Detecting Screen Presence with Activity-Oriented RGB Camera in Egocentric Videos

Amit Adate[1], Soroush Shahi[2,3], Rawan Alharbi[2,3], Sougata Sen[4],
Yang Gao[2,3], Aggelos K Katsaggelos[1,2] and Nabil Alshurafa[2,3]

Email: amitadate@u.northwestern.edu[1], soroush@northwestern.edu[2,3], rawan.alharabi@northwestern.edu[2,3]
sougatas@goa.bits-pilani.ac.in[4], yang.gao@northwestern.edu[2,3], aggk@eecs.northwestern.edu[1,2], nabil@northwestern.edu[2,3]

[1]Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL, USA
[2]Department of Computer Science, Northwestern University, Evanston, IL, USA
[3]Department of Preventive Medicine, Northwestern University, Chicago, IL, USA
[4]Department of Computer Science and Information System, BITS, Pilani, Goa, India

2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops) | 978-1-6654-1647-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/PerComWorkshops53856.2022.9767433

***Abstract*—Screen time is associated with several health risk behaviors including mindless eating, sedentary behavior, and decreased academic performance. Screen time behavior is traditionally assessed with self-report measures, which are known to be burdensome, inaccurate, and imprecise. Recent methods to automatically detect screen time are geared more towards detecting television screens from wearable cameras that record high-resolution video. Activity-oriented wearable cameras (i.e., cameras oriented towards the wearer with a fisheye lens) have recently been designed and shown to reduce privacy concerns, yet pose a greater challenge in capturing screens due to their orientation and fewer pixels on target. Methods that detect screens from low-power, low-resolution wearable camera video are needed given the increased adoption of such devices in longitudinal studies. We propose a method that leverages deep learning algorithms and lower-resolution images from an activity-oriented camera to detect screen presence from multiple types of screens with high variability of pixel on target (e.g., near and far TV, smartphones, laptops, and tablets). We test our system in a real-world study comprising 10 individuals, 80 hours of data, and 1.2 million low-resolution RGB frames. Our results outperform existing state-of-the-art video screen detection methods yielding an F1-score of 81%. This paper demonstrates the potential for detecting screen-watching behavior in longitudinal studies using activity-oriented cameras, paving the way for a nuanced understanding of screen time's relationship with health risk behaviors.**

***Index Terms*—Object Detection, Egocentric Videos, Fisheye Lens, Wearable Camera**

I. Introduction

In recent years, there has been a growing interest in the use of wearable cameras to automate the detection of screens to improve our understanding of screen time behavior. Screen time behavior is known to be associated with an increase in sedentary behavior (and hence reduction in physical activity), mindless eating activity which increases calorie intake, and it has also been shown to have negative effects on academic performance among youth [1]. Specifically, there are studies that show that there is a correlation between screen time and eating habits [2]. Being able to successfully detect screens in a scene can help understand an individual's fine-grained context better. However, in studying screen behavior, prior research has predominantly relied on controlled lab studies and/or self-report, the accuracy of which is known be affected by memory recall.

While several techniques have been proposed using thermal sensors and photovoltaic effects in PN junction, to reliably visually confirm screen behavior, researchers have turned their attention to optimizing image processing methods to detect screens from wearable cameras. Recent use of egocentric cameras (i.e., first-person view cameras oriented to approximate the wearers field of view) have shown promise in capturing screens in real-world settings, however, the majority of existing work has focused on detection of screens from high-resolution wearable cameras. While promising, high resolution cameras impose battery constraints, impacting our ability to deploy such a system longitudinally to capture behavior throughout an entire day. The utility of wearable cameras in longitudinal studies has stipulated the need for low-resolution day-long wearable cameras that mitigate privacy concerns. The performance of these models in detecting screens from such cameras has not been adequately studied in real-world environments.

In this paper, we explore the possibility of detecting the presence of screens in activity-oriented video cameras. Activity-oriented cameras have been shown to reduce privacy concerns by orienting the lens towards the wearers face, but a fish-eye lens has the potential to capture other context in the surrounding environment, such as screens. Specifically, we aim to answer the following question: "*Can we automatically detect screens from low-resolution activity-oriented video cameras worn by participants in a real-world environment?*". We define screens more generally to comprise the latest technological manifestations ranging from smartphones and tablets to laptops/desktops and infrastructure-deployed televisions. Being able to reliably detect screens in low-resolution cameras will ultimately enable improved and timely understanding of

screens as it relates to health behaviors, and also enable timely interventions to create and maintain behavior change.

