Locally-Connected, Irregular Deep Neural Networks for Biomimetic Active Vision in a Simulated Human

Masaki Nakada*, Honglin Chen*, Arjun Lakshmipathy, and Demetri Terzopoulos
Computer Science Department, University of California, Los Angeles, California, USA

***Abstract*—An advanced simulation framework has recently been introduced for exploring human perception and visuomotor control. In this context, we investigate locally-connected, irregular deep neural networks (liNets) for biomimetic active vision. Like commonly used CNNs, liNets are locally-connected, forming receptive fields, but unlike CNNs, they are suitable for spatially irregular photoreceptor distributions inspired by those found in foveated biological retinas. Compared to fully-connected deep neural networks, liNets accommodate a much greater number of retinal photoreceptors to enhance visual acuity without intractable memory consumption. LiNets serve well in the biomimetic active vision system embodied in a simulated human that learns active visuomotor control and active appearance-based recognition.**

I. Introduction

Visuomotor functionality in biological organisms refers to the process of continually acquiring and interpreting visual sensory information necessary to produce appropriate motor responses that achieve desired goals. In humans, the vision process begins with binocular optical sensing mediated by nonuniform retinal photoreceptor topology [1] feeding neural mechanisms that control eye movements and achieve higher-level visuomotor control. Biological vision has inspired computational approaches that mimic the functionality of these neural mechanisms. Recent breakthroughs in machine learning with artificial neural networks have proven effective in computer vision; however, the application of Deep Neural Networks (DNNs) to embodied biomimetic visuomotor systems has received little attention in the field.

We have recently introduced a unique human simulation framework well suited to exploring bio-inspired vision and visuomotor control [2] [3]. Our framework is unique in that it features a biomechanically simulated, human musculoskeletal model, which includes numerous skeletal muscle actuators. The virtual human perceives its environment with eyes, capable of eye movements, whose foveated retinas contain photoreceptors arranged in a nonuniform distribution like that of biological retinas. Its prototype visuomotor system includes two dozen automatically-trained, fully-connected DNNs, about half of which comprise its active vision subsystem while the other half comprise its neuromuscular motor control subsystem. In this context, our contributions in the present paper are as follows:

- In conjunction with a more realistic eye model, including cornea, iris, pupil, lens, and foveated retina (Section III), we propose *locally-connected irregular DNNs (liNets)* for active vision (Section IV). Unlike the common Convolutional Neural Networks (CNNs) that are limited to conventional images and form rectangular receptive fields on regular pixel arrays, our networks are suitable to spatially irregular photoreceptor distributions like those found in foveated biological retinas.
- Relative to the fully-connected DNNs that we used in [2], which proved manageable for retinas with a few thousand cone-like photoreceptors, our novel liNets accommodate a dramatically greater number of photoreceptors. This substantially enhances visual acuity. We investigate the memory efficiency and learning performance of liNets (Section V) applied to active, online visuomotor control as well as in a novel 3D active face recognition application (Section VI).

The significance of our work includes how the retinal sampling is achieved on the hemispherical fundus of an optically-accurate model of the eye, as well as our deep neural network solution to irregular visual processing, integrated within a comprehensive virtual human model, which can hence emulate human active vision with unprecedented realism, detail, and efficiency.

II. The Virtual Human Model

Our human model [3] includes a large number of the relevant articular bones and muscles—193 bones connected by joints comprising 163 articular degrees of freedom, plus a total of 823 muscle actuators. Each skeletal muscle is modeled as a Hill-type uniaxial contractile actuator that applies forces to the bones at its points of insertion and attachment. The human model is numerically simulated as a force-driven articulated multi-body system. Each muscle actuator is activated by an independent, time-varying, efferent activation signal, which is provided by its motor control subsystem.

Fig. 1 illustrates the architecture of our virtual human's visuomotor control system, including its improved, biomimetic eye models [4]. The system incorporates a set of 24 automatically-trained DNNs. In the motor subsystem, 12 of these DNNs comprise 6 recurrent neuromuscular motor controllers that actuate the muscle groups of the cervicocephalic complex (219 muscles), torso (443 muscles), and four limbs (29 muscles per leg, 39 muscles per arm), and 2 oculomotor DNNs control the extraocular muscles (6 muscles per eye). The remaining 10 DNNs implement the sensory subsystem, which is devoted to visual perception and continuously operates on the retinal photoreceptor outputs. A pair of foveation DNNs drive a synergy of eye, head, and torso movements, while 4 pairs of vision DNNs extract the sensory information necessary to control the arms and legs.

