Software Architecture for Automating Cognitive Science Eye-Tracking Data Analysis and Object Annotation

Karen Panetta, *Fellow, IEEE*, Qianwen Wan, *Student Member, IEEE*, Aleksandra Kaszowska, Holly A. Taylor, and Sos Agaian, *Fellow, IEEE*

***Abstract*—The advancement of wearable eye-tracking technology enables cognitive researchers to capture vast amounts of eye gaze information while participants are completing specific tasks without restrictions on their movement. However, while eye trackers can overlay a gaze indicator on the scene video, identifying the specific objects being looked at and analyzing the resulting dataset are accomplished mostly by manual annotation. This method is a cost-prohibitive and time-consuming approach that is prone to human error. Such analytic difficulty limits researchers' ability to data mine the information efficiently, ultimately restricting the number of scenarios that can feasibly be conducted within budget. Here, the first fully automated solution for eye-tracking data analysis is presented, which eliminates the need for manual annotation. The proposed software architecture, gaze to object classification (GoC), processes the gaze-overlaid video from commercially available wearable eye trackers, recognizes and classifies the specific object a user is focusing on and calculates the gaze duration time. GoC utilizes an image cross-correlation method to locate the gaze indicator and an image similarity measurement to support faster processing. The presented system has been successfully adopted by cognitive psychologists. GoC's exceptional performance in analyzing a case study spanning over 50 h of mobile eye-tracking is presented. The accuracy and a cost-analysis comparison between GoC and state-of-the-art manual annotation software are provided. GoC has game-changing potential for increasing the ecological validity of using eye-tracking technology in cognitive research.**

***Index Terms*—Cognitive research, eye-tracking data analysis software, gaze-to-object classification (GoC), wearable eye-tracking technology.**

Manuscript received June 12, 2018; revised November 2, 2018; accepted December 1, 2018. Date of publication February 4, 2019; date of current version May 15, 2019. This paper was recommended by Associate Editor H. Zhou. This work was supported by the Tufts University School of Engineering, and the Center for Applied Brain & Cognitive Sciences (CABCS) under award A452001 ARM981. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the CABCS. (*Corresponding author: Qianwen Wan.*)

K. Panetta and Q. Wan are with the Department of Electrical and Computer Engineering, Tufts University, Medford, MA 02155 USA (e-mail: karen@ece.tufts.edu; Qianwen.Wan@tufts.edu).

A. Kaszowska and H. A. Taylor are with the Department of Psychology, Tufts University, Medford, MA 02155 USA (e-mail: aleksandra.kaszowska@tufts.edu; holly.taylor@tufts.edu).

S. Agaian is with the City University of New York, New York, NY 10017 USA (e-mail: Sos.Agaian@csi.cuny.edu).

Digital Object Identifier 10.1109/THMS.2019.2892919

Fig. 1. SMI [1] head-mounted mobile eye tracker (left), the image captured from the infrared camera monitoring the eye (middle) and the recorded video with the overlaid gaze captured from the front-scene camera (right).

I. Introduction

HUMAN problem solving in its most basic form requires generating possible scenario solution paths to move between the initial problem state and the desired solution. Enhancing our understanding of the cognitive processes involved in evaluating scenarios requires psychologists to study problem solving in real-world contexts. Researchers capture these vast amounts of visual eye gaze information from eye trackers while participants are completing specific tasks.

Eye-tracking technology can be applied in various research contexts and scenarios with demonstrated usability in neuroscience [10], psychology [11], sports [12], human–robot interaction [13], geology [14]–[16], medical diagnosis [17], [18], and on-road driving applications [19], [20]. The rapid development of eye-tracking systems and technology popularized eye movement research and expanded the potential scope of research paradigms [21].

Wearable mobile eye-tracking glasses rely on infrared cameras to locate and monitor the pupil's position to estimate where the participant is looking. A front-facing scene camera records the person's field of vision, and a gaze position indicator (a cursor) is subsequently overlaid over that scene video (see Fig. 1). Hence, lightweight mobile eye-tracking technology enables the investigation of eye movements in real world, unconstrained settings, drastically improving the ecological validity of research in cognitive science.

While there have been significant advances in technology to capture eye movement data from participants, the ability to automatically process, analyze, and make informed inferences from the resulting datasets remains a challenge. The lack of reliable and user-friendly solutions limits possibilities for utilizing mobile eye-tracking technology in exploring large-sample data, effectively limiting the scope of possible advancements in

our understanding of how people interact with the surrounding world.

Automated video processing analysis addresses this problem; however, no reported work has been performed to automate the "gaze-to-object" classification for massive mobile eye-tracking data for cognitive research. This paper presents a novel software architecture for automating mobile eye-tracking data analysis and classifies the object being gazed upon.

The main contributions of the paper are as follows.

1) A solution using image analytic techniques is described to facilitate the wearable eye-tracking data analysis and gazed object classification. The novel software architecture (gaze-to-object classification (GoC)) can recognize which region of interest (ROI) participants are attending to in gaze-overlaid videos, classify the specific object within the ROI, and provide information on the gaze duration based on frame counts.
2) A user-friendly GoC prototype is designed for automatic identification and annotation (labeling) of object categories within gaze-overlaid videos relevant to the research question under investigation. It is one of the most versatile approaches to analyze mobile eye-tracking data with the advantages of being position invariant, task driven, illumination and noise tolerant (such as motion blur), and able to operate with all commercially available mobile eye-tracking equipment.
3) GoC's utility in analyzing a case study spanning over 50 h of mobile eye-tracking data is presented. During the study, participants had to conceptualize and present a solution path for a design problem. The detailed case study description is discussed in Section III. GoC's outcomes are then compared to the current standard benchmark of those achieved through manual coding by trained and inexperienced researchers.

This paper is organized as follows. Section II gives a brief review of related work and current approaches for cognitive scientists interested in utilizing mobile eye-tracking in their research. A detailed laboratory setting case study created by cognitive psychologists is described in Section III. Section IV provides a detailed description of the automatic image processing framework for the GoC. Meanwhile, GoC was tested on a large dataset from a problem-solving study, followed by an evaluation of time efficiency and accuracy of this approach. Finally, the conclusions and several future research directions are presented in Section V.

II. Background and Related Work

A. Eye Movement Properties

Eye movements are proactive. They are the conduit for gathering information, which seek to provide insight into a task's attentional demands in anticipation of a decision required for future actions [2], [3]. Eye movement properties such as blink rate, fixation duration, or saccadic amplitude and velocity are robust indicators of underlying cognitive states such as stress or high mental workload [4] and provide insight into the cognitive requirements of various tasks [5]–[7]. Analyzing the process of how people look at things informs our understanding of mechanisms guiding visual attention and influencing underlying cognitive states. Combining such information with data on where people look and what they look at provides cognitive science researchers with unique insight into how humans interact with the world to advance our understanding of human cognitive function [8], [9].