Our task of detecting screens from video has several challenges: (1) Screens can be of various shapes and sizes. So, our screen-detection system should be capable of automatically detecting screens of any size in a video. (2) The position and size of the screen in an activity-oriented camera can vary. Users can interact with their mobile phones in numerous ways and thus the system should be robust enough to consider the position and size of the screen. (3) Distant objects in low-resolution videos result in lower pixel-count on target (screen). Indeed, it is sometimes difficult for even a human to detect screens in these videos. Finally, (4) Objects, including a wearer's hand, can occlude part of the screen. Thus, it is necessary to detect screens even when they are only partially visible.

In this paper, we address these challenges and design a system that can robustly detect screens from low-resolution wearable activity-oriented cameras. This investigation makes the following contributions:

1) We present the design of a system that can robustly detect screens in real-world settings. Through a user study with 10 participants with obesity, we show that the system can detect screens with an F1-score of 0.81.
2) We present the architecture of the image processing model for detecting screens. We show that compared to existing screen detection models, our model can detect screens from low-resolution egocentric videos with 30% improvement over existing state-of-the-art methods.

This paper is organized as follows: Section II considers the related work in the space of egocentric vision and object detection using wearable devices. Section III explains the data collection hardware along with the curation of our dataset. Section IV describes the system framework which involves the network architecture and pipeline. Section V-A defines our evaluation process, shares the performance of our system, and draws comparisons with existing benchmarked methods. Section VI elaborates on the key challenges we faced during our experiment.

II. Related Works

In this section we provide an overview of research on activity recognition using computer vision algorithms and discuss the relevant works in screen activity detection and localization. There has been significant prior research in activity recognition using egocentric videos. In this realm, many studies have used screen exposure tracking and its association with health-related behaviors. Screen prediction is increasingly taking on multiple form factors. Several large-scale object-detection models have "phone," "tablet," and "television" in their label space. Recent screen-detection techniques use deep learning approaches to produce either a bounding box or object segmentation as output.

Zhang et al. used a head-mounted, egocentric camera to track head motion and detect screen watching, and attained an F-score of 0.8 in detecting screens [3]. Zhang et al. used a computational model to track head movements correlated with TV watching [3]. Their approach uses head tracking along with an attention mechanism to gauge the amount of screen exposure to the participant. In addition to understanding screen interaction, their work focused on detecting and tracking the screen within the participant's field of view. Harper et al. [4], on the other hand, used eye tracking. They statistically computed the number of times when the participant changed their attention and the activity associated with the attention shift. Apaolaza et al. developed an approach involving manual labeling to classify the gaze of a participant toward a screen [5]. They created an open source tracking technique called ABC (Automated Behavioral Coding) [6]. ABC logs participant movement and interaction and involves tracking the positional data. This approach is slow and involves excessive manual annotation effort, but its performance is very high in contrast with other approaches. Several researchers have combined gaze estimation with object detection [7], [8]. Egocentric videos are often high resolution and researchers have identified approaches to extract coarse-grained activities of the user and contextual information from them [9]. The key difference between these approaches and our approach is that our data are obtained from a low-resolution, activity-oriented RGB camera.

Fig. 1. Dataset Introduction: Indoor (left) and outdoor (right) examples from our study. Faces are blurred to preserve subject privacy

III. System Design

A. Screen Presence Definition

Our system is designed to detect the presence of a screen or part of a screen in the camera's field of view. Examples

of screens include devices with a digital display like a television, smartphone, laptop/computer. For hand-held devices (i.e., smartphones), if the participant is clearly holding the device and using it, but the screen is not visible (i.e., the viewer sees only the back of the device), we still consider it as screen presence.

Although screen presence does not directly mean screen time, they are highly correlated when the screen is oriented in the field of view of the participants of our study. The limitations about the screen detection environment are elaborated on with other key challenges in Section VI.

B. Hardware Overview

The data collection system, WildCam, is designed using an activity-oriented camera and a fisheye lens. WildCam consists of an ARM-Cortex M4 microcontroller [10], comprising a dual-stream RGB camera (OmniVision OV2640 and a 180 degree fish eye lens) [11] and a low-cost low-power 8×8 IR sensing array (Panasonic GridEye 8x8). Participants wear this device as a necklace around their neck. The data is recorded offline and is processed once they return the device.

