Indian Journal of Computer Science and Technology
https://www.doi.org/10.59256/indjcst.20240302016
Volume 3, Issue2 (May-August 2024), PP: 116-122.
www.indjcst.com
ISSN No: 2583-5300

Low-Cost Eye Gesture Communication System for People with Motor Disabilities

S. Ramya[1], Harish D[2], Tharun M[3], Kesavan SA[4], Thirumoorthy J[5]

[1]*Assistant Professor, Department of Information Technology, Er. Perumal Manimekalai College of Engineering, Hosur, Tamil Nadu, India.*

[2,3,4,5] *Department of Information Technology, Er. Perumal Manimekalai College of Engineering, Hosur, Tamil Nadu, India.*

To Cite this Article: *S. Ramya[1], Harish D[2], Tharun M[3], Kesavan SA[4], Thirumoorthy J[5], "Low-Cost Eye Gesture Communication System for People with Motor Disabilities", Indian Journal of Computer Science and Technology, Volume 03, Issue 02 (May-August 2024), PP:116-122*

Abstract: *Current eye-tracking input systems for people with ALS or other motor impairments are expensive, not robust under sunlight, and require frequent re-calibration and substantial, relatively immobile setups. Eye-gaze transfer (e-tran) boards, a low-tech alternative, are challenging to master and offer slow communication rates. To mitigate the drawbacks of these two status quo approaches, we created Gaze Speak, an eye gesture communication system that runs on a smartphone, and is designed to be low-cost, robust, portable, and easy-to-learn, with a higher communication bandwidth than an e-tran board. Gaze Speak can interpret eye gestures in real time, decode these gestures into predicted utterances, and facilitate communication, with different user interfaces for speakers and interpreters. Our evaluations demonstrate that GazeSpeak is robust, has good user satisfaction, and provides a speed improvement with respect to an e-tran board; we also identify avenues for further improvement to low-cost, low-effort gaze-based communication technologies.*

Key Words: *Eye gesture; accessibility; augmentative and alternative communication (AAC); Amyotrophic lateral sclerosis (ALS).*

I.INTRODUCTION

Eye gaze keyboards [21, 28] are a common communication solution for people with Amyotrophic Lateral Sclerosis (ALS) and other motor impairments. ALS is a neurodegenerative disease that leads to loss of muscle control, including the ability to speak or type; because eye muscle movement is typically retained, people with late- stage ALS usually rely on eye tracking input for communication. Unfortunately, the hardware for commercial gaze-operated keyboards is expensive. For example, the poplar Tobii Dyanvox [28] eye gaze hardware and software package costs between $5,000 and $10,000, depending on the specific model and configuration. Additionally, eye trackers do not work in certain conditions that interfere with infrared light (IR) (such as outdoors), and require a stand to keep the apparatus relatively static with respect to the user, which makes it difficult to use in certain situations such as in a car or in bed.

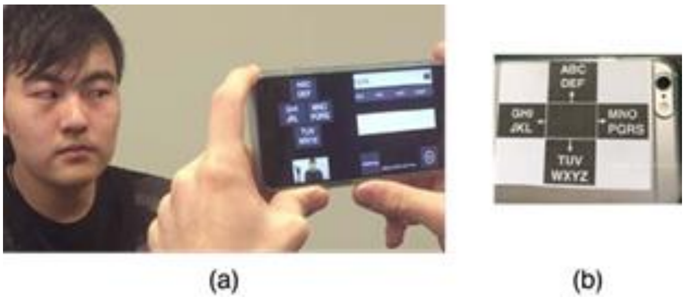

Figure 1 Communication Partner

The communication partner of a person with motor disabilities can use the Gaze Speak smartphone app to translate eye gestures into words. (a) The interpreter interface is displayed on the smartphone's screen. (b) The speaker sees a sticker depicting the four letter groupings affixed to the phone's case.

Eye-gaze transfer (e-tran) boards [3,17] are an alternative, low-tech communication solution, where clusters of letters are printed on a transparent plastic board. The communication partner holds the board, and observes the gaze pattern of the person with ALS (PALS), who selects a letter by making two coarse eye gestures: one to select which of several letter groupings contain the target letter, and a second to indicate the position within the group. Unfortunately, e-tran boards have several drawbacks: their cost is relatively low compared to gaze-tracking systems, but is not negligible (~$100); the large plastic board (e.g., one popular model [17] measures 14″ x 18″) is not easily portable; patients need to perform two eye gestures to enter one letter, which may take more than 8 seconds [25] including correcting mistakes; our survey of PALS' caregivers indicated they found e-tran to have a high learning curve, as they have to decode and remember entered characters and predict words.

