Cerebellar-Inspired Adaptive Control of a Robot Eye Actuated by Pneumatic Artificial Muscles

Alexander Lenz, Sean R. Anderson, A. G. Pipe, Chris Melhuish, Paul Dean, and John Porrill

***Abstract*—In this paper, a model of cerebellar function is implemented and evaluated in the control of a robot eye actuated by pneumatic artificial muscles. The investigated control problem is stabilization of the visual image in response to disturbances. This is analogous to the vestibuloocular reflex (VOR) in humans. The cerebellar model is structurally based on the adaptive filter, and the learning rule is computationally analogous to least-mean squares, where parameter adaptation at the parallel fiber/Purkinje cell synapse is driven by the correlation of the sensory error signal (carried by the climbing fiber) and the motor command signal. Convergence of the algorithm is first analyzed in simulation on a model of the robot and then tested online in both one and two degrees of freedom. The results show that this model of neural function successfully works on a real-world problem, providing empirical evidence for validating: 1) the generic cerebellar learning algorithm; 2) the function of the cerebellum in the VOR; and 3) the signal transmission between functional neural components of the VOR.**

***Index Terms*—Biomimetic model, cerebellum, neurorobotics, pneumatic artificial muscle (PAM), vestibuloocular reflex (VOR).**

I. Introduction

CONTROL of movement is one of the most striking features of the information processing tasks that the brain seems to perform better than its artificial rivals. The graceful and smooth movements exemplified by animals and humans look very different to the jerky and stereotyped movements that have come to be characterized as "robotic." Human movement characteristics are likely to be a product of two complementary elements, namely, musculoskeletal structure and neural signal processing, which are analogous in robotics to hardware and software. The investigation reported here is the development of a robot that is inspired by biology in terms of both the physical components [pneumatic artificial muscles (PAMs)] and the control algorithm, which is modeled on cerebellar function.

Manuscript received July 11, 2008; revised February 26, 2009. First published April 14, 2009; current version published November 18, 2009. This work was supported by the Engineering and Physical Sciences Research Council under Grant GR/T10602/01 through their Novel Computation Initiative. The work of S. R. Anderson was supported by a Wellcome Trust "Value in People" Grant. This paper was recommended by Associate Editor J. E. Bobrow.

A. Lenz, A. G. Pipe, and C. Melhuish are with the Bristol Robotics Laboratory, Bristol BS16 1QD, U.K. (e-mail: alex.lenz@brl.ac.uk; tony.pipe@brl.ac.uk; chris.melhuish@brl.ac.uk).

S. R. Anderson, P. Dean, and J. Porrill are with the Centre for Signal Processing in Neuroimaging and Systems Neuroscience, Department of Psychology, University of Sheffield, Sheffield S10 2TP, U.K. (e-mail: s.anderson@sheffield.ac.uk; p.dean@sheffield.ac.uk; j.porrill@sheffield.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2009.2018138

The region of the brain that is particularly associated with adaptive control is the cerebellum [1]. Models of cerebellar function are usually based on the ideas of Marr and Albus [2], [3], including the adaptive filter representation [4], which is at the core of a number of cerebellar models. The model used here is also based on the adaptive filter, and the learning algorithm is decorrelation control [5], so called because learning is driven by correlation of the sensory error and the motor command. There are numerous other models of cerebellar function, whose complexity often precludes analysis of control features such as theoretical limits on rates of learning and stability [6]–[8]. The advantages of the model studied here are as follows: 1) It is a simple representation, easy to interpret and understand, as well as analyze for stability using systems engineering techniques [9]; 2) it is generically applicable to different motor control tasks such as both eye and arm movements [5], [10]; 3) it directly utilizes the sensory error signal to drive adaptation, rather than the motor error signal [11] or the motor command signal [12]. This is important because electrophysiological recordings suggest that the sensory error signal is more likely to represent the neural signal transmitted along the climbing fiber (CF) in the cerebellum [13], [14].

The main objective of implementing this model of cerebellar function in a pneumatically actuated robot is to confirm that the algorithm can indeed solve real-world problems that are similar to those encountered in biology. Computational models of neural function are traditionally investigated by simulation. A complementary technique that is growing in popularity is to embody the algorithm in a robot [15], [16], of which there are a growing number of examples [17]–[21]. Embodiment tests the model in real-world conditions, which can generate unforseen insights and provide additional rigorous testing and validation of the functional understanding upon which the model is constructed.

The exemplar cerebellar control problem investigated here is image stabilization in a robotic eye. This problem is analogous to that solved by the vestibuloocular reflex (VOR) in humans, which is known to be calibrated by the cerebellum [22]. The VOR uses a head velocity signal sensed by the vestibular system to drive a feedforward inverse plant compensation scheme [5]; the controller counterrotates the eyes in response to head movements with the opposite velocity of the head, which maintains a stable visual image on the retina. The VOR is a particularly suitable test-bed system because it is a well-specified biologically important behavior that corresponds to a generic tracking control problem and is known to involve the cerebellum. Further biomimetic schemes based on the VOR have also been investigated in robotics; the key difference

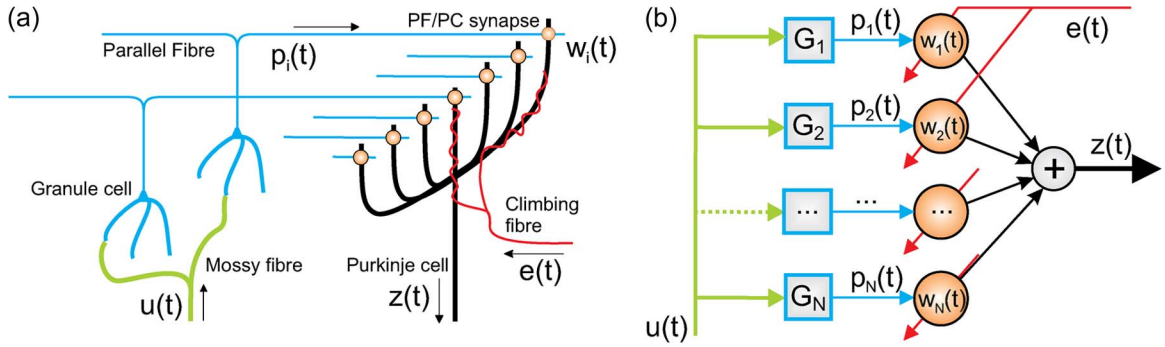

Fig. 1. (a) Schematic of the cerebellar microcircuit. The input to the cerebellum is via MFs that transmit the signal $u(t)$. GCs process the signal $u(t)$ to produce an output signal carried by the PFs, which are the axons of the GCs. PFs synapse onto the dendrites of a PC, which outputs the signal $z(t)$. PF/PC synaptic efficacy is altered by correlational firing of the CF, carrying the signal $e(t)$, and PFs. (b) Interpretation of the cerebellar microcircuit as an adaptive filter. The input to the filter is the signal $u(t)$. GC behavior is modeled in the simplest case as a tap-delay line but could be some more complex signal transformation, performed by $G_1, \ldots, G_N$. The filter weights $w_1, \ldots, w_N$, corresponding to PF/PC synapses, are adapted by the LMS rule, specifically, correlation of the signal p_i, for $i = 1, \ldots, N$, and the signal $e(t)$. Figure modified from [9].

here is that the controller is modeled on cerebellar function as opposed to being derived from statistical signal processing [23] or specialized geometric methods [24].

In relation to our previous work on the VOR in robotic systems, the robot designed, constructed, and analyzed here is a development of an electrically actuated prototype [25], [26]. The pneumatically actuated robot presents a more difficult control problem than electric actuation, requiring pulsewidth modulation (PWM) of high-speed airflow valves. The advantage of pneumatic actuators is that they are constructed from a compliant material. This is a central feature of a "soft" robot, contrasting with rigid robots, which are usually actuated by hydraulics or electrics. The construction of soft robots is primarily motivated by the need to minimize the hazard that a robotic device poses to humans; the compliant material reduces the risk of both injury to humans and damage to the robot in the event of a collision. One application area is in the care of disabled or elderly individuals, which is becoming a more prominent focus of robotics research [27], [28]. In this context, it is relevant to note that the cerebellar algorithm investigated here has been applied in simulation to the control of robotic arm movements [10].

The actuators in the robot eye were based on PAMs. The artificial muscle is a pneumatic actuator that was originally developed through the 1930s–1940s in expansive mode form [29], [30] to the 1950s–1960s in contractive mode form, where it latterly became known as the McKibben artificial muscle [31]–[33]. This actuator has evolved with technological advances into what is commonly known as the PAM [34]. The PAM is made of rubber, which contracts when filled with air and elongates when emptied. Like real muscles, PAMs usually operate in antagonist pairs because they only provide a pulling force. The dynamical properties of the PAM are nonlinear over a wide range of operation, although they are approximately linear in the region of a central operating point, suggesting that linear control is appropriate under certain conditions [35], [36]. The PAM has found wide application in robotics, inspiring a number of contributions on solving the associated control problem [8], [37]–[40].

In summary, we report here the implementation of a linear adaptive control scheme based on a model of cerebellar function in a robot eye to stabilize a visual image, under disturbances from a moving platform. In Section II, background information is given on the cerebellum, and it is described in the context of the adaptive filter model. In Section III, we provide a computational description of the VOR control problem that we replicate using the robot eye. The design and construction of the pneumatically actuated robot eye is described in Section IV. Performance of the control algorithm in simulation and in the robot are presented in Section V and discussed in Section VI, along with directions of future work. This paper is concluded in Section VII.

II. Computational Model of Adaptive Cerebellar Function: Decorrelation Control

This section briefly reviews the model structure and parameter estimation algorithm that comprises decorrelation control [5].

A. Cerebellar Model Structure

The cerebellar cortex has a similar repeating architecture throughout. Hence, it has been argued that by describing one region of the cerebellum, insight can be gained into the function of the whole of cerebellar cortex [41]. Different regions of the cerebellum are functionally distinguished by the input signals that they receive. The implication of this is that computational processing is the same throughout the cerebellum (due to the similarity of the neural architecture) and is applied to solve disparate functional tasks.

