




Article

Contactless Real-Time Eye Gaze-Mapping System Based on Simple Siamese Networks

Hoyeon Ahn ¹, Jiwon Jeon ², Donghwuy Ko ³, Jeonghwan Gwak ^{4,*} and Moongu Jeon ^{1,*}

¹ School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, Republic of Korea

² TmaxTibero, 258, Hwangsaeul-ro, Bundang-gu, Seongnam 13595, Republic of Korea

³ AhnLab, Inc., 220, Pangyoyeok-ro, Bundang-gu, Seongnam 13493, Republic of Korea

⁴ Department of Software, Korea National University of Transportation, Chungju 27469, Republic of Korea

* Correspondence: jgwak@ut.ac.kr (J.G.); mgjeon@gist.ac.kr (M.J.)

Abstract: Human–computer interaction (HCI) is a multidisciplinary field that investigates the interactions between humans and computer systems. HCI has facilitated the development of various digital technologies that aim to deliver optimal user experiences. Gaze recognition is a critical aspect of HCI, as it can provide valuable insights into basic human behavior. The gaze-matching method is a reliable approach that can identify the area at which a user is looking. Early methods of gaze tracking required users to wear glasses with a tracking function and limited tracking to a small monitoring area. Additionally, gaze estimation was restricted to a fixed posture within a narrow range. In this study, we proposed a novel non-contact gaze-mapping system that could overcome the physical limitations of previous methods and be applied in real-world environments. Our experimental results demonstrated an average gaze-mapping accuracy of 92.9% across 9 different test environments. Moreover, we introduced the GIST gaze-mapping (GGM) dataset, which served as a valuable resource for learning and evaluating gaze-mapping techniques.

Keywords: human–computer interaction; gaze mapping; facial detection; facial recognition



Citation: Ahn, H.; Jeon, J.; Ko, D.; Gwak, J.; Jeon, M. Contactless Real-Time Eye Gaze-Mapping System Based on Simple Siamese Networks. *Appl. Sci.* **2023**, *13*, 5374. <https://doi.org/10.3390/app13095374>

Academic Editor: Yu-Dong Zhang

Received: 3 March 2023

Revised: 18 April 2023

Accepted: 20 April 2023

Published: 25 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The human eye is a primary source of information in the field of human–computer interaction (HCI), providing brightness, motion, and depth information. As such, understanding the human visual system is crucial to visual HCI research [1]. The unique physical characteristics and movement patterns of the human eye allow us to determine an individual’s focus-of-attention and emotional state. By analyzing essential visual perception and cognitive organ data, we can collect core information on human behavior [2].

Research on gaze recognition has rapidly advanced since the 1970s [3]. Gaze recognition technology has been studied primarily to assist people with disabilities, leading to the development of specialized eye-tracking devices [4]. Gaze recognition has been subdivided into various fields, such as gaze-based user interfaces and human-cognition research [5–7], and has also been introduced to prevent driver drowsiness with advanced automobile driving-assistance systems (ADAS) [8]. With the growth of wearable electronic devices and computing performance, we are now able to analyze eye movements in greater detail. Existing eye-tracking and recognition methods often require participants to wear glasses-like electronic devices [9], in which an eye camera has been installed in the device to detect the field of vision for each eye. The final gaze is determined by the location of the pupil region within both of the detected eye regions. The pattern of the human eye and gaze can reveal an individual’s needs, intentions, and state of mind in the process of understanding social interactions [1]. However, most eye recognition studies to date have been performed using wearable glasses-like electronic devices [8].

These experiments have required participants to be aware of the gaze recognition test environment, which thus affected their behavior due to the hardware devices they had to wear. To reliably extract various reaction variables from the experimental test units and create a natural experimental environment that eliminated participants' psychological factors, a non-contact method was required. This method needed to record the pattern of the natural gaze, eliminating the psychological factors that contributed to human error.

In this work, we proposed a non-contact gaze-mapping method that could recognize gaze patterns without the use of a glass-like device and that could map a user's gaze regardless of their location or height. However, gaze-recognition performance in an unconstrained environment without a wearable device would be affected by external factors, such as measurement distance, lighting, and whether the pupil region had been captured, as shown in Figure 1. The coarse gaze area was acquired according to the head pose, and the fine final gaze area was mapped through a gaze-recognition module. The method proposed in this work intuitively comprehended the user's gaze via a non-contact method, that is, without using a glasses-like device, and it mapped the eye gaze regardless of the user's location.

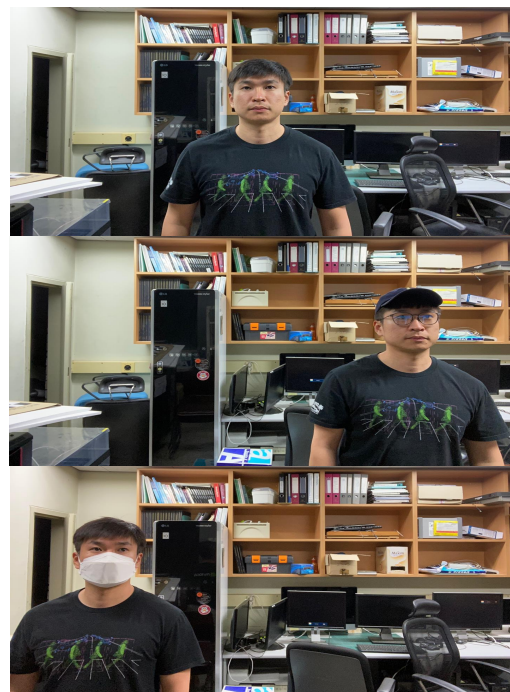


Figure 1. Description of shelf-staring environment. The gaze environment varied depending on the user's glasses, masks, hats, etc., and the gaze habits and height of the user also varied.

The proposed method consisted of a three-step process. (1) The face module was executed (detection, alignment, and recognition). (2) The eye region was extracted from the face region, and the gaze was estimated. (3) The final gaze was mapped to the target of the gaze panel by integrating the depth information obtained by the depth camera, gaze recognition information, and head pose. We implemented the entire gaze mapping process through the parallelization and optimization of the system to operate on the edge device, NVIDIA TX2, at 7.5 fps.

This paper's main contributions were the following: We created the GIST gaze-mapping (GGM) dataset for training the gaze-matching network. The GGM reflected the various user environments in which the dataset could be generalized, including a user's height, distance change, and profile gaze-posture. The depth information obtained through the depth sensor camera automated the parameter optimization, ensuring that tuning would not be required during inference. This method was easy to expand when configuring a mapping system by combining other facial detection and recognition methods. The

entire gaze-mapping system could provide real-time performance in real commercial environments, including a gaze estimator based on a deep convolutional neural network (CNN).

The remainder of this paper is organized as follows. Section 2 introduces the related fields. Section 3 describes the gaze-mapping system. Section 4 presents the experiment with our dataset, and Section 5 concludes the paper.