In this work, we investigate how to provide low-cost, portable, robust gaze-based communication for PALS that is easy for patients and caregivers to use. Our solution uses a smartphone to capture eye gestures and interpret them using computer vision techniques. Our system, Gaze Speak, consists of computer-vision-based eye gaze recognition, a text prediction engine, and text entry interfaces that provide feedback to the speaker (PALS) and interpreter (caregiver or communication partner). Gaze Speak is as portable as a mobile phone, patients perform only one eye gesture per character, and the mobile phone records the entered characters and predicts words automatically. Further, Gaze Speak does not require re-calibration under similar lighting conditions, and it does not require a bulky stand, as caregivers can hold the phone. Gaze Speak has no additional cost other than a smartphone, which most people in the U.S. (68% in 2015) [2] already own. We evaluate the error rate of our eye gesture recognition, as well as satisfaction and usability for both the speaker and interpreter. We also compare the communication speed of Gaze Speak to that of an e-tran board. We find Gaze Speak is robust, has good user satisfaction, and provides a speed improvement with respect to e-tran. Gaze Speak offers a viable low-cost, portable, lighting-robust alternative for situations in which eye-tracking systems are unaffordable or impractical.

The specific contributions of this work include:

- The Gaze Speak system, including algorithms to robustly recognize eye gestures in real time on a hand-held smartphone and decode these gestures into predicted utterances, and user interfaces to facilitate the speaker and interpreter communication roles.
- User study results demonstrating Gaze Spaak's error rate and communication speed, as well as feedback from PALS and their communication partners on usability, utility, and directions for further work.

II.RELATED WORK

Low-Tech Gaze Input Solutions

As shown in Figure 2, an e-tran board [3,17] is a low-tech AAC (augmentative and alternative communication) solution that comprises a transparent board containing groups of symbols, such as letters. An interpreter holds the board and observes and decodes the eye gestures of the speaker; the speaker gazes in the direction of a group to select a cluster of symbols and then again to disambiguate the location of a specific symbol within the cluster. Another low-tech solution, Eye Link [25], is also a transparent board printed with letters. To use Eye Link, the speaker keeps staring at the desired letter, while the interpreter moves the board until she can "link" her own eye gaze with that of the speaker, and note the letter on the board where their eyes meet. Families and occupational/speech therapists may also develop myriad custom low-tech communication solutions, such as attaching a laser pointer to a patient's head (if they have head movement control) that can be used to point at letters printed out on a poster or board. While relatively cheap (costing tens or low hundreds of dollars for materials), low-tech solutions provide low communication bandwidth (entering a letter takes 8-12 seconds [25]) and place a high learning/cognitive burden on the interpreter. Low tech solutions, while not optimal, are nonetheless important in offering a communication option in situations where other options are unavailable.

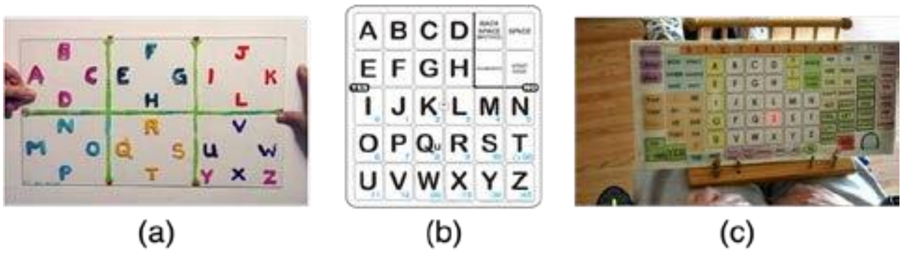

Figure 2. Low-tech gaze input solutions: a) E-Tran board; b) Eye Link; c) Laser pointer on communication board.

