Title
Biomimetic Modeling of the Eye and Deep Neuromuscular Oculomotor Control

Author
Lakshmipathy, Arjun Sriram

Publication Date
2018

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Biomimetic Modeling of the Eye and Deep Neuromuscular Oculomotor Control

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Computer Science

by

Arjun Sriram Lakshmipathy

2018

ABSTRACT OF THE THESIS

Biomimetic Modeling of the Eye and Deep Neuromuscular Oculomotor Control

by

Arjun Sriram Lakshmipathy
Master of Science in Computer Science
University of California, Los Angeles, 2018
Professor Demetri Terzopoulos, Chair

This thesis presents a novel, biomimetic model of the eye for realistic virtual human animation. We also introduce a deep learning approach to oculomotor control that is compatible with our biomechanical eye model. Our eye model consists of the following functional components: (i) submodels of the 6 extraocular muscles that actuate realistic eye movements, (ii) an iris submodel, actuated by pupillary muscles, that accommodates to incoming light intensity, (iii) a corneal submodel and a deformable, ciliary-muscle-actuated lens submodel, which refract incoming light rays for focal accommodation, and (iv) a retina with a multitude of photoreceptors arranged in a biomimetic, foveated distribution. The light intensity captured by the photoreceptors is computed using ray tracing from photoreceptor positions through the finite aperture pupil into the 3D virtual environment, and the visual information is output by the eye via an optic nerve vector. Our oculomotor control system includes a neuromuscular motor controller implemented as a locally-connected, irregular Deep Neural Network (DNN) that conforms to the irregular retinal photoreceptor distribution, plus auxiliary Shallow Neural Networks (SNNs) that control the accommodation of the pupil and lens. The neuromuscular controller is trained offline through deep learning from visual data synthesized by the eye model itself. Once trained, it operates robustly and efficiently online, innervating the extraocular muscles to produce natural eye movements in order to foveate and pursue moving visual targets. We demonstrate the operation of our eye model binocularly within a recently introduced sensorimotor control framework involving an anatomically-accurate biomechanical human musculoskeletal model.

The thesis of Arjun Sriram Lakshmipathy is approved.

Joseph M. Teran

Song-Chun Zhu

Demetri Terzopoulos, Committee Chair

University of California, Los Angeles

2018

To my parents . . .

who—among so many other things—

have been supportive in my pursuit of this degree

and in everything else I have done throughout my life

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I would like to extend special acknowledgements to Dr. Masaki Nakada and Professor Demetri Terzopoulos for their full support of this work and for the additional mentorship and guidance that they have provided throughout my Master's program. The interactions that I have had with my colleagues in the UCLA Computer Graphics & Vision Laboratory were very beneficial. I would also like to acknowledge that this thesis benefited from work done principally by Dr. Nakada and/or Honglin Chen, particularly with regards to formulation of the retinal photoreceptor distribution, DNN training, and the formulation of liNets.

CHAPTER 1

Introduction

Animating the eyes, metaphorically the "windows to the soul", is a critical yet insufficiently explored aspect of human simulation and animation. Fig. 1.1 shows a cross section through a detailed geometric model of the eye, which is available commercially. Although some pioneering work has been done in developing *functional* models of the eye, the community has not yet developed a satisfactory eye model for highly realistic autonomous human animation. This thesis takes a strongly biomimetic approach to tackling the challenge, with the conviction that greater realism in human animation will accrue from greater realism in the functional modeling of the various parts of the human body, including the eyes.

1.1 Contributions of the Thesis

More specifically, our contributions are as follows:

- We introduce a biomimetic model of the eye, which consists of the following functional components:

    1. Submodels of the 6 extraocular contractile (EOC) muscles, innervated by muscle activations, which actuate realistic eye movements.
    2. An iris submodel, actuated by pupillary sphincter and dilator muscles, that can accommodate to incoming light intensity by constricting and dilating the pupil.
    3. Corneal and deformable lens submodels that refract incoming light rays and adjust the focal distance.
    4. A retina with a multitude of photoreceptors (black dots in Fig. 1.1) arranged in

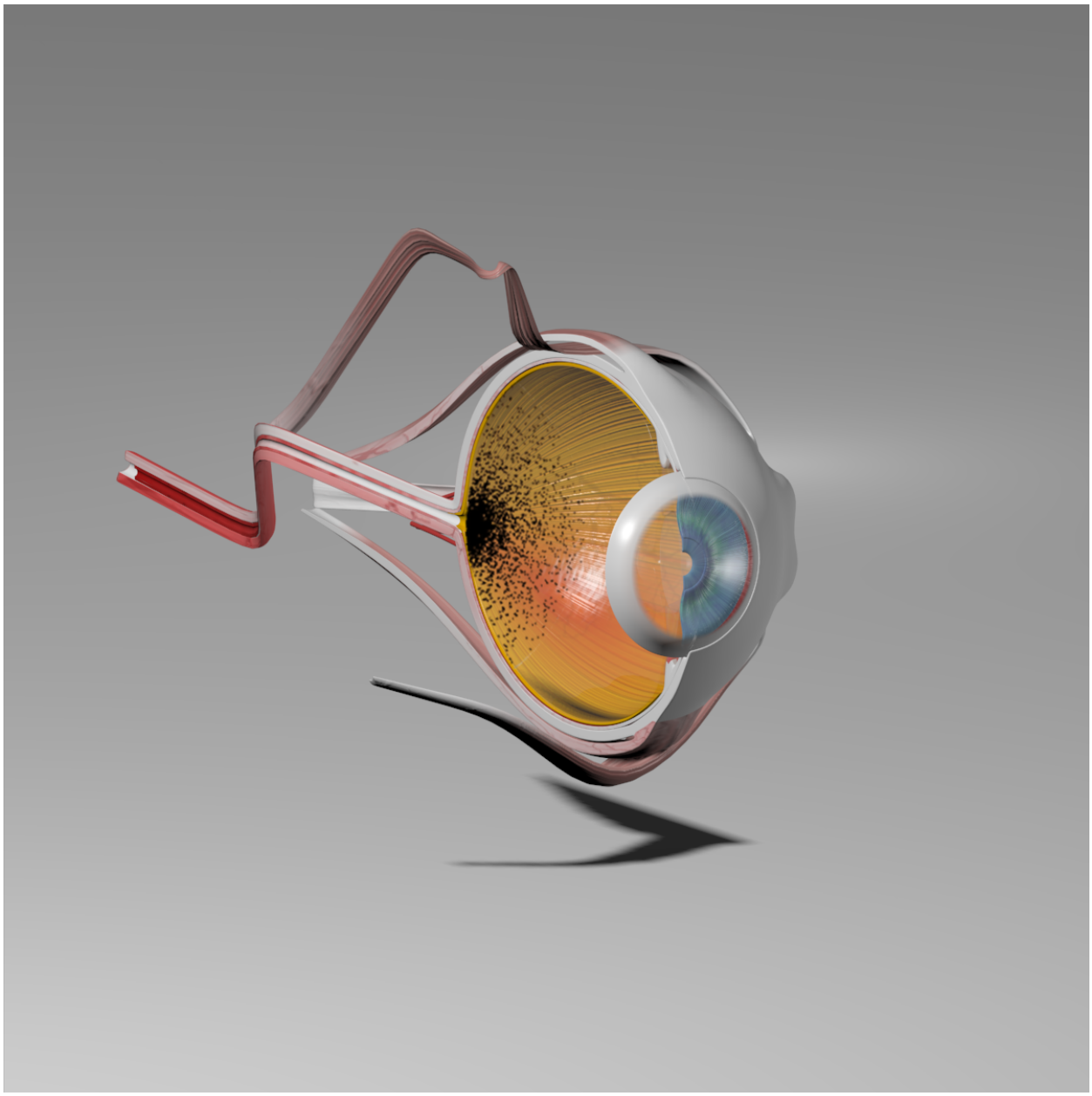


Figure 1.1: Cross section of a detailed geometric model of the eye. The black dots on the retina indicate the positions of our retinal photoreceptors.

a biomimetic, foveated distribution.

5. Retinal photoreceptor responses to 3D visual stimuli are computed using ray tracing from photoreceptor positions through the finite-aperture pupil (as opposed to an ideal pinhole camera) into the 3D virtual environment, and the visual information is output by the model eye via an optic nerve vector (ONV).

- We develop a biomimetic oculomotor control system, which is compatible with the eye model and includes the following functional components:

  1. A trained oculomotor controller, which is implemented as a locally-connected Deep Neural Network (DNN) that conforms to the irregular retinal photoreceptor distribution and drives foveation, feeding a fully-connected neuromuscular DNN that controls the EOC muscles to produce natural eye movements.

  2. Auxiliary, trained Shallow Neural Networks (SNNs) that control the accommodations of the pupil and lens.

  3. A machine learning approach to training the DNNs and SNNs offline from visual data synthesized by the eye model itself, such that the trained neuromuscular controllers can operates efficiently and robustly online, innervating the intraocular muscles for accommodation as well as the EOC muscles to produce natural eye movements for the foveation and ocular pursuit of moving visual targets.

With regard to eye movements, although there is some variability in the theories, four major types of human eye movements have been identified, as follows:

1. *Saccade:* A quick eye movement to bring a visual target from peripheral vision to the center of the retina—the fovea—where visual acuity is greatest.

2. *Fixation:* After the eyes center the fovea on the visual target with saccade movements, fixation follows, with the eyes maintaining focus on the target in place. The eyes look still and stable; however, small oscillations and drifting may be observed.

3. *Smooth pursuit:* Once a moving visual target is fixated, the eyes can pursue the target as it moves. The trajectory of the eye movement is not entirely smooth but includes random oscillation in smooth pursuit behavior, which resembles Brownian motion.

4. *Vistibulo-ocular reflex (VOR):* When the head rotates, the eye makes a compensatory rotation in the opposite direction. This reflex motion, called VOR, is necessary to stabilize the visual target in the foveal region while the head is moving.

The neuromuscular oculomotor control model that we develop synthesizes these types of eye movements using a trained DNN that takes as input the ONV from the retina and produces as output the activations for the 6 EOC muscles, and it does so naturally, without trajectory planning or the imposition of any artificial constraints on velocity and acceleration.

We demonstrate the operation of our eye model (binocularly) within a sensorimotor control framework recently introduced by Nakada et al. (2018d), which involves an anatomically-accurate biomechanical human musculoskeletal model.

1.2 Overview of the Thesis

Chapter 2: Related Work

Chapter 3: Overview of the Biomechanical Eye Model

Chapter 4: Overview of Neuromuscular Oculomotor Control

Chapter 5: Experiments and Results

Chapter 6: Conclusion

CHAPTER 2

Related Work

The motivation for our work is the prior work of Nakada et al. (2018d), who introduce an elaborate, strongly biomimetic sensorimotor system for a full-body biomechanical human musculoskeletal model, albeit with simple kinematic eyes modeled as pinhole cameras. The virtual human's sensorimotor system comprises 20 automatically-trained DNNs, half constituting its neuromuscular motor control subsystem, while the other half constitute its visual perception subsystem. A pair of perception DNNs drive eye and head movements, while 8 perception DNNs extract visual information to direct arm and leg actions. Meanwhile, 10 DNNs achieve neuromuscular control—2 DNNs control the 216 neck muscles that actuate the cervicocephalic biomechanical complex to produce natural head movements, and 2 DNNs control each limb; i.e., the 29 muscles of each arm and the 39 muscles of each leg.

It is important to note that the purely kinematic eye model of Nakada et al. (2018d) not only lacks extraocular muscles, but it also lacks other optical organs, such as a cornea, iris/pupil, and lens, which are essential for human vision. This thesis introduces a muscle-actuated biomechanical eye model including its essential optical organs. Our more complete eye model presents a much more difficult visuomotor control problem, especially when it is employed for the binocular oculomotor subsystem within the sensorimotor system of the biomechanical human musculoskeletal model of Nakada et al. (2018d).

The ensuing sections review other related work reported in the literature.

2.1 Eye Modeling and Animation

In the context of human animation, the eyes are arguably the most influential facial features (Looser and Wheatley, 2010), and the emotions of characters can be conveyed by means of gaze and eye movements. Various research efforts have focused on realistic eye animation. They can be characterized into physics-based, statistical, and data-driven modeling (Ruhland et al., 2014).

Among the numerous efforts are the following: Lee et al. (2002) pursued a statistical approach to modeling of saccadic movements of the eye using eye-tracking data. Tweed et al. (1990) demonstrated an algorithm for locating eye position and angular velocity using search coil signals. Haslwanter (1995) proposed a mathematical approach for three-dimensional analysis of the eye rotation. Neog et al. (2016) proposed a model for the eye that accounts for the movement of the eyeball as well as eyelids and soft tissues of the eye region. Buchberger (2004) developed a biomechanical model of the human eye by creating a muscle-force prediction model and a kinematic model which utilizes optimization to balance the muscle forces. Wei et al. (2010) proposed a model for human eye movement that takes into account the nonlinear kinematics of the oculomotor plant's geometry and extraocular muscle mechanics as well as its physiological properties. Komogortsev et al. (2013) employed a 2D linear oculomotor model with a simplified pulse-step neuronal control signal to account for horizontal and vertical movements of the eye ball. Trutoiu et al. (2011) proposed a model that uses a set of key-frame parameters based on tracking data from naturally occurring eye blinks.

On the one hand, data-driven approaches to eye animation are computationally inexpensive, but they tend to perform poorly in applications that require the interactions of virtual characters. On the other hand, physics-based approaches tend to be computationally expensive as they require dynamics calculations and, conventionally, inverse dynamics controllers. All these shortcomings are overcome by our biomechanical eye model and its associated neuromuscular oculomotor controller, which is trained using a deep learning approach. The result is an order of magnitude faster performance than conventional physics-based models,

avoidance of statistical modeling, as well as applicability in scenarios that require interactions of intelligent virtual characters, such as in online sensorimotor control.

2.2 Eye Models in Sensorimotor Systems

Subsequent to the seminal papers of Bajcsy (1988) and Ballard (1991), active vision systems received much attention in the computer vision literature. The "animat vision" approach of Terzopoulos and Rabie (1995) was applied within a kinematic virtual human capable of bipedal locomotion, demonstrating active, vision-guided tracking and pursuit (Rabie and Terzopoulos, 2000). A similar kinematic virtual human model, named "Walter", was employed by Sprague et al. (2007) to study visuomotor control in the context of a sidewalk navigation task. In graphics, Yeo et al. (2012) presented a visuomotor system for another kinematic, anthropomorphic virtual character capable of visual target estimation tasks and demonstrating realistic ball catching actions, albeit without any biologically-motivated visual processing. Earlier work by Lee and Terzopoulos (2006) demonstrated a more basic visuomotor system, although for a fully dynamic, biomechanical, muscle-actuated human neck-head-face model. To date, Nakada et al. (2018d) have developed the most sophisticated sensorimotor system, and the only one that has been demonstrated in a full-body biomechanical human musculoskeletal model.

2.3 Retinal Modeling and Sensing

Unlike the uniform, Cartesian grid arrangement of most artificial imaging sensors, visual sampling in the primate retina is known to be strongly space variant (Schwartz, 1977). The density of cones decreases radially from the fovea toward the periphery. A log-polar photoreceptor distribution is commonly used as a model of space-variant image sampling (Koenderink and Van Doorn, 1978; Grady, 2004; Wilson, 1983).

The virtual humans demonstrated by Rabie and Terzopoulos (2000) were equipped with crudely-foveated eyes, implemented as multiple coaxial virtual cameras that render polygon-

shaded images through the GPU pipeline, yielding composited multiresolution pyramids supporting foveal and peripheral perception. The retinal model introduced by Nakada et al. (2018d) (see also (Nakada et al., 2018b,c)) is significantly more biomimetic. Given its fundamentally nonuniform distribution of its photoreceptors, the retinas capture the light intensity of the scene using ray tracing, which better emulates how the human retina samples scene radiance from the incidence of light on its photoreceptors. Interestingly, the raytracing technique has also been used by the ophthalmology community as a methodology for synthesizing retinal images in order to predict changes in visual performance caused by changes in the eyes (Greivenkamp et al., 1995; Wei et al., 2014).