B. Eye-Tracking Research and Applications

Wearable eye-tracking technology opens new avenues for researchers to monitor eye movements and record their properties under a variety of experimental protocols and in real-life situations [22]. This allows researchers to extend the scope of investigation to more ecologically valid experimental designs [23]. In analyzing where and what people look at, scientists rely on predefined ROI within a specific stimulus or environment. Such selection is informed by a specific research question on hypothesis. As such, the ROI differs across studies in shape, size, scope, and location, and can range in complexity from large clusters of objects (an entire storefront display) to narrow areas within a single object (a specific item on the shelf). Robust identification and annotation of ROI within resulting data are therefore critical.

C. Eye-Tracking Data Analysis Methodology

Existing automatic image recognition algorithms can detect all objects within a video frame that fulfill a certain set of criteria. The relative importance of objects within a visual scene is often estimated on the basis of existing visual attention models, such as visual salience models [24], [25]. Such detection algorithms are heavily rooted in computational approaches to vision modeling [26], and allow for informed prediction on where participants would look [27], [28]. This allows researchers to delineate which objects within the visual environment should grasp participants' overt attention. Highlighting all objects of interest within a video frame is an important first step to automating eye-tracking data analysis and as such presents incredible potential for real-life applications [29].

It is not necessary to identify all objects of possible interest within a visual scene in order to support automated eye-tracking analysis. Eye-tracking equipment collects information about the participants' pupil position and calculates where the participant is looking at any given time. The gaze direction can be then cross referenced with an object within the visual scene [30]. However, eye-tracking data accuracy is prone to drift over time, and as such the calculated gaze point does not always overlap with the ROI even when the participant is in fact attending to that ROI. It is therefore necessary to introduce corrective measures [31]. Outstanding work has been done to understand the relationship between eye movements, semantic description, and computer vision [32], [33]. Gaze-tracking systems include examination of eyeball features (such as pupil, eyelid, and iris), gaze direction evaluation, a gaze mapping function, and hardware calibration as described in [34]. A benchmark for the point of gaze detection algorithms is presented by McMurrough *et al*. in [35]. The gaze detection technique is mature and commercially available; commercial eye trackers are able to calculate eye movement properties and reports gaze position estimates as a series of x–y

coordinates falling on a specific frame within an independent scene video.

The next logical step is developing a GoC tool for automated gaze annotation and ROI labeling.

To date, there is no commercially available software enabling eye-tracking researchers to perform automated object recognition on the scene video, and as such, researchers do not have a simple way of inferring what their participants are looking at.

Instead, cognitive researchers are forced to process mobile eye-tracking data by manually delineating ROI, often frame by frame, such as depicted in Fig. 2. This approach is a well-known obstacle to utilizing mobile eye-tracking technology [36].

Prior research has contributed to make ROI annotation easier. In [37], Tsang *et al*. introduced eSeeTrack, a visualization prototype that streamlines ROI annotation and allows researchers to estimate fixation duration falling within specific ROI content. However, for wearable eye-tracking data, fixated ROI annotation still heavily relies on manual human labor in completing the analysis. A similar approach was presented by Kurzhals *et al.* [38]. The proposed ISeeCube is a visual analytics tool specifically designed for visual analysis of recorded eye-tracking gaze pattern information with the ROI content. The toolbox offers the possibility of including multiple coordinated viewpoints in converting the two-dimensional (2-D) video data to a three-dimensional model supporting the analysis of ROI in motion, making ISeeCube an important tool for video analysis despite its reliance on the manual ROI annotation process. This labeling time is reduced in recent work [39] with the introduction of a user interface. This system uses the state-of-art image similarity measures to decrease the number of video frames that need to be manually annotated, but does not altogether remove the need for extensive human labor. Finally, Pontillo *et al*. proposed an object recognition-based semiautomation labeling software [40]; however, it depends heavily on manually labeling during the process of video streamlining, and its color-histogram-oriented object classification method fails to account for variance in luminance.

Other works, such as gaze-guided object recognition for a head-mounted eye tracker [41] and gaze-guided object classification utilizing deep neural networks [42], address real-time object recognition using head-mounted eye trackers. However, the classification depends heavily on providing extensive training data and the method is not position invariant, meaning the viewing distance and view perspective must remain fixed.

ROI object categories selection is hypothesis driven and therefore relies exclusively on the researcher's scientific goal. GoC identifies frames within the gaze-overlaid video where the gaze indicator overlaps with one of the predefined ROI object categories. It then automatically labels which object of interest the participant attended to.

To the best of our knowledge, there is no software architecture that can automate video data labeling of mobile eye-tracking devices, especially with the occurrence of dynamic zooming and in the presence of distortions, such as motion blurring. To date, GoC is the only automated analytic approach that does not rely on extensive human labor inherent to a manual ROI labeling approach. To clarify the capabilities of the different tools mentioned above, we have supplied the comparison table (see Table I).

TABLE I
COMPARISON BETWEEN GOC AND OTHER EYE-TRACKING DATA ANALYTIC TOOLS

Related work	Problem addressed	ROI annotation
[37]	Eye-tracking data visualization combining fixation patterns within ROI content	Manual
[38]	Visual analysis of eye-tracking data can combine the gaze pattern information with the ROI content	Manual
[39]	User-friendly interface to simplify ROI annotation by sorting similar image patches	Manual
[40]	Match labeled ROI using color-histogram intersection method	Semi-automatic
[41]	Real-time object recognition using feature matching	No annotation functionality
[42]	Real-time object classification using Deep Neural Network	No annotation functionality
GoC	Large-scale head-mounted eye-tracking data analysis	Automatic

III. CASE STUDY

A. Tool Design and Human Problem Solving

The case study explored how engineers discovered possible solution paths in aiding completion of a mundane and repetitive task: sorting an unorganized *Lego Mindstorms* NXT kit. The kit contains different types of Lego pieces, totaling over 430 pieces distributed across two trays with 4 and 13 compartments each.

The goal of the design problem was to optimize the sorting process by conceptualizing a tool (a physical piece of equipment) to be used by a particular end user in sorting a single Lego NXT kit in accordance with the instructions provided. The given instructions were to sort a disorganized Lego kit to match a pre-existing reference image of a sorted Lego kit. The envisioned tool will be used by one of three possible end users: a human (end user of known physical and cognitive capabilities), a robot (end user of unknown technical specifications), or a team consisting of a human and a robot.

As participants were given no information about specific abilities or capacities of their end user, they were expected to infer that information from previous experience or through observation.

The goal of the study was to identify the approaches that engineers employ to fill in their knowledge gaps when faced with a problem-solving task. The case study investigated whether such "knowledge patches" were based on engineers' previous experiences (as indicated by their verbal explanations during the task), or on visual examination of the end user (as indicated by the eye-tracking data).

