Quantifying Gaze Behavior during Real World Interactions using Automated Object, Face, and Fixation Detection

Leanne Chukoskie, Shengyao Guo, Eric Ho, Yalun Zheng, Qiming Chen, Vivian Meng, John Cao, Nikhita Devgan, Si Wu, and Pamela C. Cosman

***Abstract*—As technologies develop for acquiring gaze behavior in real world social settings, robust methods are needed that minimize the time required for a trained observer to code behaviors. We record gaze behavior from a subject wearing eye-tracking glasses during a naturalistic interaction with three other people, with multiple objects that are referred to or manipulated during the interaction. The resulting gaze-in-world video from each interaction can be manually coded for different behaviors, but this process requires trained behavioral coders and is extremely time-consuming. Instead, we use a neural network to detect objects, and a Viola-Jones framework with feature tracking to detect faces. The time sequence of events when the gaze lands within the object/face bounding boxes is processed for run lengths to determine "looks", and we discuss optimization of run length parameters. The performance of the algorithm is compared against a bounding box ground truth and an expert holistic ground truth.**

***Index Terms*—eye-tracking, gaze behavior, face detection, computer vision**

I. Introduction

The emergence and refinement of social communicative skills is a rich area of cognitive developmental research, but one that is currently lacking in objective assessments of real-world behavior that includes gaze, speech, and gesture. Gaze behavior is especially important during development. Shared or joint attention provides a means by which adults name objects in the child's field of view [1]. As such, joint attention is an important aspect of language development, especially word learning [2]–[4]. Gaze behavior in children with autism spectrum disorder (ASD) is atypical in terms of social looking behavior, joint attention to objects, as well as the basic timing and accuracy of each gaze shift [5]. Recent studies report that 1 in 45 individuals is diagnosed with ASD [6], and disordered visual orienting is among the earliest signs of ASD [7] identified in prospective studies of infant siblings and it persists across the lifespan [5]. Humans use gaze as one of the earliest ways to learn about the world. Any deficit in this foundational skill compounds, leading to functional difficulties in learning as well as social domains. Difficulty shifting gaze to detect and respond to these behaviors will lead to a lower comprehension of the nuanced details available in each social interaction. Although several different therapies have been designed to address social interaction [8]–[10], methods of assessing the success of these therapies have been limited.

The outcomes of social communication therapies must be evaluated objectively within and across individuals to determine clinical efficacy. They are typically measured by parent questionnaire or expert observation, both of which provide valuable information, but both of which are subjective, may be insensitive to small changes, and are susceptible to responder bias and placebo effect. Other outcome measures such as pencil and paper or computer assessments of face or emotion recognition are objective, but measure only a subset of the skills required for real-world social communication. These measures are also a poor proxy for actual social interaction. These deficits impact both research and clinical practice.

The recent development of affordable glasses-based eye trackers has facilitated the examination of gaze behavior. The glasses fuse a calibrated point of gaze, measured by a camera below the eye, with the world-view captured by a camera mounted above the eyebrows on the glasses frame. For images displayed on computer screens, many studies have used eye-tracking to examine what portions of the images ASD children attend to [11]–[15]. Eye-tracking glasses can be used during dynamic social interactions, instead of simply on computer screens. The quantification of interactions during real-

L. Chukoskie (corresponding author) is with the Institute for Neural Computation at UC San Diego. The other authors are with the Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA, 92093–0407, USA (e-mail: {lchukoskie,pcosman}@ucsd.edu).

This work was partially supported by the National Science Foundation under Grants IIS-1522125 and SBE-0542013, and also by the National Institutes of Health R21/R33 MH096967.

world activities remains challenging. Analysis of the resulting gaze-in-world video can be done manually, but labeling and annotating all the relevant events in a 30 minute video takes many hours. Computer vision and machine learning tools can provide fast and objective labels for use in quantifying gaze behavior.

Here, we report on a system that uses eye-tracking glasses to record gaze behavior in real-world social interactions. The system detects objects and faces in the scene, and calculates fixation onsets and offsets. The results are compared to the laborious manual coding. Some past work [16], [17] has also focused on gaze quantification and social orienting in naturalistic settings, making use of software for automatic tracking of areas of interest [18] but in those works the goal was not development of tools for automating gaze analysis, and all output was reviewed by human coders to ensure high detection accuracy. The closest past work to ours is [19], [20], which also involves automatic face detection methods in a social interaction using eye-tracking glasses. Their goal is different, since they have the challenging goal of detecting eye contact events. Also their setup is different, since in their scenario, the investigator wears the eye-tracking glasses rather than the subject, and can avoid excessive motion blur and maintain steady orientation (the child's face does not go into and out of the scene), and also because the detection depth remains rather constant. In addition, our scenario has multiple faces and objects (including deformable objects). Because our naturalistic setup experiences a number of frame-level detection failures, our algorithm aims to compensate for these using a runlength algorithm that can bridge gaps in the sequence of detections.

The rest of this paper is organized as follows. The system operation including methods for detecting faces, objects, and looks is described in Section II, while creation of ground truth and calibration issues are in Section III. We define evaluation metrics and provide results in Section IV, and conclude in Section V.

II. Detecting Objects, Faces, and Looks

A. System Overview

Figure 1 presents an overview of the system operation and the formation of the different types of ground truth (GT). The Pupil Labs eye-tracking glasses (Pupil Pro) produce raw video frames (24-bit color, 720 × 1280, 60Hz) from the world-view camera and raw gaze position data at 120Hz from the eye camera. The gaze data is downsampled to the video frame rate. World-view frames are input separately to object and face detection modules, whose outputs are sets of bounding boxes, intended to bound faces and objects within the chosen set of objects. Next, the hitscan algorithm inputs a bounding box and gaze position, and puts out a binary result of whether the gaze position is inside the bounding box (a "hit"). The runlength algorithm processes these binary values to determine the presence and duration of a "look" to an object or face; a minimum runlength of hits is required to declare a look, and a gap in the hits above another threshold value determines the end of that look.