2. Related Work

2.1. Gaze Estimation

Gaze recognition methods have been proposed that predominantly employ handcrafted features [10]. For example, after the fit of a model with handcrafted features was optimized according to the linear-regression equation, a method for estimating the final gaze was developed [11]. Feature-based gaze recognition has been implemented using a simple linear-regression model. However, the eye-gaze feature has made it difficult to predict the generalized performance in a real environment. Model-based methods have been proposed that could show generalized performance improvements in a real environment [12,13]. Eyeball modeling calculated a gaze vector using the feature points of a geometric eye model. As compared to feature-based gaze recognition, which extracts the local features of the eye region, model-based methods have commonly modeled the entire eyeball area to recognize the eye gaze. The eyeball-modeling method used a high-dimensional input as a feature and learned the gaze-mapping function. Appearance-based gaze recognition was performed by image matching [14,15] and modeling the entire eyeball. With the eyeball-modeling method, gaze recognition was performed by matching a 3D model with an eye image [13,16]. The method based on 2D image matching was simple, but in some cases, it had a sensitive response to changes in posture, such as the head pose or lighting, which then negatively affected the final gaze-recognition performance. However, the information required for 3D shape modeling had to be preceded by parameter measurements of the corneal radius and center; the pupil radius; the distance between the corneal center and pupil; the incidence angle; and the refractive index between the optical axis and the visual axis. A complex dataset was required, but the 3D model-based method generated more reliable gaze-recognition results than the image-matching method [17].

Since 2012, deep neural networks that showed excellent performance in the computer vision field have been proposed [18], and they have been used in various computer-vision tasks, such as object classification and object detection [19,20]. To achieve successful gaze-recognition performance, the mapping from the eye image to the gaze direction had to be well learned. Therefore, large datasets for gaze recognition, such as MPII gaze [21] and RT-GENE [22], were proposed. As deep neural networks advanced, they have also been applied to gaze recognition and shown superior results in real environments for gaze recognition, as compared to gaze recognition using feature-based methods [23,24]. Recently, a novel approach to unconstrained gaze estimation was proposed that was based on long short-term memory (LSTM) and trained on sequential datasets. As a result, the LSTM 3D gaze model was expected to be scalable, as compared to existing models, and it enabled the direct output of gaze estimates with uncertainty [25].

2.2. Facial Detection, Alignment, and Recognition

Facial detection and recognition is a procedure that automatically finds a person's face in visual media and identifies them using individual IDs, which is an essential and basic task in various facial applications. The fundamental problems of computer vision, such as occluded light, lighting changes, and pose changes, can affect the performance of facial recognition in real environments. Before the deep neural network was proposed, facial detection and recognition methods based on the handcrafted features had been studied. Related to handcrafted features, a cascade facial detector using Haar features and Adaboost was proposed [26]. Many studies proposed methods capable of real-time processing and excellent performance in environments containing occluded light and lighting changes [27,28]. As feature-based facial research has progressed, a core method

has been developed, using a deformable feature form that could perform facial recognition by modeling the relationship between parts of the face [29–31]. Deep neural-network-based object detectors, such as you-only-look-once (YOLO) [20] and a single-shot multi-box detector (SSD) [32] were developed to enable facial detection with excellent performance. Both facial detection and recognition are capable of end-to-end learning through powerful deep-learning-optimized networks, which have significantly changed facial research trends. A CNN-based method presented the possibility of over-fitting, as the network was deep, and the number of parameters was large. In addition, it had a disadvantage in that it takes time to learn, but it has the advantage of facilitating generalization. Early CNN-based facial recognition was used as an auxiliary task to improve the performance of facial alignment [33]. Since then, facial alignment issues were recognized by researchers as a major factor in detection and recognition performance, and research was conducted using a joint multi-task learning method [34].

As the performance of CNNs have improved, many application developers have anticipated excellent detection and recognition on extremely small faces that were not able to be captured in common surveillance environments, such as WIDER FACE [35]. A scale-invariant network was proposed to detect faces at different scales in each layer of a single network [36]. A facial recognition method was proposed that used anchor-level attention and showed excellent performance in occluded light [37].

3. Eye Gaze-Mapping System

The proposed system recognized users who accessed the shelves. At the same time, pupil detection, gaze recognition, and head pose were measured within the recognized user's face area. The working process of the entire gaze-mapping system is shown in Figure 2. Finally, gaze mapping without using a wearable device for gaze recognition was performed by obtaining gaze information, head pose, and depth information. Because the performance of the gaze-mapping module was dependent on the facial detection and alignment module, each module was optimized and tuned.

In addition, in order to implement a real-time gaze-mapping system that satisfied the resource limitations of the NVIDIA Jetson TX2, we carefully considered the network design and the number of parameters for each module comprehensively.

The following subsections describe the detailed implementation of the facial recognition, including the detection and alignment, the gaze recognition, and the gaze-mapping modules. The last section describes the implementation of the parallelized and optimized tasks assigned to each module.

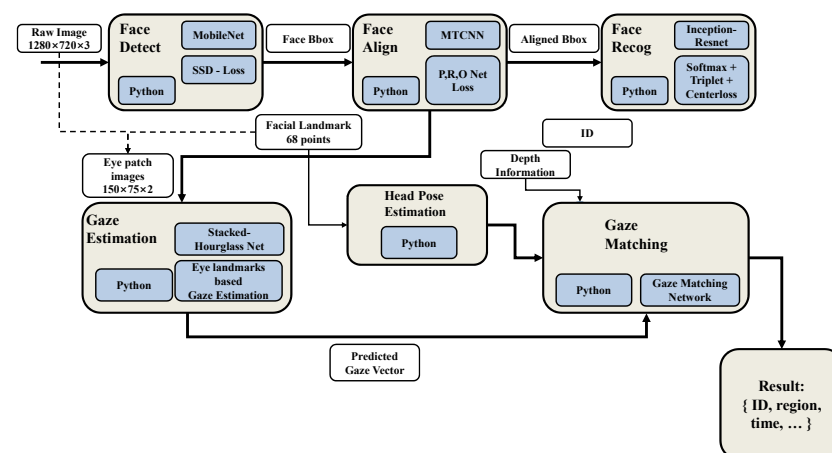


Figure 2. Overall structure of gaze-mapping system. The system consisted of the face, gaze estimation, and gaze-matching modules. The face module detected, aligned, and recognized faces, in order. The aligned facial information was transferred to the gaze-estimation module and then used as the input to the gaze-matching module, along with the head pose and the predicted gaze vector information.

3.1. Facial Detection, Alignment, and Recognition

We deployed the entire gaze-mapping system to edge devices. Therefore, facial detection and alignment suitable for low-power computing was adopted, as shown in Figure 3. A facial detection module suitable for the purpose of real-time gaze-mapping was modified and applied to MobileNet [38], which is typically used for edge-device computing.