B. Current Study

Design problem: Participants were to devise a tool that would help a specific end user sort a Lego kit more efficiently.

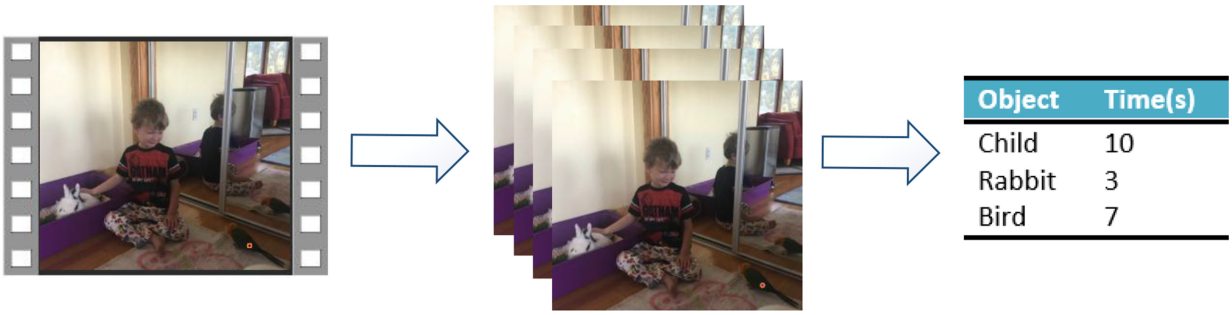


Fig. 2. Example of a gaze point-inspired ROI labeling process used in cognitive science research. White rabbit on the left, child in the middle, and bird on the right with the orange gaze indicator overlaid on the bird.

Participants: 50 undergraduate students (age mean = 19.7, standard deviation = 2.5) from the Tufts University School of Engineering completed the study for monetary compensation.

Experimental session: Each participant was randomly assigned to one of three possible end users (15 participants per end user in total). Participants were seated at a table across from their intended end user and were not allowed to interact with the user in any way other than observation. Furthermore, participants were not given any information about their end user beyond what they could observe and infer on the basis of previous knowledge.

In each session, participants received a disorganized Lego NXT kit that they could freely manipulate. The experimental task consisted of three stages: first, participants brainstormed possible solutions for 10 min. Then, they could use paper, pencils, scissors, and tape to record their design idea in any form they saw fit (sketch or model). After approximately 40 min, participants gave a short explanation of how their proposed sorting tool worked.

Eye tracking: Eye-tracking data were collected using SMI eye-tracking glasses [1] (Sensomotoric Instruments, Inc.) at 30 Hz. Gaze-overlaid videos were exported using SMI BeGaze 3.7 (Sensomotoric Instruments, Inc.).

ROI: Our cognitive psychology team was interested in observing whether participants looked at four objects: the end user, the Lego kit, the reference image of a sorted Lego kit, or the writing utensils (see Fig. 7).

Current analytic objective: Measuring the proportion of time spent looking at the end user versus time spent manipulating items on the table. Such analysis informs our understanding of how approaches to designing a tool differ as a function of the intended end user.

Analytic challenges: As participants interact with objects (Lego pieces or writing utensils), the resulting eye-tracking videos are particularly difficult to analyze due to: 1) blurring of the video resulting from a rapid shift in the field of vision from even the slightest shift in head position; 2) object manipulation and sketching during the experimental session mean that the appearance of the areas of interest is constantly changing, i.e., Lego pieces can be stuck together or separated or the initial blank paper provided gradually becomes covered in writing; and 3) participants' hand motions cross the field of vision and therefore periodically obscure the view.

IV. GoC System Framework

The overview of the GoC software architecture for automating mobile eye-tracking data analysis and scene-gazed object classification is presented in Fig. 3. There are six system process blocks.

1) *Input video:* For our software solution, we directly input a gaze-overlaid video generated by the commercially available wearable eye trackers. The gaze-overlaid scene video is converted into individual image frames. This step aims to allow the employment of image processing algorithms.
2) *Filtering process:* The number of images is reduced by taking the advantage of temporal coherence of the video data. This step can remove blurry and otherwise uninformative frames, which can also result in increasing processing speed.
3) *Utilize the eye tracker's gaze indicator:* Zooming in on the focus of the visual attention will highlight where and what the observer is looking at in each frame.
4) *Gaze directed ROI cropping:* The area around the gaze indicator is selected. Due to calibration issues, the actual gaze point may not always be in the ROI but will be close by. Cropping the area around the detected gaze indicator can help minimize the error introduced by the eye tracker calibration and help better identify the gazed upon object information.
5) *Object classification:* The classes of objects are user defined, which depends on constructions driven by the hypothesis under investigation. The goal of this step is to automate ROI annotation.
6) *Result visualization:* A scarf plot and histogram generation are used for the purpose of summarizing and visualizing analysis results. Either a scarf plot demonstrating the temporal overview for the video or a histogram showing the occurrence of different object classes can be utilized here.

All the process blocks will be explained in detail in this section.

A. Video-to-Frame Conversion

The GoC architecture starts by converting the gaze-overlaid scene video stream into frames, which means that all the sub-