Our significant advance beyond the simple, pinhole-camera eye model, to a more realistic eye model with a finite-aperture pupil positioned between transparent cornea and lens submodels that refract incoming light, necessitates the use of more advanced raytracing techniques than those applied by Nakada et al. (2018d). Furthermore, we equip the retina in our eye model with a substantially greater number of photoreceptors than the 3,600 employed in (Nakada et al., 2018d), which necessitates the use of locally-connected, irregular DNNs, dubbed liNets (Nakada et al., 2018a).

CHAPTER 3

Biomechanical Eye Model

According to human physiological data, the transverse size of an average eye is 24.2 mm and its sagittal size is 23.7 mm. The average mass is 7.5 grams. The approximate field of view of an individual eye is 30 degrees to superior, 45 degrees to nasal, 70 degrees to inferior, and 100 degrees to temporal. When the two eyes are combined, the binocular field of view becomes approximately 135 degrees vertically and 200 degrees horizontally.

We model the virtual eye as a sphere of radius 12 mm with a mass of 7.5 grams, that can be rotated with respect to its center around its vertical y axis by a horizontal angle θ, around its horizontal x axis by a vertical angle ϕ, and around its view axis z by an angle ψ. The eyes are in their neutral positions looking straight ahead when $\theta = \phi = \psi = 0°$. The horizontal and vertical fields of view is 167.5°. The anatomical model consists of the cornea, iris, lens, sclera, retina, optic nerve, and 6 EO muscles, the medial rectus, lateral rectus, superior rectus, inferior rectus, superior oblique, and inferior oblique. Regarding the mechanics of the eye, in Appendix A we provide the equations of motion of the eyeball.

3.1 Retina

The retina is the innermost, light-sensitive layer of shell tissue of the eye. It is populated by a nonuniform distribution of photoreceptors (cones and rods) that differs dramatically from the uniformly-sampled rectangular images common in computer graphics and vision.

To emulate foveated perception in our model, we use the noisy log-polar distribution employed by Nakada et al. (2018d). We include a greater number of photoreceptors—14,400

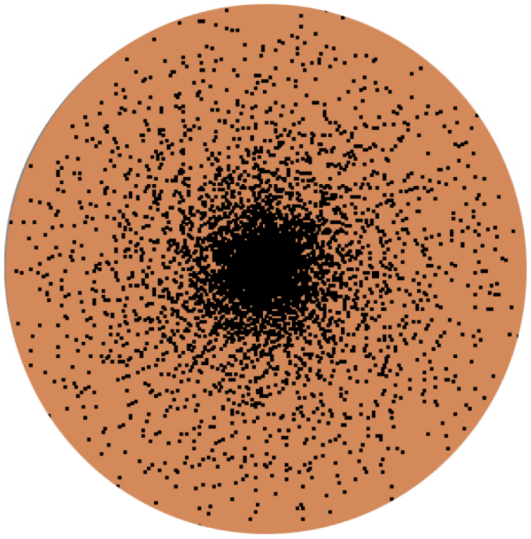


Figure 3.1: Locations of the photoreceptors (black dots) on the retina according to the noisy log-polar model.

in total—situated at

$$\boldsymbol{d}_k = e^{\rho_j} \begin{bmatrix} \cos\alpha_i \\ \sin\alpha_i \end{bmatrix} + \begin{bmatrix} \mathcal{N}(\mu, \sigma^2) \\ \mathcal{N}(\mu, \sigma^2) \end{bmatrix}, \quad \text{for } 1 \leq k \leq 14{,}400, \tag{3.1}$$

where $0 < \rho_j \leq 40$, incremented in steps of 1, and $0 \leq \alpha_i < 360°$, incremented in 1° steps, and where $\mathcal{N}$ denotes additive IID Gaussian noise. We set mean $\mu = 0$ and variance $\sigma^2 = 0.0025$. Fig. 3.1 illustrates the placement of photoreceptors on the retina. Other placement patterns are readily implementable, including more elaborate procedural models (Deering, 2005) or photoreceptor distributions empirically measured from biological eyes.[1]

3.1.1 Computing Irradiance at the Photoreceptors

The irradiance at any point on the hemispherical retinal surface at the back of the eye can be computed using ray tracing. Multiple sample rays from the positions of photoreceptors on the hemispherical retinal surface are cast through the finite-aperture pupil and out into the 3D virtual world where they recursively intersect with the visible surfaces of virtual objects

[1]This work was done principally by Dr. Masaki Nakada and/or Honglin Chen.

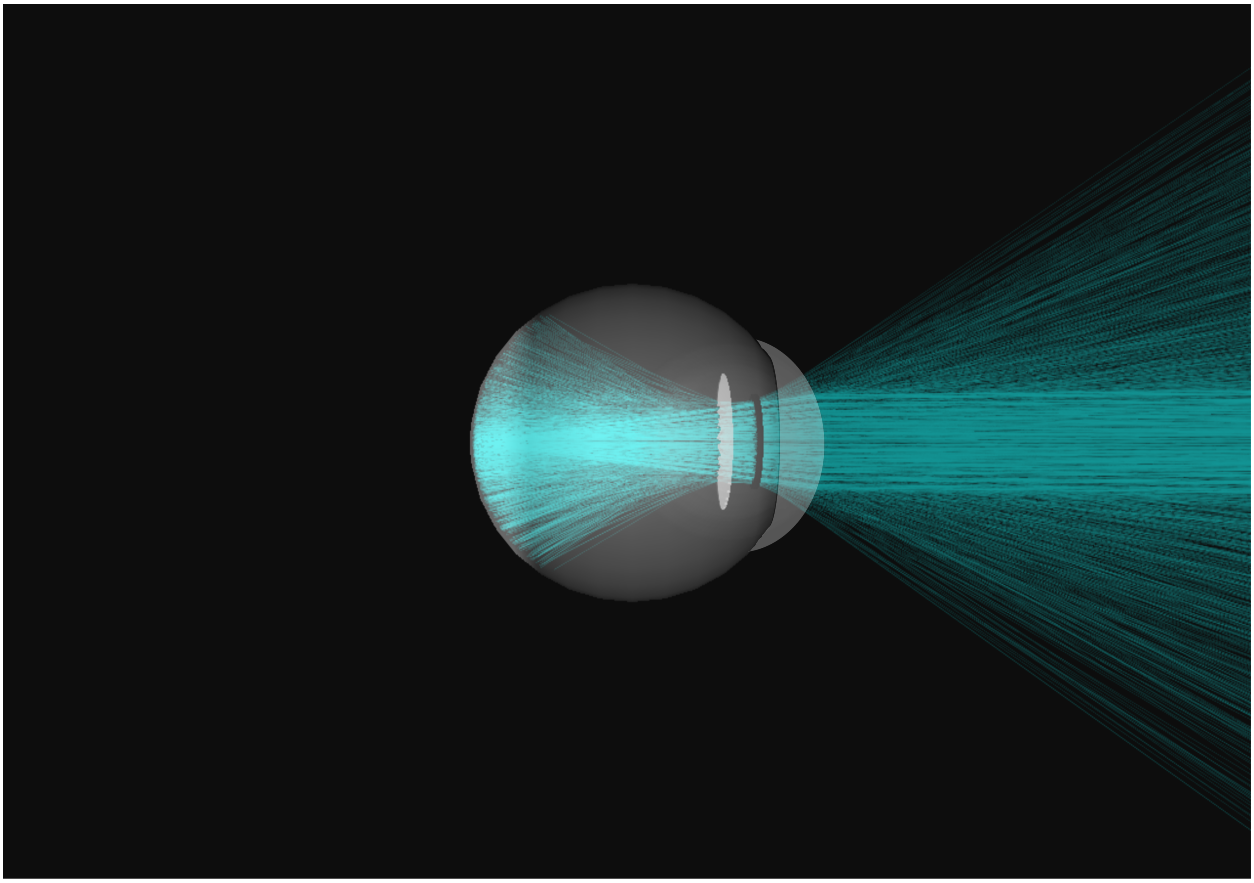


Figure 3.2: Rays cast from the positions of photoreceptors on the retina through the finite-aperture pupil and out into the scene to compute the irradiance responses of the photoreceptors according to the ray tracing process.

and query the virtual light sources in accordance with the Phong local illumination model. The irradiance values returned by these rays determine the light impinging upon the retina at the position of the photoreceptors. Fig. 3.2 illustrates the retinal imaging process.

3.2 Iris and Pupil

The iris controls how much light enters the eye through the pupil. The normal size of the human pupil is 2–4 mm diameter under bright light, and 4-8 mm in a dark room. We model the pupil as a circle with diameter ranging from 1 mm to 4 mm, and the iris as an annular region with a 6 mm external radius. The biological iris is actuated by sphincter and dilator muscles, but currently the pupil size is controlled kinematically in our model. Fig. 5.15 shows

a close-up view of our pupil model.

3.3 Lens and Cornea

Both the lens and the cornea refract light rays, resulting in a focusing of the image through the finite-aperture pupil. In the human eye, the greatest amount of refraction is caused by the cornea, with the lens making small adjustments to place the focal point onto the retina.

The cornea is modeled as a negative meniscus lens with non-uniform radii of curvature. More specifically, the outer face (closer to the outside environment) has a smaller radius of curvature than the inner face (closer to the iris-pupil complex). A uniform index of refraction is used throughout the entire organ to reduce the number of refraction computations. Spherical approximations for the two surfaces are used for the purposes of intersection computations to reduce the otherwise O(log(n)) complexity of a K-D tree search normally associated with a generic polygon mesh to O(1). The cornea is treated as a rigid body and therefore does not deform in the context of our model.

The lens is modeled as a volume preserving membrane with initial conditions determined by the approximate rest length radius of the surrounding ciliary muscle and a baseline minimum thickness. The body is connected to the surrounding muscle via a series of passive tendons known as zonules. When the ciliary muscle is completely deactivated, tension in the zonular tendons is maximized. The collective tension across all of the zonules generates a net compression force on the lens body, which in turn causes the lens to compress and achieve its minimum thickness. Conversely, at full activation the lens' restorative internal force causes the body to expand until an equilibrium is reached between the restorative force and the steadily building zonular tension caused by the expansion. The pressure force in this case is modeled as a series of damped springs between the lens' center of mass and the zonule attachment points, with the springs having an approximate rest length just slightly beyond the point at which equilibrium is achieved with the zonular tendons. We choose to model the pressure via a finite element method to exclude any computations that may apply to portions of the surface that are not directly affected by the ciliary muscle

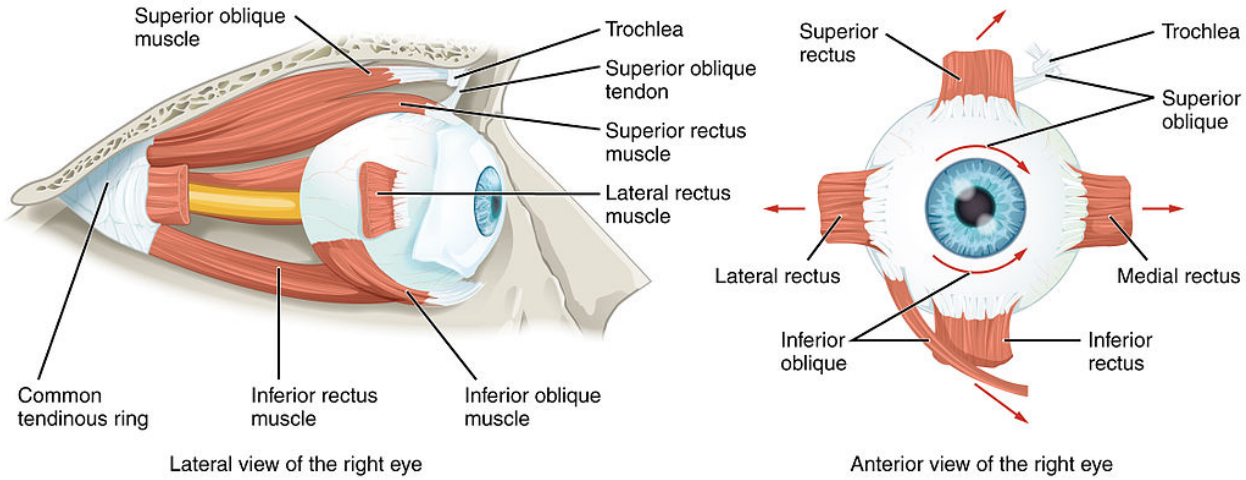


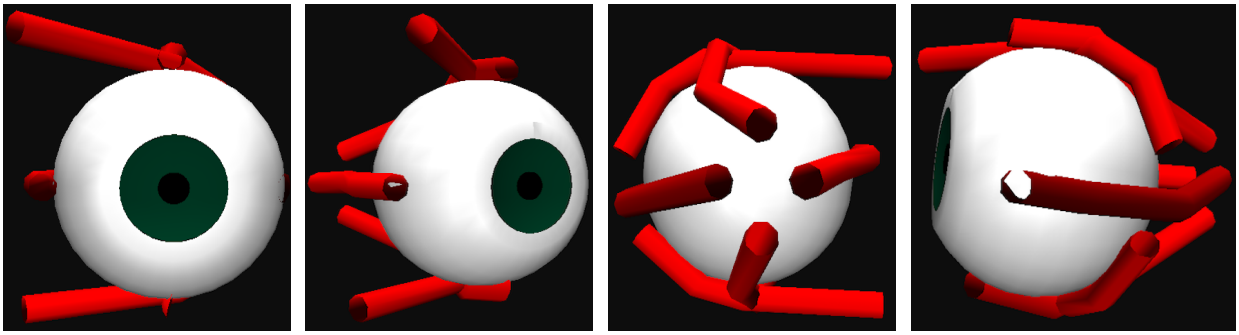
Figure 3.3: (a) Lateral view of the EOC muscles and (b) anterior view of principal muscles for the movement in each direction

activation. Spherical approximations are used to estimate the radius of curvature for both surfaces of the organ; however, as is the case in a biological eye, the radii of curvature of the surfaces typically are not equal. We again do this to reduce the computational complexity of intersection computations. As with the cornea, we use a single uniform index of refraction for the entire lens body to reduce the number of refraction computations.

The area of the lens actually used for photoreceptor sampling is determined by the radius of the pupil. Once the area has been determined, a collection of sampling points are designated on the surface. Each photoreceptor then emits a single ray through each of the sampling points; therefore, as the pupil expands, each photoreceptor is allowed to collect information from more sampling rays. The number of total rays cast during any simulation time step is therefore dependent on the luminosity of the environment and the corresponding size of the pupil which results from said environment.

3.4 Extraocular Muscles

The human eyeball is protected by the orbital bones of the skull. Fig. 3.3 is an anterior diagram of the right eye which reveals the positions of the extraocular contractile (EOC) muscles. The ocular nerves that innervate these muscles pass through an opening in the



(a) Front (b) Right (c) Back (d) Left

Figure 3.4: Eye model and EOC muscles from various view angles

superior orbital fissure of the eye. The eyeball is tightly suspended in the orbita and its motion is dictated by the direction of the six EOC muscles that surround it.

The 4 rectal muscles as well as the superior oblique all originate from a singular area towards the back of the eye socket of the skull called the orbital apex, while the inferior oblique originates from the maxilliary bone. The muscles that contribute to the movements of the eye are the lateral recti and medial for horizontal movements, the superior and inferior recti for vertical movements and the superior and inferior obliques for torsion movements.

We implemented a physics-based model of human eye movement in which the the eyeball is treated as a rigid body with a 3 rotational degree-of-freedom ball joint, which is actuated by the 6 EOC muscles. The equations of motion that govern this biomechanical model are presented in Appendix A.

