]. Sparsity is imposed by assuming that each input $\mathbf{x}_{s,j}$ is assumed to be generated as the sum of a vector sampled from one of N subspaces and a small noise vector orthogonal to the subspace. The generating subspace is assumed to evolve over time according to a first-order Markov chain. The number of subspaces $N = 900$ was larger than the number ($\sim$300) of subspaces in prior work involving instantaneous binocular input, e.g. [55, 56]. We were unable to learn a policy using only $\sim$300 subspaces, likely due to the increased dimensionality/complexity of the input, which includes both pre- and post-saccade patches. We were able to learn a policy with 900 subspaces, but the optimal number of subspaces for a given input and given task is still an open problem.

The outputs of the perceptual representation are the squared lengths of the projections onto the N subspaces. Each subspace is two-dimensional, defined by two orthonormal basis vectors contained in the columns of a matrix $\Phi_{s,n} \in \mathbb{R}^{400\times 2}$, where s denotes scale (each scale has its own set of subspaces) and $n \in \{1, 2, \ldots, N\}$. Thus, the output of the perceptual representation for $\mathbf{x}_{s,j}$ is

$$y_{s,j} = \{\left\|\Phi_{s,n}^{\mathrm{T}}\mathbf{x}_{s,j}\right\|^2\}_{n=1}^{N}. \tag{3}$$

We chose the subspace dimensionality to be two, so that the squared length of the subspace projection is similar to the binocular energy, which is commonly used to model the responses of orientation, scale, and disparity tuned binocular complex cells in the primary visual cortex [26]. The binocular energy is the sum of the squared outputs of two linear binocular simple cells, whose outputs are the dot products between their spatial receptive fields and the binocular input. In the GASSOM, each basis vector is analogous to a simple cell receptive field profile.

3.3. Cycloversion and cyclovergence control

Cycloversion and cyclovergence control are implemented by two neural networks that map the state $s = [P, T, V]^{\mathrm{T}}$ consisting of the efference copies of the pan (P), tilt (T) and vergence (V) commands to cyclorotation commands C_{s} and C_{g}. Both neural networks have the same three-layer structure. The input layers contain three neurons, corresponding to the commanded pan (P), tilt (T) and vergence (V) angles. The hidden layers contain ten neurons with hyperbolic tangent activation functions. The output layers contain a single linear neuron, which encodes the mean of a Gaussian distribution over the action space: the desired cycloversion angle $\pi_{\mathrm{s}}(\mathbf{s})$ or cyclovergence angle $\pi_{\mathrm{g}}(\mathbf{s})$. During training, actions are chosen by sampling from Gaussian distributions with fixed variance, i.e., $C_{\mathrm{s}} \sim \mathcal{N}(\pi_s(\mathbf{s}), \sigma)$, $C_{\mathrm{g}} \sim \mathcal{N}(\pi_g(\mathbf{s}), \sigma)$, where the standard deviation σ is set to 1°. This probabilistic sampling introduces exploration, which is essential in reinforcement learning. During testing, actions are set to the mean values.

Finally, the new cyclorotation angles of the left and right eyes, C_{L} and C_{R} are given by:

$$\begin{array}{c} C_{\mathrm{L}} = C_{\mathrm{s}} - C_{\mathrm{g}}/2 \\ C_{\mathrm{R}} = C_{\mathrm{s}} + C_{\mathrm{g}}/2 \end{array}. \tag{4}$$

3.4. Learning via active efficient coding

Following the AEC framework, the parameters in the GASSOM perceptual representation and the control neural networks are learned simultaneously during behavior to optimize the fidelity of the perceptual representation.

We measure the perceptual fidelity using the reconstruction error, which is defined as the squared length of residual between the input $\mathbf{x}_{s,j}$ and its projection onto the best-fitting subspace:

$$E_{s,j} = \left\|\mathbf{x}_{s,j} - \Phi_{s,m_{s,j}}\Phi_{s,m_{s,j}}^{\mathrm{T}}\mathbf{x}_{s,j}\right\|^2, \tag{5}$$

where $m_{s,j}$ is the index of the subspace that has the lowest reconstruction error for encoding the input $\mathbf{x}_{s,j}$:

$$m_{s,j} = \arg\max_n \left\|\Phi_{s,n}^{\mathrm{T}}\mathbf{x}_{s,j}\right\|. \tag{6}$$

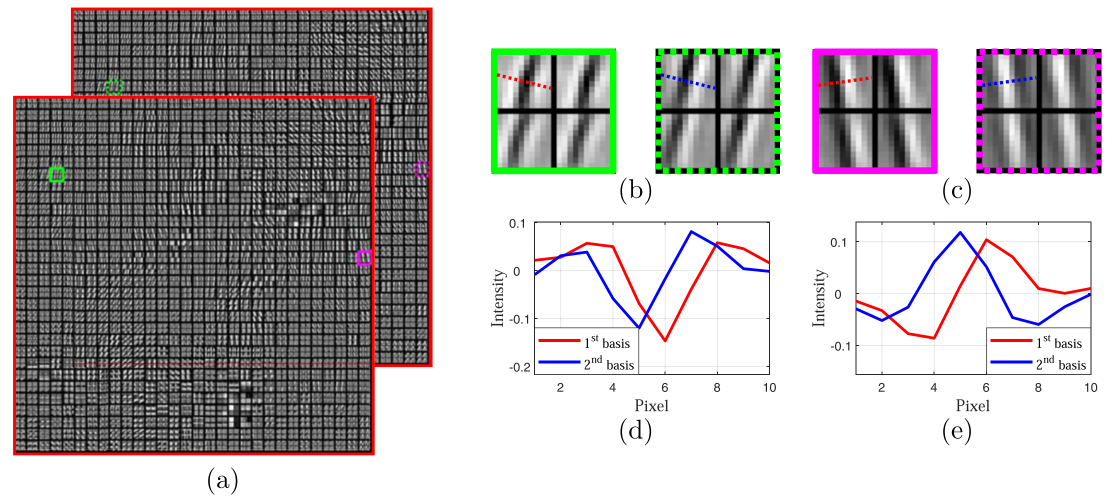

Figure 3. (a) The set of all basis vectors of the 900 learned feature extractors (coarse scale) at the end of one training trial. Each feature extractor has two basis vectors, shown at corresponding locations on the front and back layers. For each layer, the 900 learned basis vectors are presented in a 30 × 30 array. (b) and (c) Two example basis vectors taken from the positions highlighted in green (b) and magenta (c) in (a). The first basis vector is outlined with a solid line, the second with a dotted line. Each basis vector has four components. The left and right columns correspond to left and right images. The top and bottom rows correspond to pre- and post-saccade frames. (d) and (e) The cross-sectional views of one component of a basis vector taken along the red and blue lines shown in (b) and (c), respectively. Red and blue lines in (b) and (c) are in the direction perpendicular to the preferred orientation of the basis vector.

