A Visual Feedback Supported Intelligent Assistive Technique for Amyotrophic Lateral Sclerosis Patients

Zihao Wang, Aojie Zhang, Xinyue Xia, Sizhe Zhang, Haitao Li, Jiaqi Wang, and Shuo Gao*

Among diverse intelligent assistive systems developed for amyotrophic lateral sclerosis (ALS) patients, headwear eye tracking based ones trigger broad interests due to their merits such as noninvasive, cost effective, and high operation freedom. However, with headwear eye trackers, patients are easy to feel tired during human–machine interactivities (HMIs), and the operation accuracy is not satisfied compared with its counterparts. To address these two issues, herein, a visual feedback technique is developed which allows users to recognize machine’s vision by positioning a laser spot to the user watched object, according to the location information interpreted from user’s eye movement. Through the visual feedback technique, users not only obtain real-time feedback, but also can fine-tune the laser spot to the desired location before performing further operations. Experimental results demonstrate that the presented work can successfully reduce user’s fatigue and boost operation accuracy by 25.1% and 27.6%, respectively, therefore, advancing the field of intelligent assistive technologies.

1. Introduction

Amyotrophic lateral sclerosis (ALS) patients receive global attention due to their unique and horrible character that making patients gradually and irreversibly lose their abilities in controlling muscles.[1–3] During the disease development, patients are fully conscious, but cannot interact with the surrounding environment like healthy people. Hence, computer-assistive interactive (CAI) techniques are developed for making both ALS patients’ life and releasing their mental pain convenience.[4–6]

Among diverse CAI systems, eye-tracking-based architectures are broadly welcomed, owing to their merits of noninvasive, cost effective, and high accuracy,[7,8] meanwhile eye movement is one of the last losing muscle control functions supported by the ALS patients, indicating that eye trackers can accompany with patients for a longer period than other techniques.[9–12]

Fruitful results of eye-tracking-based CAI systems for ALS patients have been obtained.[13–18] For example, scientists at Technical University of Denmark presented a robotic teleoperation system based on gaze interaction, in which users can utilize gaze to remotely control the telerobot for achieving the path planning of maze map.[17] An eye-control-based wheelchair system is developed by researchers from the University of New York, in which eye movement is interpreted as control information that used for performing functions such as navigation and object recognition.[18]

In general, eye-tracking systems can be divided into desktop and headwear-based techniques.[19] Between them, the former requires a fixed relative position between the user and the desktop’s screen. Therefore, by monitoring the user’s gaze, machines can learn what content on the screen is being watched. In contrast, headwear-based techniques allow users to freely move their heads, hence are more preferred for ALS patients, as slightly muscle movement can slow down the disease progression, explore wider space (desktop-based techniques constrain all human-machine interactivities (HMIs) within the display region) and release patients’ mental pain.

However, current headwear-based eye trackers are of lower tracking accuracy compared with its counterpart. Furthermore, patients can suffer from fatigue quickly after using headwear eye trackers for a while. Although no confirmed statement has been reported in the literature about the origin of fatigue issue, from relevant studies in the area of HMI, a possible reason is that headwear-based eye trackers cannot provide users with visual feedback which can indicate where is watched by them. It has been widely reported that without a real-time feedback mechanism, users will feel fatigued during HMIs.

To address the two issues, in this article, we developed a visual feedback-supported headwear eye-tracking system for ALS patients, as shown in Figure 1. Here, the visual feedback function is enabled by integrating a laser device into a steering gear platform. After receiving the eye gaze information from the eye tracker, the computer will rotate the laser beam to the desired orientation. In addition, as the machine’s vision can be
recognized with the help of visual feedback function, users can fine-tune the laser spot location before giving further commands, therefore high operation accuracy can be reached. The experimental results validate that the presented technique can successfully remove user’s fatigue and boost operation accuracy. Details are provided in the following sections.

2. Methodology

In this section, both hardware architecture and algorithm development are explained. The former contains an eye-tracking system and a visual feedback system, and the latter involves methods for interpreting user’s intention.

2.1. Eye Tracking

As shown in Figure 2a, the eye-tracking system consists of glasses frames, an eye camera (LRCP-10190), an infrared light source (XUV-88IR), a scene camera (HBV-1466), an antislip rubber sleeve, and two USB cables. The eye camera is attached to the right leg of the glasses via a 3D-printed bracket and is about 8 cm in front of the right eye. The scene camera is secured to the left leg of the glasses by tie wraps. Both cameras are used to capture images and transmit data to the computer through USB cables. The parameters of the eye tracker are listed in Table 1.

Mainstream fixation estimation techniques include pupil central corneal reflex (PCCR), cross-ratio mapping (CRM), and neural network-based methods. Among them, PCCR is an image processing technique to extract the pupil center and Purkinje image under infrared light irradiation, and then fits the fixation point position by a second-order polynomial; CRM uses four IR LEDs to form a spot quadrilateral on the cornea, and the mapping from eye coordinate system to scene coordinate system can be realized by the cross-ratio invariance; neural networks apply a two-way network architecture to train datasets, which can approximate the line of sight state transition model. In our application scenario, PCCR is more preferred, as CRM requires complex hardware architecture, which limits its integration and transplantation. Neural networks require users to provide large training datasets, which is impractical for patients. Detailed fixation estimation based on PCCR is given below (shown in Figure 2b).