Fig. 3.4 illustrates our biomechanical model. In each of the views, specific EOC muscles are outlined, while their respective attachment points can be differentiated based on the angle of viewing. One important point to note here is that all 4 of the rectal muscles originate from a similar point towards the orbital apex, while the oblique muscles are shown to originate from elsewhere. The EOC muscles are modeled using the modified Hill-type muscle model (Lee et al., 2009), the details of which are presented in Appendix B.

There are roughly 8 tertiary gazes (in addition to the primary forward position) into which eye movements can be segmented, but each gaze orientation involves strong activations of only a couple of muscles, with the remaining muscles contributing weakly. Fig. 3.3b provides

an illustration of these poses along with the primary muscle drivers of each pose. Direct vertical movements are effected by a co-contraction of a rectal and oblique muscle, whereas horizontal movements are driven solely by the medial and lateral recti.

CHAPTER 4

Neuromuscular Oculomotor Control

Fig. 4.1 presents an overview of our neuromuscular oculomotor control system for our model of the eye, revealing its sensory and motor subsystems, and the figure caption summarizes the function and connectivity of its 4 neural controllers, including 2 SNNs and 2 DNNs.

4.1 Optic Nerve Vector

The foveated retinal RGB "image" captured by the eye is output for further processing down the visual pathway, not as a 2D array of pixels, but as a 1D vector of length $14{,}400 \times 3 = 43{,}200$, which we call the Optic Nerve Vector (ONV). The raw sensory information encoded in this vector feeds the neural networks that directly control pupil size, lens accommodation, and eye movements.

More specifically, the ONV from the retina is fed to a pupil control SNN, which outputs a control signal to the iris, which constricts or dialates the pupil. The ONV is also fed to a lens control SNNs, which outputs an activation signal to the ciliary muscle that controls lens deformation, hence optical focus. The ONV is also fed to a foveation DNN which in turn drives a neuromuscular motor controller DNN that innervates the 6 EOC muscles.

4.2 Motor Subsystem

4.2.1 EOC Muscle Control

The 6 EOC muscle actuators are controlled by specifying for each of them a time-varying, efferent activation signal $a(t)$. This is performed by a trained DNN (Fig. 4.2) that takes

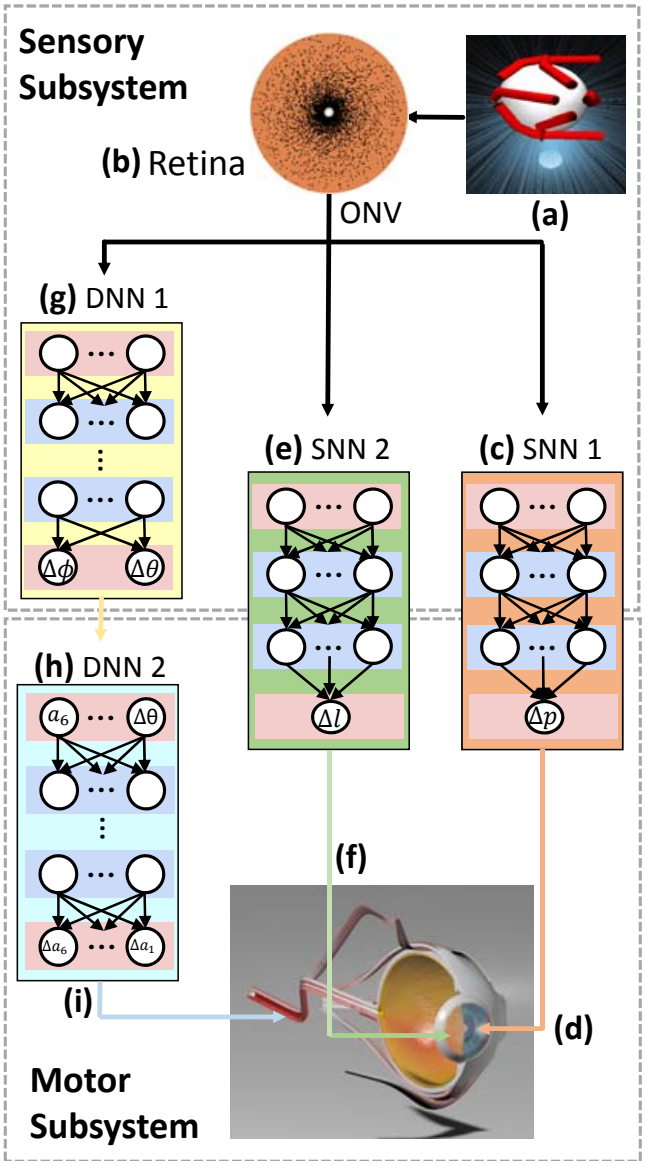


Figure 4.1: Oculomotor system architecture for the (left) eye, showing the modular neural network controllers in the sensory subsystem (top) and motor subsystem (bottom), including two DNNs (1,2) and 2 SNNs (1,2), of 4 types, colored orange, green, yellow, and blue. Within the sensory subsystem, (a) each photoreceptor on the retinas casts rays into the virtual world to compute the irradiance at the photoreceptor. (b) The arrangement of the 14,400 photoreceptors (black dots) on the foveated retina. The retina outputs a 43,200-dimensional Optic Nerve Vector (ONV). The (orange) iris SNN (c) inputs the ONV and outputs a constriction/dilation activation to the iris muscles. The (green) lens (e) controller inputs the ONV and outputs a lens muscle activation to control focal accommodation. The (yellow) foveation DNN (g) inputs the ONV and outputs angular discrepancy estimates to the (blue) neuromuscular motor controller DNN (h), which outputs activations to control the 6 EOC muscles in order to foveate and pursue visual targets.

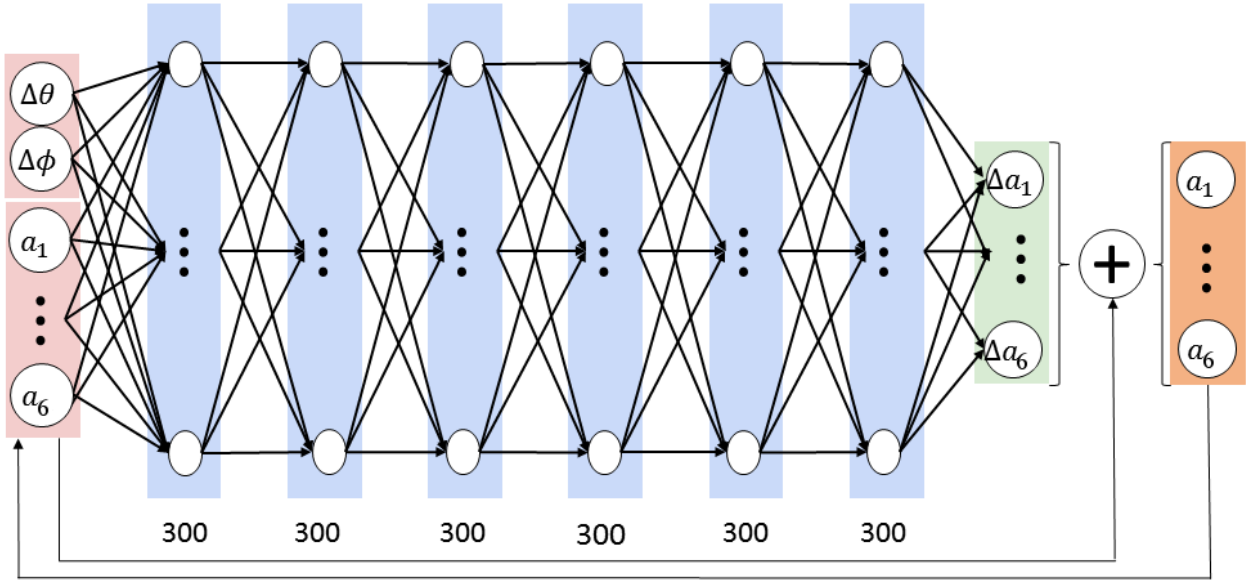


Figure 4.2: Architecture of the fully-connected voluntary controller DNN for the eye

as input the desired change in gaze direction, $\Delta\theta$ and $\Delta\phi$, and produces muscle activation adjustment signals $\Delta\boldsymbol{a}$. Hence, the output of this neuromuscular motor controller is given by

$$\boldsymbol{a}(t + \Delta t) = \boldsymbol{a}(t) + \Delta\boldsymbol{a}(t). \tag{4.1}$$

Note that the muscle activation signal $\boldsymbol{a}$ feedback loop makes the neuromuscular controller a Recurrent Neural Network (RNN).

4.2.1.1 DNN Architecture

The architecture of our DNN is consistent with those in the neuromuscular motor controllers for the biomechanical human musculoskeletal model in (Nakada et al., 2018d), for the reasons given in that article.

As shown in Fig. 4.2, the input layer of the eye voluntary motor DNN consists of 8 units that comprise 2 units for the eye rotation target angular discrepancies, $\Delta\theta$ and $\Delta\phi$, and 6 units for the activations a_i, for $1 \leq i \leq 6$, of the eye EO muscles. The output layer consists of 6 units providing the muscle activation adjustments Δa_i, for $1 \leq i \leq 6$. There are 6 hidden layers, each containing 300 units. The DNN is constructed using Rectified Linear

Units (ReLUs).

Next, we discuss the offline synthesis training data and the training of the DNN.[1]

4.2.1.2 Offline Training Data Synthesis

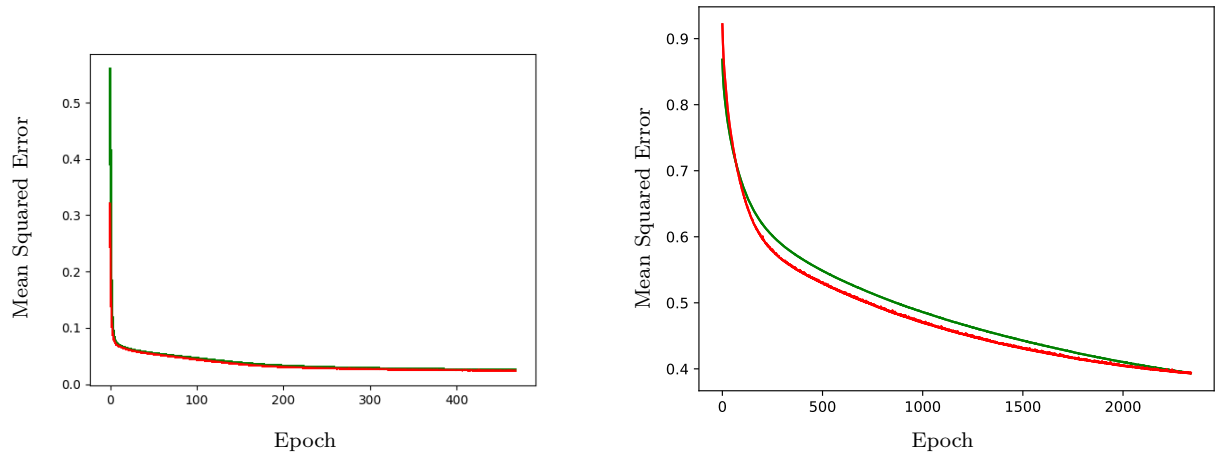
To train the DNN, we use our biomechanical eye model to synthesize training data as follows: Specifying a target orientation for the eye yields angular discrepancies $\Delta\theta$ and $\Delta\phi$ between the current eye orientation and the target eye orientation. With these angular discrepancies along with the current muscle activations a_i, $1 \leq i \leq 6$, as the input, we compute inverse kinematics followed by inverse dynamics with muscle optimization (see Appendix C) applied to the biomechanical eye model. This determines muscle activation adjustments, which incrementally reduce the angular discrepancies. These adjustments serve the desired output of the eye voluntary motor DNN. This *offline* training data synthesis process takes 0.1 seconds of computational time to solve for 0.02 simulation seconds on a 2.4 GHz Intel Core i5 CPU with 8GB Ram. Repeatedly specifying random target orientations and repeating the above synthesis process, we synthesized a large training dataset of 1M input-output pairs.

The dataset was randomly divided in a 90/10 split, with 90% of the points comprising the training set and the other 10% comprising the validation set. A test set was not partitioned since the resulting trained neural network was used as the online controller, and therefore a performance assessment of online sensorimotor control was the equivalent of gauging the test accuracy of the model.

4.2.1.3 DNN Training

The initial weights of the DNN are sampled from the zero-mean normal distribution with standard deviation $\sqrt{2/f\text{an\_in}}$, where $f\text{an\_in}$ is the number of input units in the weight tensor (He et al., 2015). For training, the loss function is mean squared error, and we apply Adaptive Moment (Adam) estimation (Kingma and Ba, 2014) as the stochastic optimization

[1]This work was done principally by Dr. Masaki Nakada and/or Honglin Chen.



(a) (b)

Figure 4.3: Progress of the backpropagation training process for (a) the eye motor DNN and (b) the pupil SNN on the training (green plot) and validation (red plot) datasets.

method, using the following parameters: $lr = 0.000001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, $\alpha = 0.001$, where β_1 and β_2 represent the exponential decay rate for momentum estimates taking an average and an average of squared gradients, respectively, ϵ prevents divide-by-zero errors, lr is the learning rate and α is the step size. An early stopping condition, negligible improvement for 10 successive epochs, is enforced to avoid overfitting.

The backpropagation DNN training process converged to a small error in 482 epochs. Fig. 4.3a shows the progress of the training process. After the DNN is trained, it serves as the *online* neuromuscular motor controller for the eye.[2]

4.2.2 Pupil and Lens Control

The pupil control network has 2 hidden layers with 2,000 and 1,000 units for the first and second hidden layers, respectively. A deeper network architecture is unnecessary. Other parameters including activation function and weight initializations are the same as for the DNN in Section 4.2.1.1.

To synthesize a training dataset, the light intensity is set randomly and the pupil size is

[2]This work was done principally by Dr. Masaki Nakada and/or Honglin Chen.

adjusted to receive a sufficient amount of light to perceive a target object clearly. The input of the network is the 1D ONV processed by the retina. The output control signal is simply to dilate/constrict the pupil with the size of the change required, or maintain the current size. We sample 200K input-output pairs of data.

The backpropagation SNN training process converged to a small error in 859 epochs. Fig. 4.3b shows the progress of the training process. The trained SNN serves *online* as a pupil controller.[3]

The lens control network has the same SNN architecture as that for the pupil, but rather than training on light intensity, it is trained on focal point error. To compute the focal point, we begin by casting rays from the object(s) in the scene in the directions of the vertex and/or surface normals. A subset of these rays will successfully penetrate the pupil and converge at a focal point determined by the refraction through the various mediums of the eye. The lens will then deform to align the resulting focal point directly on the retina, which achieves proper focus. The magnitude and direction of the focal point error (positive if in front of the retina and negative if behind) serves as the input while the ONV serves as the output of an output-input training pair. 200K input-output training pairs points are sampled to produce the training dataset.

4.3 Sensory Subsystem

4.3.1 LiNet Foveation DNN

In the human retina, there are approximately 5 million cones (Deering, 2005). Training a fully-connected neural network with that many photoreceptors is intractable due to memory constraints. Furthermore, a Convolutional Neural Net (CNN), which is designed for conventional, regularly-sampled image arrays, is inappropriate for our irregular distribution of retinal photoreceptors. Nakada et al. (2018a) propose liNets, a novel, locally-connected, irregular DNN architecture that effectively reduces memory consumption, thus enabling the

[3]The model was trained principally by Dr. Masaki Nakada.

incorporation of many more photoreceptors.

For a traditional fully-connected neural network, each unit in a hidden layer is connected to all the units in the previous layer, whereas for a locally-connected network, it is connected to a fixed number of neighboring units in the previous layer. A locally-connected network can be uniquely defined by a set of local connectivity matrices, where each matrix determines the connectivity between successive layers. Appendix D describes liNets in more detail.[4]

[4]The model was trained principally by Dr. Masaki Nakada.