*Co-primary author

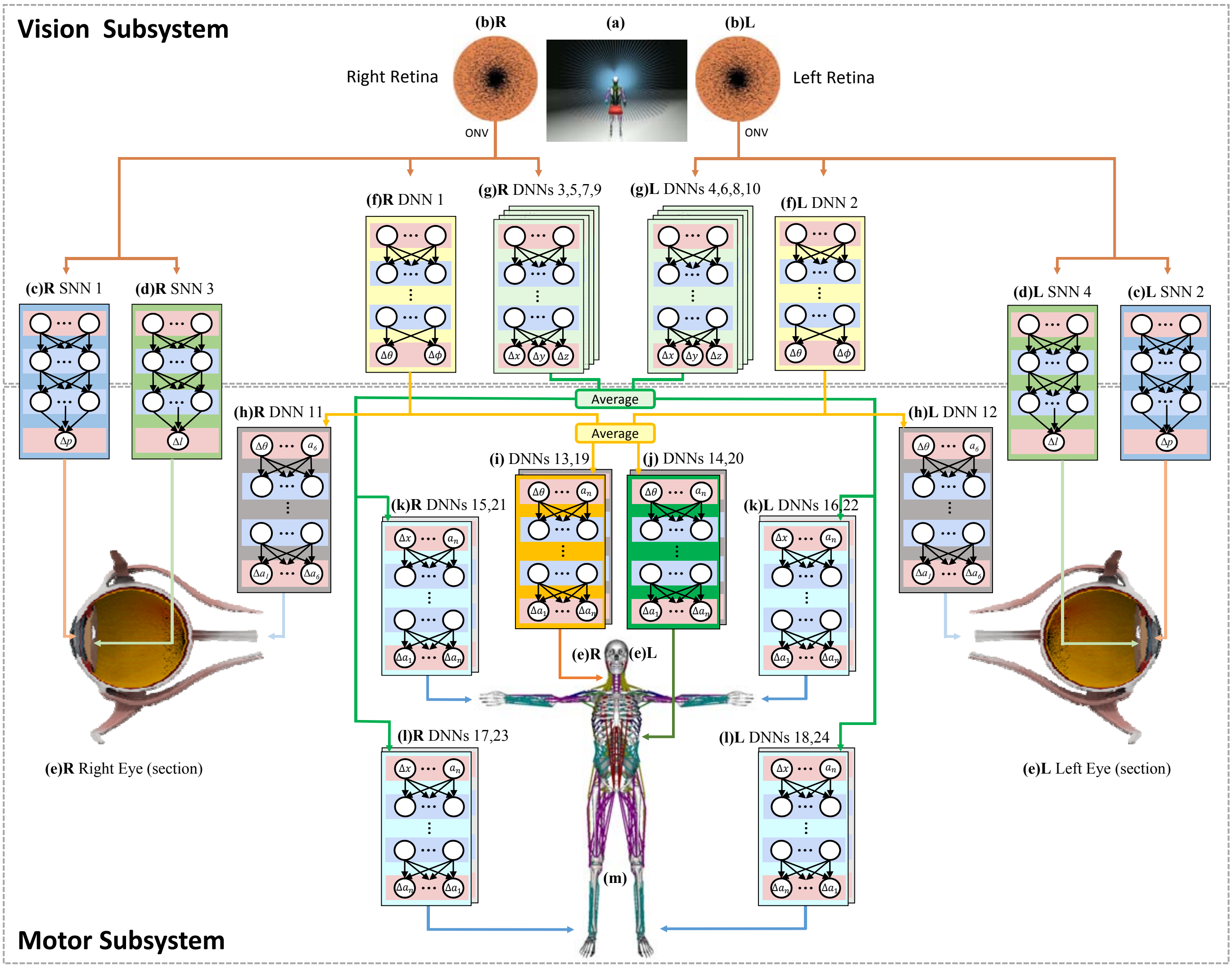
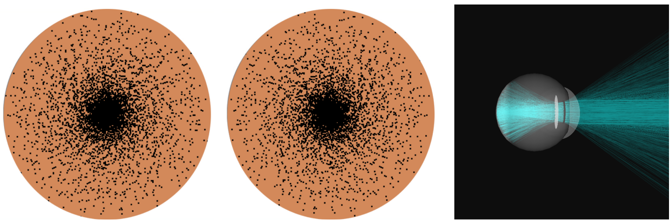


Fig. 1: Architecture of the visuomotor system. The neural controllers in the system include a total of 24 DNNs and 4 SNNs.

Vision Subsystem (top): To compute irradiance via ray-tracing, each retinal photoreceptor casts rays through the eye (Fig. 2a,b,c) and into the virtual world (a). (b) The arrangement of the photoreceptors (black dots) on the right R and left L foveated retinas. Each eye outputs an RGB Optic Nerve Vector (ONV). This feeds four trained visual accommodation SNNs (1–4); SNNs (c)R (1) and (d)R (3) control the muscles of the iris and lens of the right eye (e)R, and SNNs (c)L (2) and (d)L (4) do the same for the left eye (e)L. The ONV also feeds ten trained vision DNNs (1–10). (f) A pair of foveation DNNs (1,2) produce outputs that drive the movements of the eyes to foveate visual targets. (g) The eight limb vision DNNs (3–10) — (g)R (3,5,7,9) for the right eye and (g)R (4,6,8,10) for the left eye — output observed limb-to-target discrepancy estimates.

Motor Subsystem (bottom): Fourteen trained neuromuscular motor DNNs (11–24) comprise the motor subsystem, including eight voluntary motor DNNs (11–18) and six reflex motor DNNs (19–24). (h) The oculomotor DNNs (11,12), which are driven by the outputs of the foveation DNNs, output muscle activation signals that control the six extraocular muscles of each eye to produce eye movements. Driven by the averaged responses of the foveation DNNs, along with the current activations of the 216 neck muscles and 443 torso muscles, respectively, the cervicocephalic (i) voluntary motor DNN (13) and torso (j) voluntary motor DNN (14) each outputs muscle activation signals that contribute to actuating its associated neuromuscular complex. Driven by the bilaterally pairwise averaged responses of the limb vision DNNs, along with the current activations of the 29 muscles of each arm or 39 muscles of each leg, respectively, each of the four limb voluntary motor DNNs (k) (l) (15–18) outputs muscle activation signals that contribute to actuating its associated neuromuscular complex. Each of the six reflex motor DNNs (19–24) outputs muscle activation signals that contribute by stabilizing the muscle group of its associated musculoskeletal complex.