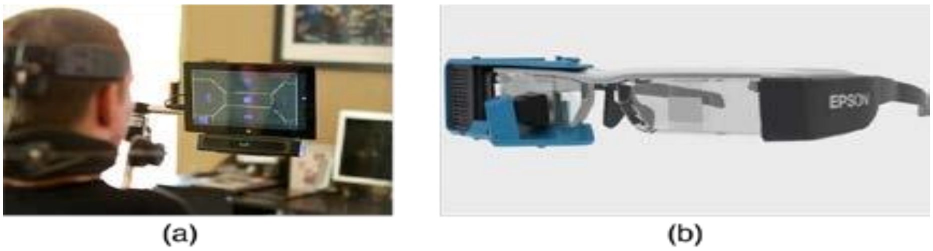

Figure 3. High-tech gaze recognition solutions: (a) Tobii Dynavox eye-tracking computer [28]; (b) Eye Speak eye- tracking glasses [18].

High-Tech Gaze Recognition Solutions

As shown in Figure 3, commercial gaze-operated keyboards allow PALS to type characters to complete a sentence [21,29], or select symbols to build sentences word by word [30]. Typically, systems are dwell-based [15,20] (i.e., the user dwells their gaze on a key for a period, typically several hundred milliseconds, in order to select that key), though dwell-free gaze systems [13,37] are an emerging area of research and commercial development that may offer further speed improvements. Other, less common, gaze input interfaces may include techniques like scanning [4] or zooming [34]. Specialized hardware and setups are used in eye gaze tracking systems. Head-mounted eye trackers [1,18] keep the eyes and the camera close and relatively static during head movement. However, such systems are expensive, and their bulk/weight is not typically comfortable for constant use as required by someone with a motor disability; head- worn systems may also interfere with eye contact and the ability to observe the environment, making them even more impractical for constant use. Other eye tracker solutions mount a camera on a computer monitor or table, and find pupil locations based on reflections. Because of the longer distance between eye and camera, commercial systems [27,28] often emit IR to increase light reflection from the eyes. IR makes eye movement more detectable, but limits outdoor usage due to interference from the strong IR in sunlight. Eye trackers range in price from hundreds to thousands of dollars depending on quality; some relatively low-cost commercial eye trackers are available (e.g., the Tobii EyeX costs around $150 USD); however, people who rely on eye gaze for AAC must also purchase a computer to connect to the eye tracker and proprietary software that allows the eye gaze to control a keyboard and other computer programs; such bundles [28] typically cost between $5,000 - $10,000. In the United States, government health insurance (Medicare) has only just begun to help pay for AAC devices [32]; some insurers do not consider access to AAC as a medical necessity.

There are also attempts at using low-cost webcams or phone cameras to recognize eye gaze location, and use direct gaze pointing as an input method. Web Gazer [24] uses a webcam to infer the gaze locations of web visitors on a page in real time, and self-calibrates while visitors interact with content on the screen with their mouse cursor. Any movement of the camera or head requires additional interactions to re-calibrate. However, PALS or other motor impairments cannot move the mouse to interact, which would break Web Gazer's self-calibration algorithm. I Tracker [12] uses an iPhone's front camera to estimate gaze location on the screen. Calibration can increase accuracy, but is not required, as it is pre-trained on a large- scale eye tracking database. However, extending this method to additional mobile devices may require collecting large eye tracking datasets for each device type. In addition, its prediction error on the iPhone 5 is almost 30% of the screen width.

Besides gaze location, eye-switches and eye gestures can also be used as an input method. Eye-switches [7] use voluntary eye blinks as binary signals, which helps in scanning input methods. In a 2D letter grid, the system moves the focus line by line, and the first eye-switch can select the line that contains the desired letter; then the system moves the focus letter by letter on that line, and the second eye-switch selects the desired letter. Eye Write [37] is the first letter-like gestural text entry system for the eyes, and uses a Tobii IR eye tracker to capture gaze input. Its letter-drawing interface is a square with four corners. The user has to move gaze to the corners to map out a letter. Eye Write does not require the dwell time to select a letter, except needing slight dwell time to signal character segmentation. Testing found Eye Write's text entry speed was around 5 wpm. Vaitukaitis and Bulling [31] presented a prototype that could recognize different continuous eye gesture patterns on a laptop and mobile phone. For example, the user can move his gaze left, then up, then right, and finally down to draw a diamond pattern. However, their evaluation was conducted in an indoor setting with controlled lighting, requiring fixed device position and distance to participants. Even with these carefully controlled conditions, the performance of this prototype was less than 5 frames per second on the phone with only 60% accuracy. In our work, we do not use compound eye gestures to draw letters or shapes; rather, we use simple, single-direction gestures (e.g., look left) to select among groups of letters.