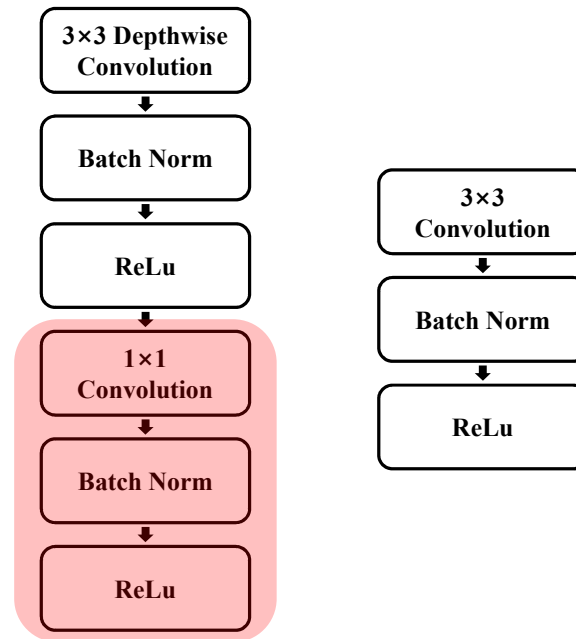


Figure 3. Core module of MobileNet was a Depth-wise Conv, followed by BN ReLU. Depth-wise Conv reduced the number of parameters and computational costs more than the normal convolutional 2D process. **(Left):** Depth-wise convolutional structure. **(Right):** Basic convolutional layer structure. Red area: Indicating the bottleneck layer.

In the following equation, D_K is the kernel size, D_F is the input channel size, M is the input channel, and N is the output channel; the general convolutional computation is shown in Equation (1):

$$FLOPs(Conv2D) = D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F \quad (1)$$

However, when the factorized depth-wise separable convolution of MobileNet was applied, the calculation was expressed, as follows:

$$FLOPs(DepthwiseSeparableConv2D) = D_K \cdot D_K \cdot D_F \cdot D_F \cdot M + D_F \cdot D_F \cdot M \cdot N \quad (2)$$

According to Equations (1) and (2), the ratios in the calculation were reduced, as compared to the general convolution.

$$\frac{FLOPs.DepthwiseSeparableConv2D}{FLOPs.Conv2D} = \frac{D_K^2 + N}{D_K^2 N} = \frac{1}{N} + \frac{1}{D_K^2} \quad (3)$$

By assuming a kernel size of 3, the MobileNet-based facial detector could achieve approximately 8 times the computational efficiency in FLOPS, and it effectively reduced the memory weight of the facial detection module in the proposed gaze-mapping system.

$$FLOPs(+ResolutionMultiplier) = D_K \cdot D_K \cdot \beta D_F \cdot \beta D_F \cdot \alpha M + \beta D_F \cdot \beta D_F \cdot \alpha M \cdot \alpha N \quad (4)$$

In addition, as shown in Equation (4), when the final calculation was applied, the width and resolution multipliers α and β , respectively, were used to reduce the facial recognition module, so it would be suitable and deployable when using the limited NVIDIA Jetson TX2 hardware. A thin network could be created based on the value of α . By adjusting β , the size of the input image and all the internal layers could be reduced by the same ratio. The facial alignment task was closely related to the detection task. Therefore, it was possible to maximize the effectiveness of the sorting when learning by multi-tasking.

The MTCNN proposed by Zhang et al. [34] consisted of three models, P-Net, R-Net, and O-Net, and used a cascaded inference structure. The network structure was designed to learn classification, landmark localization, and the multi-task loss of box regression in a joint-learning manner.

The main characteristic of MTCNN was implemented in the form of an image pyramid, which was expected to improve the detection and recognition performance by aligning faces of various sizes. In particular, it could maximize the performance of the gaze-recognition module, which was dependent on the facial detection and alignment performance. The facial recognition module applied Inception-ResNet [39]. In this study, the effect of the residual module [40] reduced the convergence speed once a large dataset had been learned. When the facial recognition module was executed, diverse features could be recognized. The triplet loss [41] for learning the discriminative features within the similar texture information was defined according to Equation (5):

$$Loss = \max(0, \alpha + d(f(X_i), f(X_{pos})) - d(f(X_i), f(X_{neg}))) \quad (5)$$

where f is an embedding function and d is a distance function that measures the distance between two inputs. The embedding distance of the data X_{pos} were similar to the anchor vector, as the reference point was expected to be greater than the distance from X_{neg} , and the distance function was trained using the $L2$ distance for the task.

3.2. Gaze Estimation and Head-Pose Estimation

Handcrafted features and model-based gaze-recognition methods tend to be sensitive to lighting changes, resolution, and occluded light. Therefore, these methods are challenged when presented with real-world images. Utilizing the stacked hourglass method, Park et al. [17] extracted eye landmarks. The landmark features contributed to the recognition of the eye appearance at multiple scales. Therefore, spatial information could be maintained by using only one skip-layer for each scale. To use these advantages, we applied UnityEyes [42] so our model could perform gaze recognition that would be suitable for a real environment. Head-pose estimation was added to compensate for the instability of the gaze estimation. To estimate the head pose, specific 2D coordinate information was required, and the 3D coordinates of the corresponding 2D feature points were required. In addition, because the 3D world was projected as a 2D image, a camera calibration process was required to remove the parameters inside the camera when converting the 2D coordinates back into 3D coordinates. The head pose was estimated using 3D coordinates and the camera matrix of the OpenCV dlib 68 facial landmarks.

3.3. Gaze-Matching Network

The gaze-mapping system has difficulty generalizing because each person has different gaze habits, such as looking to the side, and various head angles. In addition, few-shot learning was suitable because the GGM dataset did not have many training images. The Siamese network, which used two inputs and returned the similarity between two vectors, is shown in Figure 4.

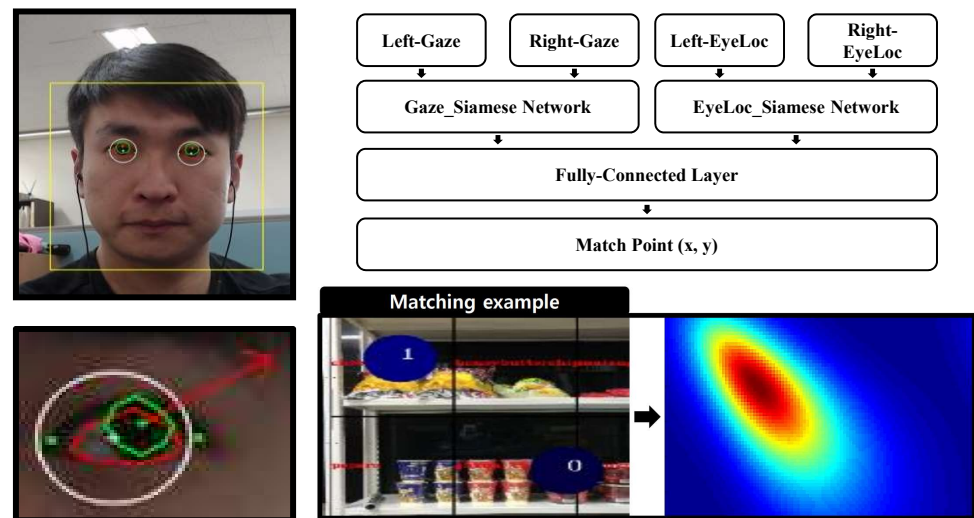


Figure 4. Description of gaze-matching module. Gaze and position information for both eyes were used as input to each Siamese network. The final matching point was calculated through the fully connected layer. In the figure, 0 is the start point of gazing and 1 is the last point.