(a) Left retina (b) Right retina (c)

Fig. 2: (a),(b) Foveated retinas with 14,400 photoreceptors (black dots) in noisy log-polar distributions. (c) Rays cast from the positions of photoreceptors on the retina through the lens and pupil out into the 3D scene by the ray-tracing procedure that computes the the irradiance on photoreceptors.

III. The Biomimetic Eye Model

The virtual human's eyes are modeled in accordance with human physiological data [5].[1] We model the virtual eye as a sphere of radius 12mm that can be rotated with respect to its center around its vertical y axis by an angle θ and around its horizontal x axis by an angle ϕ. The eyes are in their neutral positions looking straight ahead when $\theta = \phi = 0°$. The horizontal and vertical fields of view are 167.5°. To achieve dynamic eye control, the six extraocular and the intraocular (ciliary and sphincter) muscles are modeled as compound Hill-type contractile actuators [4].

A. Cornea, Iris, and Lens Submodels

The eye model includes a cornea, iris/pupil, lens, and retina. The cornea refracts incoming light rays. The adjustable iris/pupil accommodates to the brightness of the scene by controlling the quantity of light that enters the eye. Actuated by a ciliary muscle, the deformable lens further refracts incoming light rays so as to focus them on the hemispherical retinal surface at the back of the eyeball. Two shallow neural networks (SNNs) control the iris and lens accommodation [4].

B. Retina

Unlike the uniform-resolution, Cartesian grid structure of most artificial imaging sensors, visual sampling in primate retinas is well known to be strongly nonuniform. The density of cones decreases radially with eccentricity, from the foveal center toward the periphery. A log-polar photoreceptor distribution is often used as a model [6].

To emulate foveated perception, we use a noisy log-polar distribution to model the nonuniform placement of P photoreceptors on the hemispherical retina. Fig. 2a,b illustrates the placement of $P =$ 14,400 photoreceptors. Due to the additive Gaussian noise, the photoreceptors are placed in different positions on each retina.

C. Ray-Tracing-Based Irradiance Computation

The irradiance at any point on the hemispherical retinal surface at the back of the eye is computed using the conventional ray-tracing technique of computer graphics rendering [7]. Sample rays from the positions of photoreceptors on the hemispherical retinal surface are cast through the eye and out into the 3D virtual world where they recursively intersect with the visible surfaces of virtual objects 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. For our realistic eye model (Fig. 2c), each photoreceptor gathers light from the 3D environment through a finite aperture proportional to the area of the pupil. The irradiance computation by each photoreceptor requires the weighted sum of multiple cast rays refracted at the surfaces of the lens and cornea.

D. Optic Nerve Vector (ONV)

The foveated retinal RGB "image" captured by the P photoreceptors of each eye is output for further processing down the visual pathway, not as a 2D array of pixels, but as a 1D Optic Nerve Vector (ONV) of length $3P$. The raw irradiance information encoded in this vector feeds the low-level perceptual neural networks that directly control pupil size, focal accommodation and eye movements, intermediate-level networks that control cervicocephalic, torso, and limb motions, and higher-level neural networks that enable face recognition.

IV. Locally-Connected Irregular DNNs

The human retina has approximately 5 million cones [8]. Enhancing the visual acuity of the virtual retinas requires the incorporation of many more photoreceptors than the 3,600 that we employed in [2]. For instance, the ONVs are $3 \times$ 14,400 $=$ 43,200-dimensional for the retinas shown in Fig. 2. Implementing and training a *fully-connected* DNN, as we did in [2], with this many inputs is infeasible due to the exorbitant memory needed to accommodate the network's weights. To overcome this impediment, we introduce locally-connected irregular DNNs, or liNets, that effectively mitigate memory consumption, thereby enabling the use of larger numbers of irregularly distributed photoreceptors.

A. LiNet Architecture

In a fully-connected neural network, each neuron in a hidden layer is connected *globally* to all the neurons in the previous layer, whereas like CNNs [9], each neuronal unit in a hidden layer of our liNet is connected only *locally* to a fixed number of neighboring units in the previous layer. However, the implementation of a CNN conforms to conventional, regularly-sampled images, the regular arrangement of units in tensor data structures, and the sharing of weights and biases across all the units in each hidden layer (i.e., the convolutional property). The liNet shares none of these restrictions. Importantly, every unit of a liNet has its own particular weights and bias, just like in a fully-connected network.

[1] 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 g. 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 field of view becomes about 135 degrees vertically and 200 degrees horizontally.

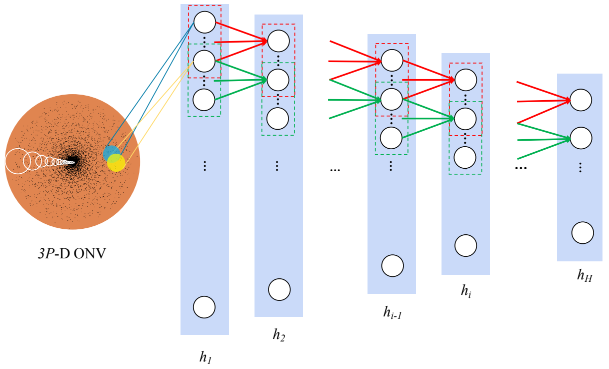


Fig. 3: The liNet architecture.