III.DESIGN GOALS

Before we started to build Gaze Speak, we conducted an online survey targeted at communication partners (spouses, caregivers, etc.) of PALS, and advertised our survey via an email list about ALS in the Seattle metropolitan area. We received 22 responses; this low number is not surprising given the low incidence rate of ALS, about 1 in 50,000 people [26].

All of the respondents indicated owning either an iPhone or Android phone. 36% of them said their companion with ALS did not own an eye-tracking system. For those whose companion did have an eye-tracker, 28% of them reported that the PALS was unable to use the eye-tracking system during more than half of the waking hours, due to issues such as system crashes, positioning at angles or locations where mounting the system is impractical (being in bed, inclined in a chair, or using the bathroom), being in situations with limited space (such as traveling in a car), or outdoors due to interference from sunlight. They also noted that when their companions with ALS only want to convey a quick communication, the start-up costs of such systems (which respondents indicated required frequent re- calibration in practice) seemed too long in proportion to the length of the communication.

64% of respondents indicated having used e-tran boards as an alternative when eye tracking was not available. Some who had not tried e-tran boards indicated they had not done so because their companion was in earlier stages of ALS's progression and still retained some speech capabilities. The self-reported learning curve for e-tran was varied; 36% of respondents reported it took a few hours to master, 14% spent a day, 29% spent a week, and 21% indicated it took more than a month. Respondents reported challenges in using e-tran boards: the interpreter may misread the eye gesture, forget the sequence of previously specified letters, and finds the board heavy/uncomfortable to hold; for the speaker, it is difficult to correct a mistake in gesture or interpretation. Respondents indicated that one or two words is the typical length of an utterance specified via e-tran board.

These survey responses added to our knowledge of the concerns facing end-users of gaze-based AAC; based on these responses and the other cost and practicality issues discussed in the Introduction and Related Work sections, we articulated several design goals for Gaze Speak:

1) Create a low-cost/high-tech alternative in an eco-system that currently offers only high-cost/high-tech and low- cost/low-tech solutions: Gaze Speak is meant to be affordable for people who may not be able to purchase an expensive, multi-thousand-dollar eye tracking solution. Since smartphone ownership is common in the U.S. [2] and was ubiquitous among our survey respondents, a smartphone app seems a reasonable way to reach a large audience at no additional cost to end-users. We do not expect that our smartphone app should exceed the performance (in terms of text entry rates) of expensive, commercial eye tracker setups; however, we do expect that Gaze Speak will offer performance and usability enhancements as compared to the current low-cost alternative. In addition to supporting a low-cost solution, the mobile phone form-factor preserves (and even exceeds some of) the advantages

of current low-tech solutions in being smaller, lighter, more portable, and more robust to varied lighting sources than high-cost commercial eye- tracking setups.

2) Simplify the e-tran process through automation: While cheap and flexible for many scenarios, e-tran still has several drawbacks that we aim to mitigate with Gaze Speak, particularly: (1) slow speed of text entry, (2) difficulty of error correction, and (3) high cognitive burden for the interpreter.

IV.SYSTEM DESIGN AND IMPLEMENTATION

Gaze Speak application currently supports iOS devices released after 2012. It uses the phone's built-in camera, touch screen, and speaker, and does not require extra hardware. To improve learnability and usability, a simple guide representing the four keyboard groupings and associated gesture directions can be printed and taped to the back of the phone's case (Figure 1b). To use this system, the interpreter holds the phone and points the back camera toward the speaker (PALS) (Figure 1a and Figure 8). The app consists of three major components: 1) eye gesture recognition, 2) a predictive text engine, and 3) a text entry interface.

4.1 Eye Gesture Recognition

Gaze Speak can robustly recognize six eye gestures for both eyes: look up, look down, look left, look right, look center, and close eyes. If the speaker can wink (closing only one eye at a time), it also recognizes winking the left eye and winking the right eye. The recognition code is written in C++ and depends on open-source libraries: Dlib [11] and Open CV [5]. To save development time, we only implemented eye gesture recognition on iOS; however, Gaze Speak could be easily extended to other platforms that support these dependencies, such as Android, Windows, Mac and Linux.