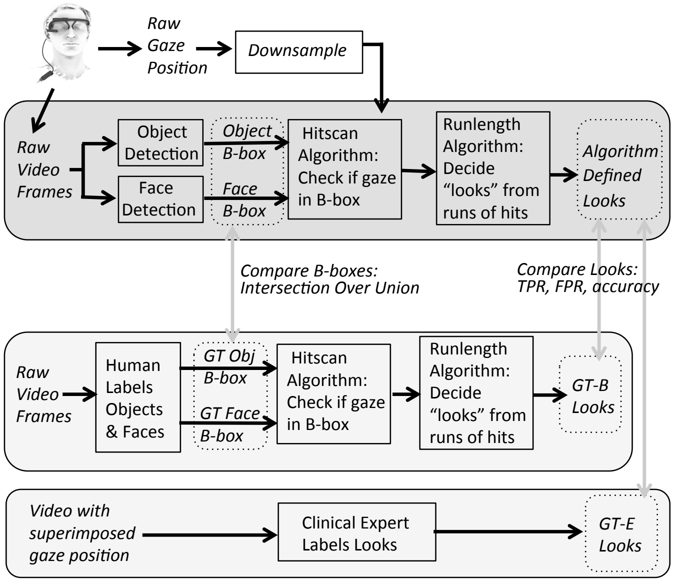

Fig. 1. Overview of the algorithm and GT formation. B-box stands for a bounding box, GT-B and GT-E are the two types of ground truth for looks, and TPR and FPR are true positive rate and false positive rate. These performance metrics will be defined in Section IV.

The lighter gray rectangles in Figure 1 depict the formation of the two GTs for looks. In one approach, humans mark bounding boxes for each object/face in each frame, without gaze position shown. These boxes and the gaze position are then passed through the hitscan and runlength algorithms to determine looks, referred to as GT-B looks. In the second approach (GT-E looks), an expert clinical neuroscientist directly labels looks by reviewing the video with superimposed gaze position in a holistic way that would be used in clinical practice. In the figure, the three vertical gray arrows show points at which the algorithm performance is evaluated. Algorithm bounding boxes are compared against the boxes marked by humans, and algorithm-defined looks are compared against GT-B and GT-E Looks.

B. Object Detection

The object detection module is based on the Faster R-CNN deep neural network [21] which classifies and localizes multiple objects in a single image. Faster R-CNN uses the Region Proposal Network for region proposal generation, improving accuracy and reducing computation compared to its predecessor Fast R-CNN.

The three objects to be detected are a photo, a spinning top, and a toy shark, shown in Figure 2. For training the neural network, images of these objects were collected using world-view video frames. At distances of 40cm and 80cm from the object, and elevation angles of 0, 30, and 60 degrees above the table, images were taken of the object on a turntable at 10 degree rotations. During testing sessions, the photo hangs on the wall unoccluded, but the top and shark might have occlusions, so top and shark images with occlusions were included in the training set. Additionally, since the shark is deformable, to make the neural net model more robust to interactions from participants, the training dataset includes images of the shark being squeezed. In total, there were 15,000 training images.

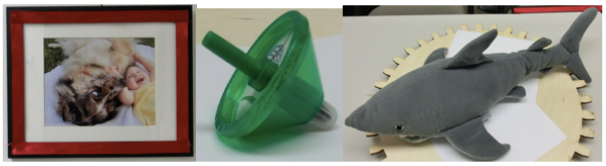

Fig. 2. Objects to be detected: photo, top, shark (shown on turntable).

In each training image, a minimum enclosing rectangle, as shown in Figure 3(a), was manually placed around the object using MATLAB's trainingImageLabeler. The object class and box coordinates serve as the GT during training. We exploit transfer learning [22] making use of pre-trained weights from VGG-16 because of its overall good performance and ability to generalize to custom datasets [23]. We used end-to-end training, and each object was trained individually to save computational time and simplify later adding of new objects if desired. Tuning the model to adapt to our custom dataset consisted of modifying the outputs of the last fully-connected layers. We trained each model with 50,000 iterations with a base learning rate of 0.001 and momentum of 0.9. The model was trained using a GeForce GTX 1080 GPU with an Intel i7-6700 CPU and 32GB RAM on Ubuntu 16.04 operating system. The software needed is OpenCV 3.1, CUDA, cuDNN, Caffe and their dependencies. OpenCV was used for image processing and manipulation, while CUDA allows for parallel computing on Nvidia GPUs such as the GeForce series, and Caffe is the framework for Faster R-CNN which uses the cuDNN library.

After this initial training, we fine-tuned the models using additional data consisting of frames from the world-view camera where some of the objects are present. GT bounding boxes were drawn for the objects, and the model performance was gauged by the intersection over union (IoU) of the bounding boxes from the human labelers and from Faster R-CNN outputs. If the IoU is less than 75%, we tuned the hyperparameters, such as the learning rate and the mini-batch size, and re-trained the corresponding object model.

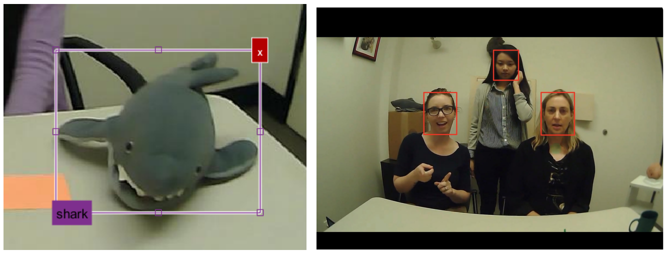

Fig. 3. (a) Minimum enclosing rectangle for a shark image, (b) Test image with manual ground truth bounding boxes drawn

C. Face Detection

The face detection and tracking system, illustrated in Figure 4, consists of five functional modules: Viola-Jones face detection, Shi-Tomasi corner detection, eigenfeature tracking with optical flow, tracking points averaging, and adjustment and reinitialization upon failure. The main function block consists of all the functional modules except for the Viola-Jones face detection. Viola-Jones face detection is executed once for each frame of the video, and the main function block is executed M times for each frame, where M (determined manually) is the number of faces appearing in the video.