The smaller the reconstruction error, the better the model matches the input.

The parameters of the GASSOM algorithm are initialized randomly, then learned according to the maximum likelihood algorithm described in [7], which is equivalent to minimizing the reconstruction error.

The weights of the cyclorotation control networks are initialized randomly, then evolve according to a modified version of the natural actor-critic reinforcement learning algorithm [2]. In our system, we include only the actor, but not the critic, network. This is because actions (torsional angles) do not affect the state (gaze direction). Thus, it is not necessary to estimate the value of each state. The reinforcement learning seeks policies that maximize the discounted sum of future rewards, where the reward at each time instant is the negative reconstruction error averaged over all scales and over all patches.

$$R = -\frac{1}{2P}\sum_{s\in \mathrm{C,F}}\sum_{j=1}^{P} E_{s,j}. \tag{7}$$

4. Experimental results

4.1. Perceptual representation

4.1.1. Learned feature extractors

Figure 3(a) shows the basis vectors of the coarse-scale feature extractors at the end of training when there was no systematic orientation offset between the left and right eyes. For most basis vectors, the four components are two-dimensional Gabor-like functions. All components have similar spatial frequencies and similar spatial orientations. Two representative feature extractors are shown in figures 3(b) and (c).

In figures 3(d) and (e), we show cross-sectional views of one component of the representative basis vectors in the direction orthogonal to their preferred orientation. For most feature extractors, corresponding components of the two basis vectors are in phase quadrature. The phase difference between the two basis functions is critical in endowing the feature detectors with invariance to small position shifts, similar to that observed in visual cortical complex cells.

4.1.2. Development of the feature extractors

To show how the perceptual representation evolves during training, figure 4 shows some representative basis vectors (only the left and right components) near the beginning (∼1% of training) and at the end of training with different systematic offsets between the left and right cameras. At the beginning of training, before cyclovergence control has developed, the average orientation offset between left and right basis vectors reflects the systematic offset. At the end of training, after the cyclovergence policy has emerged, the left and right

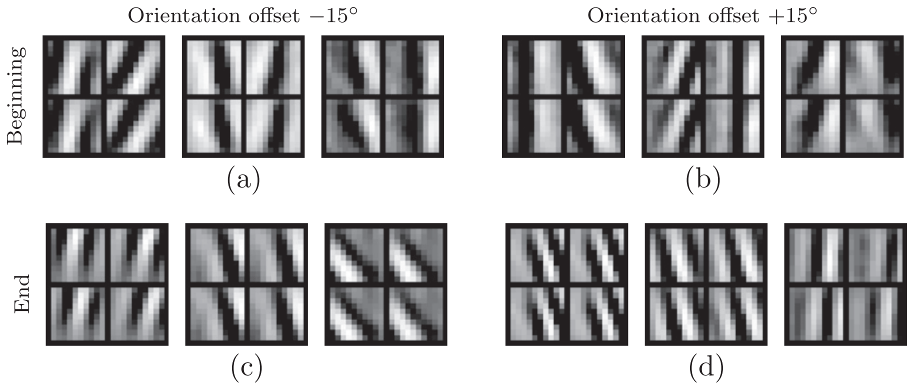

Figure 4. The first basis vectors of some example feature extractors at the beginning and the end of the training with different left/right orientation offsets. The left and right columns correspond to left and right images. The top and bottom rows correspond to pre- and post-saccade frames. (a) and (c) Example basis vectors from the training trial with a −15° left/right orientation offset near the beginning (∼1%) of training (a) and at the end of training (c). (b) and (d) Example basis vectors from the training trial with a +15° left/right orientation offset near the beginning (∼1%) of training (b) and at the end of training (d).

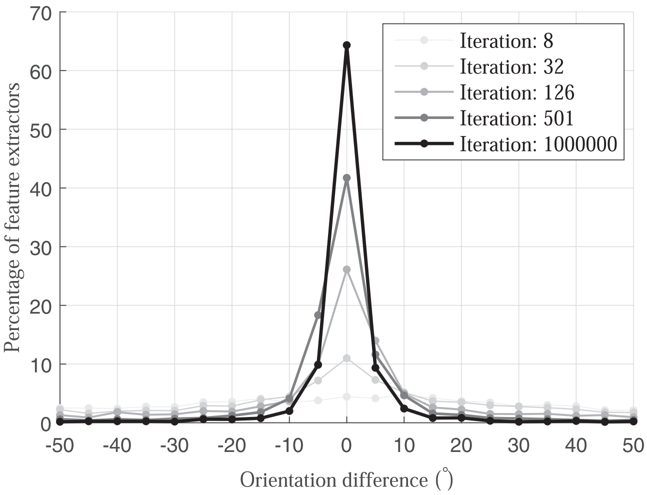

Figure 5. Statistics of the orientation difference between the pre- and post-saccade components of all the feature extractors at different points during training.

basis vectors have similar preferred orientations, since the learned cyclovergence policy has corrected for the systematic offset.

To quantify the statistics of the spatial orientation offsets, we separately fit 2D Gabor functions to each component of the learned feature extractors. Figure 5 shows the distributions of differences between the estimated orientations of the pre- and post-saccade components at different stages of the training trial where there was no systematic left/right orientation offset. The orientation differences are more concentrated around 0° at the end compared to the beginning of training.