To perform the feature extraction from the left and right eyes, the Siamese network was applied to the gaze location and gaze vector. Because the difference in the scale of each input was quite large, each input values were normalized. However, for depth, the maximum and minimum values could not be specified, so an additional single-channel, fully connected layer was added. The gaze vector and gaze location could then be shifted to a similar scale.

There were three sub-modules in the gaze-mapping module (refer to Figure 5).

- The head-pose estimator extracted the head-pose vector using information collected from the Jetson TX2, such as bounding box, landmarks, etc.
- The gaze-matching network extracted the gaze area within the image using information from various modules, including the head-pose vector, gaze vector, bounding box, and landmarks.
- The logging and visualization modules recorded the events occurring in each module in the database, visualized the logged data, and transmitted the images. As a result, the matching-point coordinates were obtained in every loop, and the gaze-matching network was inferred by the region-estimation loop during the logging process.

3.4. Gaze-Mapping System Parallelization and Optimization

The gaze-mapping system consisted of modules for facial detection, alignment, and recognition, and for gaze estimation. The system was implemented in parallel with queued and multiple processes to maximize resource utilization. We allocated the CPU resources according to the task load of each module and adjusted the overall inference speed.

Moreover, to share user IDs among different devices (see Figure 6), the ID information was synchronized in real-time using Python remote object (PYRO). All modules included in the gaze-mapping system were optimized using TensorRT, a platform for optimizing inferences in deep-learning models. In a CNN, the convolution, bias, and ReLU layers could be combined into one CBR layer to increase memory efficiency and computational speed. The platform was 45 times faster than the CPU-only implementation of INT8 and FP16 precision.

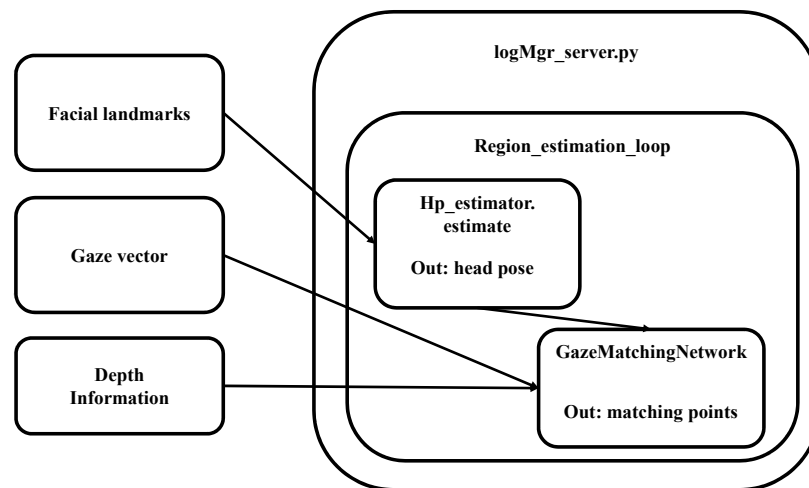


Figure 5. Gaze-matching module working structure. The module consisted of a total of three sub-modules. The head-pose estimator extracted the head-pose vector using bounding boxes and landmarks collected from the Jetson TX2. The gaze-matching network extracted the gaze area within the image by combining the head-pose vector, gaze vector, bounding box, and landmarks. The logMgr logged the events that occurred in each module and recorded them in the database.

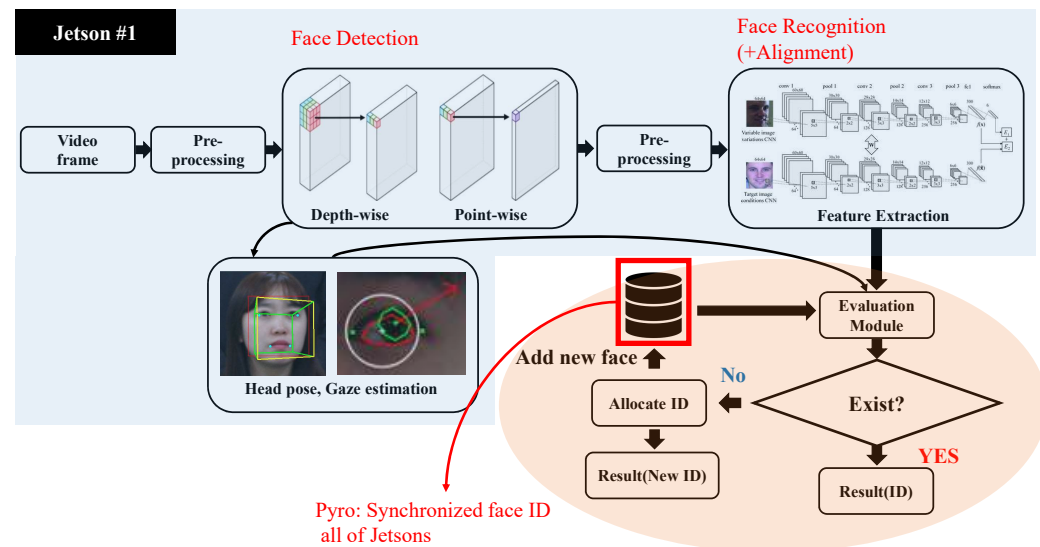


Figure 6. PyRO: A library that enabled Python objects on a remote machine by utilizing Python remote objects. Face ID was shared by the DB server in each Jetson TX2.

4. Experiments

4.1. GGM Dataset

The GGM dataset was produced by generating the test environment for 9 cases by changing the gaze-area width (0.75 m, 1 m, 1.5 m) and the user's relative position (left, right, center) from the camera, at 0.5m intervals. There were 5 users in the GGM dataset, and they had different heights (170 cm–185 cm). As shown in Figure 7, all images were acquired using the Intel RealSense depth camera D435. The depth resolution and FPS of the camera were 1280×720 and 30 fps, respectively, and the working range was from 0.11 m up to 10 m. The depth field-of-view was 85.2×58 and was utilized to calculate the relative coordinates, as shown in Figure 8. The GGM dataset consisted of a pair of depth images and labels. The file name for each record was created based on the user location and user ID of the camera. The label included the start and end frames when the user was gazing at a specific location. In each row, the attributes consisted of facial landmarks (obtained

using dlib + MTCNN), the gaze vector, and the head-pose estimation values, as illustrated in Figure 9.