C. Dataset Overview

The dataset comprises data from 10 participants wearing the WildCam for 3 days in an Institutional Review Board (IRB) approved study. Participants were instructed to wear the device throughout the day while going about their activities of daily living. We have selected 8 hours of data for each participant across the 3 day time frame. The recorded dataset has 80 hours of data, comprising of 1,200,000 frames of RGB images. We further partitioned the data into 3 folds for generalized testing and training, as shown in Table I.

Our dataset introduces new challenges with regards to screen presence. The video is captured in 240x320 pixel resolution by the camera as opposed to common HD resolution images. The subject wearing the device casually performs their errands throughout the day, therefore there is a healthy mix of indoor and outdoor activity within our dataset. Diversity of environment constitutes new challenges for screen detection as the recorded data has varied degrees of brightness, contrast, etc. Figure 5 provides example images from our dataset.

D. Data Labeling

We had an exhaustive labeling process across our dataset with 4 trained annotators. Each frame was labeled by 2 annotators working from an exhaustive rubric generated through weekly discussions. This process resulted in 300,000 frames each with a binary label for screen time across the entire dataset. Additionally, we also generated bounding box level annotations for 24,000 frames of screens/monitors/mobile phones to increase the accuracy of screen time detection.

IV. System Framework

A. Network Architecture

1) Backbone Model: Our screen dataset is captured in the egocentric field of view. Our approach is to use a backbone model that performs exceptionally well on global screen detection to extract features of screen presence. These features from the backbone model are further processed through our screen prediction network. After an extensive survey of state of the art models for screen detection, we settled on EfficientDet-D7x [12], which is pretrained on the CoCo dataset [13]. The primary reason for the selection of this backbone is the presence of an efficient feature pyramid network (biFPN layer). This network within our backbone is able to fuse features from different resolutions and caters well to our dataset. This backbone is able to learn the importance of screen detection features and is scalable across different resolutions. This functionality is to be a good fit for our problem primarily due to our input resolution and because 3/80 object categories within the CoCo dataset are ”Phone,” ”Tablet,” and ”Televisions.” Currently, this backbone model is ranked in 5th in real time object detection and 2nd in object detection on the CoCo dataset.

The FPN layer provides a control over fusing the features in a top-down approach. This feature fusion technique is a recent contribution in the domain of robust object detection [14]. The feature pyramid networks have proven to perform exceptionally well on different resolution scales [15]. This FPN layer is followed by a series of convolutional layers with the region of interest output being a detected screen.

2) Screen Prediction Network: Our approach to rely on transfer learning using the predicted features from a backbone is similar to Li et al. [14]. The screen prediction network takes the input from our backbone model and is trained fully on our dataset. In a nutshell, the output from the backbone model provides the starting point for its optimization. We used the backbone model to filter through our dataset and provide features with high concentrations of screen presence in the ground truth. Many works using the same backbone have larger input images and have stacked FPN layers. Similar to Tan et al. [12] the biFPN layer is fed into five Conv2D layers followed by two dense fully connected layers. This cascading Conv2D net is similar to the approach used by Li et al [14]. Finally, our model employs a softmax layer with binary cross entropy and measures a distance from the ground truth labels to detect screen presence. The screen prediction model was trained for 90-110 epochs until convergence on a Nvidia TITAN V GPU cluster.

V. Evaluation

A. Evaluation Summary

We have evaluated the performance of our model at both frame and episode levels. The frame level results are displayed in Table II. Further, we generated screen-level episodes using an episode creation method called DBSCAN [16]. The density based clustering provides an effective way to find high-density regions of screen time activity. Given the presence of a screen time activity per frame, the DBSCAN approach generates screen time episodes as several specific groups. DBSCAN parameters were chosen based on