First, the images obtained from the eye camera are processed by Gaussian filter and grayscale. Second, the region of interest containing the pupil is extracted to reduce the amount of calculation. Since the threshold is usually a free parameter that can be affected by changes in lighting or camera settings, an adaptive threshold algorithm based on neighborhood average is adopted to segment the gray images of human eyes, and the eye contour is extracted by Canny edge detection.

![Figure 1. Conceptional depiction of the visual feedback-supported intelligent interactive system for ALS patients.](image1)

**Table 1. The parameters of the eye tracker.**

| System parameter                  | Infrared camera                           | Scene camera                            | LED         |
|-----------------------------------|-------------------------------------------|-----------------------------------------|-------------|
| Image resolution                  | 640 × 480                                 | 640 × 480                               |             |
| Field angle                       | 60°                                       | 100°                                    |             |
| Focal length                      | 6.0 mm                                    | 2.1 mm                                  |             |
| Frame rate                        | 30 Hz                                     | 30 Hz                                   |             |
| LED wavelength                    | 850 nm                                    |                                        | 5 mW/cm²    |
| LED irradiance                    |                                           |                                        |             |

![Figure 2. a) Photograph of the headwear eye tracker. b) Simplified system block diagram and algorithm flowchart of the eye-tracking system.](image2)
Considering that most of the pupils captured by the eye camera are elliptical, we determine the pupil center point and pupil radius roughly through Hough circle transformation, and then fit the pupil contour in this area through the least-squares ellipse fitting method to achieve accurate positioning of pupil center coordinates.[29,30] Generally, the Purkinje image is located around the pupil region and has the highest gray value, therefore the preselected area can be delimited and the binarization threshold can be adjusted to effectively locate the Purkinje image. The extraction result of P-CR vector is shown in Figure 3a.

After attaining the precise location of pupil center and Purkinje image, we use the second-order polynomial regression to locate the fixation point.[31] The direction of fixation is estimated by the mapping relationship between the P-CR vector \((u, v)\) in the pixel coordinate system of eye camera and the fixation point \((x, y)\) in the pixel coordinate system of scene camera. The mapping function can be obtained by calibration and quadratic polynomial fitting,[32] the formula is as follows

\[
\begin{align*}
x &= a_0 + a_1 u + a_2 v + a_3 uv + a_4 u^2 + a_5 v^2 \\
y &= b_0 + b_1 u + b_2 v + b_3 uv + b_4 u^2 + b_5 v^2
\end{align*}
\] (1)

From Equation (1), it can be learned that at least six calibration points are needed. However, normally nine points are used for obtaining higher precision.[33,34] The calibration experiments are carried out in the laboratory environment. When a subject’s head is still, he/she is asked to gaze at the nine fixed points on the wall (3 m away from the subject), and the machine records the P-CR vector in sequence. The measured data are fed into Equation (1), and the unknown parameters are calculated by the least-squares fitting method. Based on the mapping function, we only need to obtain the P-CR vector in the current eye image, and then the eye fixation position can be real-time interpreted in the scene image, as shown in Figure 3b.

In the verification test, we carry out 30 experiments for each subject, obtaining 270 fixation points (shown in Figure 3c) calculated by the PCCR algorithm. The corresponding error distribution of estimated points is shown in Figure 3d. The results demonstrate that the calibrated eye tracker offers a low angle estimation error of average 0.92° and 0.82° in the horizontal and vertical directions, respectively.

It should be noted that besides the eye-tracking algorithm explained above, an eye gaze detection algorithm is also needed which is for locking the fixation point. The algorithm is based on human eye gaze behavior which is defined as the duration time of the fixation point in the target area exceeding the preset dwell time threshold. Generally, the dwell time threshold is set to 800 ms, while the range of target area relative to the scene camera is less than 1°.

Figure 3. a) Detection example of the pupil center and Purkinje image based on the PCCR method. b) Real-time eye fixation position calculation in the pixel coordinate system of the scene camera. c) Testing results of gaze estimation. d) Error distribution of estimated points.
2.2. Visual Feedback

The visual feedback system aims to allow users to know which object is being watched by the machine. To implement this, a laser-based system is developed. When the eye-tracking system determines the user’s focus, the laser beam will be pointed to the target. Below, the visual feedback system is explained.

2.2.1. Hardware Architecture

The laser-based visual feedback system consists of a laser, a horizontal gear, a vertical gear, and a scene camera (the one integrated in the eye-tracking system). The axis gears allow the laser to be rotated freely in two directions. The scene camera is used to build up a pixel coordinate system for the objects and the laser spot, which is necessary for the gear to control the laser’s orientation. The whole system setup is shown in Figure 4a.

The eye-tracking system first provides a location to the gears, based on the pixel coordinate system, and then the gear will adjust the laser spot to the desired position. To successfully implement this, the gear needs to know the current position of the laser spot, hence, a laser spot detection method is developed, which is explained below.