CHAPTER 5

Experiments and Results

5.1 Forward Dynamic Motor Simulation

We were able to verify that the eye was satisfactorily modeled with all 6 EO muscles and a single 3 DoF ball-link as a cohesive system by simulating realistic forward dynamic motion. Varied activation signals were sent to each of and various combinations of the EO muscles in order to test compliance with realistic eye movement based on the activation input. The forward dynamic simulation environment eventually served as a testbed for verifying the quality and integrity of the synthesized training data prior to training the eventual DNN control system.

5.2 Kinematic Motor Control With Known Visual Targets

The following kinematic control experiments were performed with known visual target locations (i.e., without the sensory subsystem). We present them first to illustrate the unrealistic eye animation that results without the use of our sensorimotor controller.

5.2.1 Saccadic Movement

A visual target was instantaneously moved to random positions within the ocular field of view. Fig. 5.1 shows a few frames from the kinematic animation—the blue trace indicates eye movement during the transition from one position to the next, and the the red line indicates the exact gaze orientation from the center of the eye. We observe that the transition occurs in one frame and fixates to the target perfectly, which is robotic behavior that lacks natural

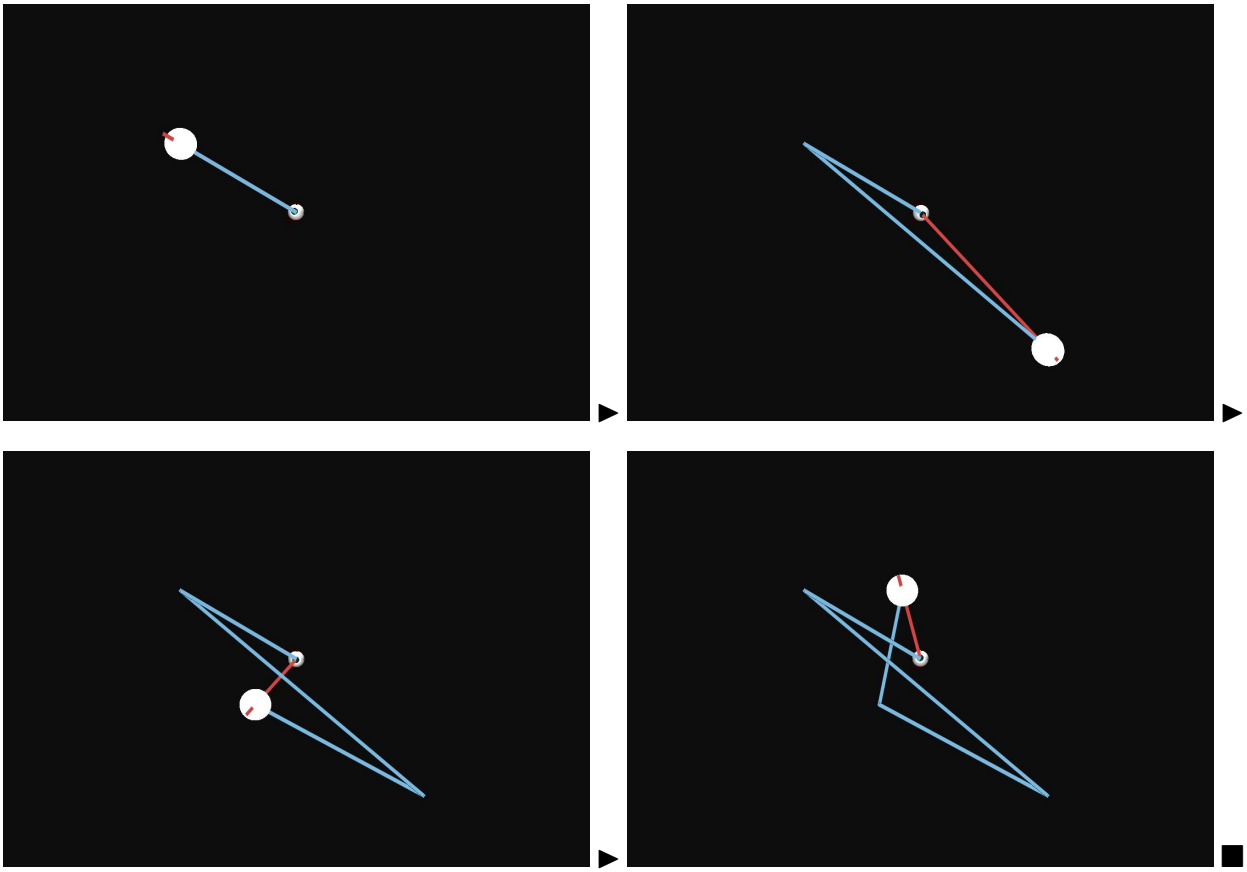


Figure 5.1: Sequence of frames from a kinematic animation of saccadic movement with a known visual target (white sphere) position using simplistic kinematic control. The blue trace indicates eye movement during the transition from one position to the next. The red line indicates the gaze direction from the center of the eye.

noise and motion latency.

5.2.2 Smooth Pursuit

A visual target was moved continuously over a variety of positions (pausing briefly at each) as opposed to instantaneously. Fig. 5.2 shows a few frames from the experiment. The eye tracks the visual target perfectly at every frame without any errors in gaze. This also looks robotic by contrast to natural human eye movement that is characterized by the presence of a certain amount of noise.

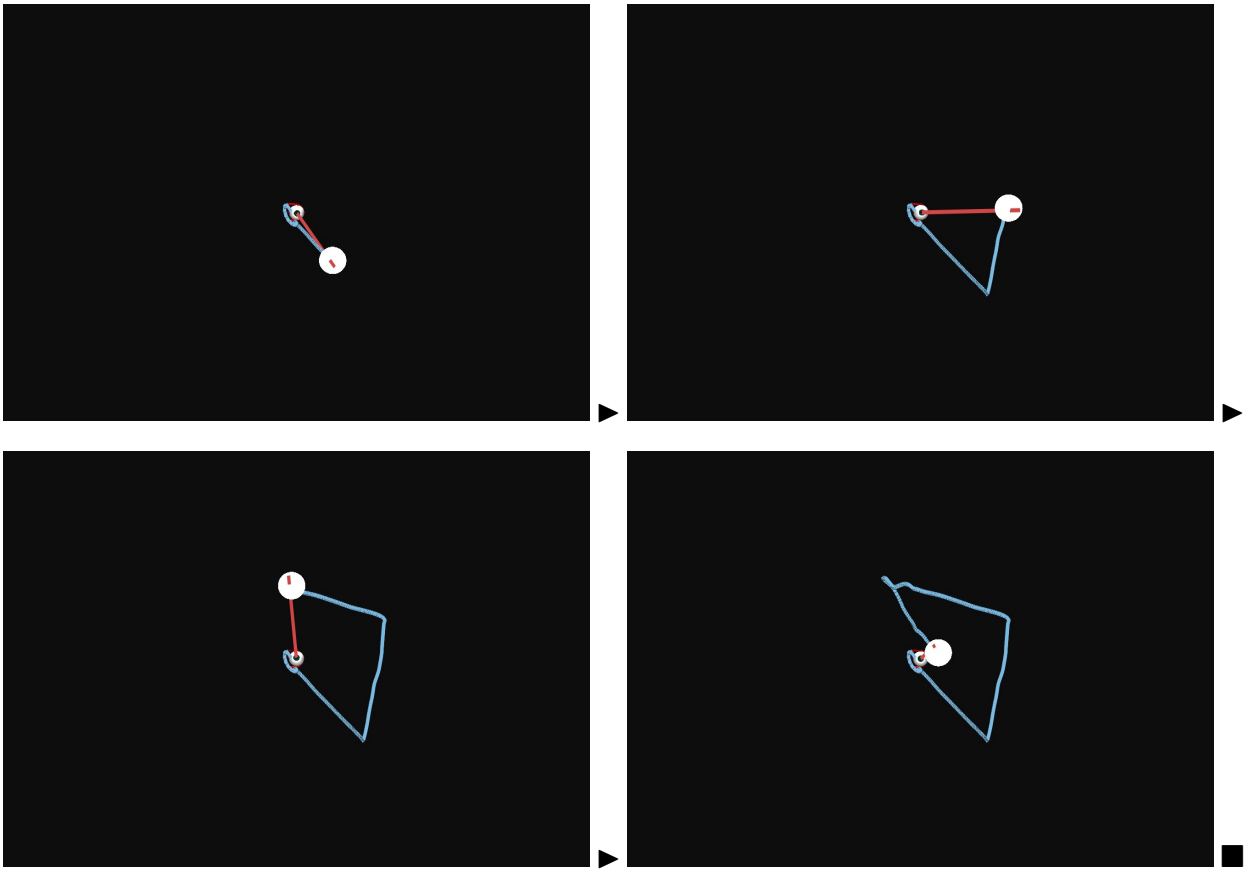


Figure 5.2: Sequence of frames from a kinematic animation of smooth pursuit with a known visual target (white sphere) position using kinematic control. The blue plot trace indicates eye movement during the transition from one position to the next. The red line indicates the gaze orientation from the center of the eye.

5.3 Non-Sensory Dynamic Control

One million training data points were synthesized by running the inverse dynamics simulation over a randomized set of target locations in order to capture as many gaze positions and transition behaviors as possible. The motor DNN was then trained via the offline training process using this training dataset.

The following experiments were performed using a known target location (i.e., independent of the sensory subsystem). We did this first to illustrate the independent robustness of the motor DNN before performing an additional round of experiments with full sensory DNN integration.

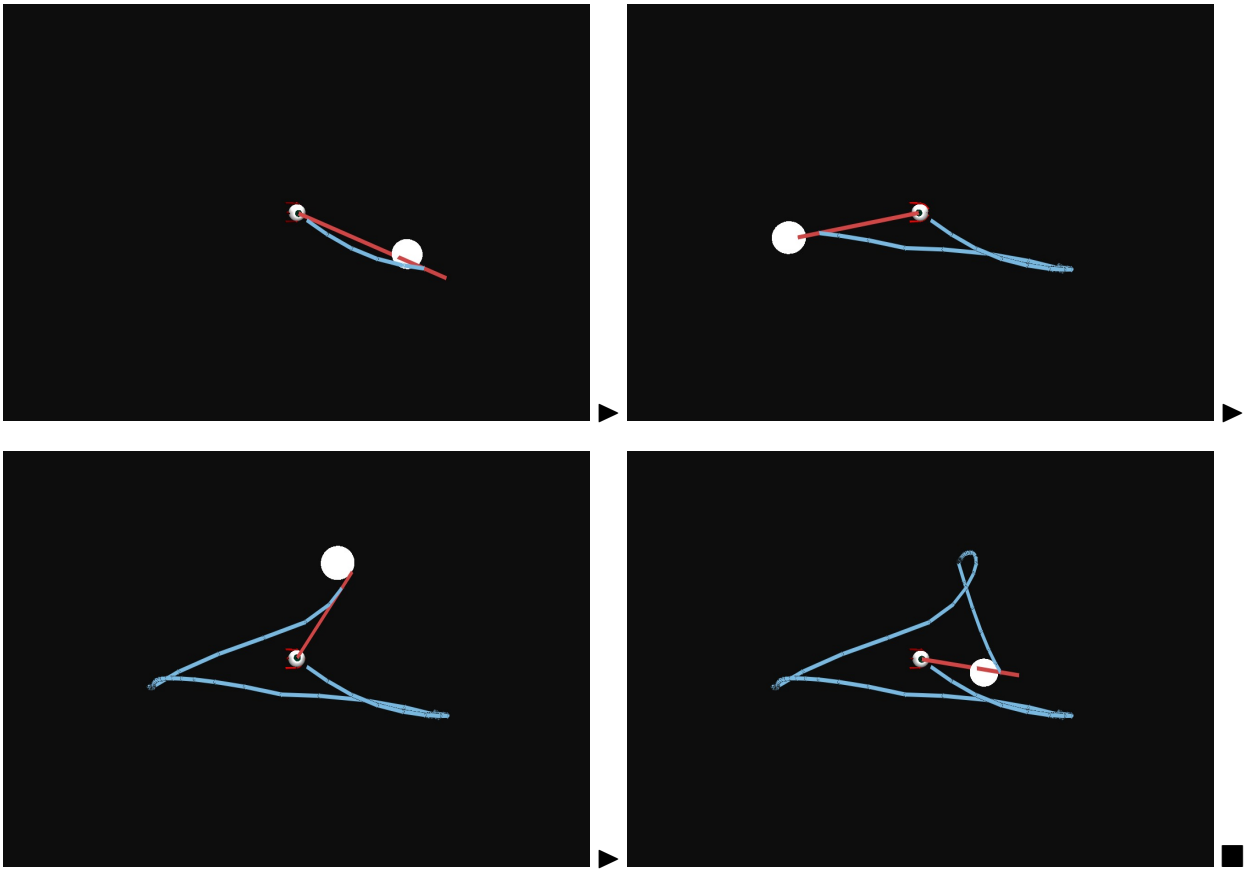


Figure 5.3: Sequence of frames from a kinematic animation of saccadic movement dynamics with a known visual target (white sphere) position and with motor DNN control. The blue trace indicates eye movement during the transition from one position to the next. The red line indicates the gaze orientation from the center of the eye.

5.3.1 Saccadic Movement

A visual target was instantaneously moved to random positions within the ocular field of view in a similar manner to the process used to synthesize the original training dataset via inverse dynamics simulation. Fig. 5.3 shows a small sample of the various target positions used. The blue line depicts the trace of the eye during the transition from one position to the next, and the red line indicates the exact gaze orientation from the center of the eye. We note here that the transitions occurred in a comparatively small number of frames and that the fixation on the target occurred extremely quickly out of the transition, consistent with our expectation of saccadic eye movement.

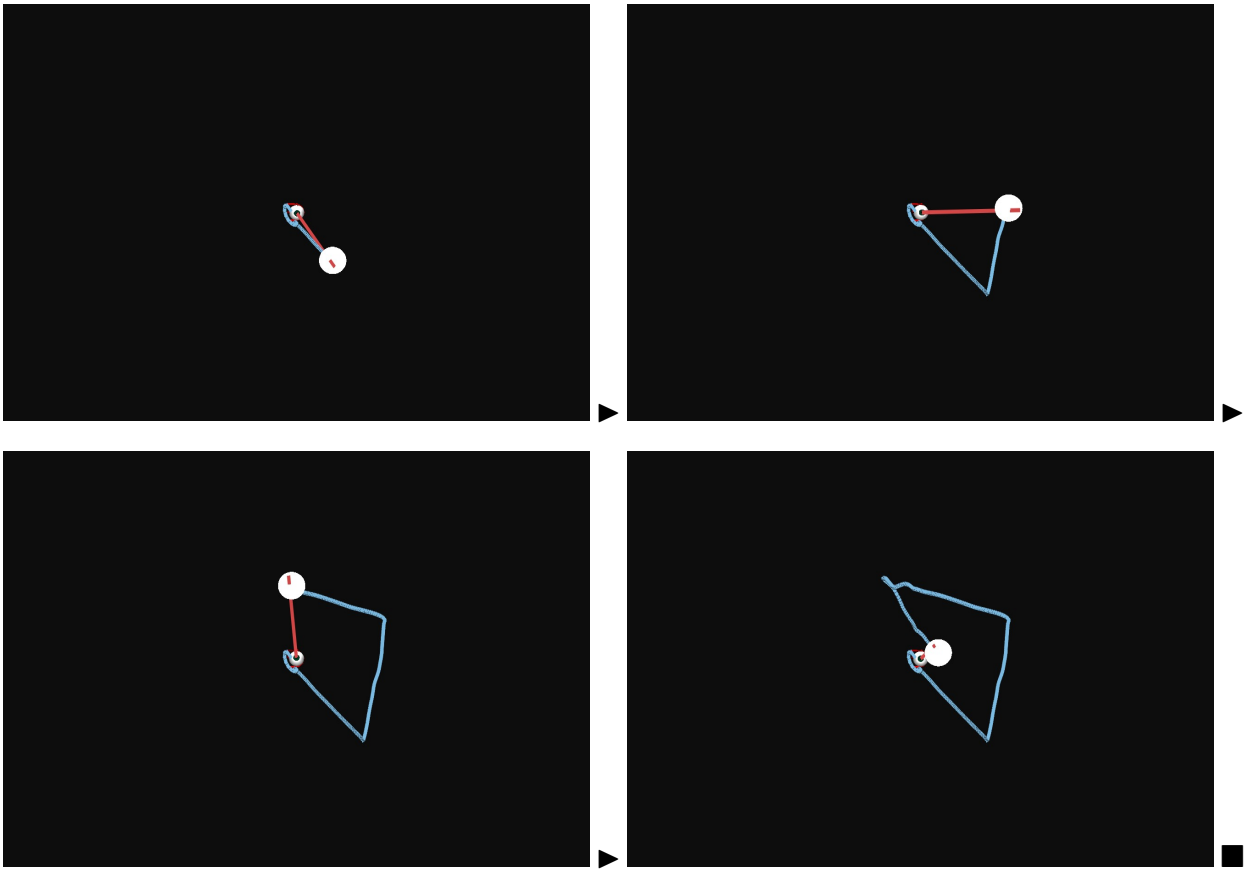


Figure 5.4: Sequence of frames from a simulation of smooth pursuit with a known visual target (white sphere) position with motor DNN control. The blue trace indicates eye movement during the transition from one position to the next. The red line indicates the gaze orientation from the center of the eye.