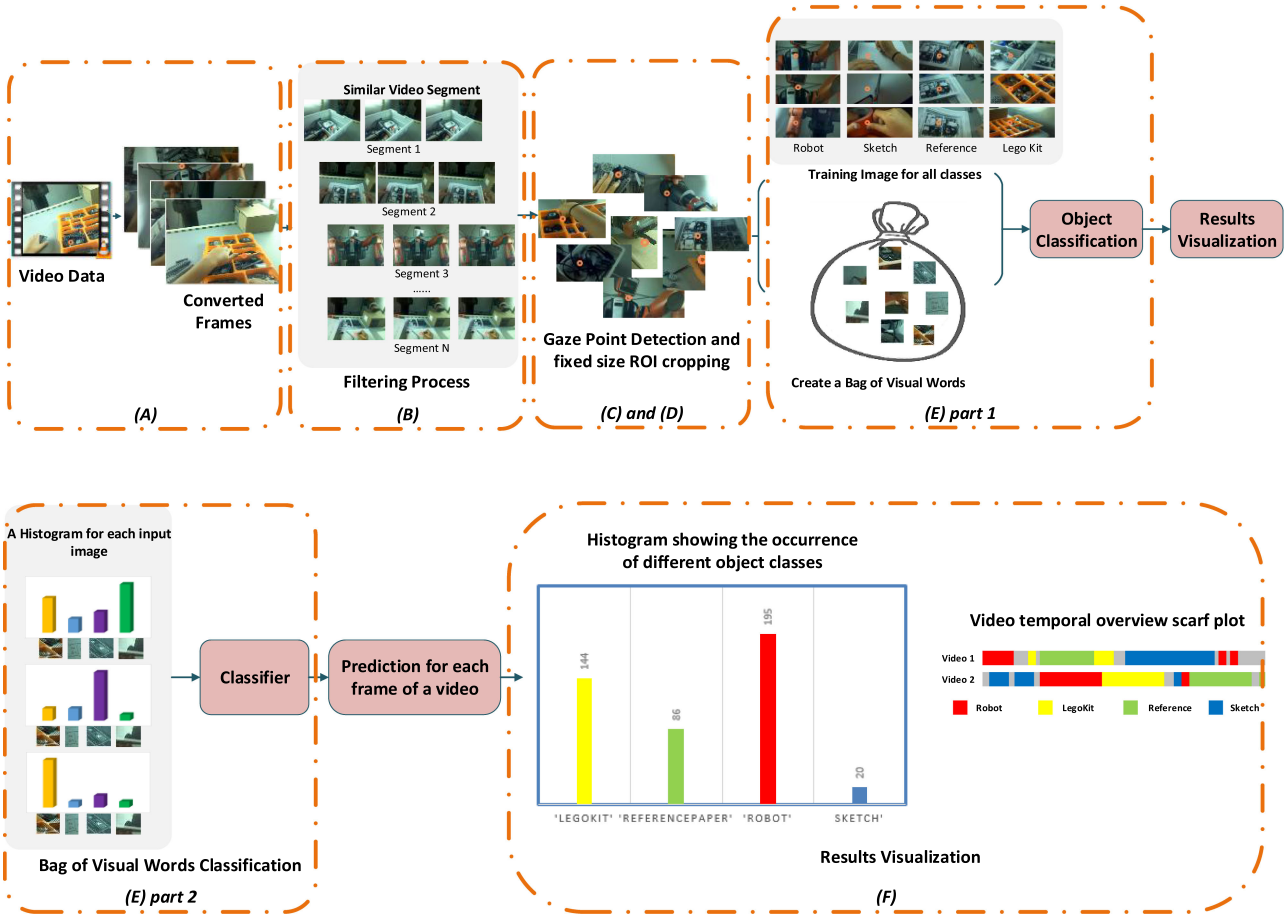


Fig. 3. Overview of GoC prototype architecture. The input mobile eye-tracking gaze-overlaid video data are (A) broken down into image frames. (B) Filtering process reduces the size of the data by taking the advantage of the temporal coherence of the gaze data. This is followed by (C) gaze indicator detection, (D) gaze-directed ROI cropping; (E) object classification and (F) result visualization. Step (E) involves building the training dataset; creating a bag of visual words, obtaining the histogram for each image and using classifiers to train the produced histogram features. (F) Resulting visualization of GoC is shown using a scarf plot demonstrating the temporal overview for the video and a histogram showing the occurrence of different object classes.

sequent image processing tools are applied directly onto the converted frames. Hence, there is no restriction on the video format and no limitation on the eye tracker devices used.

The video data captured by the SMI head-mounted mobile eye-tracking glasses were 25 frames/s; thus, after converting, there are 70 500 images in one 47-min video.

B. Filtering Process

When people focus their visual attention on a specific area, the resulting video contains a sequence of similar image frames. Relying on the temporal coherence of the underlying gaze video, a downsampling processing utilizing an image similarity measurement [43], [44] can reduce the number of frames.

The structural similarity between subsequent video frames is calculated algorithmically; images are deleted when they drop below a similarity threshold, which means that the participant is moving quickly from one object to the other, or one of the images is too blurry and noisy. The decision-making schematic is shown in Fig. 4.

In [45], an image similarity measure 4-EGSSIM using enhanced human visual system characteristics is introduced. This

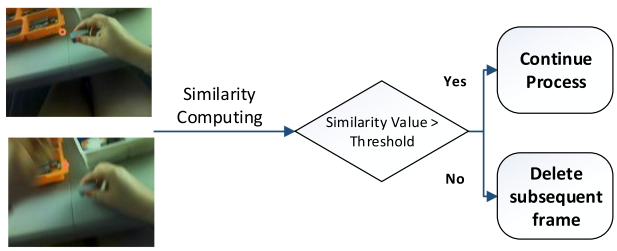


Fig. 4. Decision-making process for the filtering process to remove blurry or noisy images due to rapid eye motion. The top image frame is compared to the subsequent frame (bottom left) and shows severe motion blur. The blurry frame is deleted.

similarity measure was applied because of its high distinguishing ability in video processing [46].

For two conterminous video frames x and y, their corresponding edge map, x' and y', are calculated using a Sobel edge detector [47]. Each image was first partitioned into four subregions based on the edge information. The implementation uses the following formulas.

Preserved edge pixel region (R1)

$$\log(x'(m,n)+1) > T_1 \;\&\&\; \log(y'(m,n)+1) > T_1. \quad (1)$$

Change edge pixel region (R2)

$$\begin{aligned}&(\log(x'(m,n)+1) > T_1 \;\&\&\; \log(y'(m,n)+1) \le T_1\;))\,||\\&(\log(y'(m,n)+1) > T_1 \;\&\&\; \log(x'(m,n)+1) \le T_1)).\end{aligned} \quad (2)$$

Smooth region (R3)

$$\begin{aligned}&((\log(x'(m,n)+1) > T_1 \;\&\&\; y'(m,n) \le T_1\;))\,||\\&(\log(y'(m,n)+1) > T_2 \;\&\&\; \log(x'(m,n)+1) \le T_2)) \quad (3)\\&\text{Texture Region } (R4) : \text{Otherwise.} \quad (4)\end{aligned}$$

Here, T1 $= 0.12\ (x'_{\max})$ and T2 $= 0.06\ (x'_{\max})$, where $x'_{\max}$ is the maximum value of the gradient magnitude of x.

Then, the 4-EGSSIM value is calculated using

$$4-\text{EGSSIM}_{R_i}(x,y) = \sum_{i=1}^{4} \frac{w_i}{|R_i|} \sum_{(m,n)\in R_i} \text{GSSIM}_{x,y}(m,n) \quad (5)$$

where w_i are the weights for each region R_i, $i = 1, 2, 3$, and 4. These parameters were obtained experimentally. The GSSIM [48], a variant of SSIM [43], values are calculated in terms of local luminance, contrast, and structure, and then these local similarity measures are pooled into a single similarity 4-EGSSIM metric; the detailed explanation and calculation can be found in [45].

For this work, an applied measure threshold of 0.8 provided a good filtering result, which removed redundant frames and reduced the dataset by 10% from the original video.

The filtering process using the image similarity measure has two significant advantages: 1) efficiency is enhanced because the image assessment method is used to reduce the dataset size; and 2) accuracy is addressed because undergoing an image similarity assessment also helps delete unbefitting frames.