In the cerebellar cortex, a mossy fiber (MF) input is distributed over many granule cells (GCs), whose axons form parallel fibers (PFs), which synapse on Purkinje cells (PCs). In models of the Marr–Albus type, correlated firing of a PF and the single CF that winds around the PC alters the efficacies of PF/PC synapses [42], as illustrated in Fig. 1(a).

In the adaptive filter model of the cerebellum, as illustrated in Fig. 1(b), processing in the GC layer is modeled by a set of basis functions G_i for $i = 1, \ldots, N$, so that PFs carry expansion-recoded signals $p_i(t) = G_i[u(t)]$. At present, principles of PF encoding are not well understood. We have previously shown

that the exact choice of basis function is not critical to convergence of cerebellar adaptation (but will only affect the rate of convergence) [5], [9]. Hence, in this investigation, we represent the GC layer as tap-delay lines, similar to the original model by Fujita [4].

If the PC output is modeled as a weighted sum of PF inputs, the PC implements a "linear-in-weights" finite-impulse response filter. The CF input $e(t)$ is interpreted as an error signal, and synaptic weights $w_i(t)$ are adapted using a correlational learning rule, as described later in this paper.

In discrete time, the cerebellar filter is described as

$$z(t) = C(q,t)u(t) \tag{1}$$

where q is the forward-shift operator, i.e., $qu(t) = u(t+1)$, and

$$C(q,t) = w_1(t)q^{-1} + \cdots + w_N(t)q^{-N}. \tag{2}$$

The filter weights of $C(q,t)$ are time varying because they are adaptive. The adaptive weights directly represent the synaptic plasticity at the PF/PC synapses. This interpretation of the cerebellum as an adaptive filter, which was originally proposed by Fujita [4], is very powerful because it unifies both the gain and timing of motor control actions in a single functional object.

B. Computational Model of Synaptic Plasticity

The plasticity at the PF/PC synapses is modeled by a covariance learning rule [43]. This learning rule computationally resembles the least-mean squares (LMS) rule of Widrow and Stearns [44], further supporting the interpretation of the cerebellum as an adaptive filter.

The cerebellar learning rule is represented in discrete time as

$$\mathbf{w}(t+1) = \mathbf{w}(t) - \gamma E\left[\mathbf{p}(t)e(t)\right] \tag{3}$$

where $\mathbf{w}(t) \in \mathbb{R}^N$ is the column vector of adaptive filter weights, γ is a learning rate constant, and $E[.]$ denotes the expected value; hence, the term $E[\mathbf{p}(t)e(t)]$ is related to the correlation between the vector of PF signals $\mathbf{p}(t)$ and the error signal transmitted by the CF $e(t)$.

For the case where the GC layer is modeled as a tap-delay line, the vector of PF signals $\mathbf{p}(t)$ is simply equivalent to the delayed MF input signal $u(t)$, i.e.,

$$\mathbf{p}(t) = \left[u(t-1), \ldots, u(t-N)\right]^T. \tag{4}$$

There are variants on the learning rule in (3), based on the theory that the CF does not carry an accurate representation of the error signal. One simplified learning rule that has previously been investigated utilizes the sign of the error [5]

$$\mathbf{w}(t+1) = \mathbf{w}(t) - \gamma E\left[\mathbf{p}(t)\mathrm{sign}\left(e(t)\right)\right] \tag{5}$$

where $\mathrm{sign}(.)$ denotes the sign of the quantity in parentheses. This rule is used here because it is a conservative estimate of the computational sophistication of the system under investigation.

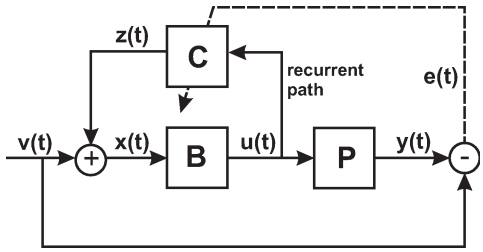

Fig. 2. Linearized model of the horizontal VOR, which is derived from the neural circuitry, where C represents the adaptive cerebellar filter, B is the fixed brainstem filter, and P is the oculomotor plant. The output of the plant $y(t)$ is the eye velocity. The feedforward control scheme is driven by the head velocity signal $v(t)$, which is sensed by the vestibular system (not shown). The motor command $u(t)$ is the input to the oculomotor plant, a copy of which is transmitted to the cerebellum via a recurrent path. The brainstem filter B is driven by the combined cerebellar and the head velocity signal, i.e., $x(t) = v(t) + z(t)$.

III. Systems-Level Computational Model of the VOR

The model of cerebellar function studied here is investigated in the context of a specific task: the VOR. The VOR is a reflex that stabilizes the visual image on the retina when the head moves, by counterrotating the eyes with the opposite velocity to the head. A sufficiently precise representation of the head velocity is available from the vestibular system [45]. This task is replicated in the robot eye, as described in the next section.

A. Overview

In the model system of the VOR considered here, the head velocity signal drives the control scheme. The head velocity signal can be regarded as a reference signal that the controller must track in a feedforward mode of operation. This reference signal $v(t)$ is added to the cerebellar output and transmitted through the brainstem, which corresponds to neurons in the medial vestibular nucleus, nucleus prepositus hypoglossi, and oculomotor nucleus. The output of the brainstem B is a motor command $u(t)$, which acts on the oculomotor plant P. A copy of $u(t)$ is sent to the adaptive cerebellar filter C via a recurrent pathway. The motor command acts on P to move the eyes with velocity $y(t)$. The head velocity is subtracted from the eye velocity to produce the retinal slip error signal $e(t)$. Retinal slip is used to adapt the filter weights of C and is carried by the CF. The VOR scheme is illustrated in Fig. 2 for horizontal movements of the eye.

B. Recurrent Architecture

A central feature of the control algorithm is that it uses a copy of the motor command signal as the input to the cerebellum via a recurrent pathway, which is supported by anatomical and electrophysiological evidence [46], [47]. Hence, in the VOR, the motor command is used as both the input signal to the cerebellum and the control input to the plant. The significance of the recurrent architecture is key to the previously described learning rule, because it allows the sensory error signal (retinal slip) to be used to drive adaptation of the filter weights [9].

C. Brainstem Model Component

The head velocity signal and cerebellar output project to the brainstem, which suggests that the brainstem plays a role in the control of the eyes during activation of the VOR. One effect of flocculectomy (removal of the region of the cerebellum associated with the VOR) is that the eyes do not maintain an eccentric position after a saccade [48]. This is widely interpreted as implying that the flocculus improves performance of a brainstem, which is undergained at low frequencies [5]. Hence, the brainstem is modeled as a fixed low-order linear controller (a lag compensator or a leaky integrator) $B(q)$, where

$$B(q) = \frac{b_0 - b_1 q^{-1}}{1 - a q^{-1}}. \tag{6}$$

The significance of this control architecture is that the brainstem provides a conservative level of control action, which the cerebellum improves upon by fine-tuning the response. This may be advantageous in an engineering scenario: 1) When the cerebellar controller is in an untrained state, the brainstem maintains a conservative level of performance; and 2) there is redundancy in the control scheme.

D. Optimal VOR Control Solution

The VOR control scheme uses sensory measurement of the head velocity to drive movements of the eyes, controlling the eye velocity. The VOR scheme, as illustrated in Fig. 2, is described in discrete time as

$$y(t) = H(q,t)P(q)v(t) \tag{7}$$

where the discrete-time filter $H(q,t)$ is a time-varying controller, i.e.,

$$H(q,t) = \frac{B(q)}{1 - B(q)C(q,t)} \tag{8}$$

and where, specifically, the cerebellar filter $C(q,t)$ is the time-varying component of the controller. The optimal solution for tracking signals at all frequencies is to set the controller $H(q,t)$ equal to the inverse of the oculomotor plant, i.e.,

$$P(q)^{-1} = \frac{B(q)}{1 - B(q)C(q,t)^*}. \tag{9}$$

Rearrangement of (9) leads to the expression of the optimal cerebellar filter, i.e.,

$$C(q,t)^* = B(q) - P(q)^{-1}. \tag{10}$$

The convergence of a recursive learning algorithm, based on the learning rule given in (3), to the optimal cerebellar filter $C(q,t)^*$ is discussed by Porrill *et al.* [9], where stability is proved using the Lyapunov's direct method.

The number of tap delays required to implement $C(q,t)^*$ in a practical implementation would be related to the decay time of the impulse response of $C(q,t)^*$. This may require more parameters than, for instance, an infinite-impulse response filter but is less restricted in the dynamics it can represent. Additionally, there are methods available for greatly reducing the number of parameters in the FIR filter, such as Laguerre functions [49].

E. Discrete-Time Implementation of the Control Algorithm

The adaptive control algorithm is simple to implement in simulation and in an embedded system. In discrete time, the control input to the plant is the output from the fixed brainstem controller driven by the combination of the cerebellar filter output and the head velocity, i.e.,

$$u(t) = B(q)x(t) \tag{11}$$

where the combined input signal to the brainstem filter is

$$x(t) = v(t) + \mathbf{w}(t)^T \mathbf{p}(t) \tag{12}$$

and where $\mathbf{w}(t)$ is the vector of time-varying adaptive filter weights defined in (3), and $\mathbf{p}(t)$ is the vector of PF signals defined in (4). A recursive implementation of (11) and (12) comprises the control algorithm. The adaptive filter weights are updated by the learning rule in (3) at each time step t. The learning rate γ and the fixed control component $B(q)$ are typically tuned in simulation using a model of the plant, which is the case here and is further discussed in Section V.