2.2.2. Laser Spot Detection

The coordinates of the laser spots should be real-time transmitted to the computer during the tracking process, indicating that both dynamic and static laser spot detection should be supported. Considering that the characteristics of laser spots under these two statuses are different, two algorithms are developed.

When the laser spot is static, it shows the following characteristic: the brightness decreases (normally following a Gaussian curve) from the center to the margin. This kind of laser spot can be extracted from the background image by using the watershed transform (WST), as explained below: 1) Perform binarization operation on the image based on continuous stepping thresholds. 2) Extract the connected regions formed under each threshold based on the contour finding algorithm, and the obtained connected regions are considered as spot candidates. 3) Set a selection standard, for spot candidate whose parameters (area, circularity, and inertia ratio) meet the standard, calculate its center pixel coordinate.

For dynamic laser spots, their circularities and inertia ratios will change greatly, which cannot be captured by the static spot detection algorithm. Therefore, an interframe difference based algorithm is proposed to extract the high-speed moving laser spot, by comparing the pixel changes between successive frame images. For a pixel position \((i, j)\) in the scene image, its grayscale values in frames \(k\) and \(k + 1\) are \(G_k(i, j)\) and \(G_{k+1}(i, j)\), and then a subtraction is performed as shown in Equation (2).

\[
D_k(i, j) = \begin{cases} 0 & G_{k+1}(i, j) - G_k(i, j) > T \\ 1 & G_{k+1}(i, j) - G_k(i, j) \leq T \end{cases}
\]

where \(T\) is the threshold value for binarization processing of different images. The movement of laser spot will make the gray value of some pixels change greatly in two successive frames of images, so we can get the position of the dynamic laser spot by extracting the coordinates of the points with strong gray-level change. To suppress the noise interference of the background environment, we experimentally determine the value of \(T\). Through many experiments, it is found that when \(T = 80\), the dynamic laser spot location achieves an excellent effect. Figure 4b shows the positioning and tracking results of static and dynamic laser spots.

The laser spot location information is then conveyed to the steering gear pan-tilt system, for further positioning the laser orientation.

**Figure 4.** a) Laser-based visual feedback system. b) Positioning and tracking results of static spot and dynamic spot. c) PID-based visual pan-tilt control method.
2.2.3. Steering Gear Pan-Tilt Control

The servo control system receives the laser spot location \((x_i, y_i)\) and fixation point position at the pixel coordinate \((x_f, y_f)\) from the computer by using a STM32 interface. As the spatial relationship between the scene camera and the steering gear pan-tilt is not fixed, the error between two locations in the pixel coordinate cannot be compensated by using a fixed mapping matrix. Hence, a PID control algorithm is developed to align the laser spot to the fixation point. The conceptual depiction of the servo control system is given in Figure 4c.

The projection of the 2D distance (which is referred to as the input error in the PID technology) of the two points in the pixel coordinate system can be expressed as

\[
\begin{align*}
\varepsilon_x &= x_f - x_i \\
\varepsilon_y &= y_f - y_i
\end{align*}
\]

(3)

where \(\varepsilon_x\) and \(\varepsilon_y\) indicate the difference between the two points at \(x\)- and \(y\)-axis, and they are employed as the input of PID control, to minimize the error between the laser spot and fixation point.

PID control algorithm involves three paralleled processing steps, which are proportion, integration and differentiation.\(^{[10]}\) Among them, the proportion controls the adjustment step length, the integration affects the steady-state error, and the differentiation improves the dynamic response of the system. As the fixation point is merely slightly altered when the user stares at the target object, hence the differentiation step is not used.

In this article, the steering gear is controlled by pulse width modulation (PWM), in which the angle of the actuator is controlled by the duty cycle.\(^{[40,41]}\) As we need to adjust the orientation at a 2D surface, PWM is actually an \(1 \times 2\) vector \([PWM_x, PWM_y]\), each element can control the steering gear to rotate from 0° to 180°. In our system, we set 200 values for each direction, indicating that the gear’s rotation resolution is 0.9°. As explained above, the distance between the eye tracker and steering gear is dynamically changed. Hence, we cannot use a fixed PWM vector. Alternatively, we develop an adaptive tracking algorithm as below

\[
PWM_{i_{\text{new}}} = PWM_{i_{\text{pre}}} + u_i \quad (i = x, y)
\]

(4)

where \(PWM_{i_{\text{new}}}\) and \(PWM_{i_{\text{pre}}}\) are the PWM vector of the next move and previous move, \(u_i\) is the change of PWM for the next move, whose value is given by

\[
u_i = \begin{cases} 
kp_1 \varepsilon_i & |\varepsilon_i| > \varepsilon_M \\
kp_2 \varepsilon_i + k_i \sum_{j=0}^{i} \varepsilon_j & \varepsilon_m < |\varepsilon_i| < \varepsilon_M \\
0 & |\varepsilon_i| < \varepsilon_m
\end{cases}
\]

(5)

where \(kp_1\) and \(kp_2\) are proportion coefficients, \(k_i\) is integration coefficient, \(\varepsilon_M\) and \(\varepsilon_m\) are maximum and minimum thresholds. When \(|\varepsilon_i|\) is greater than \(\varepsilon_M\), the proportional with a larger coefficient is used for control. When \(\varepsilon_m < |\varepsilon_i| < \varepsilon_M\), small proportion control and integration control are used to accurately approach the target. When \(|\varepsilon_i| < \varepsilon_m\), servo control is stopped.