C. Utilize the Eye Tracker's Gaze Indicator

Normalized image 2-D cross correlation is used to localize the human eye gaze indicator by estimating the similarity between the cropped template of the gaze indicator [see Fig. 5(c)] and the original video frame captured from a first-person perspective [see Fig. 5(a)].

The gaze indicator detection step starts with extracting the red color channel of the input RGB video frames. An example of a red channel image from an original video frame is shown in Fig. 5(b). Next, the 2-D normalized cross correlation between the video frame and gaze template is performed using the following formula [49]:

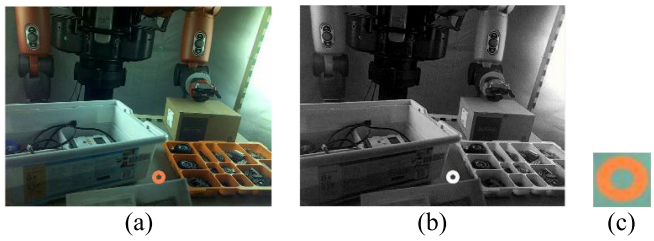


Fig. 5. (a) Original video frame captured in a laboratory setting. Gaze indicator template generated by (c) commercial eye-tracking devices that is produced by SMI BeGaze [1]. It is an orange circle with the same size in each video frame. (b) Representation of the extracted red channel of the original RGB video frame.

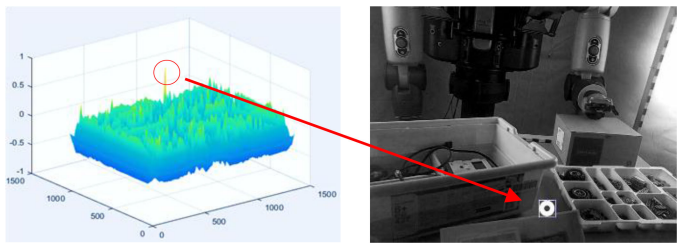


Fig. 6. Normalized 2-D correlation results for gaze indicator detection. The left image shows the 2-D correlation coefficients matrix displayed as a surface. The correlation coefficients (z-axis) can range in value from –1.0 to 1.0, and the x- and y-axis show the pixel location of the original image. The peak value indicates the image patch that is most similar to the template, which provides the location of a gaze indicator and is displayed within a bounding box (right).

$$\gamma(u,v) = \frac{\sum_{x,y}\left[f(x,y)-\bar{f}_{u,v}\right]\left[t(x-u,y-v-\bar{t})\right]}{\left\{\sum_{x,y}\left[f(x,y)-\bar{f}_{u,v}\right]^2 \sum_{x,y}\left[t(x-u,y-v-\bar{t})^2\right]\right\}^{0.5}} \quad (6)$$

where f is the image, t is the template, $\bar{t}$ is the mean of the template, and $\bar{f}_{u,v}$ is the mean of $f(x,y)$ in the region under the template.

The resulting matrix contains the correlation coefficients that range from –1.0 to 1.0, which is displayed in Fig. 6 (left). The detected gaze indicator is within a bounding box in Fig. 6 (right). Note that the accuracy of this step was approximately 99.5%, with the failure cases occurring when the predicted gaze indicator was not in a complete round shape. This occurs when the gaze falls on the edge of the video frame.

D. ROI Cropping

One of the most important tasks in analyzing mobile eye-tracking data is to distinguish between different objects in a single frame by finding the segment boundary between objects.

In designing the GoC framework architecture, the ROI cropping step can be achieved using image segmentation methods [50].

Usually, the segmentation obeys a certain criterion with respect to the same characteristics, such as color, intensity, or texture. Though many practical applications of segmentation

technology are fully explored, no segmentation algorithm is flawless and suitable for all applications. Therefore, segmentation solutions must be chosen with respect to specific analytic needs in order to maximize the efficiency and performance of the algorithm.

It was challenging to find a suitable image segmentation technique that can extract the objects of interest without losing the structural information for the subsequent scene-gazed object classification step. Instead, in the prototype system, the area around the detected gaze indicator is automatically cropped using a fixed size.

As one aspect of our future work, we will optimize the ROI cropping step by using a promising image segmentation algorithm. By combining the detected location of the gaze indicator with the location of objects within a frame, the system was able to confirm where a participant is looking during the mobile eye-tracking experiment.

E. Object Classification

Image classification refers to training a computer to determine whether an object belongs to a specific predefined category. In GoC, we chose the bag-of-visual-words (BoVW) classification [51] model (BoVW) over other computer vision classification algorithms (such as convolutional neural networks) because BoVW requires less computational complexity without sacrificing accuracy, needs less extensive training data, and has advantages of orientation invariance and scale invariance.

BoVW treats every image as 'a documentation' with many visual words. Visual words are small patches in an image that are automatically detected by feature detectors based on image structural information. Then, the similar visual words are grouped together to form the visual word vocabulary. A histogram records the visual word occurrences that represent an image, which is used to train an image category classifier. Finally, the system predicts the content using an image classifier that is encoded from the training set images.

BoVW is a leading machine learning methodology with numerous modules, such as feature extraction, feature description, unsupervised clustering, and classifier selection. Examples of selected training images are displayed in Fig. 7.

The steps for BoVW in GoC gazed object classification are as follows.

1) Extract scale-invariant feature transform [52] (SIFT) features from all training images with different categories.
2) Construct the visual words vocabulary by K-means clustering [53] ($K = 100$).
3) Generate a histogram to represent each image, counting the visual word occurrences in an image.
4) Train the classifier.
5) Predict the object's category.

The selection of different algorithms is task driven, which means that we set up different parameters for different applications [54]. For instance, in Step 1, we chose the SIFT feature extraction algorithm over the regular dense [55] feature, SURF [56] feature, and random sampling [57] feature extraction because it provided more accurate classification for eye tracker data with regard to the participants' movement that resulted in frequent zoom-in and zoom-out actions. Additionally, we chose the visual vocabulary size to be 100 and used the linear SVM classifier in the GoC prototype system.

Fig. 7. Image classification with the bag of visual words. Top left is the "LegoKit" image, top right is the reference image of a sorted Lego kit, bottom left is the "Robot" image, and bottom right is a "Sketching" image.

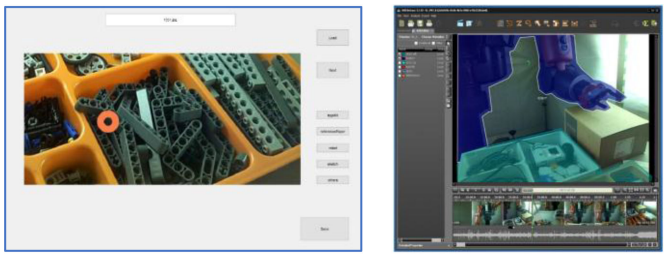
F. Scarf Plot and Histogram Generation

For visualization purposes of summarizing and presenting analysis results, a scarf plot was created to demonstrate the video streaming content with a timestamp of when specific objects are being looked at. A histogram is also output to show the occurrence of different object classes to indicate the task time duration [see Fig. 3(b)].