IV. Design and Construction of a Robot Eye Actuated by Pneumatic Muscles

The objective of this investigation was to design and construct a robot eye analogous to the human eye and associated cerebellar controller and then disturb it in a manner similar to that experienced during the VOR. The disturbances that excite the VOR control system in humans are movements of the head, which would cause the visual image of the world to shake if left uncompensated. To cause a similar excitation of the robot eye, it was mounted on a movable platform to recreate disturbances analogous to head movements. The robot eye was equipped with solid-state gyroscopes to sense the movements of the platform. To compensate for the platform disturbances, the robot eye was actuated in 2 degrees of freedom (DOFs), vertically and horizontally, referred to here as pitch and yaw, respectively. The robot eye system, including actuators, is equivalent to the plant $P(q)$ previously discussed in the context of the VOR and shown as P in Fig. 2. The pneumatically actuated robot is shown in Fig. 3. The experimental setup, including the movable platform, is shown in Fig. 4. This section discusses the design and construction of the robot and the associated control scheme.

A. Overview of Robot Eye Design

This section gives a brief overview of the robot design in terms of controller, sensors, and system integration. The overall system constitutes a distributed computing system comprising several microprocessors and peripheral hardware. A schematic of the robot is given in Fig. 5.

1) Control and Sensory Integration: The control and sensory integration of the robot for any DOF is centered

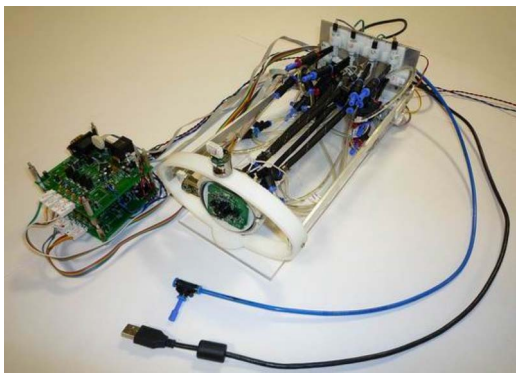

Fig. 3. (Right) Pneumatically actuated robot connected to the (left) two stacked embedded controllers corresponding to each DOF: pitch and yaw.

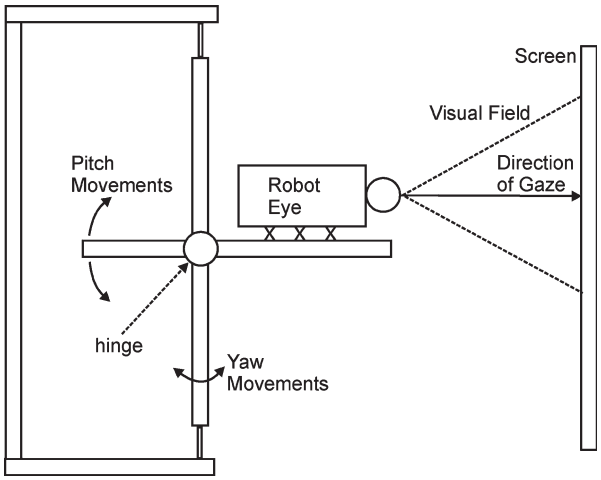

Fig. 4. Illustration of the robot eye resting on the movable platform, where the platform is analogous to the moving head of a human, actuated in 2 DOFs. The robot eye comprises a 2-DOF gimbal-mounted web camera, actuated by four PAMs. Solid-state gyroscopes attached to the robot sense the rotational disturbances of the platform on which the robot is mounted. The central vertical strut of the platform horizontally rotates about the vertical axis, and the strut on which the robot rests vertically rotates about the horizontal axis. Each strut of the platform is actuated and controlled from a system that is independent of the robot. Hence, in combination, the struts of the platform disturb the direction of gaze of the robot eye in both pitch and yaw.

around a custom-designed microcontroller board based on the dsPIC30F6012 from Microchip Technology Inc. It is assumed here that the control in each DOF is independent. The microcontroller platform, which is referred to as the neurocontroller in this paper, facilitates the integration of sensory signals, the computation involved in the adaptive control algorithm, and the control drive to the robot. The sensory integration and control calculation implemented by the neurocontroller is equivalent to the signal routing through the brainstem filter $B(q)$ and cerebellar filter $C(q,t)$, as shown in the VOR diagram in Fig. 2.

2) Motor System: The control signal output of the neurocontroller specifies the input to the motor system, which consists of the valves (with the appropriate drive circuitry) driving the PAMs, interfaced via a controller area network (CAN). All dynamic states of the system are broadcast on the CAN, and these signals are routed to a personal computer equipped with a CAN interface to monitor and record the system performance.

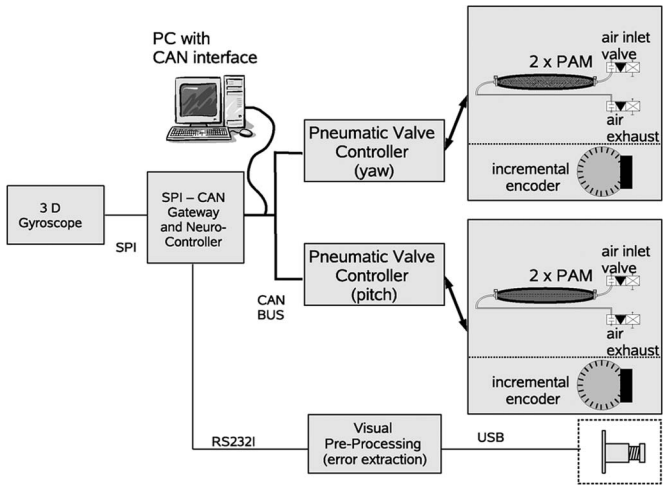

Fig. 5. Overview of the system architecture of the pneumatically actuated robotic eye. The neurocontroller integrates the sensory information from the gyroscopes (in yaw and pitch) and the visual feedback and then computes the adaptive control algorithm. The connectivity to the drive units for the PAMs is established via a CAN. The PAM drive units control the valves of the air muscles and provide the position feedback via high-resolution incremental encoders.

3) Rotational Velocity Sensors: The VOR is driven, or excited, by movement disturbances sensed as rotational velocities. To emulate this task, the robot eye direction of gaze was disturbed by mounting it on a movable platform, as shown in Fig. 4. Orthogonally mounted gyroscopes were attached to the robot eye to sense pitch and yaw velocities of the moving platform. These were integrated via the serial peripheral interface (SPI) to the neurocontroller. The excitation signal provided by the platform is analogous to $v(t)$ in Fig. 2, which is extended to a vector-valued signal for the 2-DOF case.

4) Visual processing: The visual signal from the Universal Serial Bus (USB) camera mounted in front of the robot is connected to a processing unit. This visual processor consists of a small embedded personal computer.[1] The visual motion along the x- and y-axes is conveyed via a low-bandwidth RS232 interface to the neurocontroller. The information in this visual motion signal is equivalent to the sensory error signal $e(t)$ (visual slip), as shown in the VOR diagram in Fig. 2.

5) Power Requirements: The description of the robot rig presented here focuses on a nonmobile system. However, due to the desire for future extensions to mobile robotics applications, the system was designed to run on a 12- and 5-V dc supply.

B. PAM Drive System

The robot eye is driven in pitch and yaw by a pair of antagonistic artificial muscles in each of the 2 DOFs (up to a maximum velocity of $370^\circ/\mathrm{s}$), as shown in Fig. 6. In terms of yaw control, the natural biological counterpoint to the PAMs in one eye is the two horizontal rectus muscles (medial/middle and lateral/outer). We have adopted the convention here of referring to the right muscle as medial and to the left muscle as lateral.

[1] 650-MHz Pentium with 512-MB RAM.

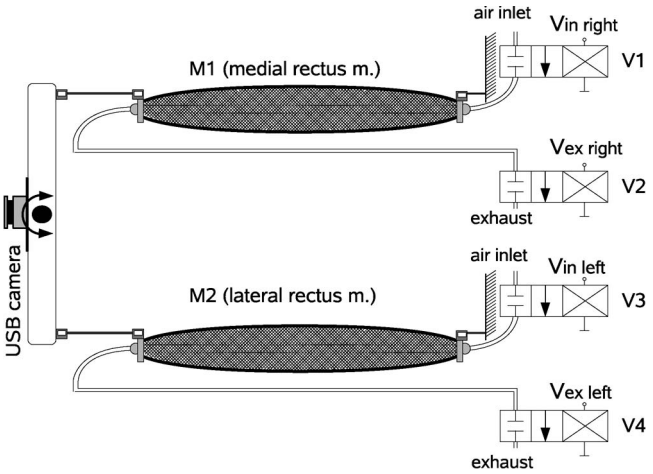

Fig. 6. Antagonistic pneumatic muscle drive scheme. V1 is the inlet valve for the pneumatic medial rectus, and V2 is the corresponding output valve. V3 and V4 are the inlet and exhaust valves for the lateral rectus, respectively.

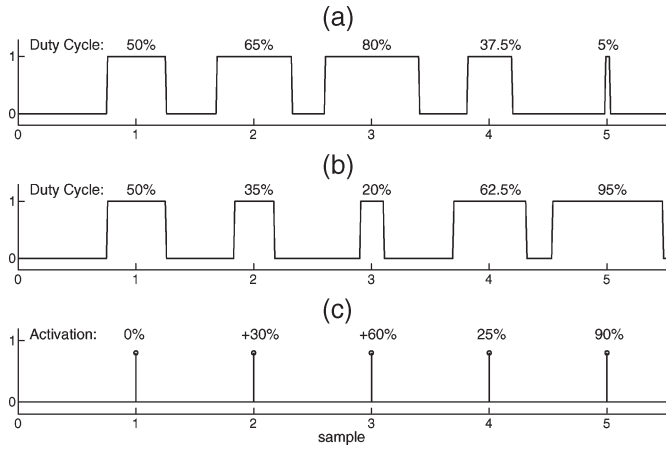

Fig. 7. Schematic waveforms of the PWM valve driving scheme for 0%, 30%, 60%, −25%, and −90% activation. (a) Opening times for the right inlet valve $V_{\text{in right}}$ and the left exhaust valve $V_{\text{ex left}}$. (b) Duty cycle of the left inlet valve $V_{\text{in left}}$ and the right exhaust valve $V_{\text{ex right}}$. (c) Center of the PWM period is displayed, and overall PWM activation is indicated in percent.