Figures 6(a) and (b) show the distributions of the differences in estimated orientation between the left and right components during training with two systematic orientation offsets. At the beginning of training, the offsets are clustered around the systematic offset. At the end of training, they are clustered around zero. Figure 6(c) shows the evolution of the median difference between the estimated orientations of the left and right components of the basis vectors during training with different systematic left/right orientation offsets. All of the curves converge to around zero orientation difference. As the cyclovergence controller learns to cancel the orientation mismatch, the statistics of the basis vectors change to reflect the new input statistics.

4.2. Torsional control policies

4.2.1. Learned policies

Figures 7(a)–(c) compares a learned cycloversion policy and Listing's law (the equation for C_s in (1)). As shown in figure 7(a), the cycloversion angle of the learned policy depends mainly upon the pan and tilt angle, with only small variation as the vergence angle (shown as error bars) changes. This is consistent with Listing's law,

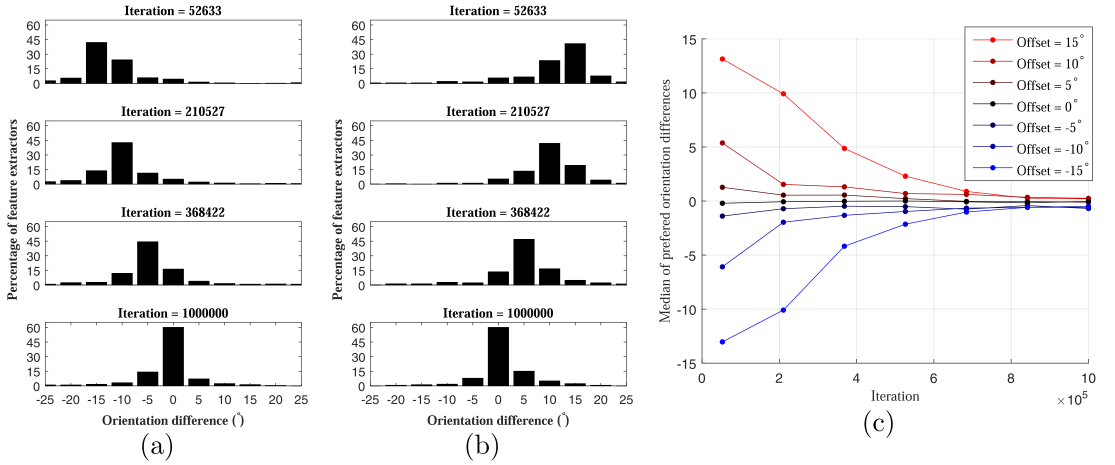

Figure 6. (a) and (b) Histograms of the orientation difference between the left and right components of all the feature extractors at the first, fourth, sixth and last of 19 checkpoints during training with constant orientation offsets of −15° (a) and +15° (b). (c) The evolution of the median orientation difference between left and right components of the feature extractors during training with different orientation offsets. Each curve shows one example simulation for each orientation offset.

which is independent of vergence. To compare the learned cycloversion policy with Listing's law, figure 7(c) plots cross sections of the policy and Listing's law along the lines indicated on the contour maps of figure 7(b). The learned policy is very similar to Listing's law. More examples of the comparison between the cycloversion polices before and after learning are available at https://youtu.be/-5gl2nIBZMs.

Figures 7(d)–(f) compares a learned cyclovergence policy with the cyclovergence predicted by L2. The cyclovergence angle of the learned cyclovergence policy depends mainly upon the vergence and tilt angle, and varies only slightly with pan angle. This is consistent with L2, where the cyclovergence angle is independent of the pan angle (shown as error bars). To compare the learned cyclovergence policy with L2, figure 7(f) plots the cyclovergence angles as a function of tilt for three different vergence angles. The learned policy is close to the one predicted by L2.

4.2.2. Development of the policies

To study the evolution of the policies, we set 119 checkpoints evenly spaced during training (1 M iterations total) and recorded the weights of the policy networks at each checkpoint.

Figure 8 shows the evolution of the root mean square error (RMSE) between the learned policy and L2 during training with different constant orientation offsets. We combine the results of positive and negative orientation offsets with the same amplitude, since we do not observe strong systematic differences between them. For all offsets, RMSE values converge to around 0.3°. The larger the amplitude of the constant orientation offset, the larger the RMSE at the beginning of learning. Because we initialize the weights of the neural network as random numbers with small absolute values, the output of the neural network is around 0° at the beginning. Thus, the magnitude of the initial error is largely due to the orientation offset. During training, the neural network approaches a policy that minimizes the orientation difference between the two eyes, leading to decreasing RMSE values over training.

To analyze the evolution of adaptive cyclovergence control behavior, we also recorded the output cyclovergence actions during training. Figure 9 presents the temporal evolution of the distribution of cyclovergence actions during training with orientation offsets −15° and 15°. The distribution of the output cyclovergence commands is concentrated around 0° at the beginning, but gradually becomes centered around +15° and −15°. More examples of the development of adaptive cyclovergence control policies are available at https://youtu.be/X7MyPVVL-ns.

4.2.3. Cyclovergence on a real iCub

The results reported above were all done in the iCub simulation platform. To demonstrate practical applicability of our model, we learned a cyclovergence control policy on data collected from a real iCub robot. During the data collection, we recorded the binocular image sequence as well as the eye joint angles (pan, tilt and vergence angles). To speed up the training, we used a set of GASSOM feature extractors learned in simulation, but trained the cyclovergence policy on real data from scratch.

