computers

MDPI

Article

Real-Time Gaze Estimation Using Webcam-Based CNN Models for Human–Computer Interactions

Visal Vidhya and Diego Resende Faria *

School of Physics, Engineering and Computer Science, University of Hertfordshire, College Lane, Hertfordshire, Hatfield AL10 9AB, UK; vv22aad@herts.ac.uk
* Correspondence: d.faria@herts.ac.uk

Abstract: Gaze tracking and estimation are essential for understanding human behavior and enhancing human–computer interactions. This study introduces an innovative, cost-effective solution for real-time gaze tracking using a standard webcam, providing a practical alternative to conventional methods that rely on expensive infrared (IR) cameras. Traditional approaches, such as Pupil Center Corneal Reflection (PCCR), require IR cameras to capture corneal reflections and iris glints, demanding high-resolution images and controlled environments. In contrast, the proposed method utilizes a convolutional neural network (CNN) trained on webcam-captured images to achieve precise gaze estimation. The developed deep learning model achieves a mean squared error (MSE) of 0.0112 and an accuracy of 90.98% through a novel trajectory-based accuracy evaluation system. This system involves an animation of a ball moving across the screen, with the user's gaze following the ball's motion. Accuracy is determined by calculating the proportion of gaze points falling within a predefined threshold based on the ball's radius, ensuring a comprehensive evaluation of the system's performance across all screen regions. Data collection is both simplified and effective, capturing images of the user's right eye while they focus on the screen. Additionally, the system includes advanced gaze analysis tools, such as heat maps, gaze fixation tracking, and blink rate monitoring, which are all integrated into an intuitive user interface. The robustness of this approach is further enhanced by incorporating Google's Mediapipe model for facial landmark detection, improving accuracy and reliability. The evaluation results demonstrate that the proposed method delivers high-accuracy gaze prediction without the need for expensive equipment, making it a practical and accessible solution for diverse applications in human–computer interactions and behavioral research.

Keywords: eye tracking; CNN; gaze estimation

Academic Editor: Xiaochen Lu

Received: 4 December 2024
Revised: 29 January 2025
Accepted: 5 February 2025
Published: 10 February 2025

Citation: Vidhya, V.; Resende Faria, D. Real-Time Gaze Estimation Using Webcam-Based CNN Models for Human–Computer Interactions. *Computers* **2025**, *14*, 57. https://doi.org/10.3390/computers14020057

1. Introduction

In the field of computer vision and human–computer interactions, gaze estimation aims to determine where a person is looking based on facial or eye images. This technology has diverse applications, including virtual reality [1], marketing research [2], and assistive technologies for individuals with disabilities [3]. Traditionally, gaze estimation relies on specialized hardware, such as infrared cameras, which are costly and cumbersome, restricting their use to controlled laboratory settings. This research addresses these challenges by developing a webcam-based gaze-tracking system powered by a convolutional neural network (CNN). The motivation lies in creating an affordable, accessible, and user-friendly alternative to traditional systems. By leveraging CNNs, this method effectively extracts

gaze information using standard webcams, significantly reducing costs and setup complexity. This innovative approach also emphasizes inclusivity, real-time processing, and adaptability, making it suitable for diverse applications, from assistive technologies to human–computer interactions, while prioritizing ethical and practical considerations.

1.1. General Overview of the Problem Domain

In recent years, eye-tracking technology has emerged as a powerful tool for enhancing accessibility, improving user interactions, and gaining insights into human behavior. While traditional eye-tracking systems offer high precision, they are often associated with significant drawbacks, such as high costs and limited accessibility. Consequently, these systems remain underutilized, especially in consumer-oriented applications where affordability and ease of use are critical.

The advent of deep learning, particularly convolutional neural networks (CNNs), has transformed computer vision, enabling the faster and more accurate processing of visual data. By leveraging CNNs, it is now possible to create gaze-tracking models using standard webcams. This democratization of eye tracking has unlocked numerous new applications and opportunities across various fields.

Despite the promise of webcam-based gaze tracking, several challenges must be addressed to achieve its successful integration into broader applications. The key issues include those listed in the following subsections.

1.1.1. Legal and Ethical Issues

Privacy Concerns: Eye tracking involves capturing sensitive data, including a user's focus, interests, and cognitive patterns. Collecting and using such data raises significant privacy concerns, particularly in consumer-facing applications where users may not fully understand the extent of the data being recorded.

Data Security: Ensuring the security of gaze data is crucial to preventing misuse or unauthorized access. Robust data protection measures and adherence to privacy regulations, such as the General Data Protection Regulation (GDPR), are essential to address these concerns.