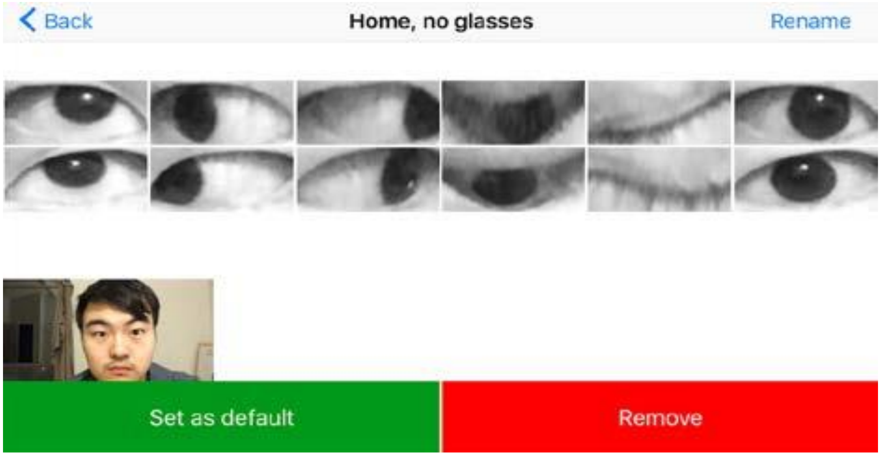

Figure 4. Calibration review screen. A photo at the bottom left serves as a reminder of context (e.g., indoors, glasses off) that can also be added to the calibration name (top). This screen also shows the calibration templates captured for the up, left, right, down, closed and center gestures for each eye.

4.2 Calibration

Gaze Speak collects a set of eye gestures from the speaker as calibration templates. When the interpreter holds the phone with the rear camera facing the speaker, they can press the "calibrate" button. The app then plays audio instructions that tell the speaker to prepare to calibrate, and then instructs him to look up, down, left, right, center, and to close both eyes. Template matching will be more robust if these six gestures are distinct from each other, so for best performance, the speaker should make eye gestures that are exaggerated (far up, to the rightmost, etc.) to the extent possible while not being uncomfortable (Figure 4). In addition, while looking down, eyelashes may cover eyes naturally, which makes it appear to be similar to closed eyes. Thus, for the look down gesture, we suggest speakers try to keep their eyes open as wide as they are able while looking down, to improve performance (Figure 4). This calibration sequence takes ten seconds. Calibration is only required for the first time using the app, or if lighting conditions vary drastically (having a separate outdoor and indoor calibration may improve performance). Calibrations taken under different circumstances can be stored, labeled, loaded as needed, and transferred between different iOS devices.

To obtain calibration templates for eye gaze recognition, Gaze Speak performs the following four steps:

1) Detect face and align landmarks: We use iOS's built-in face detector to obtain a rectangle containing the speaker's face. (Open CV's face detector could be used when extending Gaze Speak to other platforms that do not have built-in face detection support.) Then, we use dlib's implementation of fast face alignment [10] to extract landmarks on the face.

2) Extract an image of each eye: Once we get face landmarks, we calculate the bounding rectangle of eye landmarks. Then we extract images of each eye and process them separately.

3) Normalize eye images: We resize each eye image to 80x40 pixels. Then we convert the image to the HSV color space and only keep its V channel (value, i.e., brightness).

4) Store eye gesture template: We save the normalized eye images on the phone, using the filenames to indicate the eye (left/right) and gesture (up/down/left/right/center/closed).

4.3 Recognition Algorithm

Once there is a new or existing calibration for the speaker, he may perform eye gestures to communicate while the interpreter aims the rear phone camera toward the speaker (Figure 1a). The interpreter can see the camera view on the screen and ensure the speaker's face is in the view (Figure 5x). For each camera frame, Gaze Speak performs the following steps as shown in Figure 5. It detects the face and aligns landmarks, extracts an image of each eye, and normalizes the eye images (the same first three steps as during calibration). Then Gaze Speak classifies eye gestures by matching the normalized eye images extracted from the current video frame with the calibration templates. Mean squared error (MSE) [16] measures the difference between two images. We use MSE to find the closest match for the normalized eye images among the six eye gesture templates obtained during calibration. Structural similarity [33] and the sum of absolute differences [35] are also candidates for measuring the difference between two images; however, MSE achieved the best recognition rate in our iterative testing and development of the system.