Figure 10 compares no torsion control and learned torsional control. The anaglyph shown in figure 10(a) shows the rotational offset between the left and right images. The learned torsional control policy reduces the orientation offset, as shown in figure 10(b). Figures 10(c) and (d) show the differences between the

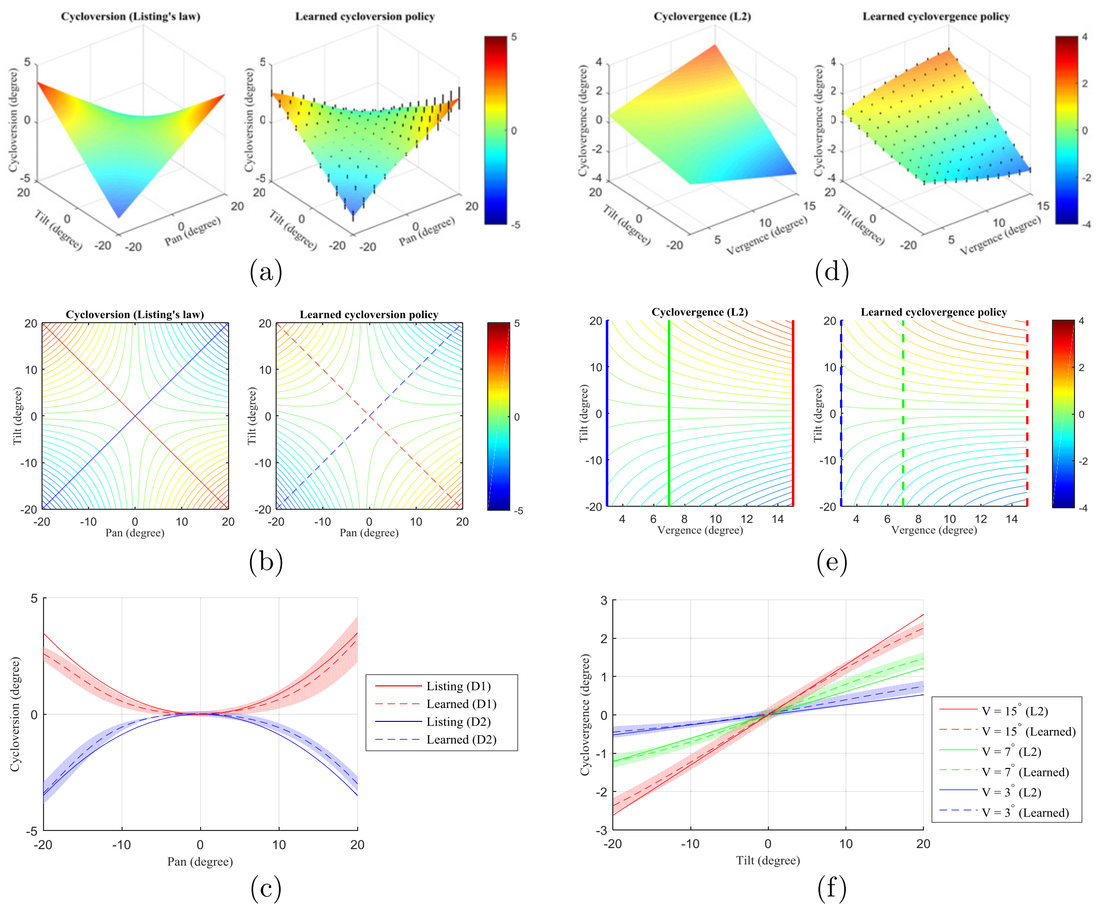

Figure 7. (a) Listing's law and learned cycloversion policy as a function of pan and tilt. Error bars for the learned policy indicate the standard deviation computed over ten vergence angles evenly sampled between 3° and 15°. (b) Contour maps corresponding to (a). The contour interval is 0.2°. (c) Cross sections of the policies along the red (D1) and blue (D2) lines in (b). Solid lines indicate Listing's law. Dashed lines/shaded areas represent the mean/standard deviation of the cycloversion angle computed over ten vergence angles evenly sampled between 3° and 15°. (d) L2 and the learned cyclovergence policy as a function of vergence and tilt. Error bars in the figure of learned policy indicate the standard deviation computed over ten pan angles evenly sampled between −20° and 20°. (e) Contour maps corresponding to (d). The contour interval is 0.16°. (f) Cross sections of the policy at different vergence angles. Solid lines represent the cyclovergence angles predicted by L2. Dashed lines/shaded areas represent the mean values/standard deviation of the cyclovergence angle computed over ten pan angles evenly sampled between −20° and 20°.

left and right eye image intensities. The remaining difference between the two images in figure 10(d) is mainly due to the horizontal misalignment, i.e. the vergence error. A video demonstration is available at https://youtu.be/17XDQf9S0U8.

4.3. Noise robustness

The experiments above assumed error free measurements of gaze direction. However, noise robustness is important. For example, biological systems may suffer from sensory noise, neuronal noise or motor noise [10]. In robotic applications, there may be mismatches between the camera positions as recorded by the encoders and the actual camera positions, due to non-idealities, such as backlash, gear slip, misalignments, mis-calibrations, and limited sensor resolution.

To test our system's robustness under more realistic conditions, we trained our system with different amounts of noise added as random offsets between the commanded eye positions, which are also input to the cycloversion and cyclovergence control networks, and the actual eye positions. We refer to this as motor noise, although it should be understood that mismatches between the actual and the sensed gaze directions may result from many different sources, such as errors in the sensory neurons in a closed loop position control system. We modelled the effect of noise in the iCub simulator by sampling the actual eye positions (pan, tilt, vergence) from three independent Gaussian distributions with means equal to the commanded eye position. We control the noise level by varying the standard deviations of the Gaussian distributions, which were equal for all components.

Figure 11 shows the difference between the learned policies and Listing's law as a function of the motor noise level. We measured the difference between the learned policies and L2 using the root mean square error

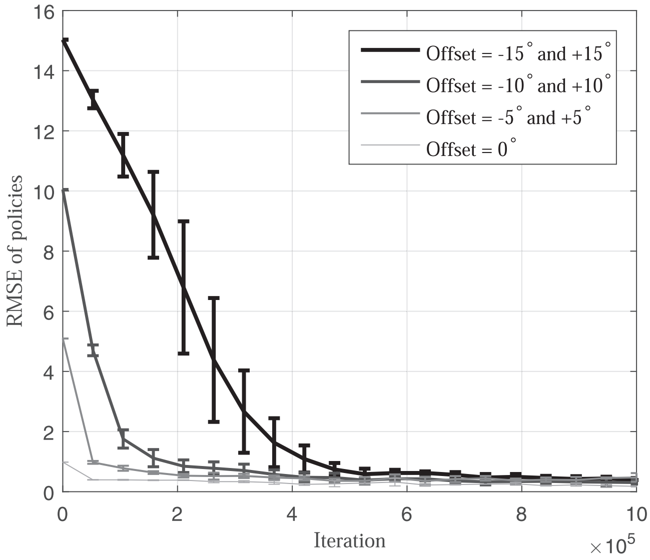

Figure 8. The evolution of the RMSE over time between the learned cyclovergence policies and the policy predicted by L2 with different orientation offsets. Error bars represent the standard deviations over three training trails.