The mechanical system was designed to minimize cross-coupling by gimbal mounting the camera via two concentric rings, as shown in Fig. 3. Although this approach alone did not completely eliminate (nonlinear) cross-coupling due to the change in lever arm length, the operation of the robot eye in a limited region (as described in Section V-A2) reduced the effects below significant levels.

The force that a PAM produces is a function of its extension and state of inflation. It is the state of inflation that is directly controlled by two high-speed valves per muscle. The inlet valves allow pressurized air to flow into the PAM in a controlled manner, and the exhaust valves are used to accordingly relax the muscle.

In the robot eye, we used PAMs with an active (i.e., inflatable) length of 200 mm and a nylon braid diameter of 6 mm. The muscles were supplied by Shadow Robot Company and were custom-made in length for this application. The PAMs were operated at 2.5 bar, as supplied by an external compressor. The muscles were mounted with a screw mechanism to allow for the adjustment in tension. This was necessary to compensate small manufacturing tolerances and to avoid slack in the system before inflation.

The solenoid-activated valves are controlled via a PWM driving scheme. The aim is to regulate the state of inflation of a PAM by adjusting the opening time of the inlet and outlet valves. For clarification, the valves do not operate over a continuous range but have two discrete states, i.e., open or closed. Consequently, both the electrical activation and the airflow are pulsewidth modulated. The choice of a suitable PWM frequency is critical for quasi-continuous airflow modulation. The PWM period must be much shorter than the fastest time constant of the pneumatic drive system. However, if the period is too short, it becomes impossible to obtain valves with fast enough opening and closing times. Finally, the electronic hardware employed must be able to provide this base frequency and a reasonable resolution for the variation of the duty cycle. To accommodate this tradeoff, we adjusted the PWM frequency to 122 Hz on our microcontroller platform. Consequently, this provided a PWM duty cycle resolution of 10 bits for positive and negative actuation from the central position.

When the camera is centrally positioned (i.e., the camera is pointing straight-ahead), the opening times for the inlet and exhaust valves are set to 50% of a period $(= 0.5 \times 1/122\,\text{Hz} = 4.1\,\text{ms})$. To change the angle of rotation in a clockwise direction, the inlet time for the medial rectus is increased, and the corresponding exhaust time is decreased. At the same time, the opposing PAM is operated in reverse; hence, the inlet time for the lateral rectus is reduced by the same percentage, and the exhaust time is increased. This corresponds to reciprocal innervation in eye muscles, where the nonlinear control problem is simplified by push–pull linearization [50]. The alteration of opening and closing times of the valves for different levels of activation is illustrated in Fig. 7.

C. Fast High-Density Valves and Microcontroller Driving Circuit

The PAMs are controlled by modulating the airflow through valves. To be able to modulate the airflow at 122 Hz, fast valves are essential. For this application, a solenoid valve, namely, LHDA0533115H, from Lee Products Ltd., was utilized because of its high-density mechanical interface and opening and closing times (when appropriately driven) in the $<$ 1-ms range. Lee specifies the flow rate of their valves by defining the valve resistance to fluid flow, which is measured in liquid ohms (Lohm).[2] LHDA0533115H creates a flow restriction of 1100 Lohm when opened. Based on medium-dependent lookup tables and a conversion formula, the flow rate of the valves is 12.5 L/min when discharging air at 2.5 bar to atmosphere at 23 °C. The valve manufacturer explicitly recommends the

[2]Lee defines Lohm in the following way: "The Lohm has been selected so that a 100 Lohm restriction will permit a flow of 250 standard liters per minute of nitrogen at a temperature of 50 deg. F, and an upstream pressure of 90 psia discharging to atmosphere." More details can be found in the Engineering Section on The Lee Company website: http://www.theleeco.com.

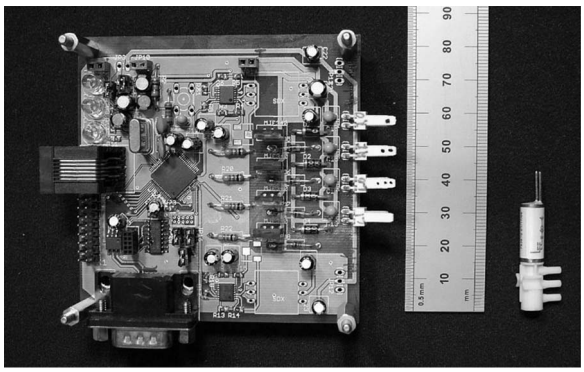

Fig. 8. (Left) Microcontroller driving board and (right) high-speed valve. One driving board controls four valves and is used to drive the robot in 1 DOF. Two stacked driving boards are used to operate the robot in 2 DOFs: pitch and yaw.

application of their valves in PWM application to control the airflow and suggest that a linear flow modulation is possible as long as the extreme ends of the duty bands are avoided.

The microcontroller driving circuits for the air muscles were custom designed and are based on the dsPIC30F5015 microcontroller from Microchip Technology Inc. The driving board is equipped with a CAN interface for seamless integration into the overall system infrastructure. An example of a driving board and a valve are shown in Fig. 8.

The driving board can control four valves and, as a consequence, is responsible for 1 DOF in an antagonistic muscle drive scheme, as shown in Fig. 6. This corresponds to one inlet valve and one outlet valve per PAM and two PAMs per DOF. Furthermore, the boards provide an interface for a high-resolution incremental encoder (80 000 counts per revolution) to monitor the position of one axis.

This choice of driving circuit reflects a modular design that allows for extension: Several boards can easily be stacked to give additional DOFs in a more complex robot. All boards can then communicate on the same CAN bus, which is designed to allow multiple nodes to simultaneously send and receive messages without the need for a master.

D. MEMS Gyroscopes Sense the Rotation Velocities

To measure the angular velocities of pitch and yaw movements, two MEMS gyroscopes (ADXRS300) from Analog Devices are utilized. The output of this device is a voltage proportional to the angular velocity about the axis normal to the top surface of the device (yaw rate sensor). The chosen device features a full-scale range of $\pm 300^\circ/\mathrm{s}$ and a sensitivity of $5\ \mathrm{mV}/^\circ/\mathrm{s}$. The upper band limit is 40 Hz.

A further microcontroller (dsPIC30F3013 microcontroller from Microchip Technology Inc.) is used to sample the rotational velocities measured by the gyroscopes, performing a 12-bit analog-to-digital conversion and providing the SPI interface for accessing the rotational velocities. The advantages of this are given as follows: 1) a clean modular design with defined interfaces and 2) local computational power that could be used in the future to transform the sensed rotational velocities of the engineered vestibular system into more biologically plausible signals.

E. Vision Processing

The control algorithm requires the visual slip signal $e(t)$ to drive the adaptation of the filter parameters (shown in Fig. 2). The visual slip signal is extracted by first interfacing the web camera with the embedded computing platform. The OpenCV computer vision library is utilized, running on Windows XP on a 633-MHz Intel Celeron embedded personal computer (RoadRunner III PC/104-Plus from LIPPERT GmbH, Mannheim). The web camera samples the scene, a few randomly organized black squares on a white background, at a rate of 30 frames/s. The resolution is 160×120 pixels. The time delay caused by visual processing is not significant at the tracking frequencies studied here (up to 1.5 Hz); however, methods for overcoming the problem of delay is the subject of continued theoretical investigation in biological eye movement control [51], which will subsequently be integrated into a future version of the robot.

The scene observed by the USB camera is passed to a feature extraction algorithm developed by Shi and Tomasi [52]. The pyramidal Lucas Kanade optical flow algorithm is then used to detect the amount of perceived image movement from frame to frame based on the detected features [53]. The average optical flow of the features along the x- and y-axes is then made available at the full frame rate for the gaze stabilization controller via an RS232 interface, as shown in Fig. 5.

F. Emulating the VOR Excitation Signal

To mimic the disturbances that drive the VOR, as caused by head movements in humans, the robot eye was mounted on a movable platform with 2 DOFs: pitch and yaw. The platform is a completely independent system that is driven by a separate embedded controller to the robot eye, which generates disturbances analogous to head movements.

The pneumatically actuated robot eye resting on top of this platform uses MEMS gyroscopes (as previously described) to sense the rotational velocities of the platform disturbances. This gyroscopic signal is equivalent to $v(t)$ in Fig. 2, that is, the sensory input that excites the control system. An illustration of the robot resting on the movable platform is shown in Fig. 4.

V. Modeling and Control Analysis

A. Analysis of Robot Dynamics

The dynamics of the robot were analyzed to determine a range of operation suitable for linear control. This led to a dynamic model of the system, equivalent to P in Fig. 2, that was used to perform a model-based control design of the robot offline, before implementation in the real-world system.