1.1.2. Social and Ethical Issues

Bias and Inclusivity: CNN-based models can inherit biases present in their training datasets, leading to inaccuracies in gaze estimation for users of different ethnicities, eye shapes, or those with visual impairments. Addressing these biases is essential to ensuring that the technology is equitable and inclusive for all users.

Impact on User Behavior: Eye-tracking technology used in consumer applications, such as advertising or social media, could enable to creation of manipulative tactics that subtly influence user behavior. This raises ethical concerns regarding the misuse of gaze data in commercial settings, potentially leading to exploitative practices.

1.1.3. Economic Impact

The adoption of webcam-based eye tracking addresses cost-related challenges in gaze estimation, making the technology more accessible for research and commercial applications. By reducing dependency on specialized hardware, this approach enhances affordability, enabling its broader adoption in fields like healthcare, marketing, and gaming. This shift fosters innovation and supports cost-effective solutions for real-time gaze tracking.

1.1.4. Commercial Risks and Risk Management

Technical Limitations: Webcam-based systems may face challenges such as reduced accuracy in low-light environments or when tracking rapid eye movements. These limitations could hinder its adoption in fields that require high precision.

Market Acceptance: Concerns regarding the accuracy, reliability, and privacy of webcam-based eye tracking could lead to hesitation among users and industries. To mitigate these risks, thorough testing and transparent communication of the technology's benefits are essential.

By addressing these challenges, the development of CNN-based webcam gaze tracking has the potential to revolutionize the field, making it more accessible, affordable, and applicable to a wide array of domains.

To address the ethical and legal aspects of human-subject experiments and data collection, the proposed method incorporates privacy-conscious design principles. By utilizing webcam-based gaze tracking, which avoids intrusive hardware setups, this approach minimizes user discomfort and fosters transparency. Participants are fully informed about the data collection process, and explicit consent is obtained, adhering to ethical guidelines. Additionally, all data are anonymized and securely stored, ensuring compliance with privacy regulations such as GDPR. This method also focuses on inclusivity by addressing potential biases in the CNN model, ensuring fair treatment across diverse user demographics, including those with visual impairments.

1.2. Application Importance of Gaze Estimation

Gaze estimation is a transformative technology with applications spanning assistive tools, digital interactions, and consumer behavior analysis. For individuals with disabilities, it enables hands-free control of devices and software, fostering greater accessibility and independence [3,4]. In gaming and virtual reality, gaze tracking enhances immersion by facilitating natural interactions driven by users' focus points [5].

Beyond these domains, gaze estimation is revolutionizing marketing, sales, and advertising by delivering actionable insights into consumer attention. Real-time gaze data allow advertisers to create personalized, adaptive ad content, optimizing its placement and design for maximum relevance and engagement [6]. In sales, gaze tracking transforms shopping experiences by providing context-sensitive product information or offers based on visual focus, boosting conversion rates. Additionally, it offers an unparalleled analysis of consumer behavior, identifying which ad elements capture attention and linking engagement to purchasing decisions.

Integrating versatile, cost-effective gaze estimation systems into digital marketing strategies enables businesses to enhance consumer engagement, improve ad effectiveness, and drive sales growth. From assistive technologies to commercial applications, gaze estimation continues to unlock innovative opportunities across industries, bridging accessibility, interactivity, and consumer insights.

2. Related Work

Gaze tracking plays a crucial role in advancing human–computer interactions (HCIs) and behavioral research. Traditional methods, such as Pupil Center Corneal Reflection (PCCR), depend on expensive infrared (IR) cameras and controlled environments. These systems, while precise, are cost-prohibitive and impractical for widespread use.

2.1. Gaze-Tracking Methods

Zhu and Ji [7] developed a gaze-tracking system capable of functioning with natural head movements. This innovation marked a significant step toward more flexible and

user-friendly solutions, demonstrating that accurate gaze estimation is possible without requiring users to remain static. This advancement paved the way for dynamic applications of gaze tracking.

Similarly, Macinnes et al. [8] explored wearable eye-tracking devices, comparing their precision and accuracy. Their findings underscored the trade-offs between device mobility and tracking performance, emphasizing the need for techniques that maintain high accuracy without relying on specialized hardware. This study highlighted the demand for accessible gaze-tracking solutions capable of operating effectively in diverse conditions. To address these challenges, the proposed method diverges from traditional IR-based techniques, offering an affordable and versatile alternative that employs standard webcams and CNNs.