5.3.2 Smooth Pursuit

A visual target was moved semi-continuously, as opposed to instantaneously, over a variety of positions (while pausing briefly at each one). Fig. 5.4 shows a subset of the frames from the experiment. Despite not being trained in smooth pursuit dynamics, we found that merely a random placing of targets in essentially saccadic-only training data synthesis suffices for the system to generalize to smooth pursuit dynamics for a proximally shifting target. This finding was surprising considering the vastly differing characteristic of the two forms of movement, but it serves to highlight the generalizability of our training process.

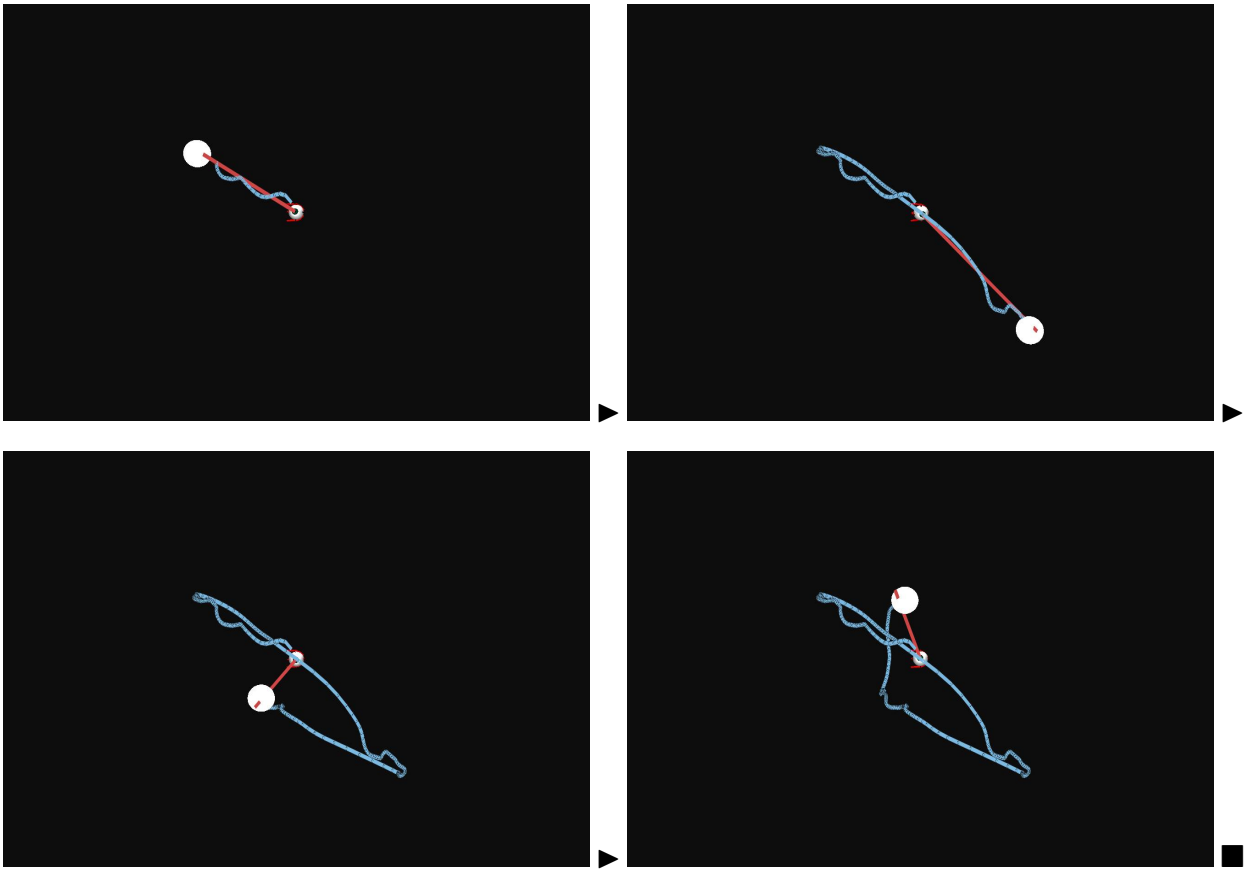


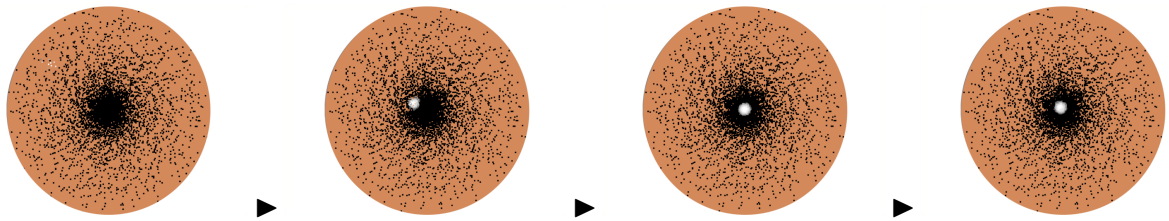
Figure 5.5: Sequence of frames from a simulation of saccadic movement dynamics with integrated motor and sensory DNN + SNNs. The blue trace indicates eye movement during the transition from one position of the visual target (white sphere) to the next. The red line indicates the gaze direction.

5.4 Dynamic Sensorimotor Control

The foregoing kinematic and non-sensory eye control experiments reveal that the resulting eye movements appear too perfect, hence unrealistic. Next, we report on several experiments related to saccadic and smooth pursuit eye movements driven by our oculomotor controller through visual stimuli processed by the trained sensory DNN and SNNs.

5.4.1 Saccadic Movement

Fig. 5.5 shows frames of a simulation that demonstrates noise in both saccadic movement and fixation, which results from the foveation DNN's observation of the visual target. This is characteristic of biological eye behavior, with which a comparison is reported below.



(a) t_0 (b) t_1 (c) t_2 (d) t_3

Figure 5.6: Time sequence (a)–(d) of photoreceptor responses in the left retina during a saccadic eye movement that foveates and tracks a moving white ball. At time t_0 the ball becomes visible in the visual periphery, at t_1 the eye movement is bringing the ball towards the fovea, the moving ball is being fixated in the fovea at time t_2, and the visual target keeps moving in the foveal area with micro-saccade movements at time t_3 and thereafter.

5.4.2 Fixation

We performed an experiment with our sensorimotor control model to compare behavior during fixation. Fig. 5.6 illustrates the behavior of a typical fixation movement. After rotating toward a target location (a)–(b) and centering the visual target on the fovea (c), the eye then exhibits small yet visible corrections (d), as the target is imperfectly centered in the fovea during fixation. In (d) we observe that the target slightly drifts away from the center of the fovea, only to be compensated for during a subsequent movement. This produces a net trace result of a cluster of gaze positions for a particular target location rather than an exact fixed point, as can be seen in the integrated saccadic movement traces in Fig. 5.5. We validate this behavior using a comparison to actual human subjects below.

5.4.3 Smooth Pursuit

Fig. 5.7 shows frames from the experiment with the aforementioned red/blue line trace. As with the saccadic movement comparison, we note here that the smooth pursuit line trace exhibits more noisy characteristics during transitions between various visual target positions.

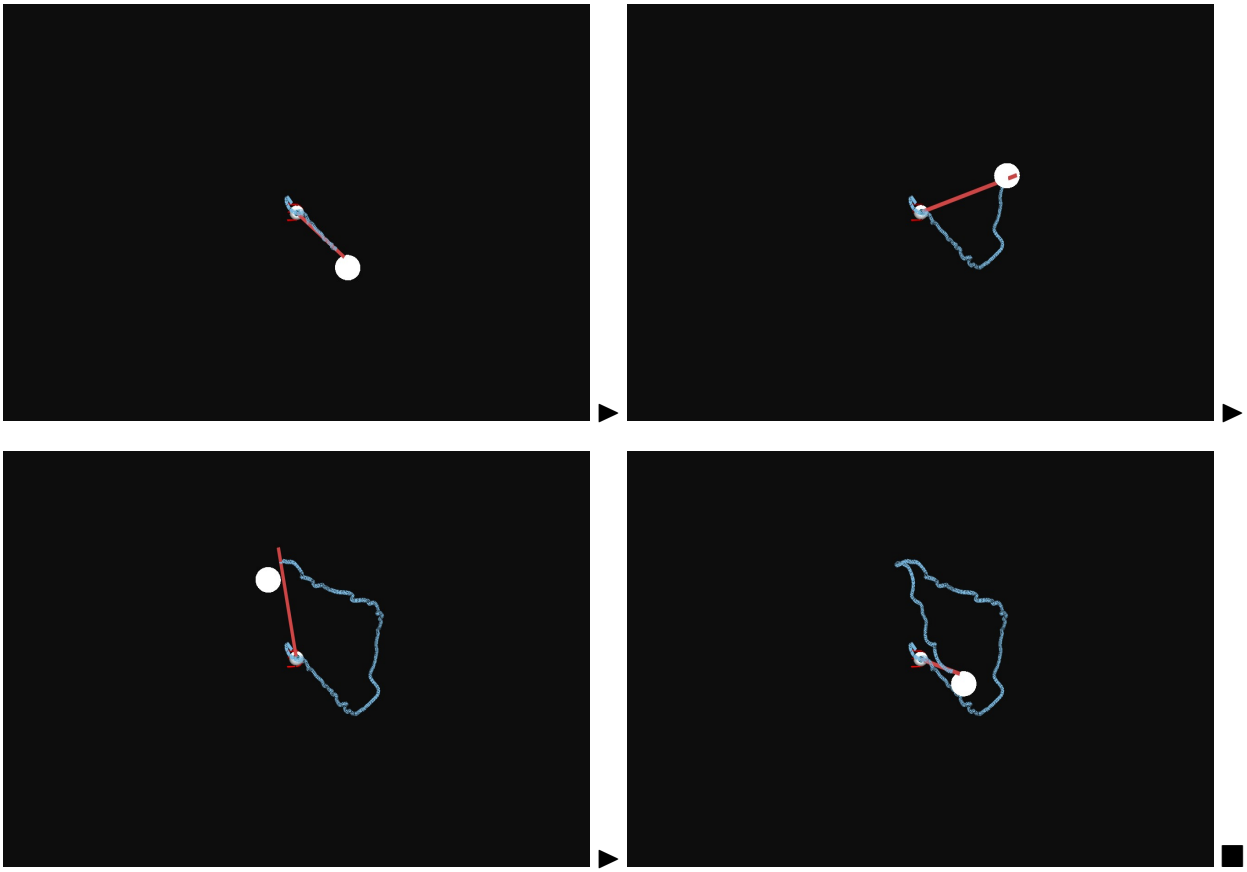


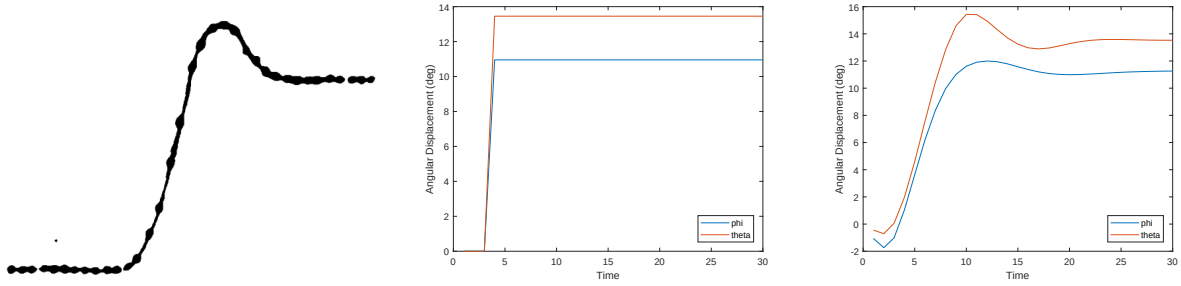
Figure 5.7: Sequence of frames from an animation of smooth pursuit movement dynamics with motor and sensory DNN + SNNs. The blue trace indicates eye movement during the transition of the visual target (white sphere) from one position to the next. The red line indicates the gaze orientation.

5.5 Comparison with Human Eye Movement

We now compare the results of our model both against the baseline kinematic human eye model as well as with biological data taken from human subjects (Thomas, 1969). We focus on saccade, smooth pursuit, and fixation.

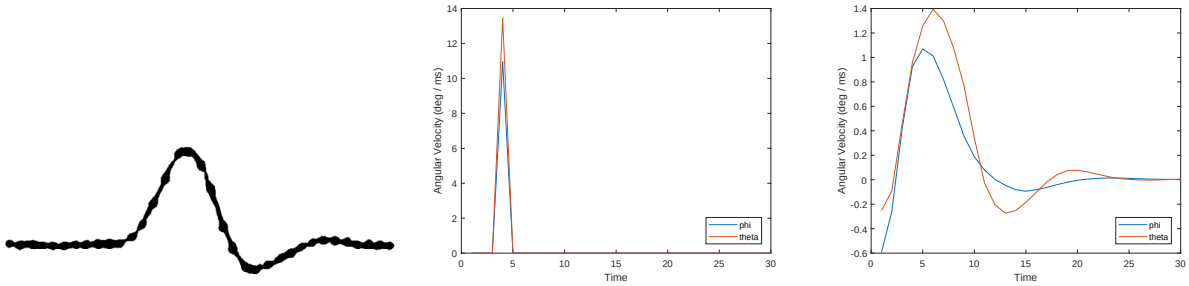
5.5.1 Saccade

Because it proved difficult to find raw data about actual muscle activations during saccadic movements, we instead focus on three accessible quantities: angular displacement, angular velocity, and angular acceleration of the eye.