1) Linear Operating Range of Robot Dynamics: The response of the pneumatically actuated robot to step inputs was strikingly similar to that of force production in eye muscles, which dominates eye movement dynamics. The step inputs for both the robot eye, showing position in degrees, and the eye muscle force from [54] are shown in Fig. 9. Although the outputs are not directly comparable (this would require a model

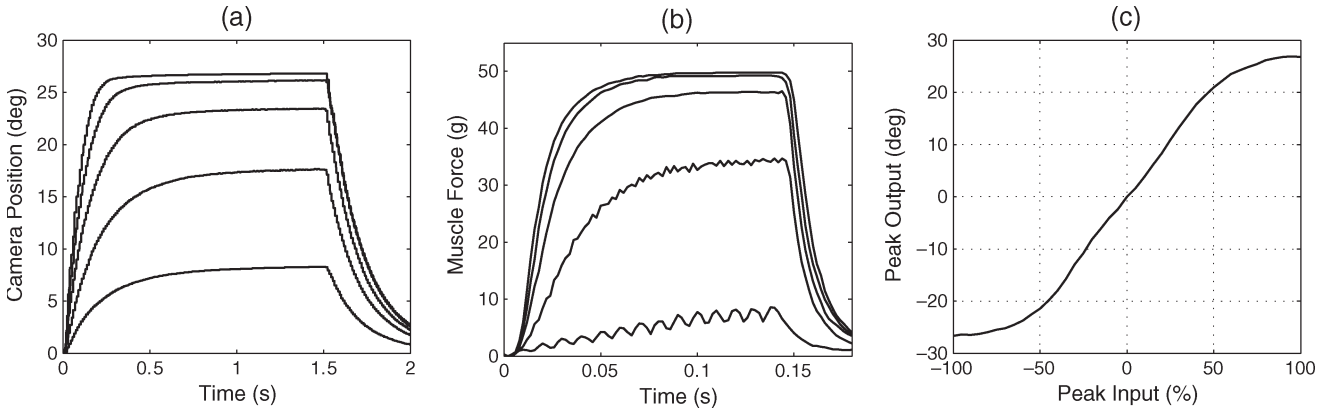

Fig. 9. Features of similar nonlinearity in robot eye dynamics and eye movement dynamics. (a) Horizontal step response of the robot at varying step sizes in only the positive direction. Step input magnitudes: 20%, 40%, 60%, 80%, and 100%. (b) Isometric force production in eye muscles (lateral rectus), as modified from [54]. Step input magnitudes in terms of the stimulation frequency: 100, 200, 300, 400, and 500 Hz. (c) Saturation effect in the horizontal step response of the robot for increasing the input step size, in both negative and positive directions.

of PAM dynamics or recordings of PAM forces) the result emphasizes the dynamic similarity in actuating a robot with PAMs to biological muscles and demonstrates that it is a useful test-bed system for a biologically-inspired control algorithm. The peak input versus peak output of the robot eye, as shown in Fig. 9(c) (derived from the robot step response), implied that a suitable linear region for controlling the robot was up to $\pm 50\%$ of the input range.

2) Linear Modeling of Robot Dynamics: The model of the robot (equivalent to the transfer function P in Fig. 2) was obtained from system identification. The robot modeling data were generated by activating the PAMs with colored noise (low-pass filtered white noise with a cutoff frequency of 2.5 Hz, via a fifth-order Butterworth filter) of duration 60 s. The input signal was limited to $\pm 50\%$ of the valve input range to avoid encountering the region of saturation at larger inputs. This led to an operational range of $\pm 22^\circ$ in position and velocities of $\pm 150^\circ/\text{s}$.

The data were sampled at 1 kHz and, for the purposes of modeling, were downsampled to 100 Hz. The model was identified in discrete-time. The structure was detected by comparing different model orders in terms of the root-mean-square (RMS) error. The output error form was used to represent the system dynamics [49]. Parameters were estimated by nonlinear regression (a quasi-Newton method, which was implemented using the optimization routine fminunc in MATLAB). The parameters were initialized for the search by first low-pass filtering the input/output signals using a fifth-order Butterworth filter, with a cutoff of 10 Hz, to attenuate the measurement noise and then estimated using least squares [55].

The structure detection showed that transfer function models of similar numerator and denominator orders were the most accurate. On inspection of the poles and zeros, it was apparent that this was due to pole/zero cancellations for model orders higher than the first order. The accuracy of the model prediction provided further support that a first-order model was sufficient to describe the system dynamics in the operational range of interest (giving a variance accounted for metric VAF $= 0.997$); the simulated first-order model is shown in comparison with the experimental data for both position and velocity in Fig. 10

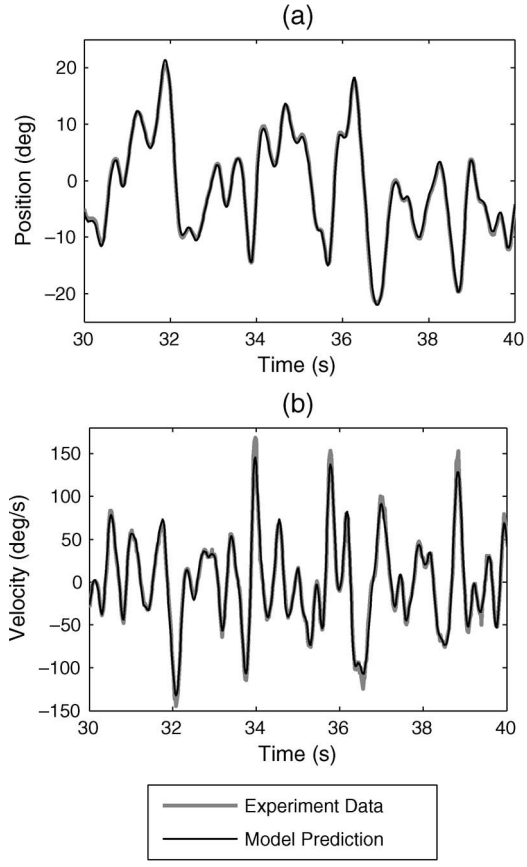

Fig. 10. Comparison of the robot movement signals with the model predicted output. (a) Position. (b) Velocity.

(for a zoomed 10-s window of representative data). The transfer function model of the system in continuous time was

$$P(s) = \frac{2.42}{s + 5.05} \tag{13}$$

where s is the Laplace operator. A notable feature of this model is that it is minimum phase and, hence, suitable for application of the inverse plant compensation control strategy, which is the basis of the model VOR control algorithm.

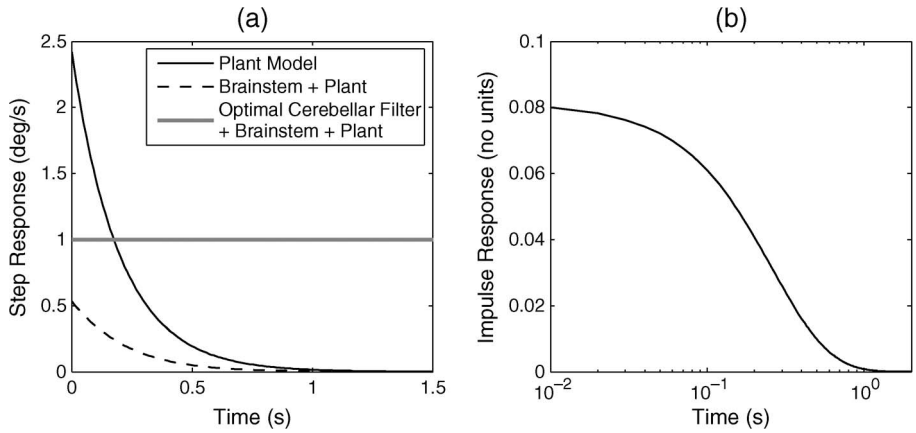

Fig. 11. (a) Simulated velocity step response of the robot model when uncompensated (i.e., plant only), compensated by the fixed brainstem controller only, and compensated by the brainstem and exact cerebellar filter required to invert the plant. (b) Impulse response of the optimal cerebellar filter required to invert the plant (in conjunction with the fixed brainstem controller).

B. Control of the Robot

1) Control Design and Analysis of 1-DOF Movements: The identified linear time-invariant model of the robot eye, as given in (13), was used to develop the control algorithm in simulation before implementation in the real-world system. The model of the robot was modified to function in velocity space rather than in position by inclusion of a derivative term in the numerator, i.e., $sP(s)$. This was because the image stabilization algorithm works by matching the velocity of the camera to the velocity of the disturbances; hence, the performance results of control are analyzed in velocity space.

The identified robot model was used to obtain a fixed controller analogous to the brainstem filter B in Fig. 2. The role of the brainstem filter here was primarily to damp the response of the uncompensated plant, which, upon construction, was significantly overgained at high frequencies, as shown in Fig. 11(a). This overshoot was a problematic feature of the robot because it meant that the valves and actuators were operated toward the extremes of their operational range, potentially leading to increased wear and damage. The brainstem filter was heuristically tuned to improve control performance by damping the overshoot; the brainstem filter was defined in continuous time as

$$B(s) = \frac{0.22s + 2.4}{s + 10}. \tag{14}$$

The step response of the simulated system controlled by this brainstem filter is shown in Fig. 11(a). For comparison, the ideal step response of the combined brainstem filter and optimal cerebellar filter, as derived from (10), is also shown in Fig. 11(a), and the ideal impulse response of this ideal cerebellar filter is shown in Fig. 11(b). Note that the gain of the optimally compensated system here is equal to 1 rather than to -1. This is because all results are reported here in terms of tracking the excitation signal $v(t)$ rather than rejecting it. This makes interpretation of the results easier from visual inspection, is the typical format of results for VOR studies, and does not affect the control algorithm, as it merely consists of switching the sign of the signal $v(t)$ to $-v(t)$.

The learning algorithm was tested in simulation and online by tracking a colored noise signal (white noise bandpassed between 0.5 and 1.5 Hz with a fifth-order Butterworth filter) sampled at 100 Hz for a duration of 600 s. The cerebellar filter defined in (2) was set up to have a finite-impulse response of 2 s, corresponding to a number of filter weights and tap-delay lines of $N = 200$ for the sample rate of 100 Hz. The learning rate in the parameter update rule was set to $\gamma = 0.015$. The simulation of the control scheme was conducted in a noise-free environment because there was no *a priori* information available on noise statistics in the robot. The brainstem controller and robot model were mapped from continuous time to discrete time, via a zero-order hold, along with the plant model. The results show that, over the period of training, the cerebellar model controller adapted to accurately track the disturbances. The untuned response of the robot primarily showing the influence of the brainstem controller is shown in Fig. 12(a). For comparison, the trained response of the robot is shown in Fig. 12(b).

The real-world robot eye system was tested under the same conditions as the simulation in 1 DOF (horizontal movements), i.e., the brainstem controller and learning rate in the parameter update rule were kept the same in each case. The training signal was also kept the same to facilitate a direct comparison between simulation and real-world systems. The results from controlling the real-world system in 1 DOF were very similar to those obtained in simulation, where, over a period of training time, the disturbances of the platform were well compensated, as shown in Fig. 12(c) and (d) in the same format as the simulation results (i.e., beginning and end of training, respectively). It is apparent that the transfer of the algorithm into the robot resulted in stable and convergent learning as expected from the simulation results and the theoretical analysis in [9]. In comparison with the simulation, it is apparent from a visual inspection of Fig. 12 that the real-world system has similar tracking performance; the response lags in phase and is undergained in the untrained system, which is substantially corrected in the trained system.