The Viola-Jones algorithm [24] is applied to detect faces. Its output is a set of bounding boxes that may contain faces, and that are unlabeled (i.e., it does not attempt to establish which face is which). At the start of the video (and again whenever there is a tracking failure) the Viola-Jones output requires human intervention to manually select and label a bounding box containing a face. After bounding box selection, the corner detection module is triggered, and the tracking loop is engaged.

The Shi-Tomasi corner detector extracts features and scores them [25] using eigenvalues of a characteristic matrix based on image derivatives. The optical flow of each extracted eigenfeature is calculated to track it in subsequent frames [26]. The average position of all the trackers is calculated and checked against the face detection boxes output from Viola-Jones for the next frame. If at least 30% of the tracked points are not lost, and if the average position of the tracked points is inside one of the detected face boxes, that is considered a tracking success

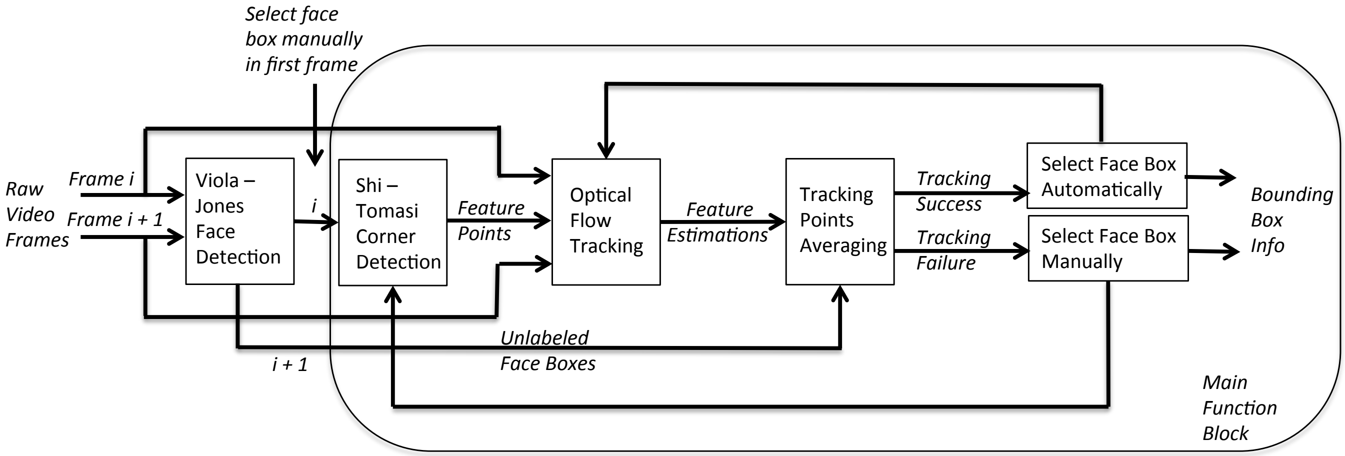

Fig. 4. Block diagram of the face detection module.

and the face bounding box is output; the algorithm then continues the tracking loop to the next frame (or moves to a different face if there is another face being tracked). Trackers will continue to function even without a valid detected area and will select the first detected area when available. If 70% of the feature points are lost, or if the average tracker position is not inside any of the detected face boxes, then it is considered a tracking failure. The system then requires human intervention to re-locate the face locations, and the algorithm automatically re-initializes the trackers. In a 3-minute video consisting of approximately 10,000 frames, there are typically 20-30 face re-initializations required (person has to click on the correct face box). The re-initialization typically happens because the subject turns her head and the face exits the field of view, needing re-initialization when it comes comes back into view, or because the face in the view gets temporarily occluded (e.g., by a hand or object).

D. Determination of Looks

The object and face detection modules produce bounding boxes around objects and faces in the video frames. For any single object (or face), if the algorithm produces a bounding box in frame i for that object, and if the gaze position is within that bounding box, the hitscan algorithm considered that frame a "hit" for that object. Otherwise, it is a "miss". The hitscan algorithm is run for each object/face in a video.

Next, the runlength algorithm processes the hit sequences to determine looks. If there is a runlength of at least T_1 hits, then that is considered a look, and the first hit position is the start of the look. Because of blinks or noise, a small runlength of misses is not considered to end the look. With a runlength of T_2 misses in a row, the look is considered terminated, and the last hit position is the end frame of the look. The choice of parameters T_1 and T_2 is discussed in Subsection III-C.

III. Ground Truth

GT represents a determination of the true presence of faces and objects and the number and length of looks. GT serves as the basis for evaluating the algorithm results. We define GT for bounding boxes, and two types of GT for looks.

A. Ground truth for bounding boxes

GT for face and object bounding boxes was established by manually placing tight axis-aligned enclosing rectangles around each face and object in the image. The protocol for drawing a face box was that the right and left limits should include the ears if visible, while the upper limit is at the person's hairline and the lower limit is at the bottom of the chin. An example of manual GT bounding boxes is in Figure 3(b). A face is not boxed if the face is turned more than 90 degrees away from the camera. A small number of faces were not boxed in the manual GT because the subject wearing the eye-tracking glasses turned his or her head rapidly, so the world-view frames had excessive motion blur. An example motion-blurred image which does not get manual GT is in Figure 5(a). It is possible, however, for the algorithm to detect a face even though it is turned more than 90 degrees or is blurry; such cases would count as false positives since they are not marked in the GT. So the results are slightly conservative on false positives.

For drawing bounding boxes for the top and photo, the box contains all of the object in the picture, and is drawn only if 50% or more of the object is judged to be present. For the shark object, a box was drawn if 50% or more is present and both eyes are present in the picture. Again, this protocol will make the algorithm results conservative on false positives.