Wood et al. [9] made contributions to this area with their appearance-based gaze estimator, which utilized a dataset of one million synthesized images. Their work underscored the importance of data diversity in training CNNs for gaze estimation. However, their use of synthetic datasets posed challenges for achieving high accuracy in real-world scenarios. Their study focused on controlled environments, limiting its applicability in dynamic settings containing natural head movements. Although the dataset was groundbreaking in addressing the scarcity of training data, it did not fully resolve issues related to generalization across varied conditions.

Building on these advancements, Krafka et al. [10] introduced a gaze-tracking system designed for mobile devices, leveraging crowdsourced data to capture a broader range of real-world scenarios. This approach significantly improved the generalizability of gaze-tracking models by incorporating diverse environmental conditions. However, the system was constrained by its reliance on specific head positions and struggled to handle the free head movements common in natural settings. While the inclusion of a larger and more diverse dataset enhanced model robustness, the variability in camera quality across mobile devices introduced inconsistencies in gaze estimation.

Deng and Zhu [11] tackled the challenge of free head movements by introducing a monocular 3D gaze-tracking system. Their deep learning model incorporated geometric constraints to improve gaze estimation accuracy under natural head movement conditions, representing a significant advancement in the realistic and practical applications of gaze tracking. However, the system demanded substantial computational resources, making it less viable for real-time use on standard consumer devices. While the introduction of geometric constraints was innovative, it added complexity to the model, limiting its applicability in resource-constrained environments.

Building on these developments, Liu et al. [12] proposed an appearance-based gaze estimation method optimized for free head movements and mobile devices. Their approach refined Deng and Zhu's [11] work by tailoring the model to the computational limitations of mobile hardware. Despite this, the trade-off between model complexity and efficiency led to a slight reduction in accuracy. The study extended gaze tracking to more practical applications but continued to face challenges such as maintaining high accuracy across varying lighting conditions and device types.

Chen et al. [13] explored the impact of different deep network architectures on CNN-based gaze tracking, providing a comparative analysis of their accuracy and computational efficiency. This research served as a guide for selecting suitable architectures for specific gaze-tracking needs. However, Chen et al. [13] did not propose a novel gaze-tracking system, instead focusing on optimizing existing models. Their findings highlighted the importance of balancing accuracy and real-time performance, especially in environments with limited resources.

Kanade et al. [14] presented a CNN-based eye-gaze-tracking system designed to enhance driver safety. Their system utilized machine learning algorithms to accurately predict eye gaze under challenging conditions, demonstrating high precision and efficiency in this specific application. However, its generalizability to other human–computer interaction (HCI) contexts was limited. The model's optimization for web-based applications left gaps in addressing broader challenges, such as variations in environmental conditions like lighting and camera angles.

Ansari et al. [15] introduced a gaze-tracking system that leveraged an unmodified webcam and a CNN, aiming to make gaze tracking more accessible by eliminating the need for specialized hardware. This approach marked a significant step in democratizing gaze-tracking technology, yet it faced limitations in achieving high precision due to the constraints of standard webcams. The lower image quality provided by these devices impacted the system's accuracy, making it less suitable for demanding applications. Nonetheless, the study represented a commendable effort to strike a balance between accessibility and performance, underscoring the need for further innovation to enhance both accuracy and practicality.

Singh and Modi [16] advanced the accessibility of gaze tracking by creating a robust real-time camera-based system powered by deep learning. Designed to analyze users' visual attention with high precision, their system demonstrated its suitability for diverse applications. By enhancing robustness across varying environmental conditions, the study addressed the limitations seen in earlier approaches. The CNN architecture utilized showcased significant improvements in performance. However, the reliance on relatively high-quality cameras posed a barrier to broader adoption, especially in scenarios where only standard webcams are available.

Narayana Darapaneni et al. [17] explored the application of CNNs in eye-tracking analyses, focusing on educational and training environments. Their system prioritized accuracy and efficiency, making it well-suited for real-time use. Despite these strengths, the research was primarily conducted in controlled settings, which limited its applicability within more dynamic, real-world contexts. This work underscored the need for further innovation to enhance adaptability and generalizability in gaze-tracking systems.

Donuk et al. [18] developed a real-time eye-tracking system tailored for web mining applications, leveraging CNNs to achieve high accuracy. Their research highlighted the potential of using gaze tracking for specialized applications, particularly in web mining. However, like many systems, it was constrained by the requirement for high-quality input data and controlled testing environments. While the study offered valuable insights into niche uses, it did not address the broader challenges of extending gaze-tracking technology to diverse and less controlled domains.