Referring to Fig. 3, suppose that we have a liNet with H hidden layers, each of which is comprised of N_h neuronal units. Using a Euclidean-distance k-nearest-neighbor algorithm, each unit in hidden layer $h = 1, \ldots, H$ forms a receptive field with R_h nearest neighbor units in the previous hidden layer $h-1$, and it inherits a position in the 2D (ρ, θ polar coordinate system) retinal domain that is the average of the positions of the units in its receptive field. The receptive fields of the units in the first hidden layer $h = 1$, are formed from R_1 neighboring retinal photoreceptors. 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. 3, naturally increase in size with eccentricity, from the denser foveal center outward to the sparser periphery.

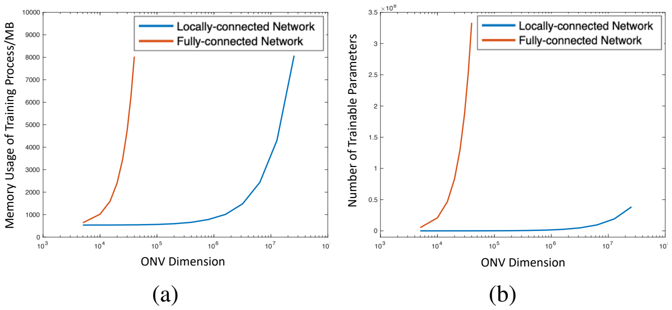
Our flexible liNet architecture is defined by a set of H connectivity matrices. Each $N_h \times R_h$ matrix $\boldsymbol{C}_h$ specifies the connectivity between units in hidden layer h and R_h units in hidden layer $h-1$. Each row in a connectivity matrix stores the indexes of the units in its receptive field. The index of a unit in hidden layer h is $0 \leq n < N_h$, while the (ONV) index of a retinal photoreceptor is $0 \leq p < P$.

Our liNets are implemented in PyTorch. We use Rectified Linear Units (ReLUs). Each ReLU computes a weighted sum of the outputs of units (or photoreceptors in the case of units in hidden layer $h = 1$) 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.

V. LiNet Efficiency and Performance

To assess the memory efficiency of the liNet, given P photoreceptors ($3P$-dimensional ONV), we compare its memory consumption to that of a fully-connected DNN with the same numbers of hidden layers and neurons in each hidden layer.

For example, consider setting the number of hidden layers to $H = 4$, and the number of units N_h in hidden layer h to the number N_{h-1} in the previous layer divided by a constant factor $f = 5$; i.e., $N_h = \lfloor N_{h-1}/f \rfloor$, such that the only difference is in the connectivity of the networks. In the liNet, every neuron in a hidden layer is locally-connected to only $R_h = 5$ nearest neighboring units in the previous layer. Fig. 4 plots the memory consumption and number of trainable parameters of the networks with increasing ONV dimensionality. Fig. 4a shows that the memory consumption of fully-connected networks increases rapidly as the input dimension increases—8,031 MB when $3P = 4.0 \times 10^4$, whereas for the liNet it is only 543 MB. For a $3P = 1.28 \times 10^7$ dimensional ONV, the liNet's memory consumption is 8,075 MB, comparable to that of the fully-connected network with a 4.0×10^4 dimensional ONV. Fig. 4b shows that the rate of increase in the number of trainable parameters (i.e., the weights and biases) of the fully-connected network is much faster than that of the liNet.



(a) (b)

Fig. 4: Semi-log plots of (a) memory consumption and (b) number of trainable parameters versus ONV dimension.

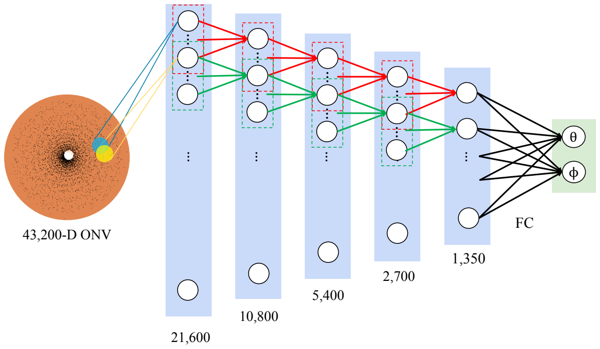
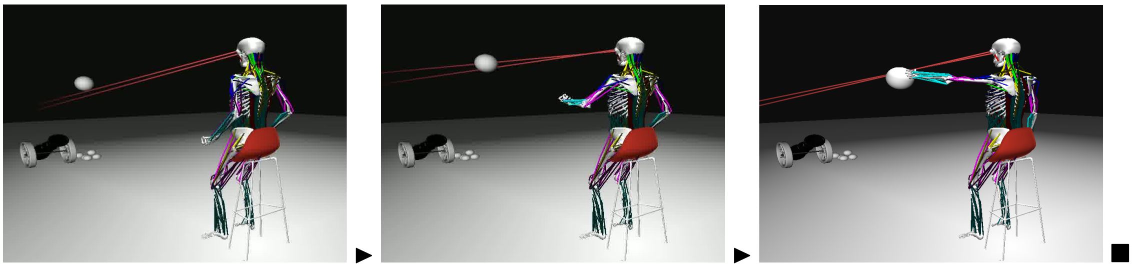


Fig. 5: Foveation DNN architecture. A liNet backbone is followed by a fully-connected layer that outputs eye orientation control signals θ and ϕ.

The above observations indicate that, unlike fully-connected networks, liNets make it feasible to perform perceptual processing on retinal models whose photoreceptor counts are on the order of the millions of cone photoreceptors in the human retina [1].