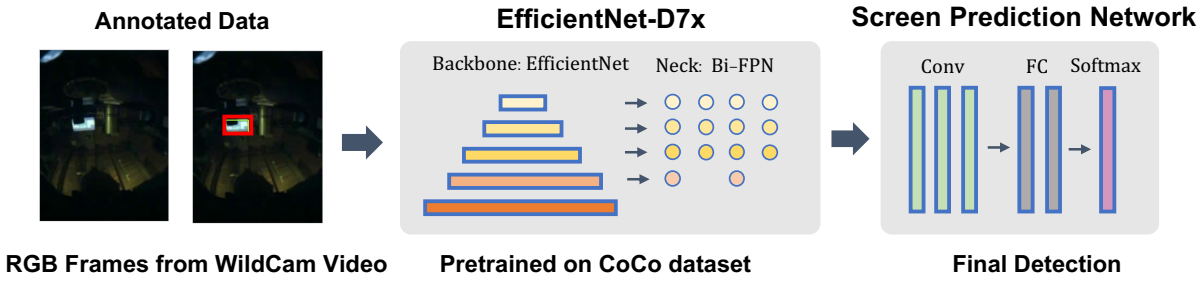

Fig. 2. Screen detection framework

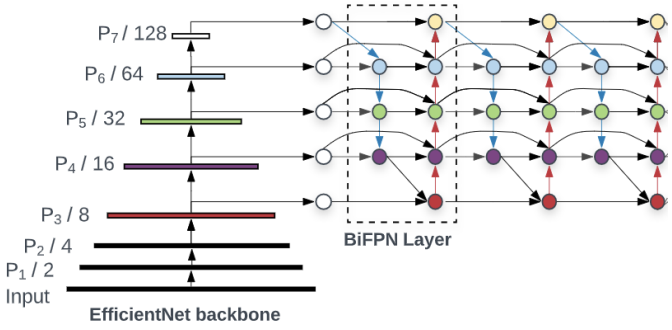

Fig. 3. EfficentDet-D7X model architecture [12]

a cumulative distribution function drawn across the dataset for frame level screen labels. Our dataset shows that each episode should at least include around 81 seconds of screen time (positive samples) to reach to a reasonable performance. Moreover, there must be no more than 142 seconds between two positive samples for them to be clustered into one episode.

As mentioned previously, we consider televisions, laptops, and mobile phones as screens. Table II shows the variation in model performance for each participant at a frame level. The F1 score varies from 0.18 to 0.54 throughout all of the participants. This is primarily because screen presence is distributed very unevenly among the participants. This prompted a need for an episode-level evaluation. To evaluate the precision, recall, and F1-score, we calculated the true positives (TP), false positives (FP), and false negatives (FN) in detecting episodes. The formulas for these computations are defined as: precision = $\frac{TP}{TP+FP}$, recall = $\frac{TP}{TP+FN}$, and F1-score = $\frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$.

Overall, we observed 112 screen episodes for the 10 participants and several screen instances for each participant. Table III presents the performance of our system in detecting screens from RGB frames. From the table, we observe that we could achieve 82.6% recall, 80.23% precision and 81.42% F1-score in detecting screens, indicating that we could detect 92 out of the 112 screen episodes.

TABLE I
DATASET SPILT

Fold	Participants	Frames
1	p1, p2, p3	402828
2	p4, p5, p6, p7	372243
3	p8, p9, p10	355911

TABLE II
FRAME LEVEL RESULTS

Fold	Participant	Precision	Recall	F1	Frames
1	p1	0.61	0.44	0.51	121566
1	p2	0.52	0.28	0.36	133630
1	p3	0.88	0.39	0.54	147632
2	p4	0.51	0.11	0.18	127404
2	p5	0.72	0.33	0.45	142495
2	p6,p7	0.68	0.28	0.40	102345
3	p8	0.41	0.56	0.47	132431
3	p9	0.26	0.43	0.32	98012
3	p10	0.71	0.32	0.44	125468

TABLE III
EPISODE LEVEL RESULTS

Participant	Precision	Recall	F1	Episodes
p1	0.78	0.78	0.78	12
p2	0.88	0.88	0.88	13
p3	0.67	0.8	0.73	10
p4	0.5	0.25	0.33	9
p5	1	1	1	16
p6	0.75	0.86	0.8	10
p7	1	1	1	10
p8	1	1	0.86	14
p9	0.8	0.8	0.73	5
p10	1	1	0.92	13
Total	**0.83**	**0.80**	**0.81**	**112**

B. Comparison with existing screen prediction models

In this section we demonstrate the performance of four additional models on a test set of 10,000 frames from our dataset. These models are publicly available benchmarks in the space of object detection with the CoCo dataset [17], [18] and activity recognition on egocentric videos [19], [20].

Our system is designed for screen detection on the WildCam dataset with custom annotations and labeling. Table IV provides the result of a comparison of our proposed system with the systems reported in the literature using our dataset. The best performing model is our system with an F1 score of 0.78, followed by our backbone model detnet-d7x [12] with an F1 score of 0.54.