G. GoC ROI Annotation Graphical User Interface (GUI)

An accompanying GUI was designed for the purpose of optimizing the BoVW parameters and evaluating the accuracy of the frame-by-frame classification. In addition, the GUI can be considered as a semiautomatic visual analytics tool for gaze data that is recorded using wearable eye trackers, which is an improved approach to manual ROI annotation. The designed GUI effectively addresses the analytic challenges and supports their resolution by: 1) observing what participants looked at (the end user, the Lego kit, the reference image of a sorted Lego kit, or the writing utensils); and 2) measuring the proportion of time spent looking at each category.

The presented GUI is shown in Fig. 8(a). The main view of the GUI contains a text header denoting the file name of the eye-tracking dataset currently being annotated. The image shown in the middle of the canvas is the cropped bounding box area with a gaze location marked as the orange circle (generated after implementing step D in GoC). The button panel on the right of the screen is divided into the following sections: 1) Load and Next buttons are used to select the starting frame and the next image frame; 2) Gaze-to-Object category buttons show the different object categories ("LegoKit", "referencePaper", "Robot" and "Sketching" image in our example case), which the user can easily customize and modify in the source code



(a) (b)

Fig. 8. (a) Evaluation GUI developed for GoC for annotating each frame of the recorded mobile eye-tracking video; the load and next button are clicked to select image frames; four categories are displayed on the buttons; by clicking the "Done" button, evaluation data are saved as a spreadsheet. (b) Screenshot of using annotation software from the eye tracker's manufacturer (SMI BeGaze version 3.7).

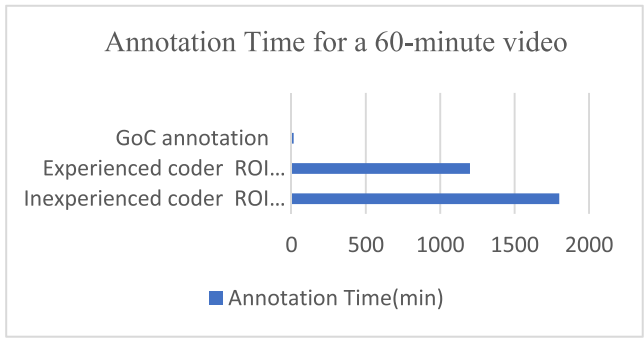


Fig. 9. Comparison of annotation human labor time for an hour-long video between manual ROI annotation (for both an experienced coder and an inexperienced coder) and the evaluation using the proposed GoC annotation.

for different applications; and 3) the "Done" button saves the annotated (evaluation) data as an Excel file.

H. Accuracy Validation

There is no ground truth for the case study (described in Section III) video annotation since no other automated process exists. The GoC-produced histogram distribution matched 83.4% compared with the human label annotation-produced histogram (commercially available SMI BeGaze version 3.7). This accuracy was consistent for case study videos spanning over 50 h (over four million image frames).

To further test the accuracy and efficiency of GoC, we conducted two additional experiments described as follows.

1) The study was a real-world outdoor walking experiment where participants' movements are completely dynamic and the location and/or scene is/are also changing in time. The study involved a campus navigation task, where participants devise routes around the Tufts University campus and then guide the researcher along the route while wearing mobile eye tracker devices. Our cognitive psychology team was interested in observing whether participants looked at common objects in the campus environment: cars, people, and street scenes. The wearable eye tracker device for data collection is described in Section III. The proposed GoC achieved 94.6% accuracy compared to human manual annotation.
2) To prove the stability of GoC under varying illumination conditions, we conducted a straightforward office environment recording while the researcher was changing the lighting intensity during mid recording. The purpose of the recording was to investigate whether the GoC recognizes common office objects (such as office utensils) and people regardless of varying illumination conditions. We started the recording in a low-lighting condition (the office lights were OFF during the late afternoon/twilight, and barely any light was coming through the office window). Then, the researcher turned ON the overhead light in the middle of the recording. The accuracy of GoC is 99.5%, which demonstrated that GoC could detect and recognize objects of interest under varying illumination conditions.

I. Time and Cost Comparison

Perhaps, the most prohibitive aspect of manual ROI annotation is the time required from researchers to complete the task. We compared how much data an inexperienced and experienced coder could annotate in 1 h. The inexperienced coder was an undergraduate student who received a 30-min tutorial on using the proprietary annotation software from the eye tracker's manufacturer [SMI BeGaze version 3.7, Sensomotoric Instruments, Inc.; Fig. 8(b)]. The experienced coder was a graduate student who had approximately five years of experience with mobile eye-tracking recordings and analysis.

The inexperienced coder reported annotating approximately 2 min of video data in 1 h. The experienced coder reported annotating approximately 3 min of video data in 1 h. On the other hand, the GoC setup took approximately 15 min of human labor, and the subsequent automated annotation process did not require supervision. The annotation evaluation using the provided GUI was possible at the rate of 500 images per hour, whereas the cost of human manual annotation rises as the amount of data to be analyzed increases. In the case of using GoC, the cost of human labor remains consistent regardless of the size of dataset to be analyzed.

The case study presented in this paper consisted of 50-h-long videos. The benefits of using GoC over both experienced and inexperienced human coders are compelling (see Fig. 9).

V. Conclusion

This paper proposed an automated solution for analyzing the perspective visual channel extracted from the mobile eye-tracking data. The software architecture, GoC, provides an efficient way for cognitive scientists to automate mobile eye-tracking data analysis by completely removing the burden of human labor. The source code of GoC can be downloaded from http://www.karenpanetta.com/download/.

A. Contributions

GoC utilizes customized image processing algorithms, including 2-D image cross correlation for utilizing the generated gaze point indicator within the gaze-overlaid video, the

4-EGSSIM image similarity measurement to the downsampling process for speeding up the procedure, and BoVW for object classification.

The software architecture has the following advantageous characteristics.

1) The position invariance means that GoC can successfully analyze visual data when participants are completing tasks without any restriction on their movement.
2) The task-driven approach ensures that cognitive researchers can design any experiments based on their needs.
3) The high tolerance to noise enables GoC to detect and delineate noisy and blurring frames.
4) Stability is achieved under varying illumination conditions because the eye tracker's scene camera adjusts illumination in under or over illuminated conditions, and GoC can still recognize and classify the gazed object with the camera adjustment.
5) Its independence of tracker manufacturers enables GoC to automate the analysis of any format of gaze-overlaid videos that are exported using the manufacturers' eye tracker proprietary software.