Next, to evaluate the learning effectiveness of the liNet, we consider the task of executing eye movements to foveate objects of interest. For this purpose, we employ retinas with $P =$ 14,400 photoreceptors (43,200-dimensional ONVs). Foveation functionality is accomplished by the foveation DNNs (denoted DNN 1 and DNN 2 in Fig. 1), each of which inputs the ONV from the eye and outputs activation signals to its 6 extraocular muscles to produce eye movements.

We construct the foveation DNNs as illustrated in Fig. 5, consisting of a liNet backbone with $H = 4$ hidden layers, a receptive field size of $R_h = 20$, and $f = 5$, followed by a



(a) (b) (c)

Fig. 8: Frames from a simulation of the biomechanically-simulated virtual human sitting on a stool (with immobilized pelvis and torso) and actively executing left-arm reaching motions to intercept a ball shot by the cannon. The ball is actively perceived by the eyes (red lines indicate the gaze directions), processed by the liNet vision DNNs, foveated and tracked through eye movements in conjunction with muscle-actuated head movements controlled by the neck-head motor DNN, and the reactive reaching motion is muscle-actuated and controlled by the left arm motor DNN.

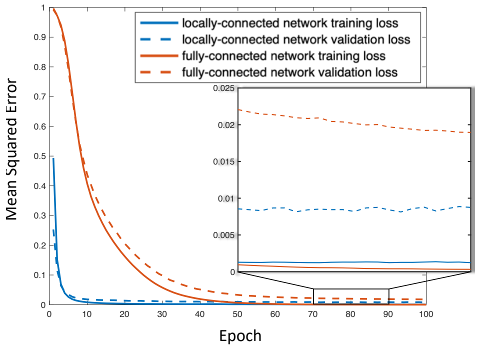
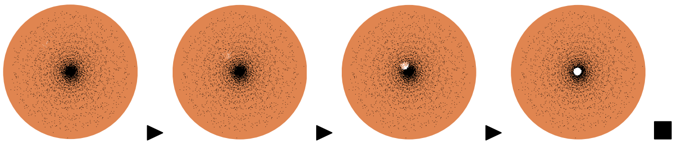


Fig. 6: Progress of the foveation DNN training process for the vision liNet (blue) and fully-connected foveation DNN (red) on the training (solid) and validation (dashed) datasets.



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

Fig. 7: 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 periphery, at t_1 the eye movement is bringing the ball towards the fovea, and the moving ball is being fixated in the fovea at times t_2 and t_3.

fully-connected (FC) output layer that produces eye rotation angles θ and ϕ.

Using a dataset of 20K training ONVs and a batch size of 64, we trained a benchmark fully-connected foveation DNN as we did in [2], as well as a liNet-based foveation DNN, to foveate a white ball by stimulating the extraocular muscles to actuate saccadic eye movements. Fig. 6 plots the progress and validation losses of the backpropagation training processes. The fully connected network converges to a small mean-squared training loss below 0.01 after 47 epochs. The liNet does so after only 9 epochs with a validation loss smaller than that of the fully-connected network.

With their trained foveation DNNs, the eyes perform natural vergence movements to converge onto visual targets. This is illustrated in Fig. 7 for a white ball in motion that enters the eye's field of view from the lower right, stimulating several peripheral photoreceptors at the upper left of the retina. The eye very rapidly foveates the visual target. Fine adjustments comparable to microsaccades can be observed during fixation.

The above results confirm that our liNets are capable of learning effective foveation with substantially smaller memory and time requirements than the fully-connected networks we employed in [2].

VI. Applications

A. Visuomotor Control

Fig. 8 presents a sequence of frames from a simulation demonstrating the visuomotor system using the trained liNet foveation DNNs and other vision DNNs (DNNs 3–10 in Fig. 1).

A cannon shoots a ball towards the virtual human, which actively perceives the ball on its foveated retinas. The ONV outputs of its eyes are continually processed by the liNet vision DNNs to perform foveation and visual tracking of the incoming ball. The motor DNNs control the extension of the arms and legs to intercept and deflect the approaching ball. Thus, given just the high level objective of deflecting the incoming ball, the virtual human successfully controls itself online to perform this nontrivial dynamic visuomotor task.

While similar to our demonstration in [2], which employed fully-connected vision DNNs to process retinas with a mere 3,600 photoreceptors, our novel liNets have enabled us to implement retinas with 14,400 photoreceptors yet effectively train the vision DNNs to perform foveation and tracking of the visual target with eye movements and cervicocephalic control.

B. Active Face Recognition

As an example of higher-level visual processing with liNets, we explore the task of 3D Active Face Recognition (AFR). Fig. 9 shows the architecture of our AFR DNN. The network includes a liNet backbone that has $H = 5$ hidden layers of

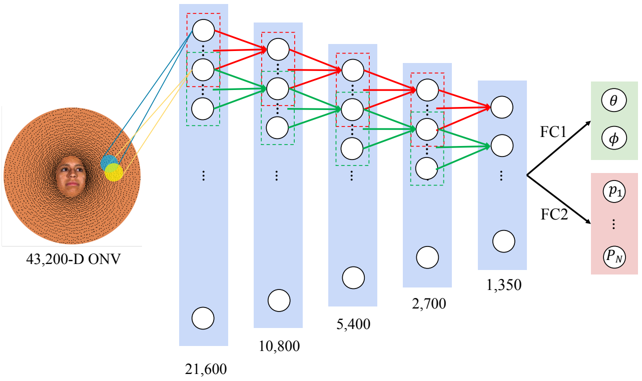


Fig. 9: LiNet architecture for active face recognition. The network outputs recognized person identity labels p_i and horizontal and vertical pose angle estimates α and β.