TABLE IV
PERFORMANCE COMPARISON

Performance (F1)	Model
0.27	coco-minival [18]
0.31	screenvoider [19]
0.42	yolov3 [17]
0.54	detnet-d7x [12]
0.78	**screen prediction framework**

VI. IDENTIFYING THE KEY CHALLENGES

A. Dataset Quality

Given the resolution and the egocentric field of view of our dataset, the screens in our dataset are different in representation as compared to a traditional screen. Once the video is processed into frames they are stored with JPEG compression. This makes modeling an object detection framework more challenging, especially for screen activity, even with a strong backbone. Due to the camera's JPEG compression and motion artifacts, there was a reduction in the capture of sharp edge features from our ground truth.

B. Screen Detection Environment

A few of the participants wore the device in an indoor setting, and their interaction with screens was relatively higher compared to that shown in data collected outdoors. The participants wore the device as a necklace, the camera's field of view centered around the participants face and body. Hence, the majority of the screen presence recorded is not centered in the camera's field of view. Furthermore, the camera lens captured the screen presence with fish eye distortion. This is very different from the traditional screen layout used for training the majority of existing screen detection models, presenting our study with a unique challenge.

Table II shows the evaluation of our system in detecting screens on a frame level for each participant. The variance on F1 score throughout the participant set is 0.36, primarily due to the variation in screen presence in the video capture. Each participant recorded screen interaction in different environments and the folds selected for evaluation were chosen accordingly in Table I.

C. Assessment of False Positives and False Negatives

1) Absent Predictions: The low resolution of the screens in the training set contributes to our model predicting screens in the absence of screens. In the future, we could maintain a

Fig. 4. Challenging Cases - Artifacts: Top Left - Edge Case, Top Right - Motion Artifact, Bottom left + Right - Bright Artifacts. Faces are blurred to preserve subject privacy

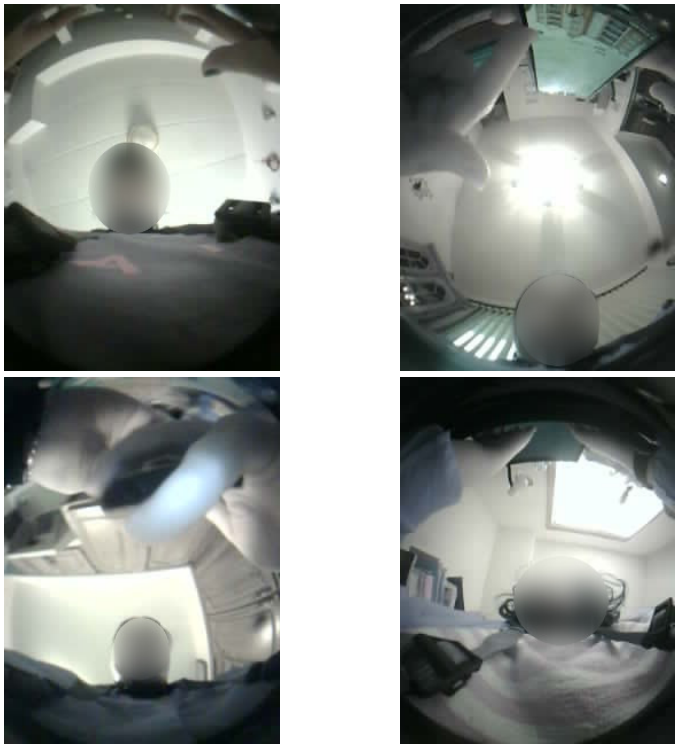

Fig. 5. Challenging Cases - Hand Held Devices: Top Left - Edge Case, Top Right - Touch Screen Interaction, Bottom left + Right - Hands in Line of Sight. Faces are blurred to preserve subject privacy

minimum number of pixels on target for detection, or slightly increase the resolution of the RGB camera to prevent these FPs.

2) Motion and Bright Artifacts - FP + FN: Each participant operates in a different environment with changes in illumination, lighting, and motion through interaction with various objects. Our screen model is sensitive to illuminated artifacts and motion artifacts. As the backbone model predicts cell phones as a label, smaller illuminated artifacts contributed to a fair share of FP screen activities.

3) Screens at the edges - FN: Because participants wear the device as a necklace, the field of view captures the scene in front of them from an egocentric perspective, as shown in Fig 5. Most of the captured screen activity involves small, handheld devices. The RGB camera has a wide field of view and captures a significant amount of screen activity in the edges of the scene.

VII. Conclusion and Future Work

We have built a system to detect screen presence in activity-oriented video captured in a privacy-preserving manner using a wearable device. This system is efficient in tracking the daily interactions with screens (televisions, laptops, and mobiles) by the subjects of our study. Our study comprises 10 individuals, 80 hours of data, and 1.2 million low-resolution RGB frames.