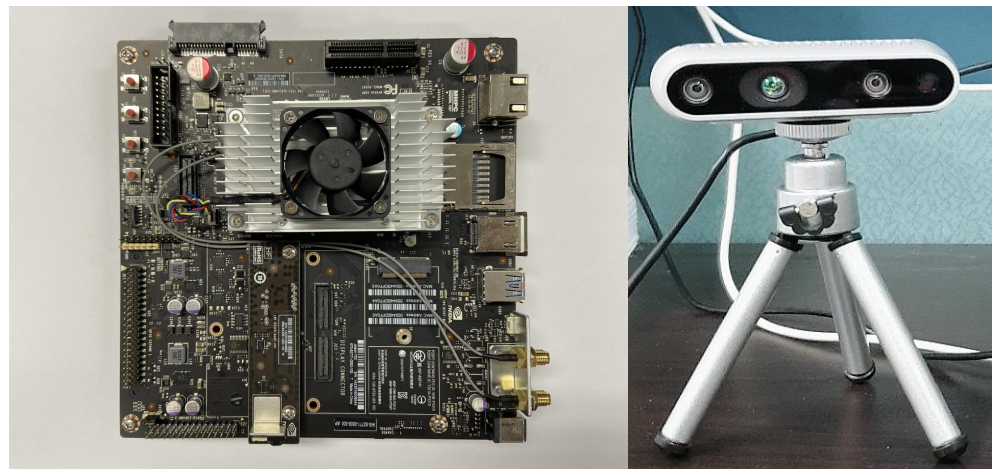


Figure 7. The gaze-mapping system hardware configuration. The Intel RealSense depth camera D435, which could acquire depth information, and NVIDIA's Jetson TX2 module were used.

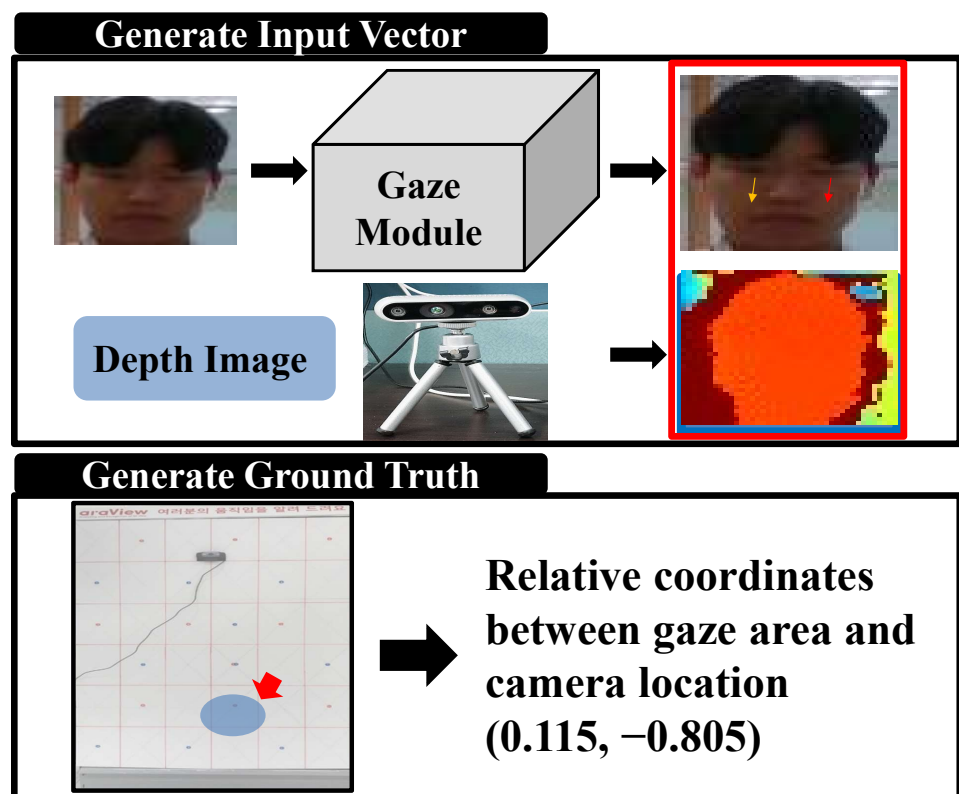


Figure 8. Description of creation of the gaze-matching network dataset. The gaze module output the gaze vector and the image depth information provided by the depth camera, which was then combined into a single vector. The final ground truth was produced based on the user's gaze area and relative coordinates, according to the camera data.

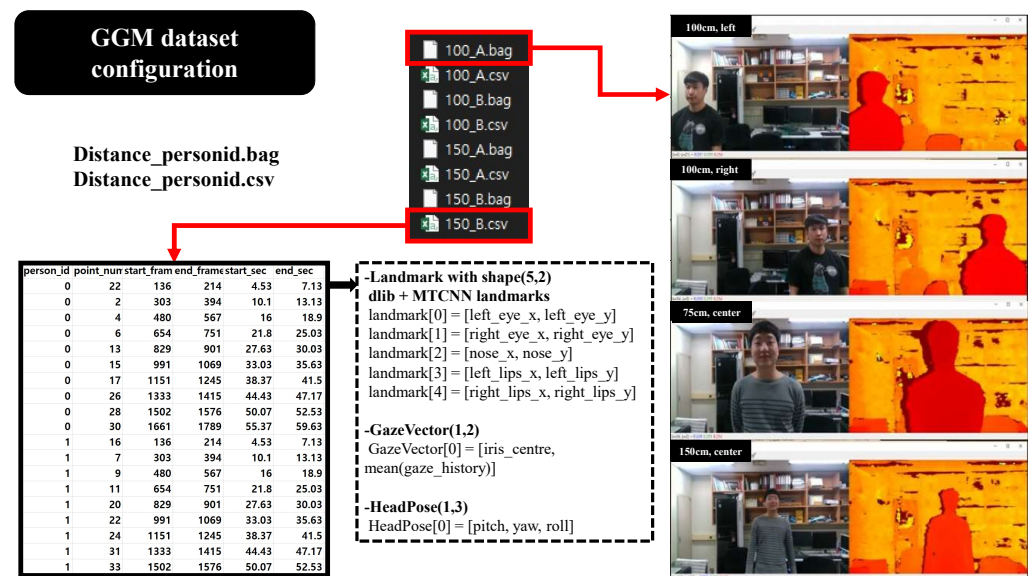


Figure 9. The configuration of the GGM dataset. The depth image acquired by D435 and the label comprised a pair, and the ID and start–end frame times were recorded.