B. Ground truth for looks

As shown in Figure 1, the GT for looks was established in two different ways. In one approach, an eye-tracking expert determined the GT for looks based on her experience with clinical gaze data, by directly viewing the world-view video in which the dot representing gaze position is superimposed on the scene (Fig. 5(b)). The gaze dot consists of a central red dot (indicating the best estimate of gaze position from the eye-tracking glasses) surrounded by a larger green dot (indicating the glasses' estimate of gaze position uncertainty). The expert does not make any use of explicit bounding boxes, but determines holistically, as they would in clinical or experimental practice, what the subject is looking at. This approach is inherently inferential, and therefore subject to a number of biases. For example, the user may consider that a set of frames corresponds to a single look to a face, despite a short temporal gap in the presence of the gaze dot on the face that may be due to the subject blinking, the subject shifting their head position and producing motion blur, or a reduction in calibration accuracy because of the glasses being jiggled on the subject's head. Indeed, it may happen in practice that the expert notices a calibration error because the gaze dot is consistently slightly below each object and face, and so marks a section as a look to an object because they know the subject "intended to look at the object" even though the gaze dot is off. Our videos were calibrated, so this level of subjectivity was not present in the expert GT (called GT-E), but some level of subjective expert judgment is inherent in this process. It is useful to include this type of GT since it is what is actually used currently in analyzing social gaze behavior.

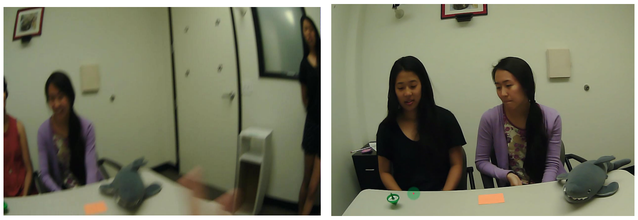

Fig. 5. (a) Example of a motion-blurred image that is not given bounding boxes, (b) test image with the gaze dot superimposed on the world view.

The other type of GT for looks, referred to as GT-B, uses the manual bounding boxes. As shown in Figure 1, GT-B is established by putting the gaze position and manually-derived bounding boxes for objects/faces through the same hitscan and runlength algorithms that are applied to the automatically-derived bounding boxes.

C. Entry and exit parameters

One approach to choosing entry and exit parameters T_1 and T_2 is based on physiology and eye behavior. At 60 fps, 5 frames represents 83 ms, and this is a reasonable lower bound duration of a single fixation. Intervals of fixation are typically identified as the period of gaze stability between the fast orienting saccadic eye movements. Typically, a standard fixation duration is about 200-300 ms in standard experimental studies with controlled target appearance and standard screen refresh rates [27]. However, we are measuring gaze behavior in the real world. Human observers typically plan sequences of saccades, especially when scanning a complex object [28] and for those sequences, the fixation duration can be quite short. Depending on task demands and the subject's level of focus, fixations can also be quite long, approaching 2 s. Physiologically speaking, the fixation need only be long enough for the visual system to extract relevant information in high resolution detail before moving to a new spot to examine. Data from visual psychophysics demonstrates that image detail can be resolved with a presentation of only 50 ms, followed by an immediate mask to prevent the use of after images [5]. Given this approximate lower bound, the T_1 value could be even lower than 5 frames, however in practice, we do not typically see fixations this brief.

From the eye physiology point of view, the T_2 exit parameter needs to be long enough to bridge a blink. We do not want to declare the end of a look because the subject blinked and the gaze position was uncertain for some frames, causing the hitscan algorithm to declare a short sequence of misses. In addition, a long "look" to a complex target such as a face can encompass several nearby fixations, for example a series of fixations to each eye and the mouth or perhaps the nose in a triangle scan path, with other fixations occurring outside this pattern but typically less common. One of these fixations might fall outside the bounding box errantly, and so a standard duration of that fixation could usefully be considered a good value for T_2.

The selection of T_1 and T_2 based solely on eye physiology ignores the fact that the detection problem is difficult due to occlusions, object deformability, rapid turning of the subject's head and other reasons. A second approach to choosing these parameters is based on making the algorithm mimic the behavior of the expert neuroscientist. That is, one could choose T_1

and T_2 so as to make the best agreement between the algorithm results and GT-E. As will be discussed later, the definition of "agreement" between the algorithm and GT-E has many possible definitions corresponding to different applications. As one case example, we measure "agreement" using the Accuracy defined in Equation (1) below. Figure 6 shows a heat map in which the color shows the accuracy of the algorithm results relative to GT-E when the algorithm uses runlength entry and exit parameters given by the values on the x and y axes.

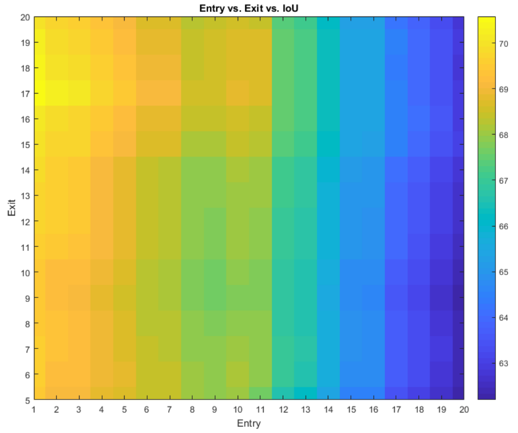

Fig. 6. Heat map of Algorithm and GT-E agreement as a function of runlength entry (T_1) and exit (T_2) parameters

From Figure 6, we see that the optimal values are T_1 = 1, and T_2 = 17. Although quite far from the values that would be suggested by physiologic reasons, these values are understandable when one considers the operation of the algorithm. Consider first the exit parameter T_2. Suppose the expert neuroscientist observes that the subject gazes at a face, and then the face turns away momentarily, and turns back. The expert judges that the gaze remains on the face the entire time, a look that spans 100 frames. The algorithm gets 30 frames of a look but loses track when the face turns. The face gets re-acquired by the Viola-Jones detection module after a gap of 12 frames. With a small value of T_2, this would be considered by the algorithm to be two distinct looks with a gap in between. A large value of T_2 = 17 bridges the gap; the entire set of frames, including the gap frames, constitute one long look, making for good agreement with GT-E. In short, choosing T_2 somewhat larger than the physiologic causes would suggest allows the runlength algorithm to compensate for deficiencies in the detection modules.