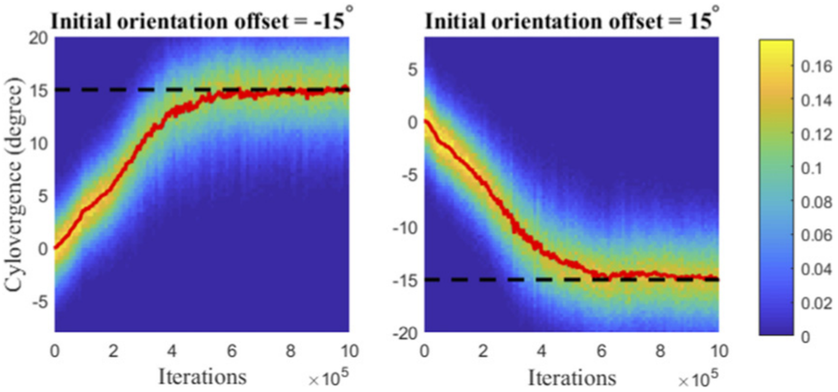

Figure 9. The temporal evolution of the distribution of cyclovergence commands recorded during training with orientation offsets −15° and 15°. The images indicate the density of the actions chosen at each time point. The values in each column sum to one. The red curves show the moving average of the cyclovergence commands. The black dashed lines indicate the negative orientation offset.

between the two outputs over 10^6 samples evenly spaced in the pan-tilt-vergence space (100 samples along each dimension). The performance of the learned policies degrades slowly as the noise level increases, leading to improvements over no torsional control for noise levels as high as 2.5°.

To study how the learned policies degrade as the encoder noise level increases, we fit the 3D surfaces of the learned policies (as shown in figures 7(b) and 8(b)) using a polynomial function of degree two in both x (pan for cycloversion policy, vergence for cyclovergence policy) and y (tilt):

$$C_{\mathrm{fit}} = a + bx + cy + \mathrm{d}x^2 + ey^2 + fxy. \tag{8}$$

For the policies that follow Listing's law for cycloversion and L2 for cyclovergence, the parameter f is −0.5 and 0.5, respectively. The rest of the parameters are zero. Figure 12 shows the changes in the fitting parameters of the learned policies as the encoder noise level increases. When there is no encoder noise introduced, the parameter f is around −0.5 for the learned cycloversion policy and around 0.5 for the learned cyclovergence policy, and the other parameters are centered on zero. As the encoder noise level increases, the parameter f shifts towards 0, indicating that the 3D surfaces of the learned policies become flatter as the encoder noise increases, i.e., the learned policies generate cyclorotation actions in the correct direction but with smaller amplitude.

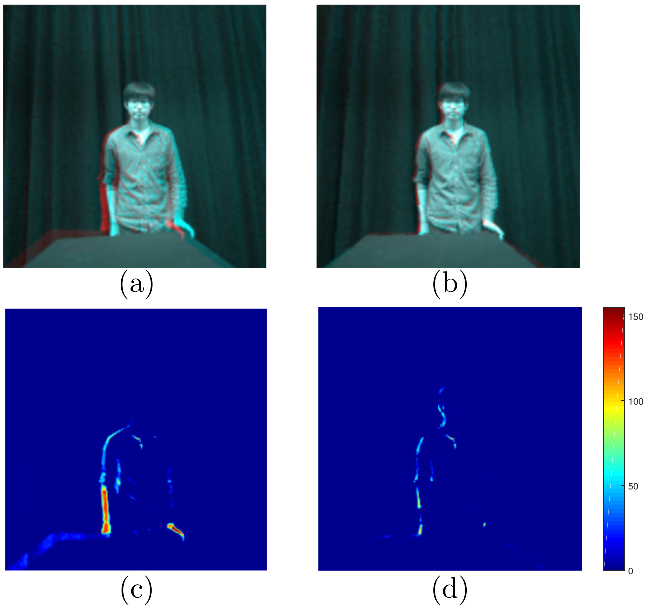

Figure 10. Qualitative comparison between no torsional control and learned torsional control on real iCub data. (a) and (b) Anaglyphs of binocular image pairs with no torsional control (a) and with learned torsional control (b). Red corresponds to the left eye image. Cyan corresponds to the right eye image. Areas that appear in gray are well matched. (c) and (d) Differences between the binocular images in (a) and (b) respectively. Both images shown in the same scale.

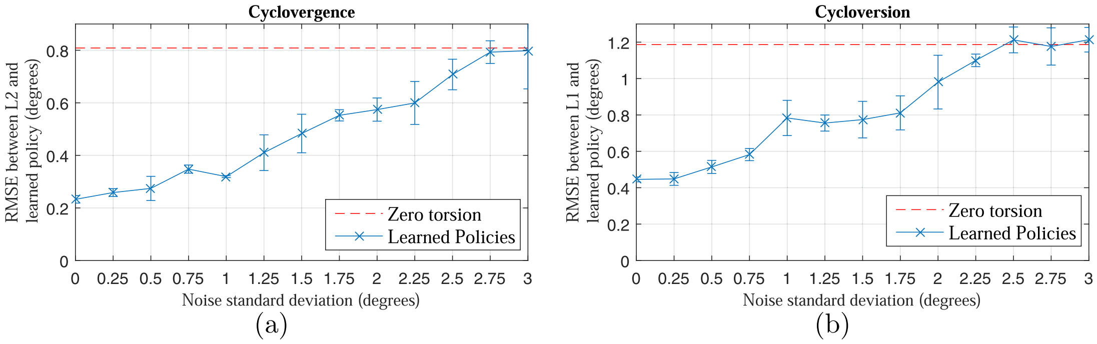

Figure 11. The system's noise robustness. (a) The RMSE value between the learned cyclovergence policies and the one described by L2 as a function of motor noise amplitude. (b) The RMSE value between the learned cycloversion policies and the one described by Listing's law as a function of motor noise amplitude. The red dashed lines represent the RMSE when torsion angles are fixed at zero degrees. Error bars represent the standard deviation over three training trials.