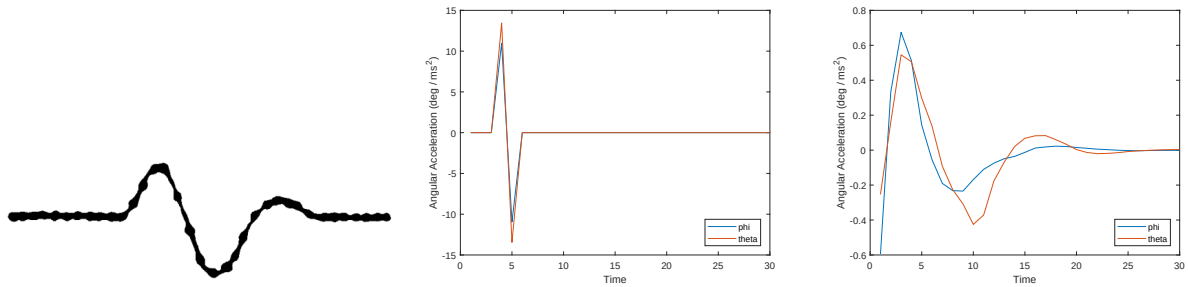
(a) Human (b) Kinematic model (c) Our DNN model

Figure 5.8: Comparison of angular displacement of saccadic eye movements. (a) Human subject. (b) Kinematic model. (c) Our DNN model. (orange: horizontal, blue: vertical)



(a) Human (b) Kinematic model (c) Our DNN model

Figure 5.9: Comparison of angular velocity of saccadic eye movements. (a) Human subject. (b) Kinematic model. (c) Our DNN model. (orange: horizontal, blue: vertical)



(a) Human (b) Kinematic model (c) Our DNN model

Figure 5.10: Comparison of angular acceleration of saccadic eye movements. (a) Human subject. (b) Kinematic model. (c) Our DNN model. (orange: horizontal, blue: vertical)

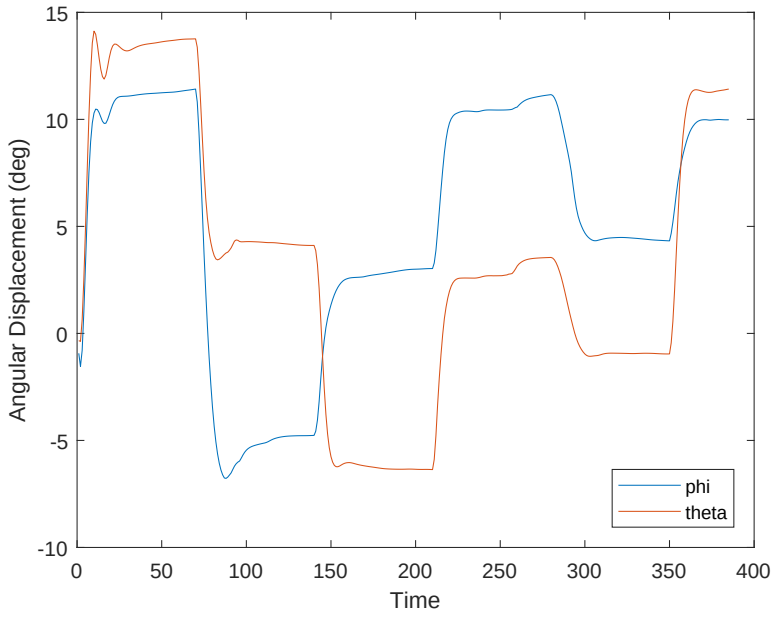


Figure 5.11: Plot of ocular angular displacement in saccadic and fixation movement over 6 random target placements. (orange: horizontal, blue: vertical)

Angular displacement: Fig. 5.8 shows a comparison between the eye angular displacement curves of a human subject, the kinematic model and our simulated model for a small saccadic movement of around 10-15 degrees. As can be seen from the plots, our simulated model curve resembles that of the human subject movement, while the instantaneous shift generated by the baseline kinematic control system is unrealistic. We therefore believe our biomechanical eye model is a firm improvement over the conventional, purely kinematic eye models of Lee and Terzopoulos (2006), Yeo et al. (2012), Nakada et al. (2018d), and many other researchers.

We performed additional experiments with various random target positions and recorded the corresponding orientations of the eye. Fig. 5.11 shows results for 6 such random orientations for both θ and ϕ ocular angles.

Angular velocity: Fig. 5.9 shows a side by side comparison between the eye angular velocity curves of the current kinematic model, a human subject, and our simulated model for a small saccadic movement of around 10-15 degrees. As can be seen from the plots, our

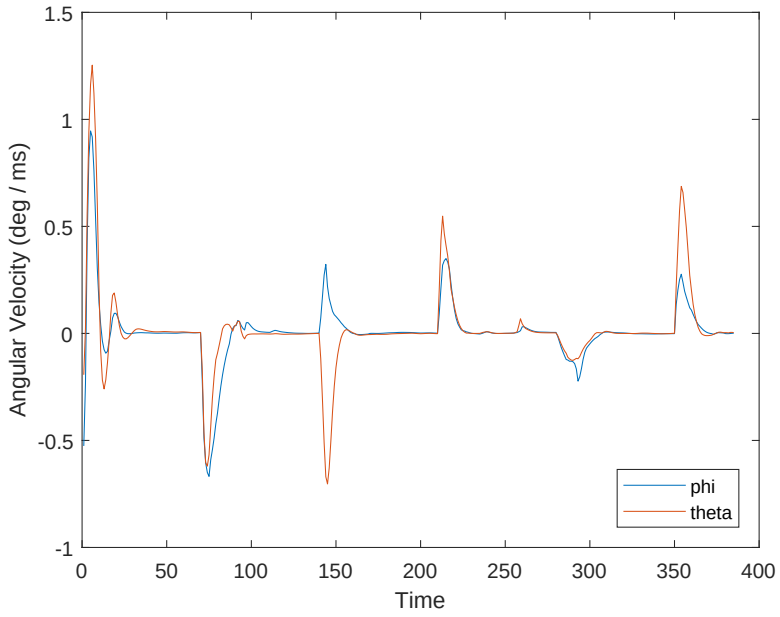


Figure 5.12: Plot of the angular velocity of saccadic eye movements during the fixation of 6 random placements of the visual target. (orange: horizontal, blue: vertical)

model curve largely resembles that of the human subject movement, while the transient spikes generated by the kinematic control system is unrealistic.

Additional experiments were performed in various random target positions, and the corresponding ocular orientations were recorded. Fig. 5.12 shows results for 6 such random orientations for both θ and ϕ ocular angles.

Angular acceleration: Fig. 5.10 shows a side by side comparison between the eye angular velocity curves of the current kinematic model, a human subject, and our simulated model for a small saccadic movement of around 10–15 degrees. As can be seen from the plots, our model curve greatly resembles that of the human subject movement, while the instantaneous spike generated by the prior kinematic control system is unrealistic.

We performed additional experiments in which various random target positions were used and the corresponding orientations of the eye recorded. Fig. 5.13 shows example results for 6 such random orientations for both θ and ϕ ocular angles.

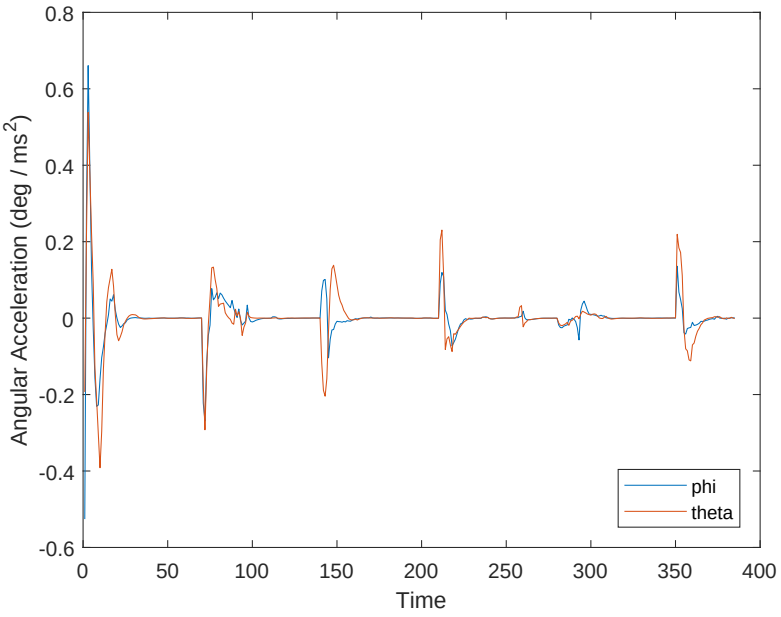


Figure 5.13: Plot of ocular angular acceleration in saccadic movements over 6 random target placements. The orange and blue line show the eye angular acceleration in horizontal and vertical direction, respectively.

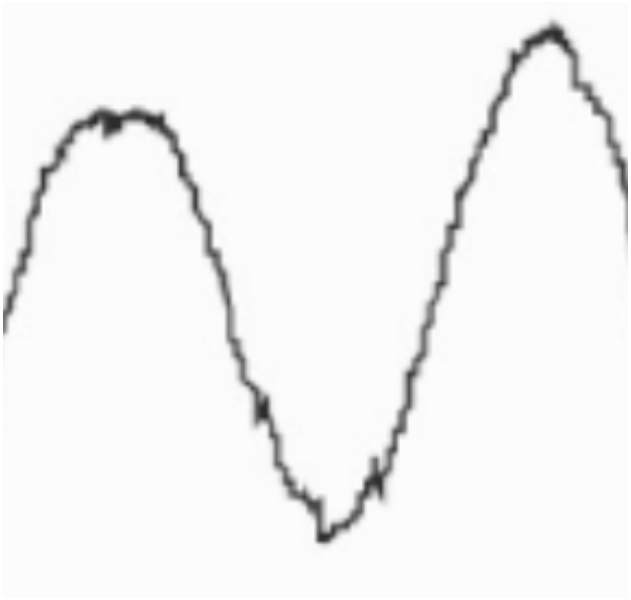
5.5.2 Smooth pursuit

Angular displacement: Fig. 5.14 shows a comparison between the eye angular displacement curves of a human subject measured by (Schraa-Tam et al., 2008) and our simulated model for a smooth pursuit following a visual target with sinusoidal pattern. As can be seen from the plots, our simulated model curve greatly resembles that of the human subject's eye movements.

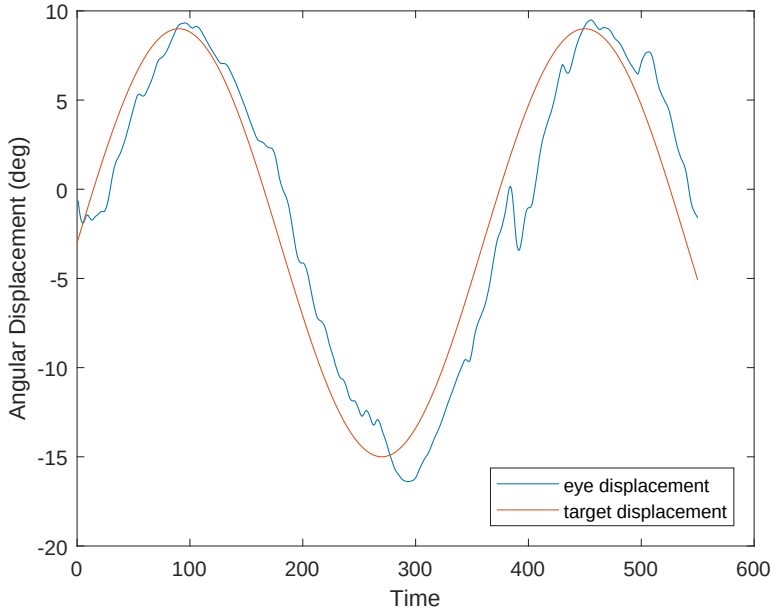
5.6 Pupil and Lens Control

To test pupil accommodation, we placed a light source in front of our eye model and changed the intensity of the illumination.

Fig. 5.15 illustrates the dilation and constriction of the pupil in response to the varying luminosity of the surrounding environment. Our experiments demonstrated that the pupil successfully made an appropriate constriction/dilation movement to adjust to the light en-



(a) Human



(b) Our DNN model

Figure 5.14: Plot of eye transition in smooth pursuit tracking a visual target, which moves with a sinusoidal curve in the horizontal direction. The plot shows the ocular angular displacement (blue line) and the visual target (red line) over time.

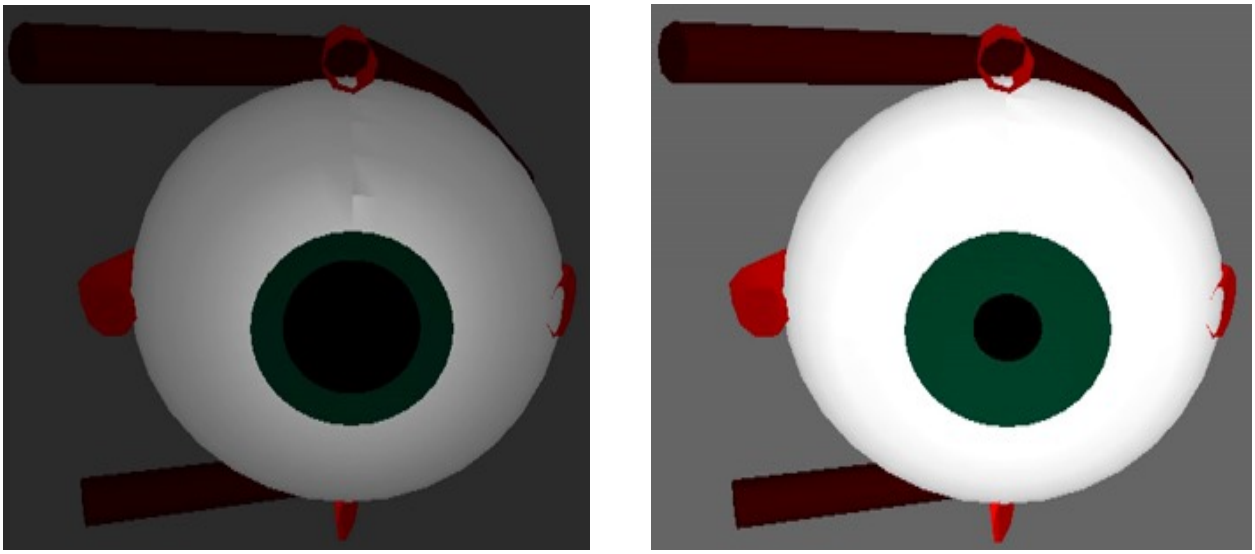


Figure 5.15: Close-up view of pupil under different illuminations.

tering the eye. The rate of constriction and expansion varied commensurately with the rate of change of light intensity.

We placed a visual target at arbitrary distances to test lens accommodation control. The lens successfully makes an appropriate accommodation to refract the incoming light so as to focus it onto the retinal surface.

5.7 Binocular Eye Control in a Musculoskeletal Model

Finally, we conducted an experiment using the neuromuscular biomechanical human model of Nakada et al. (2018d), in which we replaced their kinematic eye model with our biomechanical eye model. The frames from the full body control animation and corresponding retinal image are shown in Fig. 5.16. The two biomechanical eyes successfully foveated and pursued the visual target while the target and head are moving, which demonstrates Vestibular Ocular Reflex (VOR).