Our system leverages deep learning algorithms and detects screen presence on lower-resolution images. Our data collection system is WildCam and is designed using an activity-oriented camera with a fisheye lens. Our results outperform the existing state-of-the-art video screen detection methods, yielding an F1 score of 81%.

We have also analyzed the broad challenges in building object detection models within the egocentric domain. Additionally, we have provided a performance comparison for our approach with the existing state of the art on our data. We have performed an assessment of the performance of our model by gauging the false positive and false negative cases.

The future work for our experiment is to detect objects with integrating temporal information within the egocentric domain. Including additional infrared sensor information from the WildCam system and detecting other activities (e.g., eating) and social presence would provide even deeper insight into subject behavior.

Acknowledgment

This material is based upon work supported by the National Science Foundation (NSF) under award number CNS1915847. We would also like to acknowledge support by the National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK) under award numbers K25DK113242 and R03DK127128, and National Institute of Biomedical Imaging and Bioengineering (NIBIB) under award number R21EB030305. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the National Institutes of Health.

References

[1] W. O'Brien, J. Issartel, and S. Belton, "Relationship between physical activity, screen time and weight status among young adolescents," *Sports*, vol. 6, no. 3, 2018. [Online]. Available: https://www.mdpi.com/2075-4663/6/3/57

[2] M. G. Matias de Pinho, F. ADAMI, J. Benedet, and F. Vasconcelos, "Association between screen time and dietary patterns and overweight/obesity among adolescents," *Revista de Nutrição*, vol. 30, pp. 377–389, 06 2017.

[3] Y. C. Zhang and J. M. Rehg, "Watching the tv watchers," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 2, no. 2, jul 2018. [Online]. Available: https://doi.org/10.1145/3214291

[4] A. Apaolaza, A. Brown, C. Jay, and S. Harper, *Understanding the division of attention between TV and companion content: experiment 2, without eye-tracking*. No publisher name, Oct. 2014.

[5] V. Apaolaza, P. Hartmann, C. D'Souza, and A. López, "Mindfulness, compulsive mobile social media use, and derived stress: The mediating roles of self-esteem and social anxiety," *Cyberpsychology, Behavior, and Social Networking*, vol. 22, 05 2019.

[6] A. Apaolaza, R. Haines, A. Aizpurua, A. Brown, M. Evans, S. Jolly, S. Harper, and C. Jay, "Abc: Using object tracking to automate behavioural coding," in *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 2766–2773. [Online]. Available: https://doi.org/10.1145/2851581.2892483

[7] K. A. Funes Mora, F. Monay, and J.-M. Odobez, "Eyediap: A database for the development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, ser. ETRA '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 255–258. [Online]. Available: https://doi.org/10.1145/2578153.2578190

[8] S. Park, S. D. Mello, P. Molchanov, U. Iqbal, O. Hilliges, and J. Kautz, "Few-shot adaptive gaze estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[9] C. Cao, Y. Zhang, Y. Wu, H. Lu, and J. Cheng, "Egocentric gesture recognition using recurrent 3d convolutional neural networks with spatiotemporal transformer modules," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 3783–3791.

[10] PanasonicGridEye, "https://na.industrial.panasonic.com/products/sensors/sensors-automotive-industrial-applications/lineup/grid-eye-infrared-array-sensor," 2020.

[11] OV2640-Omnivision, "https://www.arducam.com/ov2640/," 2021.

[12] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," *CoRR*, vol. abs/1911.09070, 2019. [Online]. Available: http://arxiv.org/abs/1911.09070

[13] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[14] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "Detnet: A backbone network for object detection," *CoRR*, vol. abs/1804.06215, 2018. [Online]. Available: http://arxiv.org/abs/1804.06215

[15] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," 2017.

[16] D. Deng, "Dbscan clustering algorithm based on density," in *2020 7th International Forum on Electrical Engineering and Automation (IFEEA)*, 2020, pp. 949–953.

[17] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[18] COCO-Minival-Benchmark, "https://sotabench.com/benchmarks/object-detection-on-coco-minival," 2021.

[19] M. Korayem, R. Templeman, D. Chen, D. J. Crandall, and A. Kapadia, "Enhancing lifelogging privacy by detecting screens," *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016.

[20] K. Min and J. J. Corso, "Integrating human gaze into attention for egocentric activity recognition," *CoRR*, vol. abs/2011.03920, 2020. [Online]. Available: https://arxiv.org/abs/2011.03920