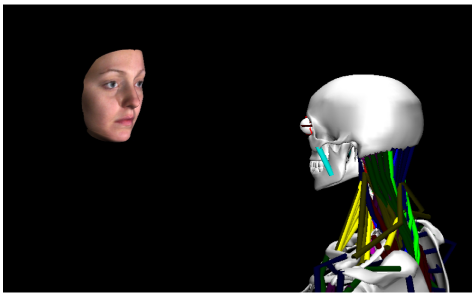


Fig. 10: The virtual human observes a 3D face model.

ReLUs. Every hidden unit forms receptive fields with $R = 20$ nearest neighbors in the previous layer. The retina has $P =$ 14,400 photoreceptors (a $3P = 43{,}000$ dimensional ONV), and the number of units in each successive hidden layer is reduced by a factor of $f = 2$. The liNet feeds two single-layer, fully-connected subnets—FC1 is a classification subnet, whose output p_i, for $i = 1, \ldots, N$, has dimensionality equal to the number N of known person identities; FC2 is a regression net that generates the estimated pose (horizontal and vertical rotation angles, α and β) of the observed face.

The AFR DNN is trained by minimizing the combined loss $\mathcal{L} = \mathcal{L}_{\text{class}} + \lambda \mathcal{L}_{\text{angle}}$, where $\mathcal{L}_{\text{class}}$ is cross-entropy person classification loss, $\mathcal{L}_{\text{angle}}$ is mean-squared pose estimation regression loss, and parameter λ trades off between them.

1) Training Data Synthesis: To train our AFR model, we use the Freiburg 3D face dataset, which comprises scans of 75 subjects, recorded using a Cyberware$^{\text{TM}}$ 3030PS laser scanner as part of the University of Freiburg 3D morphable faces database [10]. The virtual human (monocularly) observes the 3D faces of 50 different scanned subjects (Fig. 10). Fig. 11 illustrates how we synthesize the training data. Ray-tracing the textured 3D geometry, the observer's eye captures ONV "images" as the subject face is rotated by 2 degree increments from $-30°$ to $30°$ horizontally and vertically. Thus, we synthesize $50 \times 30 \times 30 = 45{,}000$ ONVs with corresponding face orientation and person ID labels. We partition the ONV dataset as follows: 70% training, 10% validation, 20% testing.

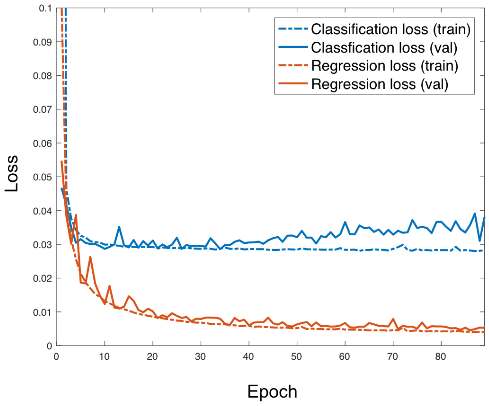


Fig. 12: Training and validation loss curves for 3D active face recognition. Classification loss (blue) is the cross-entropy loss $\mathcal{L}_{\text{class}}$ for the classification subnet. Regression loss (red) is the mean-squared loss $\mathcal{L}_{\text{angle}}$ for the regression subnet.