Consider now the entry parameter T_1. Here the optimal value equals 1, meaning a single hit frame should be considered the start of a look. This is smaller than fixation data would suggest. When the algorithm finds a single frame that has a gaze dot in the bounding box, if there are no further hit frames within a distance of T_2, then this is very unlikely to correspond to a look in GT-E. Yet the cost of calling this a look is small, since it is a single frame. On the other hand, if there are other hit frames within a distance of T_2 then the algorithm with T_1 = 1 will declare the start of the look and will connect to the later frames, so all those frames get declared to be part of the look. There is a higher chance that this look was marked by the expert in GT-E, and there might be many frames in the look. In other words, taking T_1 = 1 will cause more frames overall to be declared part of looks, so there will be both more false positives and more true positives. The false positive frames will usually incur little accuracy penalty because they are individual frames, but the true positive frames will usually be part of a larger look event, thereby gaining a significant increase in accuracy.

A different approach to computing algorithm accuracy could use whole look events, rather than frames within looks, as the basis for correctness. Consider the case where GT-E reports a single long look of 50 frames in the first 100 frames. Suppose the algorithm detects that same look exactly, and also three isolated single frames as being looks. In a frame-based approach to counting correctness, the false positive rate is 3 / (50 + 3) = 5.7%, whereas in a look-based approach to counting correctness, the false positive rate would be 3 / (1 + 3) = 75%. Taking T_1 = 1 would make sense if the first measure needs to be optimized (which is what our frame-based accuracy results are), but it would do poorly for a look-based approach. For this reason, we chose T_1 = 5 and T_2 = 17 for the runlength algorithm, which still has reasonable performance on the frame-based accuracy metric, but will do better if the goal is to count looks.

D. *Camera calibration and accuracy issues*

Calibration is required to ensure that the gaze position in the world-view scene corresponds to what the subject is looking at. Calibration is enabled through a Pupil Capture software routine; the subject wears the glasses and looks steadily at a bullseye target (in 9 different positions in a vertical plane approximately 1m from the subject) that is recognized by the Pupil Capture system. Once the calibration routine is completed, we validate it by asking the subject to look at different parts in the scene and confirm that the gaze point represented in Pupil Capture is where the subject reports looking.

Accuracy issues: The world view camera mounted on the glasses captures the world in the direction the head is facing. Typically, the eyes look forward, and so the gaze

	Acc.	FPR	FNR	Den	IoU
face1	85.24	9.58	6.3	2825	79.55
face2	77.06	8.22	17.23	2363	64.72
face3	83.23	6.56	11.6	3506	80.77
photo	68.88	21.74	3.6	4753	77.16
shark	81.43	8.18	10.21	3576	78.27
top	71.91	18.53	14.04	1431	69.32
total	77.83	12.0	9.92	18454	76.04

TABLE I
AVERAGE RESULTS ACROSS FIVE VIDEOS FOR THE ALGORITHM AND GT-B, WHERE BOTH USE T_1 = 5, T_2 = 17. DEN = NUMBER OF FRAMES IN THE DENOMINATOR OF EQUATION (1) ENTERING INTO THE ACCURACY COMPUTATION FOR EACH FACE AND OBJECT.

	Acc.	FPR	FNR	Den
face1	72.21	3.42	25.89	3242
face2	61.4	11.55	33.24	2394
face3	68.65	7.2	27.5	3757
photo	50.86	23.8	19.91	4198
shark	80.43	10.82	7.63	3055
t op	67.17	17.5	21.68	1334
total	66.06	12.01	22.82	17980

TABLE II
AVERAGE RESULTS ACROSS FOUR VIDEOS, FOR THE COMPARISON OF THE ALGORITHM AND GT-E, WHERE THE ALGORITHM USES PARAMETERS T_1 = 5 AND T_2 = 17.

position is rarely at the extreme edges of the world view scene. Gaze location data show that the eyes spend less than 5% of the time looking at the area that is within 20% of the edge of the field of view. Furthermore, when the eyes do shift to the side, the glasses have greater gaze position uncertainty. The confidence score for gaze location reported by the Pupil Pro is 98.6% for gazes to the central 10% of the world-view scene, and this sinks to 89.8% confidence for gazes to the outer 10% portion of the scene. For these reasons, bounding boxes that touch the scene border (meaning the object is cut off by the border) are ignored in the performance evaluation. That is, if the GT bounding box coincides with one generated by the algorithm, the performance evaluation does not count this as a true positive. Furthermore, if the algorithm does not output a bounding box for the object at the border, then the performance is not penalized as a false negative. Such frames are simply ignored in the performance evaluation.

IV. RESULTS

For a given face or object (e.g, the shark) we first evaluate bounding boxes. We compute for each frame the area of intersection divided by the area of union (IoU) of the algorithmic and manual bounding boxes for that object. The IoU values are averaged over frames, and over five videos, and reported in the last column of Table I.

We next evaluate the algorithm at the level of frames within looks. A sample of the results for faces in one video appears in Figure 7, and a sample of results for objects is in Figure 8.