In the proposed system, we use a total of 200 numbers to control the steering engine’s rotation angle from 0° to 180°, and experimentally determine \(kp_1\), \(kp_2\), and \(k_i\) to be 0.05, 0.01, and 0.003, respectively. With the above parameters, the tracking path diagrams and the coordinate error curves of six randomly chosen initial points (end point is the coordinate \((320, 240)\)) are shown in Figure 5a.

The error between the laser spot and the target point after tracking is within 3 pixels for both \(x\)- and \(y\)-axis. Errors from 50 experiments are given in Figure 5b. Since the scene camera is of 100° wide angle and the image resolution is \(640 \times 480\), when the operating distance is 2 m, 3 pixels will result in 0.5 cm mismatch, which is neglectable for daily use. The experimental results show that the laser tracking system controlled by the steering gear pan-tilt performs well in operation speed and accuracy.

From eye motion detection to laser beam positioning, there will be inevitable system error, indicating that the laser spot may surround the targets, instead of directly appear at them. In such cases, users can employ fine-tuning function to control the laser spot to move to the targets. More specifically, when the laser spot remains stationary after tracking, the user can know the focusing position determined by the machine. If the user
believes that the machine’s vision is different from his/hers, the user can employ eye control-based fine-tuning function, before the desired actions (e.g., turn on a light) which are also enabled by eye control.

2.3. Eye Control

Eye control function consists of eye blinking detection and eye status classification. The former is used for choosing operation mode while the latter is employed for laser spot location adjustment.

2.3.1. Blink Detection

When a person blinks, the change of pupil area is positively associated with the open degree of eyes.[42] Thus, we can detect blinks by monitoring the pupil area $S$, which is calculated as below: 1) Grayscale and blur eye image first, and then enhance image contrast by using the adaptive threshold algorithm based on neighborhood average.[27] 2) Extract the pupil edge with the Canny operator.[28] 3) Morphological expand the eye image and connect the edge of the unclosed pupil by using a $3 \times 3$ matrix. 4) Find the maximum closed-loop contour and $S$ is defined as the number of pixels surrounded by its edge.

In each frame, we can have a $S$ value. When consecutive frames are obtained during monitoring eye images, a $S$ curve can be depicted to describe the change of $S$ over time. We first search each regional minimum point of the $S$-curve, and then eliminate interruption points by setting up a reasonable area threshold of the closed pupil ($T$) and a time interval between adjacent blinks ($I$).[43] Considering that $S_{\text{close}}$ will not exceed 1000 and blinking usually lasts about 150–300 ms,[44] each eye blink can be accurately identified by setting $T$ to 1000 and $S$ to 0.3, as shown in Figure 6a.

Blinking events can be divided into conscious blinking and unconscious blinking,[45] indicating the two blinking types need to be differentiated during HMI, which can avoid “Midas touch” problems.[46] In this article, the time interval between blinks is used for recognizing conscious blinks. As intervals between unconscious blinks are normally longer than 1 s, we treat blinks with intervals less than 1 s as conscious ones.

2.3.2. Eye Status Classification

The eye blink-based interaction offers limited information which gives rise to difficulties in complex interactive scenarios. To address this issue, six eye statuses (i.e., up, down, left, right, normal, and close, as shown in Figure 6b). As different people possess diverse eye status characters (e.g., some patients suffer from strabismus), eye status detection should be customized to specific users. Therefore, a deep learning method based on convolutional neural networks (CNN) is developed.[47,48] It should be noted that the amount of training data for eye status classification is much less than fixation estimation, hence neural network-based technique will not bring inconvenience to patients during practical use.

First, we take 2100 eye images (resolution at $640 \times 480$) of the same subject at an indoor environment for six eye statuses (350 for each). Before feeding the images into the CNN, a preprocessing step is taken to release the calculation workload, by converting the collected RGB images into grayscale and normalizing them to a fixed resolution of $64 \times 64$. For enlarging the size of datasets while not increasing subjects’ burden, a data upgrading method is used through random rotation (with a range of $\pm 20^\circ$), shift (with a range of $\pm 0.2$ image width/height), and contrast (with an exponential factor between 0.25 and 4) operations. After data upgrading, in total $8400$ images are obtained, which are divided into training and testing groups, with the ratio of 7:3.

The structure of CNN is shown in Figure 6c. The network used here consists of eight layers: one input layer, four convolution layers, two pooling layers, one full connection layer, and one output layer.