The test environment of the gaze-mapping system closely resembled an actual environment of shelving in a large retail store. It was designed to mimic the eye-tracking simulations commonly used in the field of neuromarketing, as depicted in Figure 10. In particular, our gaze-mapping system was tested within a distance of 1 m, in which simple exposure effects would occur, that is, where human unconscious emotional information would be detected in a real-world retail environment.

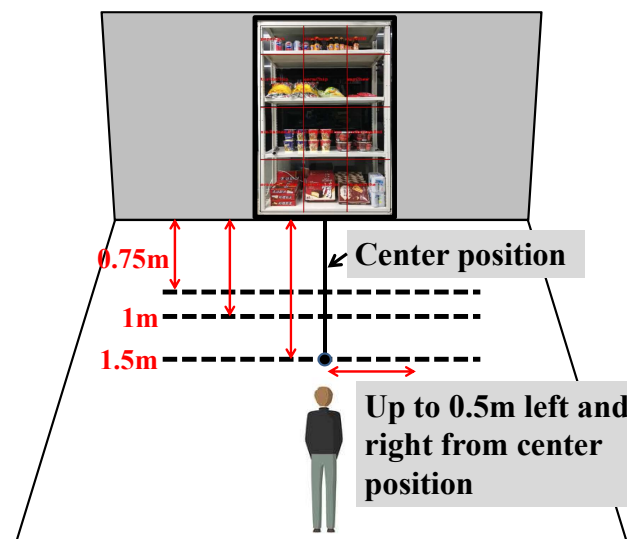


Figure 10. Description of GGM dataset acquisition method and experimental environment. The user gazed at a specific area (6 × 6 grid) in front of the shelves.

The user's appearance in the GGM dataset changed when they wore glasses, hats, and masks. In the dataset, people stared up to 30° to the left and right. The gaze panel was composed of 6 × 6 grids, as shown in Figure 11, and was divided into 36 sections. The size of each grid was 0.17 m × 0.23 m, and the size of the entire panel was 0.9 m × 1.38 m. The camera was installed at the height of 1.5 m above the ground.

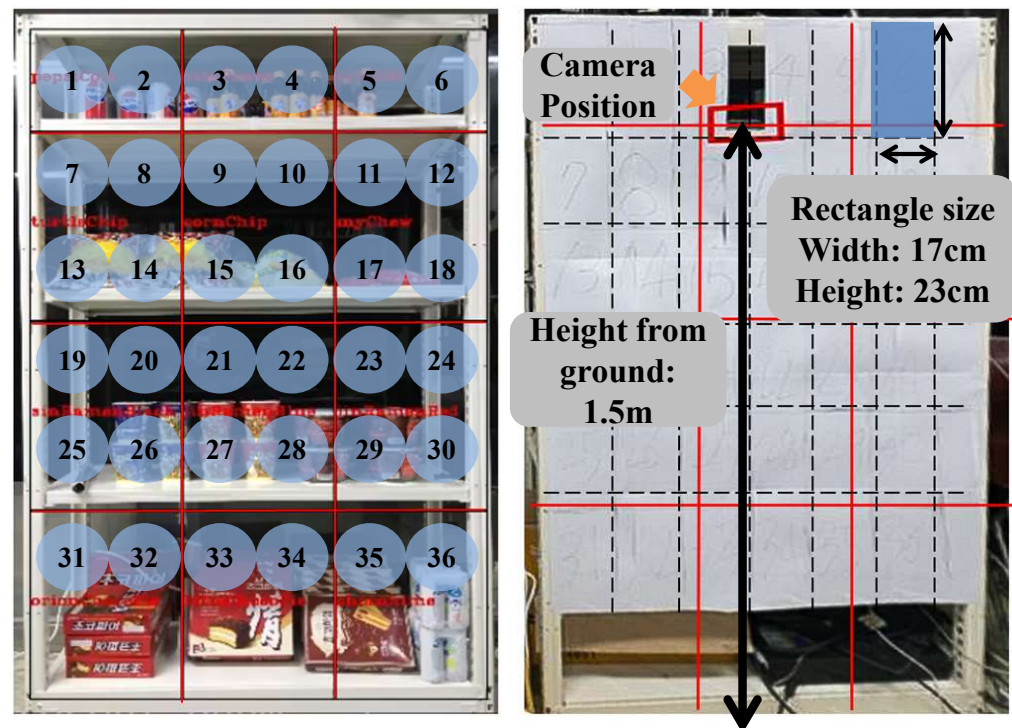


Figure 11. GIST gaze-mapping system shelving configuration. The grid area was composed of a total of 36 cells, and the same shelves used by the distribution industry were used. Each grid was 17 cm wide and 23 cm high, and the camera height was 1.5 m above the ground.

4.2. Results

A total of 90,000 GGM data-training sets, consisting of 5 users, were trained for the gaze-mapping system. The gaze-mapping system was evaluated as correct when the user gazed at a specific area for 5 s, and the calculated value of the intersection of the union (IoU) of the gaze-mapping prediction area and each grid area was 0.5 or higher.

The test environment was divided into a total of three user cases, as shown in Table 1, and the experiments included various user heights and accessories. The test environment of the gaze mapping system is shown in Figure 12. When the user gazed at the lower area of the shelves (grid numbers 25 to 36), we found that the performance of the gaze mapping was lower than when they gazed at the rest of the grid area. This tendency was that when the user looked at the top of the shelves, the region of interest was relatively easy to acquire from the whole eyeball area. Conversely, when the user looked down, the gaze estimation was limited because of the insufficient acquisition of the pupil area. When the user's gaze estimation on the shelves was incomplete, the results confirmed that the approximate gaze could be guaranteed from the head-pose information. In the case of 2 users (CASE 1 and CASE 2) according to the test setting in Table 2, an evaluation was also performed, and the processing was similar to the case of one user. An NVIDIA Jetson TX2 was used to implement the gaze-mapping system. We also tested another edge device from the same manufacturer, and the AGX Xavier showed a speed of 10 fps, as shown in Table 3. The system was composed of 5 modules, and the processing of each input and output was parallelized, as shown in Figure 13.

Table 1. GGM dataset test scenario configuration. Each user case was constructed by changes in the user height, accessories, and head pose.