The time-domain plots of robot control performance in Fig. 12(c) and (d) have a corresponding frequency-domain interpretation, which is commonly used in describing VOR performance, where the gain of the combined controller and

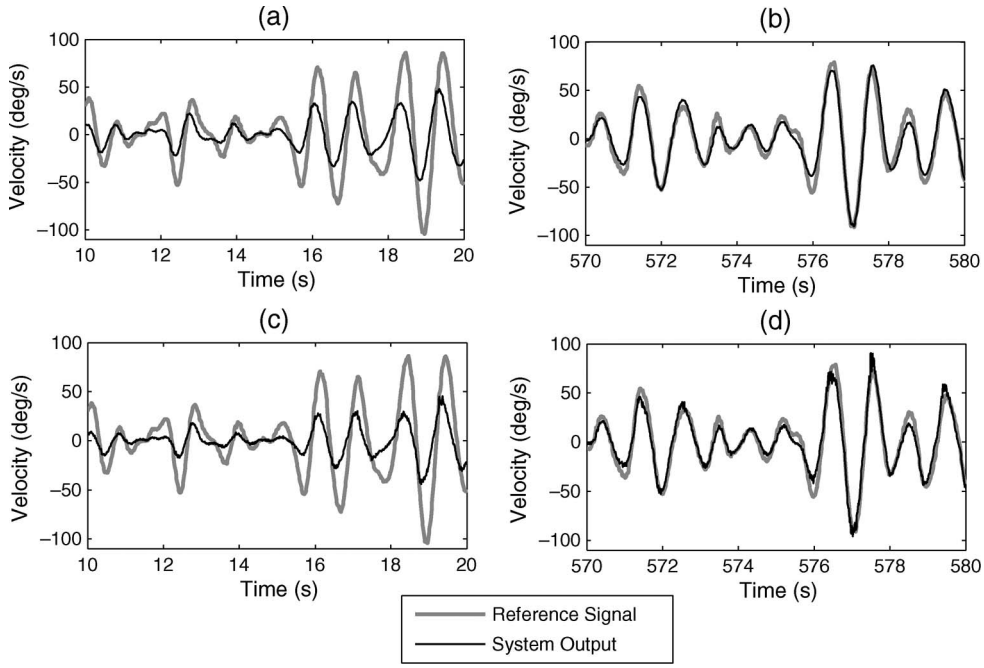

Fig. 12. Performance of the cerebellar control algorithm in simulation and on the real-world robot eye in 1 DOF only (horizontal movements). (a) Simulation: untrained controller. (b) Simulation: trained controller. (c) Robot: untrained controller. (d) Robot: trained controller.

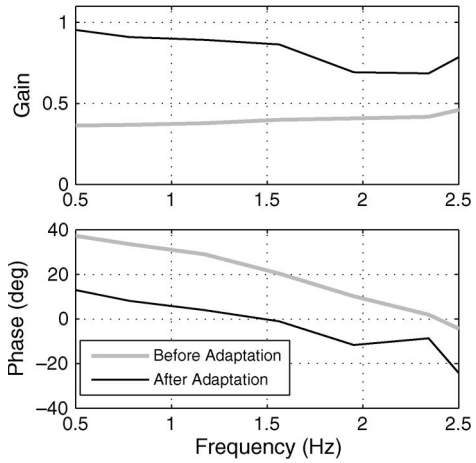

Fig. 13. Gain and phase of the robot eye before and after adaptation (obtained from the ETFE of the head-velocity-to-camera-velocity transformation).

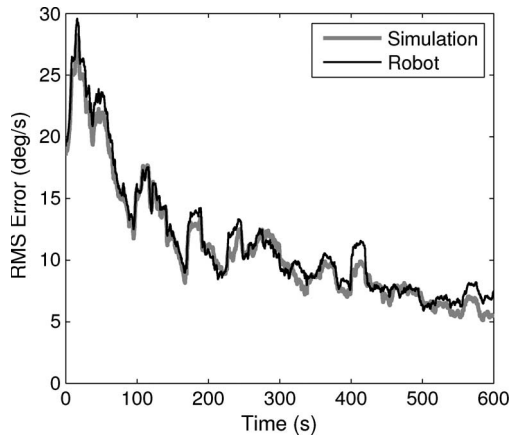

Fig. 14. Comparison of the tracking performance of the simulated and real-world robot eye in terms of the RMS tracking error windowed across 10-s intervals.

plant that transforms the head velocity into the eye velocity should be equal to 1, and the phase change should be equal to 0. Hence, we computed the gain and phase of the robotic system before and after learning [from the data shown in Fig. 12(c) and (d)] using the empirical transfer function estimate (ETFE) [49]. The data were windowed to 200 samples per window with a 50% overlap. We assumed that over the 10 s used to obtain the ETFE, the robot eye was well approximated by a stationary system due to the slow rate of parameter adaptation. The trained control system was found to improve both the gain and phase performance over the untrained system, which is shown in Fig. 13.

The windowed RMS errors, corresponding to the results in Fig. 12, are shown in Fig. 14 for both the simulation and real-world system. Both the simulation and real-world system learn with an exponential curve of a similar decay rate. The RMS errors are strikingly similar, where perturbations about the exponential learning curve in simulation are replicated in the real-world system. These results imply the following: 1) that the model of the robot is very accurate in this operational range; 2) that the effects of noise in the robot are not very significant; and, most importantly, 3) that the computational model of cerebellar function that has been tested in simulation in [5] and [9] has been validated to work as expected on a real-world system.

2) Control of the Robot Eye in 2 DOFs: The robot eye was designed to move in 2 DOFs so that disturbances in both yaw and pitch (horizontal and vertical movements) could be compensated. Hence, the fixed brainstem controller and cerebellar control algorithm were applied to the robot in both yaw and pitch to confirm that the control scheme could be extended to this case.

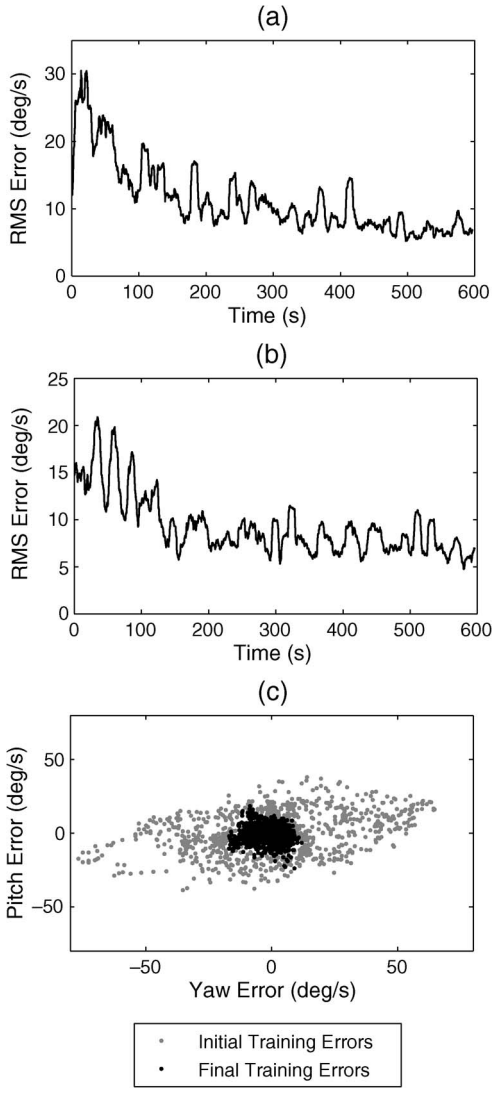

Fig. 15. Performance of the cerebellar control algorithm on the robot eye in 2 DOFs (horizontal and vertical movements). (a) Yaw RMS error signal windowed across time through the training period. (b) Pitch RMS error signal windowed across time through the training period. (c) Yaw error versus pitch error at the beginning and end of training (10-s window in each case).

The results of learning in terms of the RMS error are shown in Fig. 15(a) and (b), which demonstrates that the algorithm remains stable and convergent in both DOFs. Furthermore, the RMS error at the end of training appears similar to the 1-DOF study, as reported earlier in this paper.

One obvious feature of comparison in 2 DOFs, before adaptation, is that pitch movements are more accurately compensated than yaw movements. This is clearly shown in Fig. 15(c), where the spread of yaw errors at the start of training is much greater than in pitch. The reason for this may be that the brainstem controller performs better at compensating the system in the case of pitch, due to some difference in dynamics, although the construction is essentially the same in each DOF. It is apparent that at the end of the training period, the scatter plot of final training errors in Fig. 15(c) appears to have a similar variance in the x- and y-axes, which implies that the final trained performance is similar in pitch and yaw, and is much improved compared to the untrained control performance.

VI. Discussion

A. Analysis of Control Algorithm Performance

The control algorithm was demonstrated to effectively perform on the simulated model of the robot eye in 1 DOF (yaw movements only) and then also on the real-world system under the same conditions. The robot exhibited good performance, in general, at tracking the signals within the specified bandwidth (0.5–1.5 Hz) and appeared convergent and stable. This confirmed the linear modeling results, which suggested that the behavior of the robot in this range was well approximated by a linear function.

The adaptive controller substantially improved on the fixed controller (qualitatively modeled on the action of brainstem function in the VOR), which could be observed from the difference in tracking performance at the beginning and end of the learning period. The control scheme was extended from 1 DOF to include 2 DOFs, that is, both yaw and pitch. The performance of the brainstem controller in the pitch direction appeared to be better than the yaw direction. This might have been due to some difference in dynamics in each axis of rotation. However, the final performance appeared to be similar to the 1-DOF results in terms of the tracking error.

The main result of this investigation was that the performance of the controller, which was modeled on cerebellar function, was strikingly similar in both the simulation and real-world robotic system. This implicitly validates the cerebellar model in terms of the model structure, the learning rule, and the functional role of the cerebellum in the VOR. Successful performance of this model of brain function cannot ensure that the cerebellum has correctly been reverse engineered, but it does ensure that the adaptive filter model is a potential candidate, and no issues challenging this interpretation have been raised here.