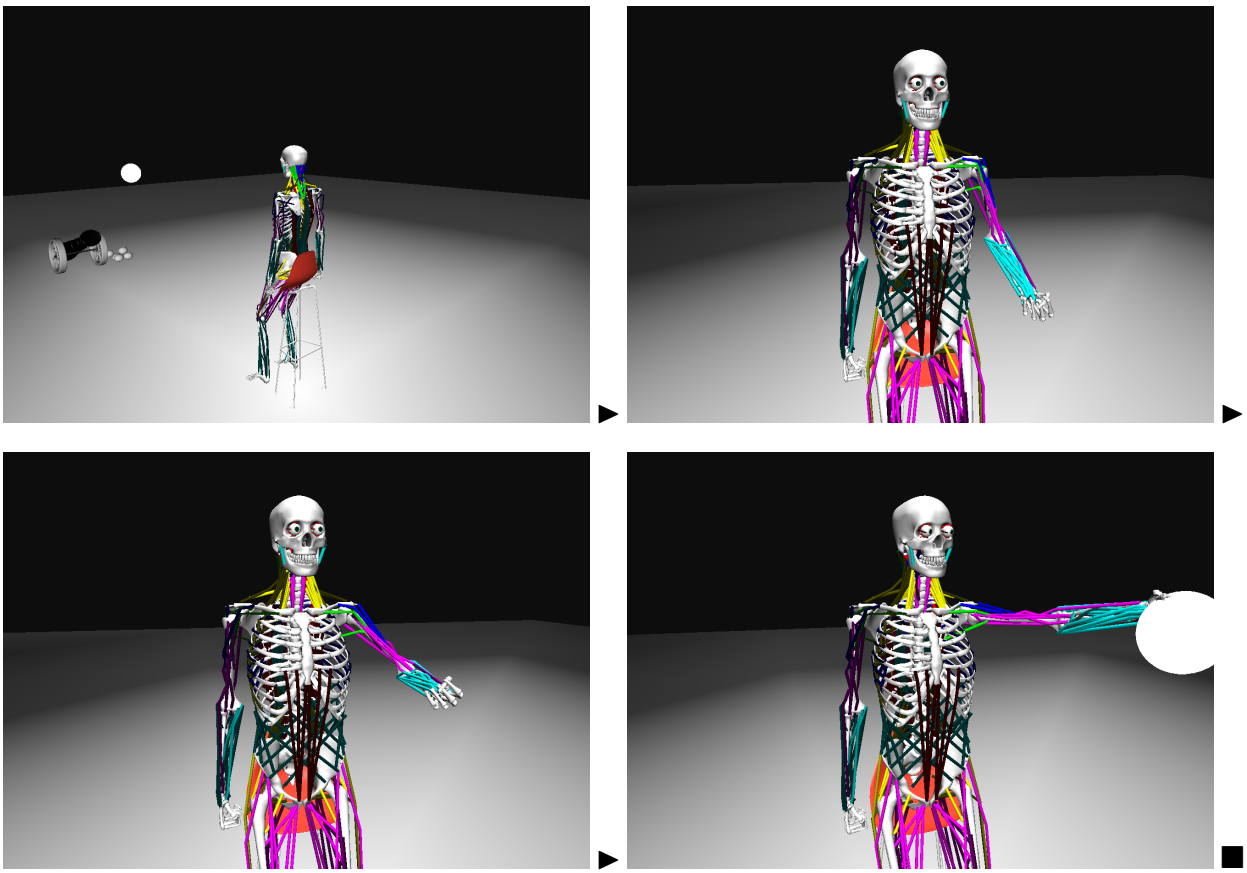


Figure 5.16: Sequence of frames from an animation of our biomechanically-simulated virtual eyes in the head of a biomechanically-simulated virtual human sitting on a stool and actively executing eye gazing movements. The ball is perceived by the eyes, processed by the vision DNNs, and foveated and tracked through eye movements, which are actuated by the neuromuscular oculomotor controller in conjunction with head movements governed by the virtual human's neuromuscular cervicocephalic motor controller. The reactive VOR-like motions stabilize the eyes on the visual target while the head is in motion.

CHAPTER 6

Conclusion

Our primary contributions in this thesis have been twofold: We have developed a novel biomechanical model of the eye with unprecedented biomimetic detail. We have also introduced a neuromuscular oculomotor controller for this model. The controller incorporates deep and shallow neural networks, which are trained offline using data synthesized by the eye model itself and, once trained, work efficiently to control the eye's extraocular and intraocular muscles. We also employed a novel, locally-connected sensory neural network and demonstrated that it can accommodate a large number of nonuniformly distributed retinal photoreceptors. Finally, we demonstrated consistency relative to human eye movements even when our eye/oculomotor model is incorporated into the sensorimotor control system of an advanced biomechanical human musculoskeletal model.

6.1 Discussion

The biomechanical model provided an ideal test platform for observing the effects of muscle placements in eye actuation, and for ensuring consistency with the muscle-optimized inverse dynamics control results. In particular, we were able to test that full activation of any particular muscle indeed actuated the eye in accordance with biological principles.

Although not perfect, the inverse dynamics and neural network controlled models both demonstrated a reasonable degree of fidelity relative to biological findings. The divergence was primarily in differences in activation levels of the superior rectus and inferior oblique, and the mirror images of the inferior rectus and superior oblique. However, because we imposed no constraint on the optimization of these activations save for a lower bound on

each muscle, there is nothing biasing the computation to favor certain muscles in particular eye orientations to ensure biological consistency. This of course is not critical if the goal is to achieve a model that works, but it is a concern if the goal is a model that accurately matches recorded EOC muscle activation level data during real human eye movement.

We have stayed true to using a purely voluntary neuromuscular control scheme since there does not seem to exist an underlying muscle reflex control layer in the oculomotor system. This does not cause much difference in the behavior of the inverse dynamic model, but it causes some noise to result in the neuromuscular control scheme. This is expected since, in an effort to avoid over-fitting, the voluntary neural network does not necessarily learn every nuance in muscle activation transitory behavior, and the noise actually appears to be more natural because it seems infeasible for the brain to compute inverse dynamics on a millisecond basis in order to obtain exact control solutions. Rather, fast compensation for errors and noise generated by the imperfect motion control is the more natural strategy.

6.2 Future Work

A simple Hill-type muscle model was employed and the inclusion of the pulleys was largely neglected. We plan eventually to introduce a robust pulley model, perhaps like the strand muscle model proposed by (Wei et al., 2010).

Secondly, instead of only enforcing energy minimization, we would like to place additional constraints on, or perhaps completely alter, the objective function in the inverse dynamics muscle optimizer such that the output activations may be more consistent with biological data taken from human subjects.

The pupil in our model is controlled kinematically in order to focus on its main function, which is to adjust the light entering the eyes. The biological pupil is controlled by pupillary sphincter and dilator muscles on the iris; however, that relationship is currently not captured. A better, physical model is called for.

APPENDIX A

Equations of Motion of the Eyeball

Following the formulation in (Lee et al., 2009), the equations of the motion for the eyeball can be written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{P}(\boldsymbol{q})\boldsymbol{f}_c + \boldsymbol{J}(\boldsymbol{q})^T \boldsymbol{f}_e, \tag{A.1}$$

where $\boldsymbol{q}$ is a state vector comprising the rotational degrees of freedom of the eyeball, $\dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{q}}$ are the corresponding velocities and accelerations, $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{C}$ is the matrix that accounts for gravity and Coriolis forces. The Jacobian matrix $\boldsymbol{J}$ transforms the external forces to joint torques, and the moment arm matrix $\boldsymbol{P}$ maps the muscle forces to joint space, $\boldsymbol{f}_c$ denotes the vector for contractile muscle forces and $\boldsymbol{f}_e$ denotes the vector for external forces. Finally, (A.1) may be written compactly as

$$\ddot{\boldsymbol{q}} = \phi(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{\tau}). \tag{A.2}$$

We solve forward dynamics to compute ϕ by calculating $\ddot{\boldsymbol{q}}$ from the generated torque produced by the muscle force. Then, we use the implicit Euler time integration method to solve the linearized equations of motion. We can compute velocity at the next time step, where Δt is the step size, by solving

$$\dot{\boldsymbol{q}}(t + \Delta t) - \dot{\boldsymbol{q}}(t) = \Delta t \phi(\boldsymbol{q}(t + \Delta t), \dot{\boldsymbol{q}}(t + \Delta t), \tau). \tag{A.3}$$

The problem here is that this equation has the variable at the next time step on the right hand side of the equation. We use a first-order approximation and rewrite the equation as

follows:

$$\begin{aligned}\delta\dot{\boldsymbol{q}} &= \Delta t \left[\phi(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \tau) + \frac{\partial\phi}{\partial\boldsymbol{q}}\delta\boldsymbol{q} + \frac{\partial\phi}{\partial\dot{\boldsymbol{q}}}\delta\dot{\boldsymbol{q}}\right] \\ &= \Delta t \left[\phi(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \tau) + \frac{\partial\phi}{\partial\boldsymbol{q}}\Delta t(\dot{\boldsymbol{q}}(t) + \delta\dot{\boldsymbol{q}}) + \frac{\partial\phi}{\partial\dot{\boldsymbol{q}}}\delta\dot{\boldsymbol{q}}\right].\end{aligned} \tag{A.4}$$

where

$$\delta\dot{\boldsymbol{q}} = \dot{\boldsymbol{q}}(t + \Delta t) - \dot{\boldsymbol{q}}(t) \tag{A.5}$$

Thus, we can compute the joint velocities at the next time step. Then the joint angles at the next time step can be computed using time integration.

APPENDIX B

Extraocular Muscle Model

The muscle force $f_m = f_P + f_C$ is the combination of two components. The passive element f_P, which produces a restoring force due to the material elasticity to the deformation, is represented as a uniaxial exponential spring, as follows:

$$f_P = \max(0, k_s(\exp(k_c e) - 1) + k_d \dot{e}), \tag{B.1}$$

where k_s and k_d are the stiffness and damping coefficient, respectively, e is the strain of the muscle and $\dot{e}$ is the strain rate. The contractile element f_C, which actively generates the contractile force of the muscle, is computed as

$$f_C = aF_l(l)F_v(\dot{l}), \tag{B.2}$$

where F_l is the force-length relation, F_v is the force-velocity relation, and a is the muscle activation level ($0 \leq a \leq 1$), which serves as the control input to the actuator. Fig. B.1

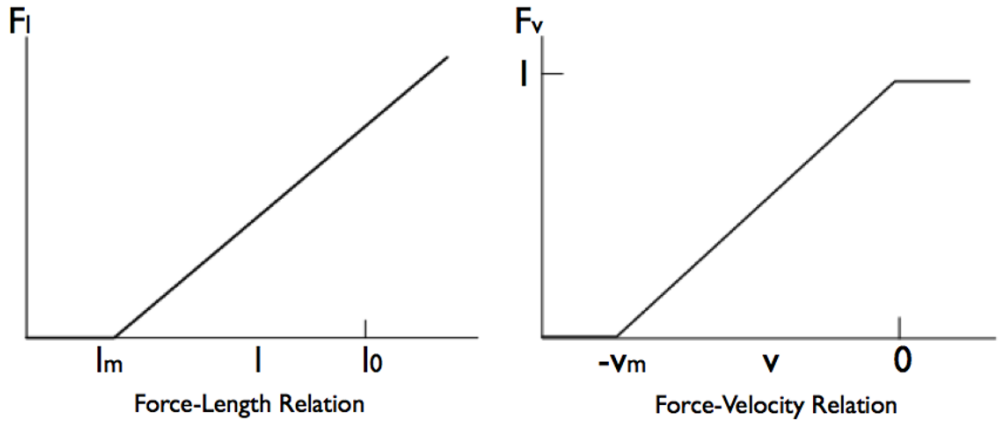


Figure B.1: The force-length and force-velocity relations of the Hill-type muscle model.

plots the force-length and force-velocity relations. The former, F_l is represented as $F_l(l) = \max(0, k_{max}(l - l_m))$, where l_m is the minimum length for muscle to generate the force, k_{max} is the maximum stiffness of activated muscle. The latter, $F_v(\dot{l})$ is represented as $F_v(\dot{l}) = \max(0, 1 + min(\dot{l}, 0)/v_m)$, where v_m is the maximum contraction velocity with no load. The coefficient k_c is set to 7 for all the muscles, and I_m is set to $0.5l_0$ and v_m is $l_0 sec^{-1}$. The other coefficients k_s, k_d and k_{max} are scaled to be proportional to the strength of weight factor of each muscle, which is calculated as roughly proportional to the cross sectional area of the muscle. The list of weights can be found in (Lee et al., 2009).

In the modified Hill-type model, $F_l(l)$ increases monotonically. This works for our biomechanical model because the extraocular muscles stretch only a modest amount. Thus, we avoid negative stiffness, which potentially causes instability in the numerical simulation of the musculoskeletal system.

APPENDIX C

Muscle Optimization

Inverse kinematics is first solved to compute the desired joint angles for the eyes to look at a given target object.

With the target kinematic pose computed, the desired acceleration needed to reach the target is computed. Because the eye is able to move at the astonishingly fast rate of 900° per second, it is not necessary to impose a strict upper bound on the maximal velocity; hence, the acceleration is permitted to be high enough to reach the target orientation in very few steps. By integrating over successive time steps using these desired accelerations, we are able to compute the torque necessary to actuate the eye to the new orientation.

The desired torque is then used as the basis of a muscle optimizer, which ultimately computes the activations for each muscle. A numerical solution is computed as an energy minimization problem, where the objective is to minimize the total energy output by the combined activations. This may not necessarily be the ideal problem to optimize in the case of the eye, as it may be more desirable to optimize speed of actuation instead (though neither of these approaches is supported directly by biological evidence and it is at best just speculation); however, the results empirically demonstrate that the energy minimization optimization generates reasonable eye movements.

An additional non-zero constraint is imposed on all of the activations to prevent any of the muscles from being completely deactivated during a movement, owing to the belief that all the EOC muscles contribute at least minimally to all eye movements. Intuitively, this is expected only to enhance biological accuracy without loss of significant solution space considering that any agonist-antagonist muscle pair could simply be equally co-activated to maintain the eye in a neutral position.

All activations are then converted to muscle forces, allowing for the resulting system to be time integrated once again using the implicit Euler method to compute the new positions at each time step.

APPENDIX D

LiNets

In a fully-connected neural network, each neuron in a hidden layer is connected globally to all the neurons in the previous layer, whereas as in the case of a CNN, each neuron in a hidden layer of a liNet is connected locally to a fixed number of neighboring units in the previous layer. However, whereas the implementation of a CNN assumes conventional, regularly-sampled images and regular placements of units in array and tensor data structures, the liNet does not.[1]

D.1 Architecture

Referring to Fig. D.1, suppose that we have a liNet with H hidden layers, each of which is comprised of N_h neuronal units. Each unit in hidden layer $h = 1, \dots H$ forms a receptive field with R_h nearest neighbor units in the previous hidden layer $h-1$. Each unit is assigned a position in the 2D retinal domain that is the average of the positions of the units in its receptive field, and nearest neighbors are determined using a Euclidian-distance k-nearest-neighbor algorithm. The receptive fields of the units in the first hidden layer $h = 1$, are formed from R_1 neighboring retinal photoreceptors, whose positions $\boldsymbol{d}_p$ are specified in (3.1). Given an irregular, foveated photoreceptor distribution, the overlapping receptive fields, which are illustrated by the white circles in the retinal domain at the left of Fig. D.1, naturally increase in size from the denser fovea out to the sparser periphery.

Our flexible liNet architecture is defined by a set of H connectivity matrices. Each

[1]This appendix describes work done principally by Dr. Masaki Nakada and Honglin Chen. The full details are found in Nakada et al. (2018a).