Frame i represents a true positive event for a look to face 1 if frame i is part of a look to that face according to GT and frame i is also part of a look to that face in the algorithm output. Recall that for frame i to be part of a look to a face does not require that the gaze is within the face bounding box for frame i, or even that the face was detected in that frame. If the face was detected and the gaze was inside its bounding box for earlier and later frames, and frame i is part of a sufficiently short gap, then frame i can still be considered part of the look. The following provides an example. The first row is the binary sequence of hits for algorithm bounding boxes, that is, a value of 1 represents a case where the gaze position is inside the algorithm-derived bounding box:
row 1: 00011111100111100000000000000000000001111111

Although in this sequence, frames 10 and 11 do *not* have the gaze inside the bounding box, the runlength algorithm considers that two frames is too short to be interpreted as looking away. So those frames are considered part of the look, and the frame-level algorithm output for looks is as follows:
row 2: 00011111111111100000000000000000000001111111

So performance measures, such as true positive events, are counted between this sequence for the algorithm and GT-E (or between the algorithm and the corresponding runlength-processed bounding box ground truth GT-B). Let TP represent the total number of true positive events for that video and that object. A false positive event (for a given object) occurs when frame i is not part of a look to that object according to GT but is part of a look to that object for the algorithm. FP denotes the number of false positive events. Similarly FN is the number of false negative events, when a frame is part of a look according to GT but the algorithm does not mark it. Lastly TN represents the number of true negative events, where neither the GT nor the algorithm considers a look to be occurring in a given frame. False Positive Rate and False Negative Rate are defined as $FPR = FP/(FP + TP)$ and $FNR = FN/(TP + FN)$. One standard definition of accuracy is $A = (TP+TN)/(TP+FP+TN+FN)$ however since the subject is often not looking at any of the objects or faces under consideration, TN is large, and including it in both the numerator and denominator would obscure the trends. Instead we use the definition

$$A = TP/(TP + FP + FN) \tag{1}$$

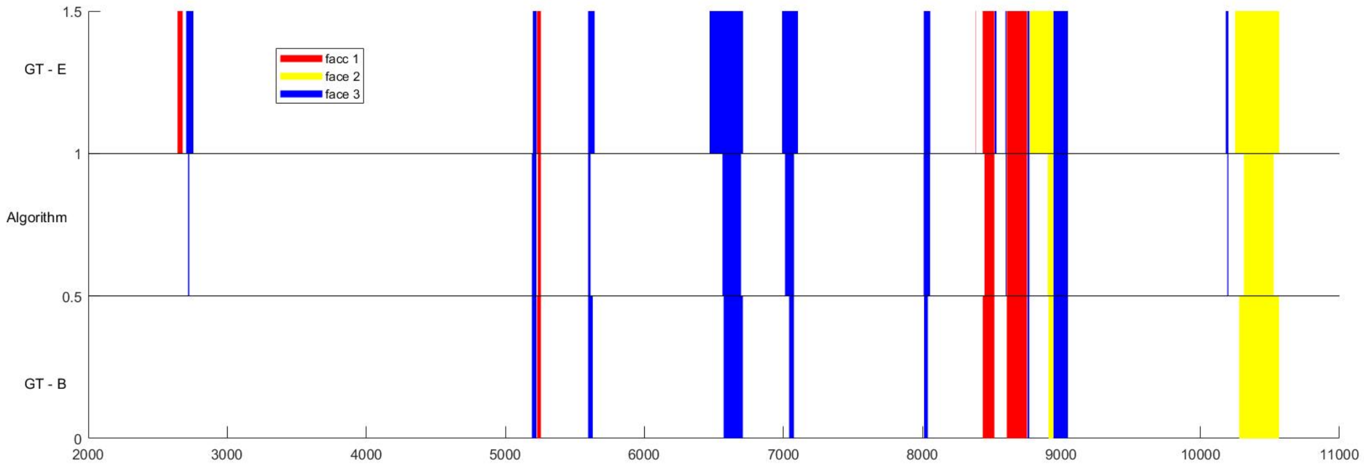

Fig. 7. Example of algorithm results and both GTs for three faces in one video. The x-axis shows the frame number. The y-axis shows, from top to bottom, GT-E, algorithm looks, and GT-B for faces.

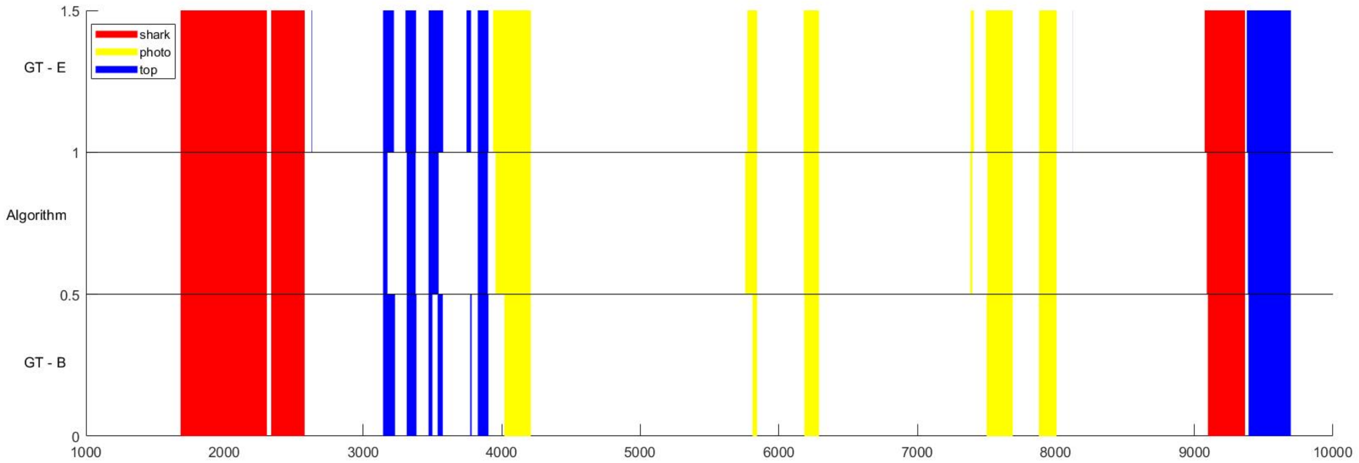

Fig. 8. Example of algorithm results and both GTs for objects in one video. The x-axis shows the frame number. The y-axis shows, from top to bottom, GT-E, Algorithm results, and GT-B for objects.

The values for accuracy, FPR, and FNR for each object/face, averaged across videos, are in Table 1 (relative to GT-B) and in Table 2 (relative to GT-E).

A. Discussion

The faces are labeled 1,2,3 from left to right, and the middle face (face2) is usually farther back in the scene. We see that the average IoU values for faces 1 and 3 are very similar (79.55 and 80.77) but the average IoU is worse for face2. Among the three objects, the average IoU values are similar for the shark and the photo (78.27 and 77.16) and the value is lower for the top (69.32) likely because the top is a much smaller object.