B. Extensions to Nonlinear Control

The robot eye was demonstrated to have significant nonlinearities at the extremes of the operational range. Hence, the investigation presented here focused on linear control over a subregion of the robot dynamics, which turned out to be relatively large given the actuator nonlinearities. We found that a low-order linear model of the robot dynamics was an accurate descriptor of the system in the operational range investigated ($\pm 22^\circ$ in yaw and pitch). However, the nonlinear control problem is important to be addressed in future work because accurate control is typically desirable over the full operational range of a system.

It is apparent from the comparison of eye muscle and robot eye dynamics in Fig. 9 that the human control system must overcome similar nonlinearities to the robotic scheme. In addition, modeling studies of the oculomotor plant also suggest that eye movement dynamics are nonlinear [56], [57]. Hence, it is plausible that further investigations of biological motor control will lead to biomimetic solutions to nonlinear control problems in robotics. It has been suggested that 1) push–pull linearization in eye muscles [50] and 2) organization of motor unit recruitment in muscle force control (of the rabbit

nictitating membrane) [58] are potential major factors in the linearization of muscle dynamics. The latter point has led to the idea of a "virtual plant," where the adaptive controller (e.g., cerebellum) is presented with a linearized control problem at the systems level, by virtue of an appropriate recruitment strategy of motor units.

Additionally, further study of cerebellar function may lead to a biomimetic solution of the nonlinear control problem: The GC layer, which is modeled here as a tap-delay line, could in fact perform some nonlinear transformation of the input signal analogous to basis functions in artificial neural networks. The granular layer contains 98% of all cerebellar neurons. Indeed, GCs, which are the major constituent of the granular layer, make up at least half of all neurons in the entire mammalian brain [59]. However, little is known about what signal transformations are actually performed in the granular layer. This is a key question to answer for understanding human control at the systems level, as well as for technology transfer to other disciplines such as robotics.

C. Extensions to Mobile Robotics

For the successful application of many parallel adaptive neural control structures in multi-DOF systems (i.e., the task that the cerebellum seems to perform), it is crucial that the computational requirements are reduced to a minimum. The fact that the adaptive algorithm could successfully be implemented as a real-time system on a medium-range microcontroller platform suggests that the approach presented here is suitable for such applications. Furthermore, the relatively inexpensive hardware, combined with low power consumption, makes our system a possible candidate for cost-driven (mobile) mass-market products.

D. Applications in Neural Prosthetics

The architecture of the cerebellar model investigated here is based on the current understanding of neural connectivity from the neuroscience literature. This realistic adherence to the known connectivity is of great importance for the future development of neural prosthetics. The reason is that artificial control systems will be far simpler to integrate within the nervous system if they only require input/output signal pathways that are actually utilized by the cerebellum in natural operation. From this perspective, the use of the CF pathway to transmit the sensory error is a key feature of the model. Furthermore, the successful embodiment of the cerebellar algorithm in a robot reported here can be viewed as a necessary development step toward integrating an embedded digital controller within a biological nervous system.

VII. Conclusion

The aim of this paper was to investigate a generic model of cerebellar function in a specific task implemented in a real-world robotic system, i.e., image stabilization in a robot eye. This human image stabilization task, known as the VOR, has been emulated by construction of a soft robot eye, where the actuators were PAMs. The control algorithm has been designed on an identified linear model of the robot system and then successfully implemented online in the robot. Control in the linear region of robot operation exhibited good performance; the adaptive controller converged to a filter that successfully reduced tracking errors over a timescale similar to that indicated by simulations. The high degree of similarity in simulation and real-world system performance validated the investigated model of the cerebellum, suggesting that it was an accurate interpretation of the neural connectivity, the functional role of the cerebellum in the VOR, and the adaptive capabilities of this brain structure.

References

[1] M. Ito, *The Cerebellum and Neural Control*. New York: Raven, 1984.

[2] D. Marr, "A theory of cerebellar cortex," *J. Physiol. (London)*, vol. 202, no. 2, pp. 437–470, Jun. 1969.

[3] J. S. Albus, "A theory of cerebellar function," *Math. Biosci.*, vol. 10, pp. 25–61, 1971.

[4] M. Fujita, "Adaptive filter model of the cerebellum," *Biol. Cybern.*, vol. 45, no. 3, pp. 195–206, Oct. 1982.

[5] P. Dean, J. Porrill, and J. V. Stone, "Decorrelation control by the cerebellum achieves oculomotor plant compensation in simulated vestibulo-ocular reflex," *Proc. R. Soc. Lond. B*, vol. 269, no. 1503, pp. 1895–1904, Sep. 2002.

[6] R. E. Kettner, S. Mahamud, H.-C. Leung, N. Sitkoff, J. C. Houk, B. W. Peterson, and A. G. Barto, "Prediction of complex two-dimensional trajectories by a cerebellar model of smooth pursuit eye movement," *J. Neurophysiol.*, vol. 77, no. 4, pp. 2115–2130, Apr. 1997.

[7] K. Yamamoto, Y. Kobayashi, A. Takemura, K. Kawano, and M. Kawato, "Computational studies on acquisition and adaptation of ocular following responses based on cerebellar synaptic plasticity," *J. Neurophysiol.*, vol. 87, no. 3, pp. 1554–1571, Mar. 2002.

[8] S. Eskiizmirliler, N. Forestier, B. Tondu, and C. Darlot, "A model of the cerebellar pathways applied to the control of a single-joint robot arm actuated by McKibben artificial muscles," *Biol. Cybern.*, vol. 86, no. 5, pp. 379–394, May 2002.

[9] J. Porrill, P. Dean, and J. V. Stone, "Recurrent cerebellar architecture solves the motor-error problem," *Proc. R. Soc. Lond. B*, vol. 271, no. 1541, pp. 789–796, Apr. 2004.

[10] J. Porrill and P. Dean, "Recurrent cerebellar loops simplify adaptive control of redundant and nonlinear motor systems," *Neural Comput.*, vol. 19, no. 1, pp. 170–193, Jan. 2007.

[11] M. Kawato and H. Gomi, "The cerebellum and VOR/OKR learning models," *Trends Neurosci.*, vol. 15, no. 11, pp. 445–453, Nov. 1992.

[12] S. Khemaissia and A. Morris, "Use of an artificial neuroadaptive robot model to describe adaptive and learning motor mechanisms in the central nervous system," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 28, no. 3, pp. 404–416, Jun. 1998.

[13] J. I. Simpson, D. R. Wylie, and C. I. De Zeeuw, "On climbing fiber signals and their consequence(s)," *Behav. Brain Sci.*, vol. 19, no. 3, pp. 384–398, 1996.

[14] C. I. De Zeeuw, J. I. Simpson, C. C. Hoogenraad, N. Galjart, S. K. E. Koekkoek, and T. J. H. Ruigrok, "Microcircuitry and function of the inferior olive," *Trends Neurosci.*, vol. 21, no. 9, pp. 391–400, Sep. 1998.

[15] M. Kawato, "From "Understanding the brain by creating the brain" towards manipulative neuroscience," *Philos. Trans. Roy. Soc. B, Biol. Sci.*, vol. 363, no. 1500, pp. 2201–2214, Jun. 2008.

[16] B. Webb, "Robots in invertebrate neuroscience," *Nature*, vol. 417, no. 6886, pp. 359–363, May 2002.

[17] C. G. Atkeson, J. G. Hale, F. Pollick, M. Riley, S. Kotosaka, S. Schaul, T. Shibata, G. Tevatia, and A. Ude, "Using humanoid robots to study human behavior," *IEEE Intell. Syst.*, vol. 15, no. 4, pp. 46–56, Jul./Aug. 2000.

[18] O. Holland, "A strongly embodied approach to machine consciousness," *J. Conscious. Stud.*, vol. 14, no. 7, pp. 97–110, 2007.

[19] A. Karniel, M. Kositsky, K. M. Fleming, M. Chiappalone, V. Sanguineti, S. T. Alford, and F. A. Mussa-Ivaldi, "Computational analysis *in vitro*: Dynamics and plasticity of a neuro-robotic system," *J. Neural Eng.*, vol. 2, no. 3, pp. S250–S265, Sep. 2005.