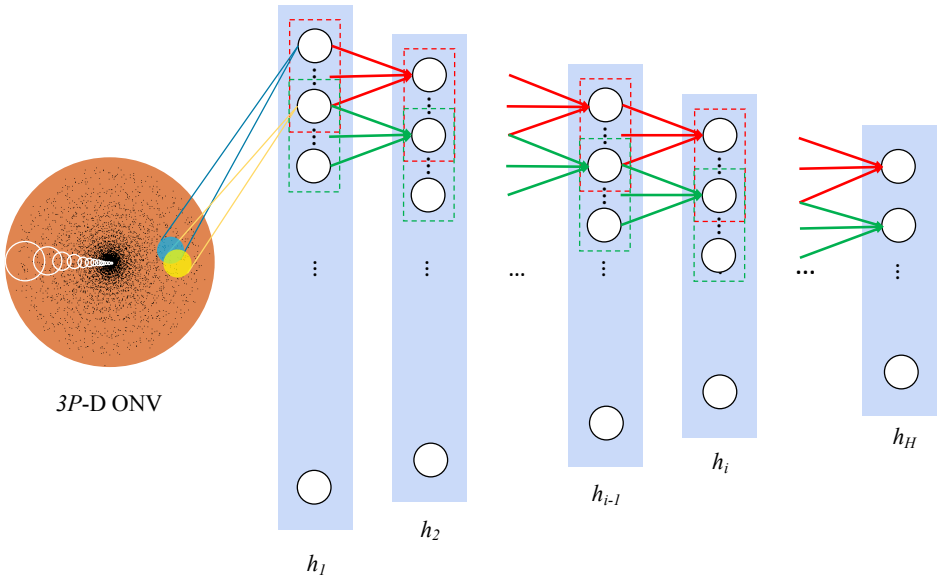


Figure D.1: LiNet architecture

$N_h \times R_h$ matrix $\boldsymbol{C}_h$ specifies the connectivity between units in hidden layer h and units in hidden layer $h-1$. Each row in a connectivity matrix stores the indexes of the units in its receptive field, where the index of a unit in hidden layer h is $n = 1, \ldots, N_h$, while the (ONV) index of a retinal photoreceptor is $p = 1, \ldots, P$.

In practice, we use Rectified Linear Units (ReLUs) in our liNet. Each ReLU computes a weighted sum of the outputs of units (or photoreceptors in the case of units in the first hidden layer) within its receptive field. The weights, which are stored in a set of weight matrices $\boldsymbol{W}_h$ of size $N_h \times R_h$, and the biases, which are stored in bias vectors of size N_h, are learned from training data using standard backpropagation learning techniques. It is important to realize that, unlike a CNN in which weights and biases are shared by all the units in each hidden layer, each and every unit of an liNet has its own weights and bias.

D.2 Memory Efficiency and Training Performance

To examine the memory efficiency of the locally-connected deep neural network, we compare its memory consumption to that of the fully-connected network. Given a fixed input dimension, we construct both networks with a fixed number of hidden layers and fixed number of units in each hidden layer. In particular, we fix the number of hidden layers $h = 4$, and the number of units in each hidden layers is equal to the number of units in the previous layer divided by a constant factor $f = 5$. The only difference lies in the connectivity of the networks. In locally-connected networks, every unit in a hidden layers is connected to $k = 5$ nearest units in the previous layer.

Once the network architectures are defined, we train both networks using the same batch of dataset, with batch size set to be 64, and compare their respective memory consumptions on the GPU. Furthermore, we vary the input dimensions to investigate the correlation between memory consumption and input dimensions. The experimental results are shown in Fig. D.2 and Fig. D.3.

Fig. D.2 shows that the memory usage of training fully-connected networks increases rapidly as the input dimension increases. When the input dimension is 4.0×10^4, the memory consumption reaches 8,031 megabytes. In comparison, when the input dimension is 4.0×10^4, training the locally-connected network only consume 543 megabytes. When the input dimension is 1.28×10^7, the memory consumption is 8,075 megabytes, which is comparable to the memory usage of fully-connected network at an input dimension of only 4.0×10^4.

This plot therefore clearly shows that locally-connected neural networks are more memory-efficient than fully-connected networks. Fig. D.3 shows the rate of increase in the number of trainable parameters (i.e., weights and biases) of a fully-connected network is much faster than that of a locally-connected network, which accounts for the differences in the memory consumptions of the networks.

In addition to memory efficiency, we also run experiments to examine the training performance of the locally-connected network and its capability to make accurate predictions.

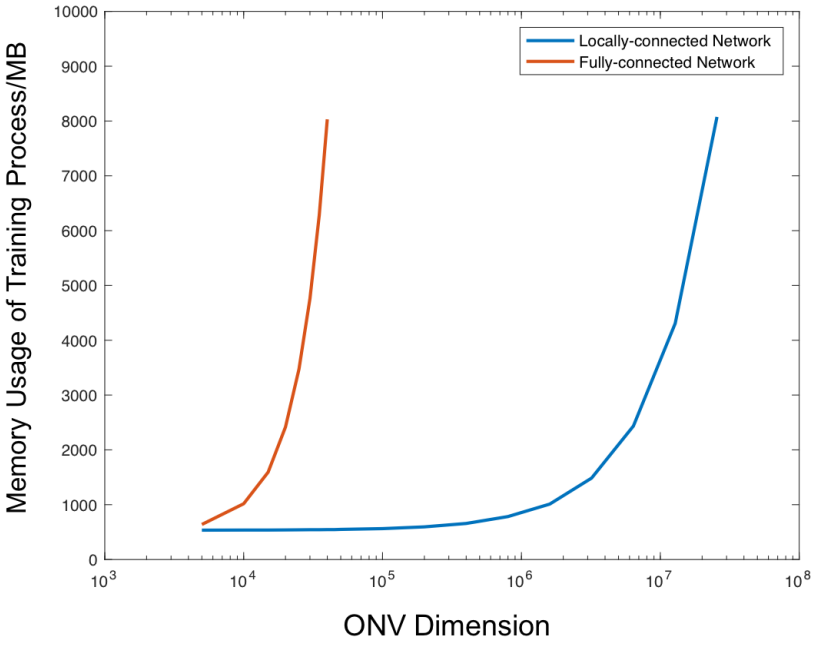


Figure D.2: Semi-log plot of memory consumption versus the dimension of the optic nerve vector (ONV).

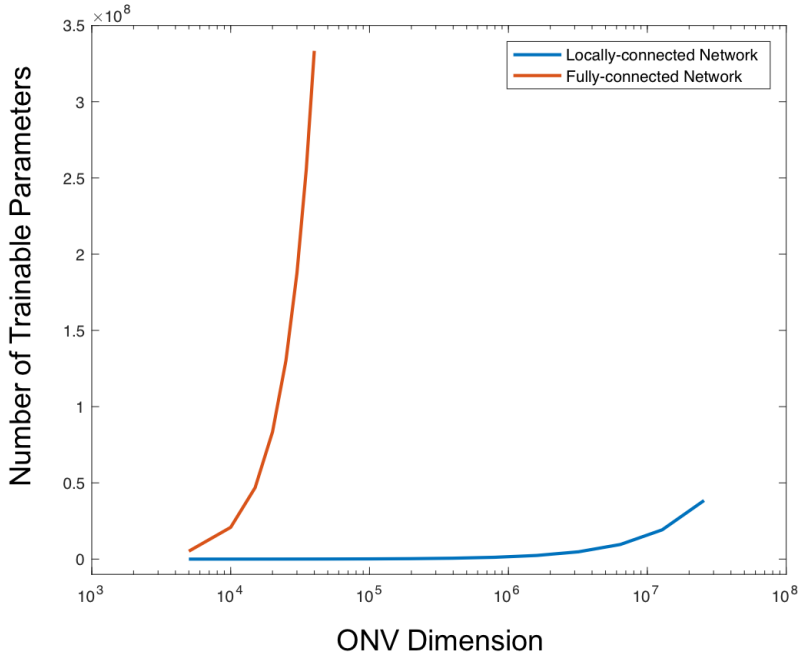


Figure D.3: Semi-log plot of number of trainable parameters versus the dimension of the optic nerve vector (ONV).

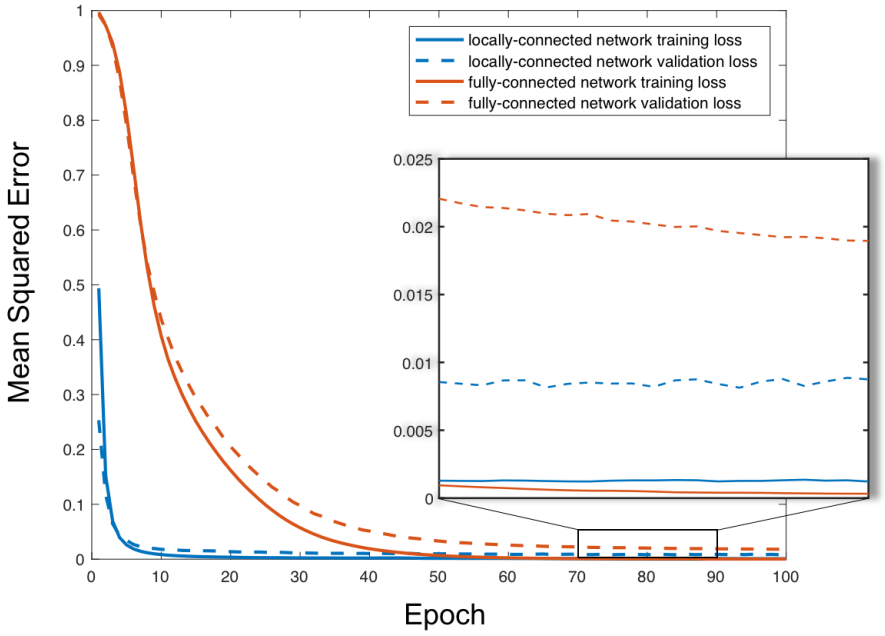


Figure D.4: Progress of the backpropagation training process for the eye sensory DNN on the training (solid line) and validation (dotted line) datasets. The red plot corresponds to the fully-connected neural network while the blue plot corresponds to the liNet.

We synthesized a training dataset of 20K input-output pairs. Each instance is an ONV with 14,400 photoreceptors. We trained both fully-connected and locally-connected networks independently using the same dataset. The training and validation losses of the training processes are shown in Fig. D.4. The fully-connected network converges to a small mean-squared training loss below 0.01 after 47 epochs, whereas the locally-connected network converges more quickly to a loss below 0.01 after 9 epochs. Furthermore, the validation loss of the locally-connected network is smaller than that of the fully-connected network, which demonstrates that our locally-connected network is capable of making more accurate predictions despite its lower memory and time cost.

REFERENCES

Bajcsy, R. (1988). Active perception. *Proceedings of the IEEE*, 76(8):996–1005. 7

Ballard, D. (1991). Animate vision. *Artificial Intelligence*, 48:57–86. 7

Buchberger, M. (2004). *Biomechanical modelling of the human eye*. PhD thesis, Johannes Kepler University, Linz, Austria. 6

Deering, M. F. (2005). A photon accurate model of the human eye. *ACM Transactions on Graphics (TOG)*, 24(3):649–658. 10, 21

Grady, L. J. (2004). *Space-variant computer vision: A graph-theoretic approach*. PhD thesis, Boston University. 7

Greivenkamp, J. E., Schwiegerling, J., Miller, J. M., and Mellinger, M. D. (1995). Visual acuity modeling using optical raytracing of schematic eyes. *American Journal of Ophthalmology*, 120(2):227–240. 8

Haslwanter, T. (1995). Mathematics of three-dimensional eye rotations. *Vision Research*, 35(12):1727–1739. 6

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proc. IEEE International Conference on Computer Vision*, pages 1026–1034. 19

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. 19

Koenderink, J. and Van Doorn, A. (1978). Visual detection of spatial contrast; influence of location in the visual field, target extent and illuminance level. *Biological Cybernetics*, 30(3):157–167. 7

Komogortsev, O., Holland, C., Jayarathna, S., and Karpov, A. (2013). 2D linear oculomotor plant mathematical model: Verification and biometric applications. *ACM Transactions on Applied Perception (TAP)*, 10(4):27. 6

Lee, S.-H., Sifakis, E., and Terzopoulos, D. (2009). Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics (TOG)*, 28(4):99. 14, 40, 43

Lee, S.-H. and Terzopoulos, D. (2006). Heads up! Biomechanical modeling and neuromuscular control of the neck. *ACM Transactions on Graphics (TOG)*, 25(3):1188–1198. 7, 32

Lee, S. P., Badler, J. B., and Badler, N. I. (2002). Eyes alive. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, pages 637–644, New York, NY, USA. ACM. 6

Looser, C. E. and Wheatley, T. (2010). The tipping point of animacy: How, when, and where we perceive life in a face. *Psychological Science*, 21(12):1854–1862. 6

Nakada, M., Chen, H., Lakshmipathy, A., and Terzopoulos, D. (2018a). Linets: Locally-connected irregular deep neural networks for biomimetic vision. Submitted to the 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019). 8, 21, 46

Nakada, M., Chen, H., and Terzopoulos, D. (2018b). Deep learning of biomimetic visual perception for virtual humans. In *ACM Symposium on Applied Perception (SAP 18)*, pages 20:1–8, Vancouver, Canada. 8

Nakada, M., Chen, H., and Terzopoulos, D. (2018c). Learning biomimetic perception for human sensorimotor control. In *Proc. Second IEEE CVPR Workshop on Mutual Benefits of Cognitive and Computer Vision (MBCC 2018)*, pages 2030–2035, Salt Lake City, UT. 8

Nakada, M., Zhou, T., Chen, H., Weiss, T., and Terzopoulos, D. (2018d). Deep learning of biomimetic sensorimotor control for biomechanical human animation. *ACM Transactions on Graphics*, 37(4). Proc. *ACM SIGGRAPH 2018*, Vancouver, Canada, August 2018. 4, 5, 7, 8, 9, 18, 32, 36

Neog, D. R., Cardoso, J. L., Ranjan, A., and Pai, D. K. (2016). Interactive gaze driven animation of the eye region. In *Proceedings of the 21st International Conference on Web3D Technology*, pages 51–59. ACM. 6

Rabie, T. F. and Terzopoulos, D. (2000). Active perception in virtual humans. In *Proc. Vision Interface 2000*, pages 16–22, Montreal, Canada. 7

Ruhland, K., Andrist, S., Badler, J., Peters, C., Badler, N., Gleicher, M., Mutlu, B., and Mcdonnell, R. (2014). Look me in the eyes: A survey of eye and gaze animation for virtual agents and artificial systems. In *Eurographics State-of-the-Art Report*, pages 69–91. 6

Schraa-Tam, C. K., Van Der Lugt, A., Frens, M. A., Smits, M., Van Broekhoven, P., and Van Der Geest, J. N. (2008). An fMRI study on smooth pursuit and fixation suppression of the optokinetic reflex using similar visual stimulation. *Experimental Brain Research*, 185(4):535–544. 34

Schwartz, E. (1977). Spatial mapping in the primate sensory projection: Analytic structure and relevance to perception. *Biological Cybernetics*, 25(4):181–194. 7

Sprague, N., Ballard, D., and Robinson, A. (2007). Modeling embodied visual behaviors. *ACM Transactions on Applied Perception (TAP)*, 4(2):11. 7

Terzopoulos, D. and Rabie, T. F. (1995). Animat vision: Active vision with artificial animals. In *Proc. Fifth International Conf. on Computer Vision (ICCV'95)*, pages 840–845, Cambridge, MA. 7

Thomas, J. (1969). The dynamics of small saccadic eye movements. *The Journal of Physiology*, 200(1):109–127. 30

Trutoiu, L. C., Carter, E. J., Matthews, I., and Hodgins, J. K. (2011). Modeling and animating eye blinks. *ACM Transactions on Applied Perception (TAP)*, 8(3):17. 6

Tweed, D., Cadera, W., and Vilis, T. (1990). Computing three-dimensional eye position quaternions and eye velocity from search coil signals. *Vision Research*, 30(1):97–110. 6

Wei, Q., Patkar, S., and Pai, D. K. (2014). Fast ray-tracing of human eye optics on graphics processing units. *Computer Methods and Programs in Biomedicine*, 114(3):302–314. 8

Wei, Q., Sueda, S., and Pai, D. K. (2010). Biomechanical simulation of human eye movement. In *International Symposium on Biomedical Simulation*, pages 108–118. Springer. 6, 39

Wilson, S. W. (1983). On the retino-cortical mapping. *International Journal of Man-Machine Studies*, 18(4):361–389. 7

Yeo, S. H., Lesmana, M., Neog, D. R., and Pai, D. K. (2012). Eyecatch: Simulating visuomotor coordination for object interception. *ACM Transactions on Graphics (TOG)*, 31(4):42. 7, 32