5. Discussion

We have described a model for self-calibration of torsional eye movement control based on the AEC framework. Efficient coding hypothesizes that neurons provide the best encoding of the sensory inputs subject to internal resource constraints, such as metabolic costs. This is often formulated as 'predictive coding', where the nervous system generates a prediction of the sensory input and transmits the difference between the actual and predicted inputs [33, 38]. AEC extends this by hypothesizing that the organism simultaneously structures its behavior so that its sensory input can be efficiently encoded. In this sense, the system we propose here can be considered to be a concrete implementation of Friston's free energy principle [12].

In the visual representation mechanism of AEC, we model the receptive fields of neurons using a set of basis vectors. The components of the basis vectors corresponding to the pre- and post-saccade images are analogous to receptive fields that respond to different visual field locations, but the same locations relative to the saccade target. The shifts in visual field model the predictive remapping of receptive fields before saccades observed in cortical neurons [8]. This remapping preserves the constancy of the visual world, since the same object excites the cortical neuron both before and after the saccade [50].

AEC formulates the visual calibration/image orientation alignment as an active and adaptive process by modeling the sensorimotor contingencies that result from interaction with the environment [31]. If the torsion

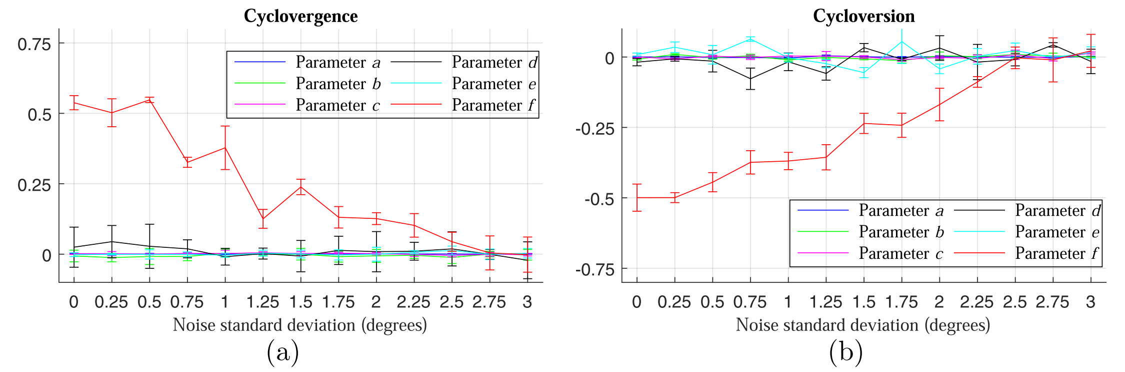

Figure 12. Fitting parameters as functions of the encoder noise level. (a) Shows the fitting parameters of the 3D surfaces of the learned cyclovergence policies trained with different levels of encoder noise. (b) Shows the fitting parameters of the 3D surfaces of the learned cycloversion policies trained with different levels of encoder noise.

is not actively controlled, there will be orientation mismatch between images of the target before and after a saccade, and between images of the target in the left and right eyes. By minimizing reconstruction error, the agent learns torsional control strategies that reduce orientation mismatch across saccades and between the two eyes.

Our model does not require any external rewards or supervision. The sensory representation and behavioral control are both learned by following the same intrinsic motivation: to minimize the reconstruction error of the input stimuli. This enables the system to adapt without supervision to non-idealities, such as noise and changes in physical configuration. The reconstruction error is a generic cost function, not specifically designed for the cyclorotation control task. The exact same cost function has been used to account for the emergence and self-calibration of other eye movement behaviors, such as vergence control [54] and smooth pursuit [53].

AEC provides the most parsimonious explanation to date for Listing's law. Minimizing sensory prediction error is a more general principle than the functional criteria commonly assumed to be optimized by Listing's law, e.g. torsional change, rotational eccentricity or binocular search zone size. Because it emphasizes primarily the sensory consequences of movement, our formulation of AEC is more aligned with the sensorimotor explanation [14, 15], than the motor explanation [11, 49], of Listing's law. Motor considerations could be incorporated as additional resource constraints. For example, we could include the metabolic costs of the muscles [18] in addition to the neural metabolic costs, which are captured by the sparsity constraint in the GASSOM.

Our system does not require any constraints on the motor policy in order to ensure the emergence of Listing's law. For example, although the equation for the cycloversion angle in (1) only depends on pan and tilt angles and the equation for the cyclovergence angle only depends on vergence and tilt angles, we do not make this constraint in our system. All three angles determining the binocular eye gaze (pan, tilt, and vergence) were available to the cycloversion and cyclovergence policy networks. The cycloversion/cyclovergence policy networks automatically learn to ignore the vergence/pan angles (figure 7).

Because our model accounts for the entire sensorimotor chain, from visual input to commanded motor output, it is directly applicable to robotic systems. We performed most experiments using a commonly used and widely available humanoid robotic simulation platform: the iCub simulator. We have also presented experimental results on data collected from a real iCub robot. Biological and artificial agents share the need to adapt to changes in physical configuration and to compensate for noise. Changes in physical configuration in biological systems are introduced by growth or injury. In artificial agents, they are introduced by mechanical misalignment or damage by collision. Regardless of the root cause, the AEC model described here can self-calibrate these systems to compensate, based on the sensorimotor contingencies that are evoked naturally by interaction between agent and environment.

Moving forward, this model can be extended by adding other torsional control inputs besides the efference copy of gaze direction, such as visually induced components of torsion. Hooge *et al* show that visually and gaze direction induced components of cyclovergence add linearly, suggesting that they are governed by two independent systems [16]. To account for visually induced cyclorotation, our model could be extended with another neural network, which maps the output of the perceptual representation to actions. The final cyclorotation would be a linear combination of the gaze-driven and the visually-driven actions. The integration of visual information into eye control would also enable us to model learning of saccadic and torsional components simultaneously.