User Case	Conditions
Common conditions	1. A/B/C/D Sub-CASE (total no. of points-per-user $80 = 10 \text{ points} \times 4 \text{ cases} \times 2 \text{ trials}$) -A: No accessories, stared at front of shelves -B: No accessories, 30° side view -C: Wore accessories (glasses, hats, masks), stared at front of shelves -D: Wore accessories (glasses, hats, masks), stared at 30° to the left and right 2. Complied with grid setting value (6×6 grid shelf)
Different conditions (CASE 1–3)	1. The height of the two users must be different (user heights: 155 cm, 175 cm, ± 5 cm) 2. The height of the two users must be different (user heights: 165 cm, 185 cm, ± 5 cm) 3. Turn the head as far as possible and stare

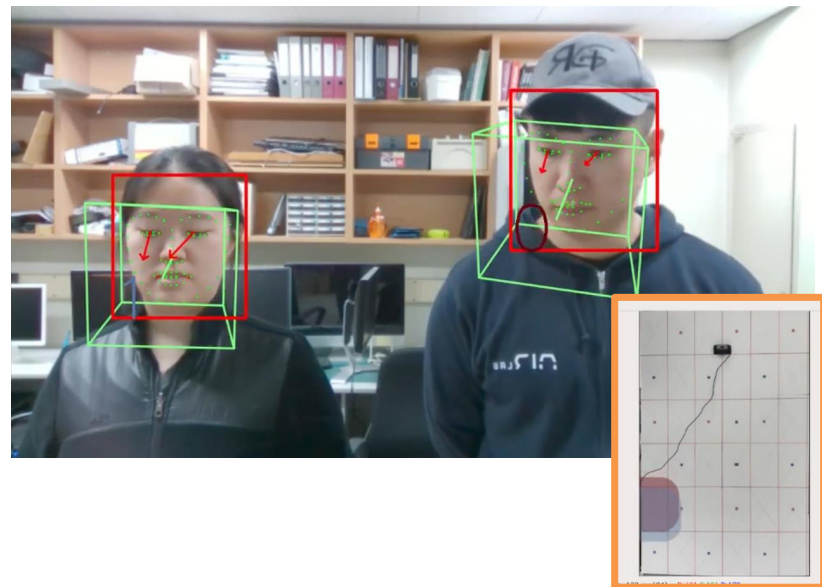


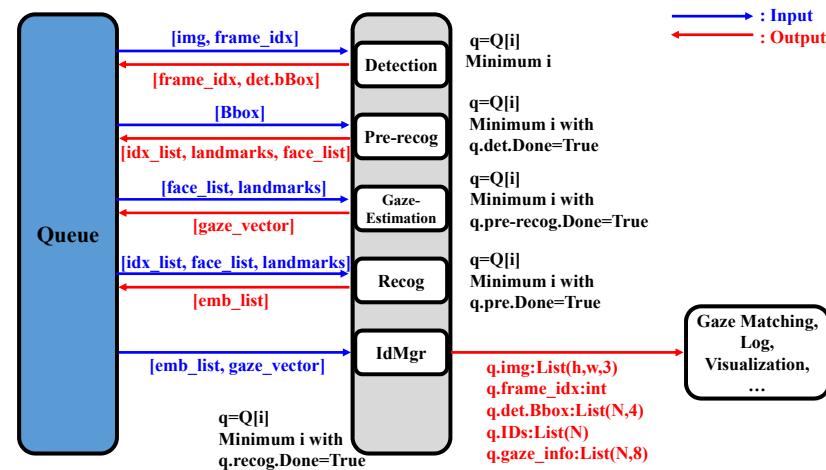
Figure 12. Example of the gaze-mapping system in the test environment. This figure shows the area at which two users are staring. Each part is displayed in a different color on the panel (USER CASE 1).

Table 2. Experiments were performed according to the defined test environment. Various tests were performed according to the number of users and the distance from the user to the shelves.

Test Definition	User Case	No. of Trials	(Accuracy, %)
2 users stared at 0.75 m	CASE 1	80	93.75
2 users stared at 0.75 m	CASE 2	80	90.00
2 users stared at 1 m	CASE 1	80	96.25
2 users stared at 1 m	CASE 2	80	91.25
2 users stared at 1.5 m	CASE 1	80	96.25
2 users stared at 1.5 m	CASE 2	80	91.25
1 user stared at 0.75 m	CASE 3	40	95.00
1 user stared at 1 m	CASE 3	40	92.50
1 user stared at 1.5 m	CASE 3	40	90.00

Table 3. Comparison of speculation and performance (fps) on Jetson TX2 and Jetson AGX Xavier.

Hardware	Jetson AGX Xavier	Jetson TX2
CPU(ARM)	8-core Carmel ARM CPU @ 2.26 GHz	6-core Denver and A57 @ 2 GHz
GPU	512 Core Volta @ 1.37 GHz	256 Core Pascal @ 1.3 GHz
Memory	16 GB 256-bit LPDDR4x @ 2133 MHz	8 GB 128-bit LPDDR4
Speed	10 fps	7.5 fps

**Figure 13.** Diagram of gaze-mapping system parallelization. Based on queued and multi-processing, resource use was maximized to improve speed, and CPU resources could be dynamically allocated according to the load of each task in order to improve overall inference speed.

5. Conclusions

Gaze mapping is a method used to measure the movement of the eyeball in order to determine the location at which a person is looking and how long they fixate on a certain point. Since our eyes are one of the primary organs used for decision-making and learning, accurate measurements and the understanding of visual attention are essential. This study aimed to construct an experimental environment for an eye-tracking simulation configuration used in neuromarketing to measure the distance at which human unconscious emotional information operates (i.e., the distance at which simple exposure effects occur). The resulting gaze-mapping (GGM) dataset provided various possibilities for customization and training for generalized performance, including a person's height, distance changes, and profiles of head poses. In contrast to previous studies, the proposed non-contact gaze-mapping system could map the user's eye gaze without the need for wearable hardware devices. With real-time recognition and the mapping of the natural gaze without the user's awareness, the system has numerous potential applications for researchers studying human behavior as well as in neuromarketing and retail companies. In addition, if the mapping system was used when watching a television program, it could measure the effectiveness of advertising on viewers or apply it to training programs for athletes and pilots. It could be installed in a car and applied as a program that analyzes driver driving patterns and detects drowsiness. However, future research is necessary to extend the distance range for gaze mapping beyond the current implementation, which was limited to the distance in a retail shelving environment. Eye tracking accuracy could be improved by using a large gaze-estimation model, as the real-time processing via edge devices was difficult.

In our method, accurate measurements were difficult at 40 cm above and below the camera height, and accurate gaze-mapping was difficult when the gaze vector could not be resolved due to a user's eyes being obfuscated or when a user wore glasses. Furthermore, by enabling communication between installed Jetson modules in different commercial section, a large-scale gaze-mapping system could be developed. Finally, extending the facial

recognition module with a facial ID re-identification function could enhance the mapping system's capabilities by identifying the user's trajectory in each commercial section.