[20] M. J. Pearson, A. G. Pipe, C. Melhuish, B. Mitchinson, and T. J. Prescott, "Whiskerbot: A robotic active touch system modeled on the rat whisker sensory system," *Adapt. Behav.*, vol. 15, no. 3, pp. 223–240, 2007.
[21] D. Vernon, G. Metta, and G. Sandini, "A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents," *IEEE Trans. Evol. Comput.*, vol. 11, no. 2, pp. 151–180, Apr. 2007.
[22] M. Ito, "Cerebellar learning in the vestibulo-ocular reflex," *Trends Cogn. Sci.*, vol. 2, no. 9, pp. 313–321, Sep. 1998.
[23] T. Shibata and S. Schaal, "Biomimetic gaze stabilization based on feedback-error-learning with nonparametric regression networks," *Neural Netw.*, vol. 14, no. 2, pp. 201–216, Mar. 2001.
[24] F. Panerai, G. Metta, and G. Sandini, "Visuo-inertial stabilization in space-variant binocular systems," *Robot. Auton. Syst.*, vol. 30, no. 1, pp. 195–214, Jan. 2000.
[25] A. Lenz, T. Balakrishnan, A. G. Pipe, and C. Melhuish, "Eye-robot: A biologically inspired adaptive controller," in *Proc. TAROS*, Aberystwyth, U.K., Sep. 2007, pp. 231–238.
[26] A. Lenz, T. Balakrishnan, A. G. Pipe, and C. Melhuish, "An adaptive gaze stabilisation controller inspired by the vestibulo-ocular reflex," *Bioinspir. Biomim.*, vol. 3, no. 035001, Sep. 2008.
[27] A. Bicchi and G. Tonietti, "Dealing with the safety–performance tradeoff in robot arms design and control—Fast and soft-arm tactics," *IEEE Robot. Autom. Mag.*, vol. 11, no. 1, pp. 22–33, Mar. 2004.
[28] K. Ikuta, H. Ishii, and M. Nokata, "Safety evaluation method of design and control for human-care robots," *Int. J. Rob. Res.*, vol. 22, no. 5, pp. 281–297, 2003.
[29] R. C. Pierce, "Expansible cover," U.S. Patent No. 2 041 950, May 1936.
[30] H. De Haven, "Tensioning device for producing a linear pull," U.S. Patent No. 2 483 088, Sep. 1949.
[31] M. M. Gavrilovic and M. R. Maric, "Positional servo-mechanism activated by artificial muscles," *Med. Biol. Eng.*, vol. 7, no. 1, pp. 77–82, Jan. 1969.
[32] R. H. Gaylord, "Fluid actuated motor system and stroking device," U.S. Patent No. 2 844 126, Jul. 1958.
[33] H. F. Schulte, "The characteristics of the McKibben artificial muscle," *The Application of External Power in Prosthetics and Orthotics*, pp. 94–115, 1961. no. 874.
[34] D. G. Caldwell, G. A. Medrano-Cerda, and M. J. Goodwin, "Braided pneumatic actuator control of a multi-jointed manipulator," in *IEEE Int. Conf. Syst., Man Cybern.*, Le Touquet, Paris, 1993, pp. 423–428.
[35] C. P. Chou and B. Hannaford, "Measurement and modeling of McKibben pneumatic artificial muscles," *IEEE Trans. Robot. Autom.*, vol. 12, no. 1, pp. 90–102, Feb. 1996.
[36] D. B. Reynolds, D. W. Repperger, C. A. Phillips, and G. Bandry, "Modeling the dynamic characteristics of pneumatic muscle," *Ann. Biomed. Eng.*, vol. 31, no. 3, pp. 310–317, Mar. 2003.
[37] T. Hesselroth, K. Sarkar, P. P. van der Smagt, and K. Schulten, "Neural network control of a pneumatic robot arm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 24, no. 1, pp. 28–38, Jan. 1994.
[38] P. van der Smagt, F. Groen, and K. Schulten, "Analysis and control of a rubbertuator arm," *Biol. Cybern.*, vol. 75, no. 5, pp. 433–440, 1996.
[39] B. Tondu and P. Lopez, "Modeling and control of McKibben artificial muscle robot actuators," *IEEE Control Syst. Mag.*, vol. 20, no. 2, pp. 15–38, Apr. 2000.
[40] J. H. Lilly and L. Yang, "Sliding mode tracking for pneumatic muscle actuators in opposing pair configuration," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 4, pp. 550–558, Jul. 2005.
[41] M. Ito, "Cerebellar circuitry as a neuronal machine," *Prog. Neurobiol.*, vol. 78, no. 3–5, p. 272, Feb.–Apr. 2006.
[42] M. Ito, "Long-term depression," *Annu. Rev. Neurosci.*, vol. 12, pp. 85–102, 1989.
[43] T. J. Sejnowski, "Storing covariance with nonlinearly interacting neurons," *J. Math. Biol.*, vol. 4, no. 4, pp. 303–321, Dec. 1977.
[44] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
[45] R. H. S. Carpenter, *Movements of the Eyes*, 2nd ed. London, U.K.: Pion, 1988.
[46] J. A. Buttner-Ennever and A. K. E. Horn, "Pathways from cell groups of the paramedian tracts to the floccular region," in *New directions in vestibular research*, vol. 781, S. M. Highstein, B. Cohen, and J. A. Buttner-Ennever, Eds. New York: New York Acad. Sci., 1996, pp. 532–540.
[47] T. Belton and R. A. McCrea, "Role of the cerebellar flocculus region in cancellation of the VOR during passive whole body rotation," *J. Neurophysiol.*, vol. 84, no. 3, pp. 1599–1613, Sep. 2000.
[48] D. S. Zee, A. Yamazaki, P. H. Butler, and G. Gucer, "Effects of ablation of flocculus and paraflocculus on eye movements in primate," *J. Neurophysiol.*, vol. 46, no. 4, pp. 878–899, Oct. 1981.
[49] L. Ljung, *System Identification—Theory for the User*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1999.
[50] P. Dean, J. Porrill, and P. A. Warren, "Optimality of position commands to horizontal eye muscles: A test of the minimum-norm rule," *J. Neurophysiol.*, vol. 81, no. 2, pp. 735–757, Feb. 1999.
[51] J. Porrill and P. Dean, "Cerebellar motor learning: When is cortical plasticity not enough?" *PLoS Comput. Biol.*, vol. 3, no. 10, pp. 1935–1950, Oct. 2007.
[52] J. Shi and C. Tomasi, "Good features to track," in *IEEE Conf. CVPR*, Seattle, WA, Jun. 1994, pp. 593–600.
[53] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Image Understanding Workshop*, Washington, DC, Apr. 1981, pp. 121–130.
[54] A. Fuchs and E. Luschei, "Development of isometric tension in simian extraocular muscle," *J. Physiol.*, vol. 219, no. 1, pp. 155–166, 1971.
[55] L. Ljung, "Initialisation aspects for subspace and output error identification methods," in *Proc. 5th Eur. Control Conf.*, Cambridge, U.K., 2003.
[56] S. R. Anderson, N. J. Gandhi, D. L. Sparks, J. Porrill, and P. Dean, "Nonlinearities in primate oculomotor plant revealed by effects of abducens microstimulation," *Society for Neuroscience Abstracts*, 2006, Atlanta, GA. no. 345.15.
[57] P. A. Sylvestre and K. E. Cullen, "Quantitative analysis of abducens neuron discharge dynamics during saccadic and slow eye movements," *J. Neurophysiol.*, vol. 82, no. 5, pp. 2612–2632, Nov. 1999.
[58] E. Mavritsaki, N. Lepora, J. Porrill, C. H. Yeo, and P. Dean, "Response linearity determined by recruitment strategy in detailed model of nictitating membrane control," *Biol. Cybern.*, vol. 96, no. 1, pp. 39–57, Feb. 2007.
[59] B. B. Andersen, L. Korbo, and B. Pakkenberg, "A quantitative study of the human cerebellum with unbiased stereological techniques," *J. Comp. Neurol.*, vol. 326, no. 4, pp. 549–560, Dec. 1992.

Alexander Lenz received the microelectronics engineering degree from the University of Applied Sciences Trier, Trier, Germany, in 2001 and the M.Sc. degree in advanced technologies in electronics from the University of the West of England, Bristol, U.K., in 2002. He is currently working toward the Ph.D. degree with the University of the West of England.

He has worked for Siemens VDO, Germany, as an Embedded Software Engineer for the automotive industry and is currently a Research Associate with the Bristol Robotics Laboratory, Bristol, addressing control and safety issues in human–robot cooperation scenarios. His research interests include distributed real-time systems, embedded sensing and control, robotics, and bioinspired algorithms.

Sean R. Anderson was born in Reading, U.K. He received the M.Eng. degree in control systems engineering and the Ph.D. degree from the University of Sheffield, Sheffield, U.K., in 2001 and 2005, respectively.

He is currently a Research Associate with the Centre for Signal Processing in Neuroimaging and Systems Neuroscience, Department of Psychology, University of Sheffield. He studies the central nervous system, using systems engineering techniques to interrogate methods of biological adaptive control. His research interests include identification of continuous- and discrete-time dynamic systems, nonlinear systems modeling, oculomotor plant dynamics, computation and function of the cerebellum, and the study of adaptive and optimal control in biological systems.

Dr. Anderson is a member of the Society for Neuroscience.

A. G. (Tony) Pipe received the Ph.D. degree from the University of Sheffield, Sheffield, U.K., in 1997.

He is the Deputy Director of the Bristol Robotics Laboratory, Bristol, U.K., and a Reader of robotics and autonomous systems. He has 15 years of experience of carrying out research in biologically inspired robotics, machine learning, and adaptive behavior, applied to intelligent and distributed control/monitoring systems. Although he is an Electronics/Control Engineer, he extensively works with other engineers and Life Scientists, such as neurobiologists, neuroscientists, and anatomists, since this is crucial to the advancement of his work on modern robotics.

Chris Melhuish received the degree in geology from Durham University, Durham, U.K., the M.Sc. degree in computer science from Bristol University, Bristol, U.K., and the Ph.D. degree in collective robotics from the University of the West of England, Bristol.

He is a Professor of robotics and autonomous systems with the University of Bristol and a Professor of intelligent and autonomous systems with the University of the West of England. He is the Director of the Bristol Robotics Laboratory, Bristol. His research interests include collective mobile robotics, robot–human interaction, neuroinspired systems, and energy autonomy in robots.

Prof. Melhuish is a Fellow of the British Computer Society and the Institute of Engineering and Technology. He is also a Chartered Engineer.

Paul Dean received the M.A. degree in physiology with psychology from the University of Cambridge, Cambridge, U.K., and the D.Phil. degree from the University of Oxford, Oxford, U.K.

He is currently a Professor with the Department of Psychology, University of Sheffield, Sheffield, U.K., and is a member of the Centre for Signal Processing in Neuroimaging and Systems Neuroscience. His major research interest is in producing computational models of neural systems that are based on both biological data and current developments in control engineering, signal processing, and robotics. These models are intended to serve as a vehicle for two-way communication between biological and physical sciences, allowing roboticists to use new discoveries in biology, and biologists to interpret their findings in the light of current developments in signal processing.

John Porrill received the Ph.D. degree from the University of Cambridge, Cambridge, U.K., working with J. Stewart on topics in classical general relativity.

He joined the Department of Psychology University of Sheffield, Sheffield, U.K., to work on human and computer vision with J. Mayhew and J. Frisby and has a continuing interest in the psychophysics of human stereo vision. His recent research projects include video tracking of eye movements and building the EyeLab open-source model of the extraocular muscle system. He is currently working on computational models of the role of the cerebellum in the control of eye movements.