We use the softmax classifier to predict the probability $p_k$ ($k = 1,2,\ldots,6$) that the eye image belongs to each status by the following formula

$$p_k = \frac{\exp(w_i^T x)}{\sum_{j=1}^{6} \exp(w_j^T x)}$$  \hspace{1cm} (6)

where $w_i$ is the parameter learned by the back-propagation algorithm.[49] Softmax defines the loss function ($L$) in terms of cross-entropy as

$$L = - \sum_{i=1}^{n} \sum_{j=1}^{k} t_{ij} \ln y_{ij}$$  \hspace{1cm} (7)

where $t_{ij}$ represents the true probability that sample $i$ belongs to the category $j$, and $y_{ij}$ represents the prediction probability of the model. The training objective is to minimize the loss of cross-entropy. Since the loss function is convex, the momentum gradient descent method is employed for training optimization. It calculates an exponentially weighted average of the gradients and then uses it to update the weights.[50,51] When the optimization is completed, CNN outputs the status of the eye image in real time according to the maximum probability.

The algorithm is used to train the model for 40 iterations, the classification accuracy, loss curves, and confusion matrix of the dataset are shown in Figure 6d,e. The accuracy of eye status classification is 98.84%, while the cross-entropy loss value is 0.0469. The results show that CNN can recognize eye status images effectively.

To avoid “Midas touch,” we develop an interactive method based on eye status detection, to map four different eye status combinations into corresponding control commands. The mapping relationship is given in Table 2, in which $C_S$ represents the initial eye status, $C_M$ the transition eye status, and $C_E$ the end eye status of the behavior. When a certain eye status combination is detected, the corresponding control command will be triggered, and the fixation point in the pixel coordinate system of scene camera will translate 5 pixels to the desired direction accompanied by the automatic tracking of laser.

The system described above needs to cooperate with action functions in practical scenarios. Below in this article, we employ...
a robotic arm developed in our previous work for collaborating with the visual feedback system to grasp objects for users. To achieve this, a subject will use the fixation detection function, laser feedback function, and eye control function explained above. More specifically, after the system detects the fixation point of the subject, a laser beam will be pointed in that direction; and then the user can recognize if the machine’s vision is the same as his/hers or not. If not, eye control-based laser beam fine-tuning will be enabled. Finally, the robotic arm will grab the object when the subject satisfies with the position of the laser spot.

Table 2. Correspondence between eye status transition and pan-tilt control.

| Control instruction | Eye status | Left | Right | Up | Down | Normal | Close |
|---------------------|------------|------|-------|----|------|--------|-------|
| Translate left      |            | C_r  |       |    |      |        |       |
| Translate right     | C_r        |      |       |    |      |        |       |
| Translate up        |            |      |       |    |      |        | C_r   |
| Translate down      |            |      |       |    |      |        | C_r   |

Figure 6. a) Relationship between the pupil area and eye statuses for detecting eye blinks. b) Conceptual depiction of six eye statuses (normal, right, left, up, down, close). c) Architecture of the CNN eye status classifier model. d) Loss and accuracy for training and validation. e) Confusion matrix for 6 classes.
3. Experimental Section

For evaluating the effect of visual feedback, a robotic arm-based human–machine cooperation system is constructed, which two experiments are designed and a questionnaire study is performed. The overall framework of the interaction is shown in Figure 7.

3.1. Environment Construction

All experiments are carried out in the laboratory environment. The entire system includes an eye tracker, a steering gear pan-tilt, a robotic arm, and a computer. The robotic arm is installed on the desk, while other components are integrated with a wheelchair. Four blocks representing routine objects are distributed on the desk. During the experiment, the subjects sit on the wheelchair. The whole setup is shown in Figure 8a.

3.2. Task Design

The subject needs to complete the task of grasping the specified color block, using an eye movement interaction system with and without visual feedback. When visual feedback is not in use, laser feedback, and eye status fine-tuning functions are also disabled. Three distributions based on different spacing distances between blocks are designed: case 1 (>10 cm), case 2 (5–10 cm), and case 3 (<5 cm). The aim of using three different distance ranges is to evaluate the performance of the developed technique in diverse scenarios.

3.3. Experiment Protocol

Thirty healthy subjects (details of subjects are given in Table 3) at Beihang University are recruited. Before the formal experiment, all subjects are required to be familiar with the operation method of the system, and the system is calibrated for each subject. All provided written informed consent, and all study procedures were approved by the Committee for Medical Research Ethics at the First Hospital of Shijiazhuang City, China, with the assigned project number of 2020036.

The procedures of the two experiments are described as follow:

3.3.1. Experiment I

1) Grab the blocks from left to right in sequence, with and without the visual feedback system; 2) After successfully grasping all four blocks, take 5 min rest; 3) Repeat step 2 for 10 times; Adjust the block spacing and repeat steps 2 to 4; Conduct a predesigned questionnaire survey.

3.3.2. Experiment II

1) Set the block distribution to case 3; 2) The subject operates the robotic arm to grab the block freely by using the system with and without visual feedback; 3) After all four blocks are successfully grabbed, refresh blocks; 4) Repeat steps 2 and 3 contentiously for grabbing blocks as many as possible until they report suffering fatigue; 5) Record the number of successfully grabbed blocks every 3 min; 6) Conduct a questionnaire survey.

3.4. Results and Discussion

In this section, quantitative (interaction time and success rate) and qualitative (subjective feeling represented by questionnaire survey score) results are provided to comprehensively evaluate the influence of the visual feedback technique in human–machine collaboration. The interaction time is defined as the average time used to complete a block grabbing action successfully, while the success rate is the number of successful attempts over all attempts.