A user-friendly evaluation interface is provided to test the GoC accuracy and to tune the BoVW object classification and filtering algorithms parameters.

The GoC architecture testing achieved remarkable accuracy and vastly outperformed the traditional manual annotation process in cognitive research. Moreover, GoC has the potential to measure the visual attention of humans across a broad range of areas such as: psychology, cognition, usability, and marketing.

B. Future Work

In the future, the system's accuracy can be improved by investigating image segmentation approaches for object contouring. Furthermore, the BoVW model can be enhanced by adding additional feature information such as the color, shape, edge, and corner.

References

[1] Eye Tracking Solutions by SMI, 2012. [Online]. Available: https://www.smivision.com/

[2] M. Land and B. Tatler, *Looking and Acting: Vision and Eye Movements in Natural Behaviour*. Oxford, U.K.: Oxford Univ. Press, 2009.

[3] Z. Kang and S. J. Landry, "An eye movement analysis algorithm for a multielement target tracking task: Maximum transition-based agglomerative hierarchical clustering," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 1, pp. 13–24, Feb. 2015.

[4] N. M. Moacdieh and N. Sarter, "The effects of data density, display organization, and stress on search performance: An eye tracking study of clutter," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 6, pp. 886–895, Dec. 2017.

[5] K. F. Van Orden, T.-P. Jung, and S. Makeig, "Combined eye activity measures accurately estimate changes in sustained visual task performance," *Biol. Psychol.*, vol. 52, pp. 221–240, 2000.

[6] J. M. Henderson and W. Choi, "Neural correlates of fixation duration during real-world scene viewing: Evidence from fixation-related (FIRE) fMRI," *J. Cogn. Neurosci.*, vol. 27, pp. 1137–1145, 2015.

[7] J. M. Henderson and G. L. Pierce, "Eye movements during scene viewing: Evidence for mixed control of fixation durations," *Psychonomic Bull. Rev.*, vol. 15, pp. 566–573, 2008.

[8] A. Mishra, Y. Aloimonos, and C. L. Fah, "Active segmentation with fixation," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 468–475.

[9] D. P. Papadopoulos, A. D. Clarke, F. Keller, and V. Ferrari, "Training object class detectors from eye tracking data," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 361–376.

[10] A. T. Duchowski, "A breadth-first survey of eye-tracking applications," *Behav. Res. Methods Instrum. Comput.*, vol. 34, pp. 455–470, Nov. 1, 2002.

[11] J. Currie, R. R. Bond, P. McCullagh, P. Black, D. D. Finlay, and A. Peace, "Eye tracking the visual attention of nurses interpreting simulated vital signs scenarios: Mining metrics to discriminate between performance level," *IEEE Trans. Human-Mach. Syst.*, vol. 48, no. 2, pp. 113–124, Apr. 2018.

[12] K. Chajka, M. Hayhoe, B. Sullivan, J. Pelz, N. Mennie, and J. Droll, "Predictive eye movements in squash," *J. Vis.*, vol. 6, pp. 481–481, 2006.

[13] B. S. Yoo and J. H. Kim, "Evolutionary fuzzy integral-based gaze control with preference of human gaze," *IEEE Trans. Cogn. Develop. Syst.*, vol. 8, no. 3, pp. 186–200, Sep. 2016.

[14] J. B. Pelz, T. B. Kinsman, and K. M. Evans, "Analyzing complex gaze behavior in the natural world," *Proc. SPIE*, vol. 7865, pp. 1–11, 2011, Art no. 78650Z.

[15] T. P. Keane, N. D. Cahill, J. A. Tarduno, R. A. Jacobs, and J. B. Pelz, "Computer vision enhances mobile eye-tracking to expose expert cognition in natural-scene visual-search tasks," *Proc. SPIE*, vol. 9014, pp. 1–12, 2014, Art. no. 90140F.

[16] K. M. Evans, R. A. Jacobs, J. A. Tarduno, and J. B. Pelz, "Collecting and analyzing eye tracking data in outdoor environments," *J. Eye Movement Res.*, vol. 5, pp. 1–19, 2012.

[17] Y.-M. Cheung and Q. Peng, "Eye gaze tracking with a web camera in a desktop environment," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 4, pp. 419–430, Aug. 2015.

[18] M. Vidal, J. Turner, A. Bulling, and H. Gellersen, "Wearable eye tracking for mental health monitoring," *Comput. Commun.*, vol. 35, pp. 1306–1311, 2012.

[19] M. Sodhi, B. Reimer, J. Cohen, E. Vastenburg, R. Kaars, and S. Kirschenbaum, "On-road driver eye movement tracking using head-mounted devices," in *Proc. Symp. Eye Tracking Res. Appl.*, 2002, pp. 61–68.

[20] R. Zheng, K. Nakano, H. Ishiko, K. Hagita, M. Kihira, and T. Yokozeki, "Eye-gaze tracking analysis of driver behavior while interacting with navigation systems in an urban area," *IEEE Trans. Human-Mach. Syst.*, vol. 46, no. 4, pp. 546–556, Aug. 2016.

[21] G. Andrienko, N. Andrienko, M. Burch, and D. Weiskopf, "Visual analytics methodology for eye movement studies," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2889–2898, Dec. 2012.

[22] Y. B. Eisma, C. D. Cabrall, and J. C. de Winter, "Visual sampling processes revisited: Replicating and extending senders (1983) using modern eye-tracking equipment," *IEEE Trans. Human-Mach. Syst.*, vol. 48, no. 5, pp. 526–540, Oct. 2018.

[23] N. M. Moacdieh and N. Sarter, "Using eye tracking to detect the effects of clutter on visual search in real time," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 6, pp. 896–902, Dec. 2017.

[24] K. Shanmuga Vadivel, T. Ngo, M. Eckstein, and B. Manjunath, "Eye tracking assisted extraction of attentionally important objects from videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3241–3250.

[25] A. Torralba, A. Oliva, M. S. Castelhano, and J. M. Henderson, "Contextual guidance of eye movements and attention in real-world scenes: The role of global features in object search," *Psychol. Rev.*, vol. 113, pp. 766–786, 2006.

[26] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 10, pp. 1915–1926, Oct. 2012.

[27] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 2106–2113.

[28] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 185–207, Jan. 2013.

[29] T. Walber, A. Scherp, and S. Staab, "Can you see it? Two novel eye-tracking-based measures for assigning tags to image regions," in *Proc. Int. Conf. Multimedia Model.*, 2013, pp. 36–46.

[30] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl, "State-of-the-art of visualization for eye tracking data," in *Proc. EG/VGTC Conf. Vis.*, 2014, pp. 63–82.