We see that the Accuracy results for looks to faces 1 and 3 are again better than those for face 2. Of the objects, the photo has a high FPR. This is driven by the fact that the photo framing colors (red, black, white) are common clothing colors worn by the participants, and the photo itself shows faces, causing non-photo items to be detected as photos, or causing the algorithm's photo bounding box to be drawn too large. With the exception of the photo, the Accuracy rates for looks are all higher than the IoU measures of bounding box accuracy, suggesting that lack of precision in the bounding boxes can to some degree be compensated for by the runlength algorithm that declares looks.

Counting FP and FN events at the level of entire looks, rather than, as we do, at the level of frames within looks, would change the numeric results. For example, in examining the results in Figure 7, we see that the photo has 5 entire "look events" in the GT but 6 in the algorithm, leading to a FPR of 0.17 if one counts entire look events. However the FPR is different if one counts at the frame level, since several of the look events have extra FP frames at the leading edge of the event. Whether or not it is desirable to count FP and FN events at the level of entire look events or at the level of frames, or indeed whether some completely different metrics are needed, will depend on the application.

B. Consideration of Different Applications

There are research, educational, and clinical applications for which it would be useful to have a system that can automatically identify looks to faces and objects as part of a real-world interaction. These applications vary in their spatial and temporal demands in terms of what constitutes a look, which has a bearing on the values of T_1 and T_2 and on other aspects of the system.

Consider a child reading a middle school science textbook. One might want to identify when the student is reading the columns of text, approximately at what point in that text the student jumps to a figure box, how long the student spends in the figure box and where the student's gaze goes after the box (ideally back to the point in the text where she left off). Since the primary interest is in mapping gaze onto the textbook, we would want to optimize the spatial accuracy of looks within the book (and not worry about the background). We would not be as concerned about temporal precision in this case. It is useful to know that the child spent about 3.2 seconds reviewing the figure, but it is not necessary to know that she entered it on frame 80 and left on frame 272. In a clinical example, an adolescent with ASD might wear the gaze glasses and engage in a conversation and a game with two other people. We could identify all looks to faces and quickly calculate the proportion of time spent looking at faces during the interaction as a whole, a potentially useful measure in a social evaluation. For cases such as these where overall time spent looking at a face or object is important but not the number or precise onset of looks, the T_1 and T_2 parameters can be set to the values which optimize the accuracy with GT-E for that task (in our specific case, that would be the values T_1 = 1 and T_2 =17).

Separate from the total time, it may also be useful to know the *number* of separate looks. If the child looks back and forth ten times between the text and the figure box, it may be a sign that the figure is confusing, or that the figure has insufficient labeling. When counting correctly the number of looks is the primary goal, the optimization of T_1 and T_2 relative to GT-E would use the count as the optimization goal, which would lead to a larger value of T_1.

Taking the clinical example further, the adolescent and two research assistants might all wear gaze glasses and the data streams are synchronized. One might like to know how quickly after one assistant turns to the other does the adolescent also turn to look at the assistant. The latency to orient to a social cue is a useful part of a social evaluation since slow orienting behavior can result in missed information. However, whenever we intend to calculate latency, the temporal precision in the onset and offset of a look matters a great deal.

These various applications with various requirements suggest that the algorithm parameters can usefully be tailored for different scenarios. For cases where the spatial precision is important, a restricted region of interest (e.g., the textbook) can be precisely calibrated. Also, allowing some padding region outside the algorithm bounding box for where the gaze location counts as a “hit”, or conversely, tightening up the region which counts as a 'hit” might allow for greater accuracy optimization between the algorithm and GT-E.

V. Conclusions

This project brings together multiple different technologies to enhance our understanding of gaze behavior in real-world situations. Currently, the use of real-world eye-tracking is limited because the first-to-market glasses-based eye-trackers were expensive, and the resulting gaze-in-world data was difficult to analyze in any automated or semi-automated way. The open source model offered by Pupil Labs has made glasses-based eye-tracking both affordable and customizable. The system developments described here allow us to automate the count and duration estimate of looks to faces and objects during a social interaction. Because of the prevalence of ASD and its social interaction challenges, together with the subjectivity and difficulty in current methods for assessing the success of therapeutic efforts, investing in objective and quantitative social outcome measures can be useful to measure efficacy of social therapies.

One contribution of this work is the system integration involving both face and object detection in the context of naturalistic social interactions with varied motion of the subject and other participants. But the main contribution is the approach to determining looks, involving a runlength algorithm whose parameters are set by optimizing a suitable definition of agreement between the algorithm looks and an expert ground truth. The definition of agreement can be modified depending on the application. The detection accuracy of our modules is already sufficiently high for many clinical or educational evaluation purposes, and superior detection algorithms could be substituted in a modular way for the current methods, retaining the optimized runlength approach to determining looks as a postprocessing method after any detection algorithm.

Our long-term goal is to develop a system using gaze glasses and analytic software to assess change in social and communicative behavior in individuals at a range of ages and levels of function. We plan to include methods for automated sound and voice detection as well as gesture detection. Our next steps include identifying instances in time (trigger points) from which one might want to calculate latencies. Audio triggers might include a knock on the door, the onset of speech in general, or when a participant's name is spoken. Visually identifiable trigger points include pointing movements, head turns and other gestures.

Acknowledgments: We thank Ms. Sarah Hacker who helped with data collection. An early version of this work has been submitted to ISBI 2018. The current paper extends that by adding expert ground truth GT-E (Sec. III.B), algorithm parameter optimization (Sec. III.C) new results (in Sec. IV), as well as Section IV.B.

References

[1] P. Mundy, "A Review of Joint Attention and Social-Cognitive Brain Systems in Typical Development and Autism Spectrum Disorder," *European Journal of Neuroscience,* Sep 18., 2017.