It would also be interesting to deploy the learning system on the real robot and to learn with hardware in the loop. In figure 10(d), we observe a horizontal misalignment between the left and right images, which indicates a mismatch between the pre-trained vergence policy and the real robot. We believe that this misalignment could be eliminated by fine tuning the parameters on the real robot, instead of directly running the system trained in simulation.

Finally, a very promising extension of this work is to adapt it to work on input from event-based neuromorphic vision sensors [21]. As discussed in section 3.1, the image rendering parameters in our experiments, i.e. the image resolution, window sizes and focal length, were chosen so that they are compatible with current neuromorphic sensors, which tend to have lower resolution than frame-based RGB cameras due to the integration of extra circuitry within each pixel. In addition, the GASSOM perceptual representation can be applied to the event streams [6]. The results of our experiments using these choices provide confidence that an implementation of this system using neuromorphic hardware will be effective in endowing robots with the capability of self-calibrating behavior. The neuromophic iCub [46] would be an ideal platform on which to verify this. However, much work remains to be done, e.g. how to handle the combination of pre- and post-saccade images. One possibility would be to use address filtering and re-mapping to route address events from the target sub-windows before the saccade into storage buffers, then collect events from the central foveal sub-windows after the saccade into separate buffers while ignoring events generated during the saccade. This would exploit the unique properties of the address event representation in a novel implementation of the shifting receptive fields discussed by [8, 50].

Acknowledgments

This work was funded in part by the Hong Kong Research Grants Council under Grant Number 16214821. JT was supported by the Johanna Quandt Foundation.

Data availability statement

No new data were created or analysed in this study.

ORCID iDs

Qingpeng Zhu https://orcid.org/0000-0003-3038-1863
Bertram E Shi https://orcid.org/0000-0001-9167-7495

References

[1] Barlow H B 1961 *Possible Principles Underlying the Transformations of Sensory Messages* (Cambridge, MA: MIT Press) pp 217–34
[2] Bhatnagar S, Sutton R S, Ghavamzadeh M and Lee M 2009 Natural actor-critic algorithms *Automatica* **45** 2471–82
[3] Bommasani R *et al* 2021 On the opportunities and risks of foundation models (arXiv:2108.07258)
[4] Borghuis B G, Borghuis B G, Ratliff C F, Sterling P and Balasubramanian V 2008 Design of a neuronal array *J. Neurosci.* **28** 3178–89
[5] Cannata G and Maggiali M 2006 Implementation of Listing's law for a tendon driven robot eye *2006 Int. Conf. Intelligent Robots and Systems* (IEEE) pp 3940–5
[6] Chandrapala T N and Shi B E 2016 Invariant feature extraction from event based stimuli *2016 6th IEEE Int. Conf. Biomedical Robotics and Biomechatronics (BioRob)* (IEEE.) pp 1–6
[7] Chandrapala T N and Shi B E 2015 Learning slowness in a sparse model of invariant feature detection *Neural Comput.* **27** 1196–529
[8] Duhamel J-R, Colby C L and Goldberg M E 1992 The updating of the representation of visual space in parietal cortex by intended eye movements *Science* **255** 90–2
[9] Eckmann S, Klimmasch L, Shi B E and Triesch J 2020 Active efficient coding explains the development of binocular vision and its failure in amblyopia *Proc. Natl Acad. Sci. USA* **117** 6156–62
[10] Faisal A A, Selen L P J and Wolpert D M 2008 Noise in the nervous system *Nat. Rev. Neurosci.* **9** 292–303
[11] Fick A 1854 Die bewegungen des menschlichen augapfels, *Zeitschrift fur Rationelle Medicin* vol 4 (Heidelberg: C. F. Wintersche Verlagshandlung) pp 101–28
[12] Friston K 2010 The free-energy principle: a unified brain theory? *Nat. Rev. Neurosci.* **11** 127
[13] Hansard M and Horaud R 2010 Cyclorotation models for eyes and cameras *IEEE Trans. Syst. Man Cybern.* B **40** 151–61
[14] von Helmholtz H 1925 *Treatise on Physiological Optics* vol 3 (Rochester, NY: Optical Society of America)
[15] Hering E 1868 *Die lehre vom binocularen sehen* (Frankfurt am Main: Salzwasser-Verlag Gmbh)
[16] Hooge I T C and van den Berg A V 2000 Visually evoked cyclovergence and extended Listing's law *J. Neurophysiol.* **83** 2757–75
[17] Hubel D H and Wiesel T N 1962 Receptive fields, binocular interaction and functional architecture in the cat's visual cortex *J. Physiol.* **160** 106

[18] Klimmasch L, Lelais A, Lichtenstein A, Shi B E and Triesch J 2017 Learning of active binocular vision in a biomechanical model of the oculomotor system *2017 Joint IEEE Int. Conf. Development and Learning and Epigenetic Robotics* pp 21–6

[19] Lee C, Zee D S and Straumann D 2000 Saccades from torsional offset positions back to Listing's plane *J. Neurophysiol.* **83** 3241–53

[20] Lelais A, Mahn J, Narayan V, Zhang C, Shi B E and Triesch J 2019 Autonomous development of active binocular and motion vision through active efficient coding *Front. Neurorob.* **13** 49

[21] Lichtsteiner P, Posch C and Delbruck T 2008 A 128 × 128 120 dB 15 μs latency asynchronous temporal contrast vision sensor *IEEE J. Solid-State Circuits* **43** 566–76

[22] Lonini L, Zhao Y, Chandrashekhariah P, Shi B E and Triesch J 2013 Autonomous learning of active multi-scale binocular vision *IEEE 3rd Joint Int. Conf. Development and Learning and Epigenetic Robotics* (Osaka: IEEE) pp 1–6

[23] Lonini L, Forestier S, Teulière C, Zhao Y, Shi B E and Triesch J 2013 Robust active binocular vision through intrinsically motivated learning *Front. Neurorob.* **7** 20

[24] Mok D, Ro A, Cadera W, Crawford J D and Vilis T 1992 Rotation of Listing's plane during vergence *Vis. Res.* **32** 2055–64

[25] Nikolić D, Häusler S, Singer W and Maass W 2009 Distributed fading memory for stimulus properties in the primary visual cortex *PLoS Biol.* **7** e1000260