![Figure 7. Block diagram of the visual feedback and robotic arm-integrated intelligent interactive system.](image-url)
3.4.1. Experiment I

The interaction time and success rate of the 30 subjects (in three different block placement scenarios) are shown in Figure 8b,c. It can be seen that when the blocks are placed in case 1, the introduction of visual feedback will increase the additional adjustment time by 26.8%, but the improvement of interaction accuracy is not significant. In contrast, when the blocks are placed in case 3, visual feedback can boost the success rate by 27.6% and save 8.0 s of interaction time. The experimental results yield that, small spacing can give rise to a high possibility of misoperation when visual feedback is disabled.

From experimental results, we also learn two phenomena. First, background diversity does not demonstrate a significant statistical difference in interaction time and operation accuracy. Second, age has a strong influence in the above mentioned two indexes. One example is given in Figure 8d,e, which depicts the data of the case 3 of Experiment I. It is observed that senior people performed not as good as students in terms of both interaction time and accuracy, while students in different disciplines
have close operation abilities (details of operation time and accuracy are provided in Table 4). Nevertheless, it should be noted that the senior are satisfied with their performance and the user experience with the system, through questionnaire answers below.

In terms of the questionnaire survey I shown in Table 5, the average score of all problems with visual feedback eye movement interaction system is 3.95, which is 0.86 points higher than that of without using the visual feedback system. From the result, we find that the subjects agree that visual feedback plays a positive role in the eye movement interactive system. This is because without visual feedback, the invisibility of system state increases subjects’ doubts and cognitive burden. In contrast, the subjects can get a clear gaze position through visual feedback during the interaction process, which provides confidence of interactive operation for users. The results also show that the subjects generally believed that the eye movement interaction system with visual feedback can better meet the comfort needs of users and reduce user’s fatigue by 25.1%.

3.4.2. Experiment II

The number of blocks captured by the subjects in different time periods is shown in Figure 8d. Our findings suggest that visual feedback can eliminate the fatigue of the subjects to a certain extent, so that they can concentrate longer time in the interactive experiment, improving both efficiency and accuracy. Considering senior people’s physical conditions are not as good as students, they are prone to suffering fatigue, explaining the curve trends in Figure 8f.

From the questionnaire survey II, we can almost make the same conclusion as the questionnaire survey I. However, it demonstrates that under pressure, users are more preferred to have a visual feedback function.

During both experiments, we find out that the frequency of using fine-tuning function is highly user dependent. Through the developed technique, once the laser spot is shown at the object surface, and then the vision-based technique can recognize the object’s position (this has been informed to the subjects). Nevertheless, some users prefer to align the laser to the center position of the objects, while others may stop the fine-tuning step after seeing the laser spot at the edge of desired objects. From our record, for successfully grabbing an object averagely 2.75 times fine-tuning actions need to be taken, with a variance of 1.03 times.

Table 3. The detail information about the background of 30 subjects.

| Major Identity | Number of people | Age distribution |
|----------------|------------------|------------------|
| Instrumentation and Optoelectronic Engineering | Undergraduate | 8 | 18–22 |
| Senior faculty member | 5 | 52–59 |
| Electronic and Information Engineering | Undergraduate | 2 | 19 |
| Graduate | 1 | 23 |
| New Media Art and Design Humanities and Social Sciences | Undergraduate | 6 | 19–21 |
| Undergraduate | 4 | 18–21 |

Table 4. The success rate and interaction time of the subjects with different identities.

| Experiment | Success rate/interaction time [s] |
|------------|----------------------------------|
|            | Engineering students | Liberal arts | Senior faculty members |
| Case 1     | With visual feedback | 68.9%/18.3 | 68.1%/17.9 | 62.2%/22.5 |
|            | Without visual feedback | 62.8%/14.3 | 63.2%/13.7 | 57.7%/18.9 |
| Case 2     | With visual feedback | 64.1%/20.3 | 65.3%/20.1 | 58.3%/25.6 |
|            | Without visual feedback | 42.5%/25.4 | 40.3%/24.2 | 36.1%/31.2 |
| Case 3     | With visual feedback | 55.4%/28.4 | 56.9%/28.8 | 49.1%/35.1 |
|            | Without visual feedback | 28.1%/36.2 | 28.4%/36.4 | 22.1%/44.4 |

Table 5. Questionnaire. (The scoring range is 1–5. The higher score, the better interaction experience.).

| Questions | Mean value with feedback (score I/score II) | Mean value without feedback (score I/score II) |
|-----------|-------------------------------------------|-----------------------------------------------|
| How much confidence did you have when grasping the cube block? | 4.16/4.01 | 3.21/2.85 |
| How effective did you think when grasping the cube block? | 3.90/3.78 | 3.23/3.03 |
| How stable did you think when grasping the cube block? | 4.54/4.63 | 3.61/2.52 |
| How comfortable did your eyes feel when controlling the interactive system? | 3.65/3.70 | 3.33/2.71 |
| How much fatigue did you feel when controlling the interactive system? | 3.38/2.91 | 2.56/2.15 |
| How easily did you control the interactive system? | 3.02/3.29 | 3.83/3.45 |
| How would you evaluate your satisfaction with the system interaction? | 4.29/4.34 | 3.90/3.01 |
| How practical do you think the interactive system is? | 4.89/4.68 | 2.88/3.11 |
4. Conclusion

ALS patients suffer from unimaginable mental pain, hence receive increasing attentions and cares. The work presented in this article showcases a feasible means for ALS patients to effectively communicate with surrounding intelligent devices and systems. Compared with conventional desktop eye-tracking and electroencephalogram (EEG)-based techniques, the proposed visual feedback-supported intelligent system can provide an enhanced user experience, indicating its strong potential in releasing ALS patients' mental stress and making their daily lives convenient.