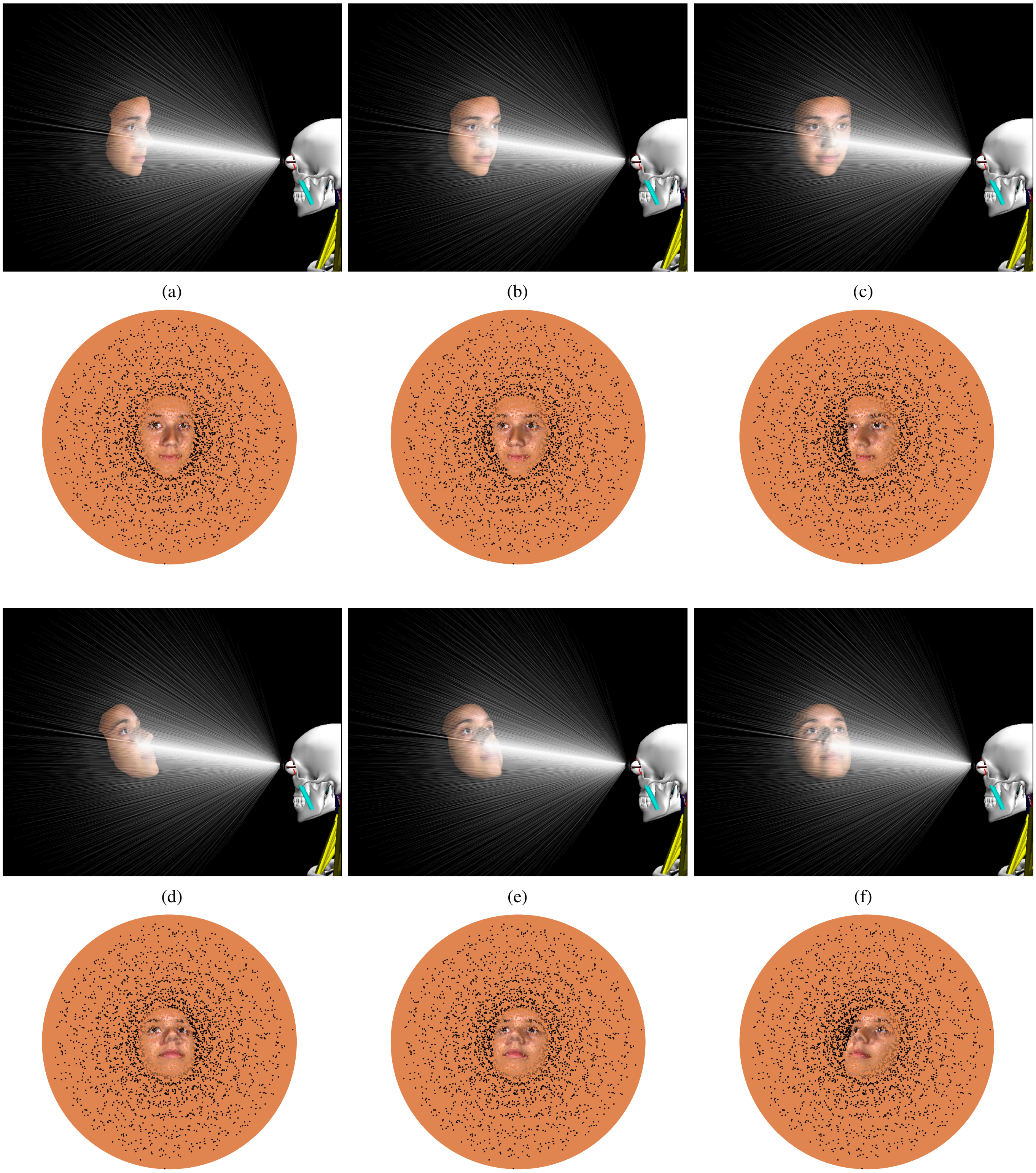
2) Training and Validation: To train our network, we use the Adaptive Moment Estimation (Adam) stochastic optimizer with a learning rate $\eta = 0.01$, step size $\alpha = 10^{-3}$, and forgetting factors $\beta_1 = 0.9$ for gradients and $\beta_2 = 0.999$ for second moments of gradients. Overfitting is avoided by using a validation split of 0.1 along with an early stopping condition. We standardized the α and β outputs of the regression network by computing the means and standard deviations over the entire training dataset.

The hyper-parameter in the aforementioned loss function $\mathcal{L}$ is set to $\lambda = 6.0$. The model parameters are recorded only when the model simultaneously attains best regression loss and classification accuracy on the validation set after each epoch. Fig. 12 shows the training and validation loss curves. The training process converged to a small error after 57 epochs, which triggered the early stopping condition (no improvement for 30 successive epochs).

3) Accuracy: The testing classification accuracy for the model was 97.97%, and the testing mean squared errors for the α and β angles were 5.52×10^{-3}. This corresponds to an average error of 0.7° (after adjusting for the effect of standardizing the output angles).[2]

Note that there exist no other datasets against which to benchmark the performance of the 3D active face recognition ability of our virtual human. Futhermore, the biomimetic retina+liNet combination in its active vision system is not meaningfully comparable against conventional CNNs trained on regular, passively-acquired, uniform-resolution, facial images.

[2] Let a_t denote the target output and a_p the prediction of the network, e_{mse} denote the mean squared error of the angle, e_{deg} denote the corresponding error in degrees, and let μ and β denote the mean and standard deviation of the target angles in the dataset. We have $e_{\text{mse}} = \left[\left(\frac{a_t - \mu}{\sigma}\right) - \left(\frac{a_p - \mu}{\sigma}\right) \right]^2$ and we obtain $e_{\text{deg}} =\mid a_t - a_p \mid= \beta \sqrt{e_{\text{mse}}}$. The standard deviation β is 8.96.



(a) (b) (c)

(d) (e) (f)

Fig. 11: The virtual human foveating a 3D face in several different poses. Irradiance at the retinal photoreceptors is computed by ray-tracing; the white lines indicate rays cast from the positions of the photoreceptors, through the lens, pupil, and cornea, and out into the 3D scene. Underneath, the photoreceptor responses comprising the ONV are visualized as retinal “images”.

VII. Related Work

A number of researchers have proposed (locally-connected) neural network architectures whose neuronal units have receptive fields; the best known are Fukushima's "Neocognitron" [11] and the currently popular deep CNNs [9]. CNNs are characterized by the regular structure of their receptive fields, enabling the sharing of weights across all units in a hidden layer (i.e., the convolutional property). Closer to our liNets is the non-convolutional LCNP [12]. However, unlike the liNet, the aforecited networks are intended for use in conventional, regularly-sampled pixel images common to computer vision, which fundamentally differ from biological retinas.

Our biomimetic, foveated retina model employs a noisy log-polar photoreceptor distribution. Other placement patterns are readily created, including more elaborate biomimetic procedural models [8] or photoreceptor distributions empirically measured from biological eyes [1], [13], all of which depart fundamentally from the uniform-resolution, Cartesian images common to computer vision/graphics. However, foveated sensors have also been of interest in computer vision and image processing [14]. Recently, Ozimek et al. [15] explored foveated sampling and log-polar maps [6] to compute regular Cartesian cortical representations of images to which deep CNNs may be applied with significant efficiency gains.