[26] Ohzawa I, DeAngelis G C and Freeman R D 1990 Stereoscopic depth discrimination in the visual cortex: neurons ideally suited as disparity detectors *Science* **249** 1037–41

[27] Olmos A and Kingdom F A A 2004 A biologically inspired algorithm for the recovery of shading and reflectance images *Perception* **33** 1463–73

[28] Olshausen B A and Field D J 1996 Emergence of simple-cell receptive field properties by learning a sparse code for natural images *Nature* **381** 607

[29] Olshausen B A and Field D J 1997 Sparse coding with an overcomplete basis set: a strategy employed by V1? *Vis. Res.* **37** 3311–25

[30] Van Opstal J, Hepp K, Suzuki Y and Henn V 1996 Role of monkey nucleus reticularis tegmenti pontis in the stabilization of Listing's plane *J. Neurosci.* **16** 7284–96

[31] O'Regan J K and Noë A 2001 A sensorimotor account of vision and visual consciousness *Behav. Brain Sci.* **24** 939–73

[32] Porrill J, Ivins J P and Frisby J P 1999 The variation of torsion with vergence and elevation *Vis. Res.* **39** 3934–50

[33] Rao R P N and Ballard D H 1999 Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects *Nat. Neurosci.* **2** 79

[34] Raphan T 1998 Modeling control of eye orientation in three dimensions: I. Role of muscle pulleys in determining saccadic trajectory *J. Neurophysiol.* **79** 2653–67

[35] Renninger L W, Verghese P and Coughlan J 2007 Where to look next? Eye movements reduce local uncertainty *J. Vis.* **7** 6

[36] Van Run L J and Van den Berg A V 1993 Binocular eye orientation during fixations: Listing's law extended to include eye vergence *Vis. Res.* **33** 691–708

[37] Schreiber K, Crawford J D, Fetter M and Tweed D 2001 The motor side of depth vision *Nature* **410** 819

[38] Srinivasan M V, Laughlin S B and Dubs A 1982 Predictive coding: a fresh view of inhibition in the retina *Proc. R. Soc.* B **216** 427–59

[39] Strasburger H, Rentschler I and Jüttner M 2011 Peripheral vision and pattern recognition: a review *J. Vis.* **11** 5

[40] Teulière C, Forestier S, Lonini L, Zhang C, Zhao Y, Shi B and Triesch J 2015 Self-calibrating smooth pursuit through active efficient coding *Robot. Auton. Syst.* **71** 3–12

[41] Tikhanoff V, Fitzpatrick P, Metta G, Natale L, Nori F and Cangelosi A 2008 An open-source simulator for cognitive robotics research: the prototype of the iCub humanoid robot simulator *Proc. 8th Workshop on Performance Metrics for Intelligent Systems* (Gaithersburg: ACM) pp 57–61

[42] Triesch J, Eckmann S and Shi B 2017 A computational model for the joint development of accommodation and vergence control *J. Vis.* **17** 162

[43] Tweed D and Vilis T 1990 Geometric relations of eye position and velocity vectors during saccades *Vis. Res.* **30** 111–27

[44] Tweed D, Fetter M, Andreadaki S, Koenig E and Dichgans J 1992 Three-dimensional properties of human pursuit eye movements *Vis. Res.* **32** 1225–38

[45] Van Rijn L J, Van der Steen J and Collewijn H 1992 Visually induced cycloversion and cyclovergence *Vis. Res.* **32** 1875–83

[46] Vasco V, Glover A, Tirupachuri Y, Solari S, Chessa M and Bartolozzi C 2016 Vergence control with a neuromorphic iCub *2016 IEEE-RAS 16th Int. Conf. Humanoid Robots (Humanoids)* (IEEE) pp 732–8

[47] Vikram T N, Teulière C, Zhang C, Shi B E and Triesch J 2014 Autonomous learning of smooth pursuit and vergence through active efficient coding *4th Int. Conf. Development and Learning and on Epigenetic Robotics* (IEEE) pp 448–53

[48] Wong A 2004 Listing's law: clinical significance and implications for neural controll *Surv. Ophthalmol.* **49** 563–75

[49] Wundt W M 1862 *Beiträge zur Theorie der Sinneswahrnehmung* (Leipzig: C.F. Winter)

[50] Wurtz R H 2008 Neuronal mechanisms of visual stability *Vis. Res.* **48** 2070–89

[51] Yarbus. A L 1967 *Eye Movements During Perception of Complex Objects* (New York: Plenum) pp 171–211

[52] Zhang C, Triesch J and Shi B E 2016 An active-efficient-coding model of optokinetic nystagmus *J. Vis.* **16** 10

[53] Zhang C, Zhao Y, Triesch J and Shi B E 2014 Intrinsically motivated learning of visual motion perception and smooth pursuit *Int. Conf. Robotics and Automation* (Hong Kong: IEEE) pp 1902–8

[54] Zhao Y, Rothkopf C A, Triesch J and Shi B E 2012 A unified model of the joint development of disparity selectivity and vergence control *IEEE Int. Conf. Development and Learning and Epigenetic Robotics* (San Diego: IEEE) pp 1–6

[55] Zhao Z, Triesch J and Shi B E 2018 Learning hierarchical integration of foveal and peripheral vision for vergence control by active efficient coding *Int. Conf. Simulation of Adaptive Behavior* (Springer) pp 78–89

[56] Zhu Q, Triesch J and Shi B E 2017 Autonomous, self-calibrating binocular vision based on learned attention and active efficient coding *2017 Joint IEEE Int. Conf. Development and Learning and Epigenetic Robotics* (Lisbon, Portugal)

[57] Zhu Q, Triesch J and Shi B E 2020 Integration of vergence, cyclovergence, and saccades through active efficient coding *2020 Joint IEEE Int. Conf. Development and Learning and Epigenetic Robotics* pp 1–6

[58] Zhu Q, Triesch J and Shi B E 2017 Joint learning of binocularly driven saccades and vergence by active efficient coding *Front. Neurorob.* **11** 58

[59] Zhu Q, Zhang C, Triesch J and Shi B E 2018 Autonomous learning of cyclovergence control based on active efficient coding *2018 Joint IEEE Int. Conf. Development and Learning and Epigenetic Robotics* pp 251–6