[2] D.A. Baldwin, "Understanding the link between joint attention and language," In C. Moore & P.J. Dunham (Eds.) *Joint attention: Its origins and role in development,* (pp.131-158). Hillsdale, NJ: Erlbaum, 1995.

[3] M. Hirotani, M. Stets, T. Striano, and A.D, Friederic, "Joint attention helps infants learn new words: event-related potential evidence," Neuroreport. 2009 Apr 22;20(6):600-5.

[4] B.R. Ingersoll and K.E. Pickard, "Brief report: High and low level initiations of joint attention, and response to joint attention: differential relationships with language and imitation," *Journal of Autism and Developmental Disorders,* 45(1):262-8, January 2015.

[5] J. Townsend, E. Courchesne, and B. Egaas, "Slowed orienting of covert visual-spatial attention in autism: Specific deficits associated with cerebellar and parietal abnormality," *Development and Psychopathology,* 8(3): 503-584, 1996.

[6] B. Zablotsky, L.I. Black, M.J. Maenner, L.A. Schieve, and S.J. Blumberg, "Estimated Prevalence of Autism and Other Developmental Disabilities Following Questionnaire Changes in the 2014 National Health Interview Survey," National Health Statistics Report, 87, November, 2015.

[7] L. Zwaigenbaum, S. Bryson, T. Rogers, W. Roberts, J. Brian, and P. Szatmari, "Behavioral manifestations of autism in the first year of life," *International Journal of Developmental Neuroscience,* Apr-May;23(2-3), pp. 143-152, 2005.

[8] L. Schreibman and B. Ingersoll, "Behavioral interventions to promote learning in individuals with autism," In Handbook of autism and pervasive developmental disorders: Vol., Edited by: F. Volkmar, a. Klin, R. Paul, and D. Cohen, Vol. 2, pp. 882-896, 2005, New York, NY: Wiley.

[9] E.A. Laugeson, F. Frankel, C. Mogil, and A.R. Dillon, "Parent-Assisted Social Skills Training to Improve Friendships in Teens with Autism Spectrum Disorders," *J Autism Dev Disord,* 39:596-606, 2009.

[10] P.J. Crooke, L. Olswang, and M.G. Winner, "Thinking Socially: Teaching Social Knowledge to Foster Behavioral Change," *Topics in Language Disorders*, July/September, Volume 36, Issue 3, pp 284-298, 2016.

[11] M. Chita-Tegmark, "Social attention in ASD: a review and meta-analysis of eye-tracking studies," *Research in Developmental Disabilities,* 48:79-93, 2016.

[12] K. Chawarska and F. Shic, "Looking but not seeing: Atypical visual scanning and recognition of faces in 2 and 4-year-old children with autism spectrum disorder," *J. Autism and Developmental Disorders,* Vol. 39, No. 12 p. 1663, 2009.

[13] M. Hosozawa, K. Tanaka, T. Shimizu, T. Nakano, and S. Kitazawa, "How children with specific language impairment view social situations: an eye tracking study," *Pediatrics,* Vol. 129, No. 6, e1453-e1460, 2012.

[14] K. Pierce, D. Conant, R. Hazin, R. Stoner, and J. Desmond, "Preference for geometric patterns early in life as a risk factor for autism," *Archives of General Psychiatry,* Vol. 68, No. 1 pp. 101-109, 2011.

[15] A. Klin, W. Jones, R. Schultz, F. Volkmar, and D. Cohen, "Visual fixation patterns during viewing of naturalistic social situations as predictors of social competence in individuals with autism," *Archives of General Psychiatry,* Vol. 59, No. 9, pp. 809-816, 2002.

[16] S. Magrelli, P. Jermann, B. Noris, F. Ansermet, F. Hentsch, J. Nadel and A. Billard, "Social orienting of children with autism to facial expressions and speech: a study with a wearable eye-tracker in naturalistic settings," *Frontiers in Psychology,* Vol. 4, Nov. 2013.

[17] B. Noris, J. Nadel, M. Barker, N. Hadjikhani, and A. Billard, "Investigating Gaze of Children with ASD in Naturalistic Settings," *PLOS One,* Vol. 7, Issue 9, Sept. 2012

[18] B. Noris, K. Benmachiche, J. Meynet, J.P. Thiran, and A.G. Billard, "Analysis of head mounted wireless camera videos," *Comp. Recognit. Syst. 2*, 663-670, 2007.

[19] E. Chong, K. Chanda, Z. Ye, A. Southerland, N. Ruiz, R.M. Jones, A. Rozga, and J.M. Rehg, "Detecting Gaze Towards Eyes in Natural Social Interactions and Its Use in Child Assessment," Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, Vol. 1, No. 3, Article 43, Sept. 2017.

[20] Z. Ye, Y. Li, A. Fathi, Y. Han, A. Rozga, G.D. Abowd, and J.M. Rehg, "Detecting Eye Contact using Wearable Eye-Tracking Glasses," UbiComp 2012, Sep. 5-8, Pittsburgh, USA.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," NIPS, Montreal, Canada, Dec. 2015.

[22] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both Weights and Connections for Efficient Neural Network," Neural Information Processing Systems Conf., Montreal, Canada, 2015.

[23] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", arXiv: 1409.1556v6[cs.CV], Apr. 2015.

[24] P. Viola and M.J. Jones, "Robust Real-Time Face Detection," Intl. Journal of Computer Vision 57(2), pp. 137-154, 2004.

[25] J. Shi and C. Tomasi, "Good Features to Track", IEEE Conf. on Computer Vision and Pattern Recognition, Seattle, June 1994.

[26] B.D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," in Proc. 7th international Joint Conference on Artificial Intelligence (IJCAI), Vancouver, B.C., pp. 674–679, August 24-28, 1981.

[27] D. D. Salvucci and J. H. Goldberg, "Identifying Fixations and Saccades in Eye-Tracking Protocols," ETRA Proc. Symp. on Eye Tracking Research & Application, pp. 71-78, 2000.

[28] G.T. Buswell, *How People Look at Pictures: a Study of the Psychology of Perception in Art*, Chicago, IL: University of Chicago Press; 1935.