Acknowledgements

This work was supported in part by the National Natural Science Foundation under grant 61803017 and grant 61828702, and in part by the Beihang University under grant KG12090401 and grant ZG216S19C8 and in part by the Beihang University under First-class Undergraduate Course (Digital Image Processing) Construction Project.

Conflict of Interest

The authors declare no conflict of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Keywords

amyotrophic lateral sclerosis patients, eye movement tracking, human–machine interactivity, intelligent assistive system, visual feedback

Received: May 17, 2021  
Revised: November 18, 2021  
Published online: December 10, 2021

[1] L. I. Grad, G. A. Rouleau, J. Ravits, N. R. Cashman, Cold Spring Harbor Perspect. Med. 2017, 7, a024117.
[2] B. J. Traynor, M. Alexander, B. Corr, E. Frost, O. Hardiman, J. Neurol. Neurosurg. Psychiatr. 2003, 74, 1258.
[3] Z. Simmons, Neurotherapeutics 2015, 12, 394.
[4] R. Rahnama-ye-Moqaddam, H. Vahdat-Nejad, in 2015 5th Int. Conf. on Computer and Knowledge Engineering, IEEE, Piscataway, NJ 2015, pp. 218-221.
[5] D. R. Lingegowda, K. Amrutesh, S. Ramanujam, in 2017 IEEE Int. Conf. on Consumer Electronics-Asia, IEEE, Piscataway, NJ 2017, pp. 36-40.
[6] S. Gao, Y. Dai, A. Nathan. Adv. Intell. Syst. 2021, 2100074, https://doi.org/10.1002/asi.2100074.
[7] C. J. Lin, Y. T. Prasetyo, R. Widyaningrum, J. Eye Mov. Res. 2018, 11, 3.
[8] R. J. Jacob, K. S. Karn, in The Mind’s Eye (Eds.: J. Hyönä, R. Radach, H. Deubel), North-Holland, 2003, pp. 573-605.
[9] C. S. Hwang, H. H. Weng, L. F. Wang, C. H. Tsai, H. T. Chang, J. Mot. Behav. 2014, 46, 233.
[10] K. Linse, W. Ruger, M. Joos, H. Schmitz-Peiffer, A. Storch, A. Hermann, Ann. Neurol. 2017, 81, 310.
[11] A. Pessia, A. Ilardi, V. Pasian, C. Moglia, S. Giacone, S. Cammarosano, A. Montuschi, A. Canosa, S. Gallo, A. Calvo, F. Corno, A. Chio, Eur. J. Neurol. 2010, 17, 303.
[12] A. Calvo, A. Chio, E. Castellina, F. Corno, L. Farinetti, P. Ghiglione, V. Pasian, A. Vignola, in 11th Int. Conf. on Computers Helping People with Special Needs, Springer, Linz 2008, pp. 70-77.
[13] L. Wöhle, M. Gebhard, Sensors 2021, 21, 1798.
[14] C. Jyotsna, J. Armudha, in 7th Int. Conf on Advances in Computing, Communications and Informatics, IEEE, Bangalore 2018, pp. 1588-1593.
[15] X. Wang, A. H. Fathaliyan, V. J. Santos, Front. Neurorobot. 2020, 14, 567571.
[16] E. R. Jones, W. Chinthammit, W. Huang, U. Engelke, C. Lueg, Symmetry 2018, 10, 680.
[17] G. T. Zhang, J. P. Hansen, K. Minakata, A. Alapetite, Z. Y. Wang, in HRI 2019–14th ACM/IEEE Int. Conf. on Human-Robot Interaction, IEEE, Daegu 2019, pp. 574-575.
[18] M. A. Eid, N. Giakoumidis, A. El. Saddik, IEEE Access 2016, 4, 558.
[19] Y. M. Cheung, Q. M. Peng, IEEE Trans. Hum.-Mach. Syst. 2015, 45, 419.
[20] C. H. Morimoto, D. Koons, A. Amir, M. Flickner, Image Vis. Comput. 2000, 18, 331.
[21] J. N. Chi, P. Y. Zhang, S. Y. Zheng, C. Zhang, Y. Huang, in 2009 WRI Global Congress on Intelligent Systems, IEEE, Xiamen 2009, pp. 133-138.
[22] N. M. Arar, H. Gao, J. P. Thiran, in 2015 15th IEEE Winter Conf. on Applications of Computer Vision, IEEE, Waikoloa 2015, pp. 642-648.
[23] D. H. Yoo, M. J. Chung, Comput. Vis. Image Underst. 2005, 98, 25.
[24] T. Harada, H. Iwasaki, K. Mori, A. Yoshizawa, F. Mizoguchi, in 12th IEEE Int. Conf. on Cognitive Informatics and Cognitive Computing, IEEE, New York 2013, pp. 428-434.
[25] J. Lemley, A. Kar, A. Drimbarean, P. Corcoran, IEEE Trans. Broadcast Telev.Receivers 2019, 65, 179.
[26] C. Lu, P. Chakravarthula, Y. Tao, S. Chen, H. Fuchs, in 19th IEEE Int. Symp. on Mixed and Augmented Reality, IEEE, Virtual 2020, pp. 320-331.
[27] A. Youssif, A. Darwish, A. M. Madbouly, IFCSNS Int. J. Comput. Sci. Netw. Security 2010, 10, 322.
[28] W. Fuhl, M. Tonsen, A. Bulling, E. Kasneci, Mach. Vis. Appl. 2016, 27, 1275.
[29] A. Al-Rahayfeh, M. Faezipour, in 9th Annual Conf. on Long Island Systems, Applications and Technology, IEEE, Farmingdale 2013, pp. 1-6.
[30] W. Fuhl, T. C. Santini, T. Kubler, E. Kasneci, in 9th Biennial ACM Symposium on Eye Tracking Research and Applications, ACM, Charleston 2016, pp. 123-130.
[31] J. Park, T. Jung, K. Yim, in 9th Int. Conf. on Innovative Mobile and Internet Services in Ubiquitous Computing, IEEE, Blumenau 2015, pp. 431-435.
[32] A. T. Duchowski, B. Pelfrey, D. H. House, R. Wang, in 8th Annual Symposium on Applied Perception in Graphics and Visualization, ACM, Toulouse 2011, pp. 15-22.
[33] P. Blignaut, K. Holmqvist, M. Nyström, R. Dewhurst, Current Trends in Eye Tracking Research, Springer, Cham 2014.
[34] D. B. Liston, S. Simpson, L. R. Wong, M. Rich, L. S. Stone, in 9th Biennial ACM Symp. on Eye Tracking Research and Applications, ACM, Charleston 2016, pp. 221-224.
[35] D. Krstinic, A. K. Skelin, I. Milatic, Sensors 2014,14, 20112.
[36] C. L. Sun, X. M. Wang, in Proc. 2010 3rd IEEE Int. Conf. on Computer Science and Information Technology, IEEE, Chengdu 2010, pp. 63-66.
[37] Z. Q. Cao, D. Y. Sun, W. Y. Huang, Z. H. Wang, Y. Q. Zhou, W. H. Gu, in 10th Int. Conf. on Communications, Circuits and Systems, IEEE, Chengdu 2018, pp. 387-391.
[38] Y. H. Cheng, J. Wang, in 2013 2nd Int. Conf. on Mechanical Design and Power Engineering, TTP, Beijing 2014, pp. 1283-1286.