[31] R. Netzel, M. Burch, and D. Weiskopf, "Interactive scanpath-oriented annotation of fixations," in *Proc. 9th Biennial ACM Symp. Eye Tracking Res. Appl.*, 2016, pp. 183–187.

[32] M. Hayhoe and D. Ballard, "Eye movements in natural behavior," *Trends Cogn. Sci.*, vol. 9, pp. 188–194, 2005.
[33] K. Yun, Y. Peng, D. Samaras, G. J. Zelinsky, and T. L. Berg, "Studying relationships between human gaze, description, and computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 739–746.
[34] S. Sheela and P. Vijaya, "Mapping functions in gaze tracking," *Int. J. Comput. Appl.*, vol. 26, no. 3, pp. 36–42, 2011.
[35] C. D. McMurrough, V. Metsis, J. Rich, and F. Makedon, "An eye tracking dataset for point of gaze detection," in *Proc. Symp. Eye Tracking Res. Appl.*, 2012, pp. 305–308.
[36] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, and J. Van de Weijer, *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford, U.K.: Oxford Univ. Press, 2011.
[37] H. Y. Tsang, M. Tory, and C. Swindells, "eSeeTrack—Visualizing sequential fixation patterns," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 953–962, Nov./Dec. 2010.
[38] K. Kurzhals, F. Heimerl, and D. Weiskopf, "ISeeCube: Visual analysis of gaze data for video," in *Proc. Symp. Eye Tracking Res. Appl.*, 2014, pp. 43–50.
[39] K. Kurzhals, M. Hlawatsch, C. Seeger, and D. Weiskopf, "Visual analytics for mobile eye tracking," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 1, pp. 301–310, Jan. 2017.
[40] D. F. Pontillo, T. B. Kinsman, and J. B. Pelz, "SemantiCode: Using content similarity and database-driven matching to code wearable eyetracker gaze data," in *Proc. Symp. Eye-Tracking Res. Appl.*, 2010, pp. 267–270.
[41] T. Toyama, T. Kieninger, F. Shafait, and A. Dengel, "Gaze guided object recognition using a head-mounted eye tracker," in *Proc. Symp. Eye Tracking Res. Appl.*, 2012, pp. 91–98.
[42] M. Barz and D. Sonntag, "Gaze-guided object classification using deep neural networks for attention-based computing," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput. Adjunct*, Heidelberg, Germany, 2016, pp. 253–256.
[43] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[44] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.
[45] S. Nercessian, S. S. Agaian, and K. A. Panetta, "An image similarity measure using enhanced human visual system characteristics," *Proc. SPIE*, vol. 8063, pp. 1–9, 2011, Art. no. 806310.
[46] Q. Wan, K. Panetta, and S. Agaian, "A video forensic technique for detecting frame integrity using human visual system-inspired measure," in *Proc. IEEE Int. Symp. Technol. Homeland Security*, 2017, pp. 1–6.
[47] H. I. Works, "Sobel edge detector," cse. secs. oakland. edu., 2004.
[48] G.-H. Chen, C.-L. Yang, and S.-L. Xie, "Gradient-based structural similarity for image quality assessment," in *Proc. 2006 IEEE Int. Conf. Image Process.*, 2006, pp. 2929–2932.
[49] J. P. Lewis, "Fast normalized cross-correlation," *Vis. Interface*, vol. 95, pp. 120–123, 1995.
[50] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
[51] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Workshop Statist. Learn. Comput. Vision*, 2004, pp. 1–2.
[52] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, 2004.
[53] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probability*, 1967, pp. 281–297.
[54] I. Diamant, E. Klang, M. Amitai, E. Konen, J. Goldberger, and H. Greenspan, "Task-driven dictionary learning based on mutual information for medical image classification," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 6, pp. 1380–1392, Jun. 2017.
[55] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit.*, 2005, pp. 524–531.
[56] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.
[57] S. Ullman, M. Vidal-Naquet, and E. Sali, "Visual features of intermediate complexity and their use in classification," *Nature Neurosci.*, vol. 5, pp. 682–687, 2002.

Karen Panetta (S'84–M'85–SM'95–F'08) received the B.S. degree in computer engineering from Boston University, Boston, MA, USA, in 1985, and the M.S. and Ph.D. degrees in electrical engineering from Northeastern University, Boston, in 1987 and 1994, respectively.

She is currently a Dean of Graduate Engineering Education and a Professor with the Department of Electrical and Computer Engineering, Tufts University, Medford, MA, and the Director of the Dr. Panetta's Vision and Sensing System Laboratory. She is the President-Elect of the IEEE Eta Kappa Nu. Her research interests include developing efficient algorithms for simulation, modeling, signal, and image processing for biomedical and security applications.

Qianwen Wan (S'13) received the B.S. degree in information engineering from the Wuhan University of Technology, Wuhan, China, the B.S. degree in law from Wuhan University, Wuhan, in 2013, and the M.S. degree in electrical and computer engineering in 2015 from Tufts University, Medford, MA, USA, where she is currently working toward the Ph.D. degree in electrical and computer engineering.

Her research interests include mobile eye tracking, computer vision, machine learning, and augmented reality.

Aleksandra Kaszowska received the B.A. degree in psychology from Clark University, Worcester, MA, USA, in 2012, and the M.S. degree in psychology in 2017 from Tufts University, Medford, MA, where she is currently working toward the Ph.D. degree in experimental psychology and cognitive science.

Her research interests include utilizing eye tracking and think aloud protocols for investigating cognitive processes underlying decision making, problem solving, and strategy selection in a variety of contexts.

Holly A. Taylor received the B.A. degree in mathematics from Dartmouth College, Hanover, NH, USA, in 1987, and the Ph.D. degree in cognitive psychology from Stanford University, Stanford, CA, USA, in 1992.

She is currently a Full Professor of psychology, an Adjunct Professor of mechanical engineering, and the Co-Director of the Center for Applied Brain and Cognitive Science, Tufts University, Medford, MA, USA. Her research interests include spatial cognition, spatial language, and spatial visualization.

Sos Agaian (M'98–SM'00–F'16) received the M.S. degree in mathematics and mechanics from Yerevan University, Yerevan, Armenia, in 1968, and the Ph.D. degree in math and physics in 1975 from the Steklov Institute of Mathematics, Russian Academy of Sciences, Moskva, Russia, where he also received the Doctor of Engineering Sciences degree from the Institute of the Control System in 1985.

He is currently a Distinguished Professor with the City University of New York, New York, NY, USA. His research interests include computational vision and machine learning, multimodal data fusion, signal/image processing modeling, multimodal biometric and digital forensics, three-dimensional imaging sensors, information processing and security, and biomedical and health informatics.

Dr. Agaian is a Fellow of SPIE, Society for Imaging Science & Technology, and American Association for the Advancement of Science.