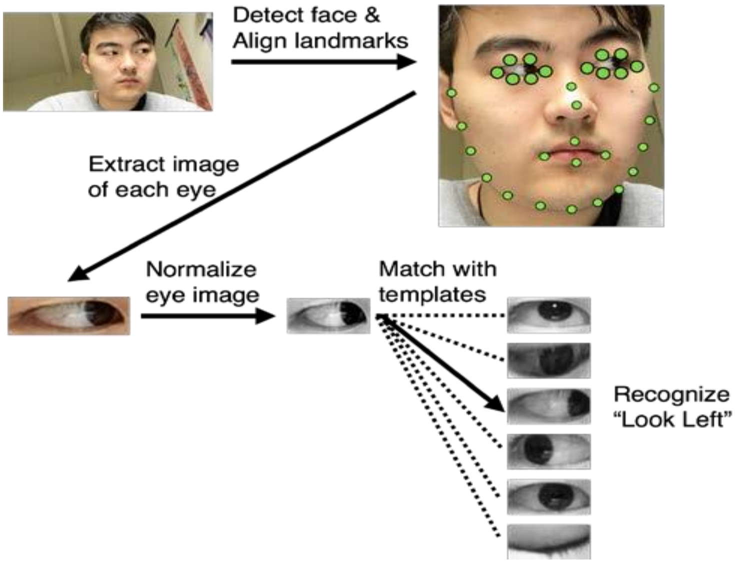

Figure 5. Flowchart of eye gesture recognition algorithm

4.4 Performance

We tested our eye gesture recognition on recent models of the iPhone and iPad. For the iPhone, the recognition speed ranges from 27 frames per second (fps) (iPhone 6s Plus) to 17 fps (iPhone 5s). For the iPad, it ranges from 34 fps (iPad Pro 9.7) to 16 fps (iPad mini 2). The slowest device still processes enough frames to confirm gestures for text entry.

4.5 Robustness

Our algorithm works in a variety of lighting conditions, including indoors and outdoors. Since it uses an RGB rather than IR camera, its performance is unlikely to be degraded under sunlight. In low-light conditions, Gaze Speak can turn on the phone's flashlight to make the speaker's face visible. To avoid flash burn to the eyes, we need to apply a diffuser- like tape such as 3M tape (less than $1) over the flashlight. Our algorithm tolerates two major transformations: Scaling (e.g., if the interpreter moves the phone closer to the speaker), Translation (e.g., if the speaker moves his head a bit, or the interpreter slightly moves the phone while holding it), and Scaling + Translation (e.g., the interpreter puts down the phone, and later holds it in a slightly different position). Rotation of speaker's head or the phone significantly changes perceived face shape and thus reduces recognition accuracy. Our app shows the face image and recognition results to visually assist the interpreter in self- correcting the positioning.

4.6 Accuracy

To assess the accuracy of our gesture recognition system, we recruited 12 participants through email lists within our organization, and paid each participant $5 for a 30-minute session. They reported a mean of 29 years old (min 20, max 44); five were male and seven were female. Six had a normal (uncorrected) vision, one wore contact lenses, and five wore glasses. Participants had varied skin and eye colors. During the study, each participant performed each of up/down/left/right/closed eye gestures 30 times; after each gesture, the participant looked back to center. In total, we recorded 300 eye gestures for each participant (30 x 5 + an additional 150 gazes toward the center).

	Up	Down	Left	Right	Close	Center
Mean	88.6%	75.3%	87.8%	86.9%	77.5%	98.6%
Stdev	5.2%	17.5%	4.8%	7.4%	13.5%	1.8%

Table 1. Gaze speaks recognition rate of each eye gesture.

4.7 Predictive Text Engine

To avoid fatigue from making complex eye gestures, improve learnability, and improve recognition rates, Gaze Speak uses a small number of simple eye gestures (up/down/left/right) to refer to all 26 letters of the English alphabet, using only one gesture per character entered (to reduce fatigue and increase throughput). This design leads to an ambiguous keyboard (Figure 6a) in which the letters of the alphabet are clustered into four groups that can each be indicated with one of the up, down, left, or right gestures. Gaze Speak implements a predictive text engine [22,23] to find all possible words that could be created with the letters in the groups indicated by a gesture sequence. Our implementation uses a trie [14] data structure to store common words and their frequencies in oral English: we select the most common 5,000 words (frequency min=4,875, max=22,038,615, mean=12,125) from this wordlist [6]. This trie can also be extended by an interpreter, who can add the speaker's frequently-used words (including out of dictionary words, such as names). For a series of eye gestures with length n, our trie structure allows us to rapidly look up its matching words and word frequencies in O(n) time. Gaze Speak can also look up high-frequency words whose initial characters match the gesture sequence thus far.