In the context of human active vision and visuomotor control [16], Terzopoulos and Rabie deployed their "animat vision" framework within a kinematic virtual human capable of bipedal locomotion, demonstrating active, vision-guided tracking and pursuit [17], and a similar kinematic virtual human model, dubbed "Walter", was employed by Sprague et al. [18] to study visuomotor control in the context of sidewalk navigation tasks. The virtual humans demonstrated in [17] were equipped with crudely-foveated eyes, implemented as coaxial virtual cameras rendering composite polygon-shaded square images through the GPU pipeline, quite unlike our biomimetic ocular model that samples light in the 3D virtual environment using ray-tracing so as to realistically emulate how irregularly distributed photoreceptors respond to light irradiating the retina.

VIII. Conclusions

Within a unique simulation framework for investigating biomimetic human vision and visuomotor control, we have introduced and evaluated locally-connected, irregular deep neural networks, or liNets. Unlike CNNs, our novel liNets are well suited to a bio-inspired retinal model with irregularly distributed photoreceptors, enabling greater numbers of photoreceptors resulting in enhanced visual acuity.

We empirically demonstrated the utility, efficiency, and performance of trained liNets in foveation, visuomotor control, and in a prototype 3D active facial recognition subsystem incorporated into the brain of a simulated biomechanical human musculoskeletal model with biomimetic eyes. Our approach has been to train the DNNs with large quantities of training data that are synthesized by the virtual human itself.

Inevitably, our current retinal model is a gross simplification of the biological human retina. Modeling the retinal ganglion cells, as well as circular excitatory-inhibitory center-surround and oriented receptive fields, the M-cell pathway, and the retinocortical map [6] are worthwhile avenues for future research with our framework. We believe that our embodied, biomimetic modeling approach is suitable for such extensions. Our foveation liNet generates saccadic eye movements to foveate interesting objects in a variety of different scenarios; hence, our model can be valuable in human visual attention research [16], a topic that we wish to explore in future work.

References

[1] C. A. Curcio, K. R. Sloan, R. E. Kalina, and A. E. Hendrickson, "Human photoreceptor topography," *Journal of Comparative Neurology*, vol. 292, no. 4, pp. 497–523, 1990. 1, 4, 8

[2] M. Nakada, H. Chen, and D. Terzopoulos, "Learning biomimetic perception for human sensorimotor control," in *Proc. Second IEEE CVPR Workshop on Mutual Benefits of Cognitive and Computer Vision (MBCC 2018)*, Salt Lake City, UT, June 2018, pp. 2030–2035. 1, 3, 5

[3] M. Nakada, T. Zhou, H. Chen, T. Weiss, and D. Terzopoulos, "Deep learning of biomimetic sensorimotor control for biomechanical human animation," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 56:1–15, 2018, proc. *ACM SIGGRAPH 2018*, August 2018. 1

[4] M. Nakada, A. Lakshmipathy, H. Chen, N. Ling, T. Zhou, and D. Terzopoulos, "Biomimetic eye modeling & deep neuromuscular oculomotor control," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–14, 2019, proc. *ACM SIGGRAPH 2019*, Brisbane, Australia, Nov 2019. 1, 3

[5] P. Riordan-Eva and E. Cunningham, *Vaughan & Asbury's General Ophthalmology*. McGraw Hill Professional, 2011. 3

[6] E. L. Schwartz, "Spatial mapping in the primate sensory projection: Analytic structure and relevance to perception," *Biological Cybernetics*, vol. 25, no. 4, pp. 181–194, 1977. 3, 8

[7] P. Shirley and R. K. Morley, *Realistic Ray Tracing*, 2nd ed. Natick, MA, USA: A. K. Peters, Ltd., 2003. 3

[8] M. F. Deering, "A photon accurate model of the human eye," *ACM Trans. on Graphics (TOG)*, vol. 24, no. 3, pp. 649–658, 2005. 3, 8

[9] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10, p. 1995, 1995. 3, 8

[10] V. Blanz and T. A. Vetter, "Morphable model for the synthesis of 3D faces," in *Proc. ACM SIGGRAPH 99 Conf.*, 1999, pp. 187–194. 6

[11] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Networks*, vol. 1, no. 2, pp. 119–130, 1988. 8

[12] R. Uetz and S. Behnke, "Large-scale object recognition with cuda-accelerated hierarchical neural networks," in *IEEE International Conference on Intelligent Computing and Intelligent Systems*, vol. 1. IEEE, 2009, pp. 536–541. 8

[13] L. J. Grady, "Space-variant computer vision: A graph-theoretic approach," Ph.D. dissertation, Boston University, 2004. 8

[14] M. Yeasin and R. Sharma, "Foveated vision sensor and image processing: A review," in *Machine Learning and Robot Perception*. Springer, 2005, pp. 57–98. 8

[15] P. Ozimek, N. Hristozova, L. Balog, and J. P. Siebert, "A space-variant visual pathway model for data efficient deep learning," *Frontiers in Cellular Neuroscience*, vol. 13, 2019. 8

[16] J. M. Findlay, I. D. Gilchrist *et al.*, *Active vision: The psychology of looking and seeing*. Oxford University Press, 2003, no. 37. 8

[17] T. F. Rabie and D. Terzopoulos, "Active perception in virtual humans," in *Proc. Vision Interface 2000*, Montreal, Canada, 2000, pp. 16–22. 8

[18] N. Sprague, D. Ballard, and A. Robinson, "Modeling embodied visual behaviors," *ACM Transactions on Applied Perception (TAP)*, vol. 4, no. 2, p. 11, 2007. 8