Author Contributions: Methodology, H.A.; Software, H.A.; Validation, J.G.; Formal analysis, H.A., J.G. and M.J.; Investigation, J.J., D.K. and M.J.; Data curation, J.J. and D.K.; Writing—original draft, H.A., J.G. and M.J.; Writing—review & editing, J.G. and M.J.; Supervision, M.J.; Project administration, M.J.; Funding acquisition, M.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Argyle, M. Non-verbal communication in human social interaction. In *Non-Verbal Communication*; Cambridge U. Press: Cambridge, UK, 1972; Volume 2.
2. Goldin-Meadow, S. The role of gesture in communication and thinking. *Trends Cogn. Sci.* **1999**, *3*, 419–429. [[CrossRef](#)] [[PubMed](#)]
3. Rayner, K. Eye movements in reading and information processing: 20 years of research. *Psychol. Bull.* **1998**, *124*, 372. [[CrossRef](#)] [[PubMed](#)]
4. Jacob, R.J.K.; Karn, K.S. Eye tracking in human–computer interaction and usability research: Ready to deliver the promises. In *The Mind's Eye*; North Holland: Amsterdam, The Netherlands, 2003; pp. 573–605.
5. Vicente, F.; Huang, Z.; Xiong, X.; De la Torre, F.; Zhang, W.; Levi, D. Driver gaze tracking and eyes off the road detection system. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2014–2027. [[CrossRef](#)]
6. Massé, B.; Ba, S.; Horaud, R. Tracking gaze and visual focus of attention of people involved in social interaction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 2711–2724. [[CrossRef](#)]
7. Ramirez Gomez, A.; Lankes, M. Towards designing diegetic gaze in games: The use of gaze roles and metaphors. *Multimodal Technol. Interact.* **2019**, *3*, 65. [[CrossRef](#)]
8. Khan, M.Q.; Lee, S. Gaze and eye tracking: Techniques and applications in ADAS. *Sensors* **2019**, *19*, 5540. [[CrossRef](#)] [[PubMed](#)]
9. Jen, C.L.; Chen, Y.L.; Lin, Y.J.; Lee, C.H.; Tsai, A.; Li, M.T. Vision based wearable eye-gaze tracking system. In Proceedings of the 2016 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 7–11 January 2016; pp. 202–203.
10. Huang, M.X.; Kwok, T.C.; Ngai, G.; Leong, H.V.; Chan, S.C. Building a self-learning eye gaze model from user interaction data. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 1017–1020.
11. Sesma, L.; Villanueva, A.; Cabeza, R. Evaluation of pupil center-eye corner vector for gaze estimation using a web cam. In Proceedings of the Symposium on Eye Tracking Research and Applications, Santa Barbara, CA, USA, 28–30 March 2012; pp. 217–220.
12. Sun, L.; Liu, Z.; Sun, M. Real time gaze estimation with a consumer depth camera. *Inf. Sci.* **2015**, *320*, 346–360. [[CrossRef](#)]
13. Wood, E.; Baltrušaitis, T.; Morency, L.P.; Robinson, P.; Bulling, A. A 3d morphable eye region model for gaze estimation. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016, Proceedings, Part I 14*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 297–313.
14. Mansanet, J.; Albiol, A.; Paredes, R.; Mossi, J.M.; Albiol, A. Estimating point of regard with a consumer camera at a distance. In *Pattern Recognition and Image Analysis: 6th Iberian Conference, IbPRIA 2013, Funchal, Madeira, Portugal, 5–7 June 2013. Proceedings 6*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 881–888.
15. Xu, L.; Machin, D.; Sheppard, P. A Novel Approach to Real-time Non-intrusive Gaze Finding. In Proceedings of the British Machine Conference, Southampton, UK, 14–17 September 1998; pp. 1–10.
16. Wang, K.; Ji, Q. Real time eye gaze tracking with 3d deformable eye-face model. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1003–1011.
17. Park, S.; Zhang, X.; Bulling, A.; Hilliges, O. Learning to find eye region landmarks for remote gaze estimation in unconstrained settings. In Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications, Warsaw, Poland, 14–17 June 2018; pp. 1–10.
18. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [[CrossRef](#)]
19. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.

20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
21. Zhang, X.; Sugano, Y.; Fritz, M.; Bulling, A. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *41*, 162–175. [[CrossRef](#)] [[PubMed](#)]
22. Cortacero, K.; Fischer, T.; Demiris, Y. RT-BENE: A dataset and baselines for real-time blink estimation in natural environments. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27–28 October 2019.
23. Wood, E.; Baltrusaitis, T.; Zhang, X.; Sugano, Y.; Robinson, P.; Bulling, A. Rendering of eyes for eye-shape registration and gaze estimation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3756–3764.
24. Krafska, K.; Khosla, A.; Kellnhofer, P.; Kannan, H.; Bh, arkar, S.; Matusik, W.; Torralba, A. Eye tracking for everyone. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2176–2184.
25. Kellnhofer, P.; Recasens, A.; Stent, S.; Matusik, W.; Torralba, A. Gaze360: Physically unconstrained gaze estimation in the wild. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6912–6921.
26. Viola, P.; Jones, M.J. Robust real-time facial detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. [[CrossRef](#)]
27. Pham, M.T.; Gao, Y.; Hoang, V.D.D.; Cham, T.J. Fast polygonal integration and its application in extending haar-like features to improve object detection. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 942–949.
28. Zhu, Q.; Yeh, M.C.; Cheng, K.T.; Avidan, S. Fast human detection using a cascade of histograms of oriented gradients. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2006; pp. 1491–1498.
29. Mathias, M.; Benenson, R.; Pedersoli, M.; Van Gool, L. Face detection without bells and whistles. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014, Proceedings, Part IV 13*; Springer International Publishing: Berlin/Heidelberg, Germany, 2014; pp. 720–735.
30. Yan, J.; Lei, Z.; Wen, L.; Li, S.Z. The fastest deformable part model for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2497–2504.
31. Zhu, X.; Ramanan, D. Face detection, pose estimation, and landmark localization in the wild. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2879–2886.
32. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016, Proceedings, Part I 14*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
33. Zhang, Z.; Luo, P.; Loy, C.C.; Tang, X. Facial landmark detection by deep multi-task learning. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014, Proceedings, Part VI 13*; Springer International Publishing: Berlin/Heidelberg, Germany, 2014; pp. 94–108.
34. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint facial detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [[CrossRef](#)]
35. Barbu, A.; Lay, N.; Gramajo, G. Face detection with a 3d model. In *Academic Press Library in Signal Processing*; Academic Press: Cambridge, MA, USA, 2018; Volume 6, pp. 237–259.
36. Zhang, S.; Zhu, X.; Lei, Z.; Shi, H.; Wang, X.; Li, S.Z. S3fd: Single shot scale-invariant facial detector. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 192–201.
37. Wang, J.; Yuan, Y.; Yu, G. Face attention network: An effective facial detector for the occluded faces. *arXiv* **2017**, arXiv:1711.07246.
38. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Wey, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
39. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
40. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
41. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for facial recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 815–823.
42. Wood, E.; Baltrušaitis, T.; Morency, L.P.; Robinson, P.; Bulling, A. Learning an appearance-based gaze estimator from one million synthesised images. In Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications, Charleston, SC, USA, 14–17 March 2016; pp. 131–138.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.