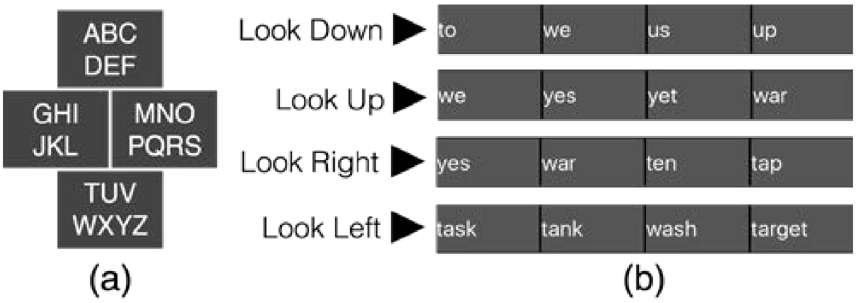

Figure 6. Word predictions update after each gesture in this example four-gesture sequence to spell the word "task".

4.8 Text Entry Interface

The speaker and interpreter use different interfaces. The speaker sees the back of the phone, and thus the screen-less text entry interface consists of two major components: 1) a sticker displaying the four-key keyboard, and 2) audio feedback. The interpreter sees the phone screen, which shows four major components as in Figure 7: 1) the four- key keyboard, 2) the input box and word predictions, 3) the sentence box and 4) the camera preview.

4.9 Speaker Interface

On the back of the phone, a four-key sticker (figure 1b) reminds the speaker of the letter groupings associated with each of the four gesture directions. To enter one character, the speaker moves his eyes in the direction associated with that letter's group. Once Gaze Speak detects that the eyes have settled in one direction, it speaks aloud the direction it detected (e.g., "Up"). The speaker can then move his eyes to enter the next character. When the speaker mistypes or hears feedback indicating an incorrect gesture recognition, he can wink his left eye (if the speaker cannot wink, an alternative is to close both eyes for at least two seconds); this gesture removes the last character from the current sequence. When the speaker finishes a sequence for an entire word, he can wink his right eye (or alternatively look center for at least two seconds) to indicate the end of the word; then the system will speak aloud the first word prediction based on the entire series of eye gestures. The speaker can wink his right eye again to confirm this prediction, or perform a look right gesture to hear the next prediction. After a word has been confirmed, it is added to the sentence being constructed (Figure 7d). After the speaker confirms the last word of the sentence, he can wink his right eye again to confirm the end of the sentence, and the system will play the whole sentence aloud.

V.PROCEDURE

We employed a within-subjects design to examine the input speed, usability, and user preference among three input methods: an e-tran board, Gaze Speak's default operation style, and Gaze Spaak's front-facing mode. We used a Latin Square design to counterbalance the ordering of the three input methods across participant pairs. For sentences to be entered during testing, we randomly picked a set of eighteen five-word-long sentences (29-31 characters each) from the Mackenzie and Sourke off phrase sets commonly used for evaluating text entry techniques [19] to use as our testing corpus. Participation began with a brief introduction of the purpose of the study. We then randomly assigned the speaker and interpreter roles to the members of a pair (these roles were held constant for all three input methods). Participants sat face to face (in chairs set 18-inches apart).

For each of the three conditions, we presented a tutorial on how to use the communication method, and let the pair practice until they felt comfortable using the method to communicate a two-word example utterance (e.g., "hello world"). We then privately showed the speaker a sentence from the testing corpus, and instructed them to communicate that sentence as quickly and accurately as possible to their partner without speaking, using only the current communication method. We then started a timer and stopped the timer when the interpreter correctly decoded the sentence. This procedure continued until either six sentences had been successfully communicated or ten minutes had elapsed, at which point we stopped the session in order to avoid excessive fatigue. Participants then completed a short questionnaire providing feedback about their experience using that communication method, and took a short break if they felt fatigued. After repeating this procedure for all three communication methods, participants completed a final questionnaire ranking their preferences among all three methods.