[39] J. Q. Yang, D. Y. Wang, W. H. Zhou, Optik 2017, 131, 994.

[40] X. H. Lv, Y. X. Liu, in 8th World Congress on Intelligent Control and Automation, IEEE, Jinan 2010, pp. 3737-3740.

[41] M. N. Rudzuan, D. Hazry, W. A. N. Khairunizam, A. B. Shahriman, M. Saifizi, M. N. Ayob, Proc. Eng. 2012, 41, 189.

[42] C. W. Cheng, W. L. Ou, C. P. Fan, in 2016 IEEE Int. Conf. on Consumer Electronics, IEEE, Las Vegas 2016, pp. 445-446.

[43] K. Grauman, M. Betke, J. Gips, G. R. Bradski, in 2001 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, IEEE, Kauai 2001, pp. I1010-I1017.

[44] A. Krolak, P. Strumillo, Univers. Access Inf. Soc. 2012, 11, 409.

[45] Y. Sugano, Y. Matsushita, Y. Sato, H. Koike, IEEE Trans. Hum.-Mach. Syst. 2015, 45, 750.

[46] R. J. K. Jacob, in Proc. of the SIGCHI Conf. on Human Factors in Computing Systems, ACM, Seattle, 1990, pp. 11-18.

[47] A. George, A. Routray, in 11th Int. Conf. on Signal Processing and Communications, IEEE, Bangalore 2016, pp. 1-5.

[48] M. M. Rahman, M. S. Islam, M. K. A. Jannat, M. H. Rahman, M. Arifuzzaman, R. Sassi, M. Aktaruzzaman, in 22nd Int. Conf. on Advanced Communications Technology, IEEE, Pyeongchang 2020, pp. 84-90.

[49] X. Z. Liang, X. B. Wang, Z. Lei, S. C. Liao, S. Z. Li, in 24th Int. Conf. on Neural Information Processing, Springer, Guangzhou 2017, pp. 413-421.

[50] D. P. Kingma, J. L. Ba, in 3rd Int. Conf. on Learning Representations, ICLR, San Diego 2015.

[51] Y. Cao, H. Wang, F. Jing, J. Han, M. Xie, in 2018 Int. Conf. on Electronics Technology, IEEE, Chengdu 2018, pp. 387-391.