VI.RESULTS

For each 10-minute session using a given interface, we prepared six sentences for the speaker to communicate to the interpreter. In the two Gaze Speak conditions, all pairs successfully communicated all six phrases. However, in the e-tran condition, pairs completed an average of 4 phrases. Participants, on average, spent 137.8 seconds to complete a sentence using the e-tran board, 80.9 seconds using Gaze Spaak's default mode and 77.1 seconds using Gaze Spaak's front-facing mode. A one-way repeated measures ANOVA indicates that mean input time differed significantly between input methods (F (1.067, 9.602) = 21.032, p = 0.002). Follow-up pairwise paired-samples t-tests show that both modes of Gaze Speak bring a statistically significant reduction in input time as compared to the e-tran board (default mode vs. e-tran: t (9) = 4.136, p = 0.003, and front-facing mode vs. e-tran: t (9) = 3.983, p = 0.003). To make Gaze Speak even more robust and suitable to a wider range of end-user needs and abilities, we include a manual input mode for use when eye gestures cannot be recognized, and a front-facing mode that can be used when a phone stand is available. The front-facing mode allows Gaze Speak to be used without the help of an interpreter, offering the possibility of a lower-cost, but lower-fidelity and less general-purpose, alternative to commercial eye- tracking solutions if someone cannot afford one. In following principles of ability-based design [36], Gaze Speak gracefully adds/degrades functionality depending on a user's capabilities, such as allowing backspacing if left-winking is possible, offering dual-eye closing as an alternative to winking, and still functioning without the backspace capability at all if the speaker can perform neither of those gestures. We have also implemented a head-tracking mode that can be used as an alternative to eye gestures depending on the capabilities and preferences of the speaker.

Our feedback sessions with PALS and their communication partners found areas for further improvement of Gaze Speak. To improve eye gesture recognition availability, we can improve the face detection and eye detection algorithms to have robustness in cases of face-worn medical equipment. We can also further improve the word prediction by automatically updating the weight of prediction from daily usage, and making word prediction based on already-typed words and other contextual information (e.g., location information from the phone's GPS [9]), and offering a larger n-best list for interpreters to select from. While our evaluations gave insight into usability and performance for novice users, additional types of evaluation can offer additional value. For example, longer-term studies of use would give insight into the relative learning curves of Gaze Speak versus other AAC methods, and would allow characterization of expert-level performance. More formal, long-term data collection from PALS and their communication partners, while logistically complex, is particularly important for understanding in situ use of this new type of AAC. Gaze Speak may also hold value for other audiences besides PALS (e.g. people with cerebral palsy, spinal cord injury, stroke, traumatic brain injury, etc.). Because PALS can have different eye movement pattern than other motor-impaired users, additional testing may be necessary to modify Gaze Speak to support such users.

VII.CONCLUSION

In this paper, we introduce Gaze Speak, a smartphone app to facilitate communication for people with motor disabilities like ALS. In real time, Gaze Speak robustly recognizes eye gestures and decodes them to words. Our user studies show that Gaze Speak surpasses e-tran boards (a commonly-used low-tech solution) in both communication speed and usability, with a low rate of wrong recognition. Feedback from PALS and their communication partners affirmed the value of offering a portable, low-cost technology to supplement IR-based eye tracking systems in situations where they are impractical to use (or are unaffordable to purchase).

REFERENCES

1. *Javier S. Agustin, Henrik Skovsgaard, John P. Hansen, and Dan W. Hansen. 2009. Low-cost Gaze Interaction: Ready to Deliver the Promises. In Conference on Human Factors in Computing (CHI), 4453–4458. http://doi.org/10.1145/1520340.1520682*
2. *Monica Anderson. 2015. Technology Device Ownership: 2015. Retrieved September 15, 2016 from http://www.pewinternet.org/2015/10/29/technology- device-ownership-2015/*
3. *Gary Becker. 1996. Vocal Eyes Becker Communication System. Retrieved September 15, 2016 from http://jasonbecker.com/eye_communication.html*
4. *Pradipta Biswas and Pat Langdon. 2011. A new input system for disabled users involving eye gaze tracker and scanning interface. Journal of Assistive Technologies 5, 2: 58–66. http://doi.org/10.1108/17549451111149269*
5. *Gary Bradski. 2000. The OpenCV Library. Doctor Dobbs Journal 25: 120–126.*
6. *Mark Davies. 2008. The Corpus of Contemporary American English: 520 million words, 1990-present. Retrieved from http://corpus.byu.edu/coca/*
7. *Kristen Grauman, Margrit Betke, Jonathan Lombardi, James Gips, and Gary R. Bradski. 2003. Communication via eye blinks and eyebrow raises: Video-based human-computer interfaces. Universal Access in the Information Society 2, 4: 359–373. http://doi.org/10.1007/s10209-003-0062-x*
8. *Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. Advances in Psychology 52, C: 139–183. http://doi.org/10.1016/S0166-4115(08)62386-9*