2.2. Applications of Gaze Tracking

Zhang et al. [19] pioneered the use of eye tracking for analyzing viewer engagement with video advertisements. By examining fixation duration and focus points, their system offered a deeper understanding of consumer attention and preferences. This foundational study helped advertisers identify the most captivating elements of their ads. However, reliance on hardware-based eye-tracking devices restricted accessibility and scalability, making the system less practical for widespread use.

Building on this, Lee et al. [20] introduced a gaze-data visualization system, enabling a more granular analysis of user engagement by mapping gaze patterns onto specific advertisement elements. Despite its innovative approach, the study shared similar limitations, as it required specialized equipment and controlled environments. Moreover, it

focused primarily on static content, leaving the dynamic nature of video advertisements largely unaddressed.

Okano and Asakawa [21] bridged this gap by analyzing attention consumers paid to product messages across web ads and TV commercials. Their study revealed that different media formats significantly influence consumer perceptions and the retention of product messages, emphasizing the importance of context in advertising. However, the lack of machine learning techniques limited their ability to extract deeper insights from their gaze data.

Expanding on these findings, Zhang and Yuan [22] conducted a comprehensive analysis of video advertisements, correlating specific ad elements with their effectiveness. This study provided actionable insights for optimizing content but remained dependent on traditional hardware, making it cost-prohibitive. Additionally, their work focused on descriptive analyses, leaving predictive modeling unexplored.

Muñoz-Leiva et al. [23] conducted a thematic analysis of eye-tracking applications in marketing, identifying emerging trends and research gaps. They highlighted the lack of studies leveraging deep learning to improve the accuracy and efficiency of gaze tracking, particularly in dynamic advertising environments. Their work underscored the potential of combining eye tracking with machine learning for real-time applications.

Modi and Singh [24] addressed some of these challenges by developing a real-time CNN-based gaze-tracking system using standard webcams. This innovation eliminated the need for specialized hardware, making the technology more accessible and scalable. While their study focused on social media applications, it set the stage for applying CNNs in broader marketing and advertising contexts.

Onwuegbusi et al. [25] explored gaze behavior among young audiences exposed to gambling and non-gambling advertisements. Their study underscored the importance of understanding how various ad types capture attention, offering valuable insights for regulatory policies. Using deep learning techniques could have enhanced the granularity and accuracy of their analysis.

Xie et al. [26] advanced the field by incorporating machine learning into gaze tracking for mobile advertisements. Their ambulatory eye-tracking study improved the precision of consumer attention analysis, emphasizing the relevance of machine learning in dynamic and real-world settings.

Finally, Tsubouchi et al. [27] introduced an innovative approach to personalized web advertising on smartphones, aligning real-time advertisements with the user's gaze. This novel application demonstrated the potential for gaze tracking to transform targeted advertising. However, the study was limited to smartphones and did not fully utilize CNNs to enhance tracking accuracy or explore their applications in other advertising media.

2.3. Advancing Beyond the State of the Art

The proposed research aims to address the limitations of traditional eye-tracking methods by developing an affordable, real-time gaze-tracking system using a standard webcam and a CNN. Traditional eye-tracking systems often depend on specialized hardware and controlled environments, limiting their accessibility and scalability. Furthermore, the application of CNN-based gaze tracking in marketing and advertising remains underexplored. The proposed system enhances real-time gaze analysis by incorporating features such as gaze heatmaps, fixation analysis, and blink rate detection, providing deeper insights into consumer behavior. By integrating these advanced features with the real-time processing capabilities of CNNs, the system offers marketers and advertisers a powerful tool to evaluate campaign effectiveness and optimize content based on consumer engagement. This approach not only overcomes the challenges of traditional methods but also fills a critical

gap in the current research by fully exploring the potential of CNNs in marketing and advertising. Ultimately, this research has the potential to provide more accurate, actionable insights, significantly enhancing advertising strategies and contributing to the growing body of knowledge on gaze-tracking technology.

3. Materials and Methods

In this section, we propose a shallow CNN-based approach to gaze tracking, prioritizing computational efficiency and reduced complexity over deeper architectures like VGG16 or DenseNet. While deeper CNNs excel in feature extraction, their increased parameters can lead to overfitting with limited data. Our shallow architecture effectively captures essential features from the eye region, making it well suited for real-time gaze tracking with minimal computational resources. To provide a clearer understanding of the overall process, Figure 1 illustrates the flow of the proposed method, outlining the steps involved in data collection, data pre-processing, and data analysis.

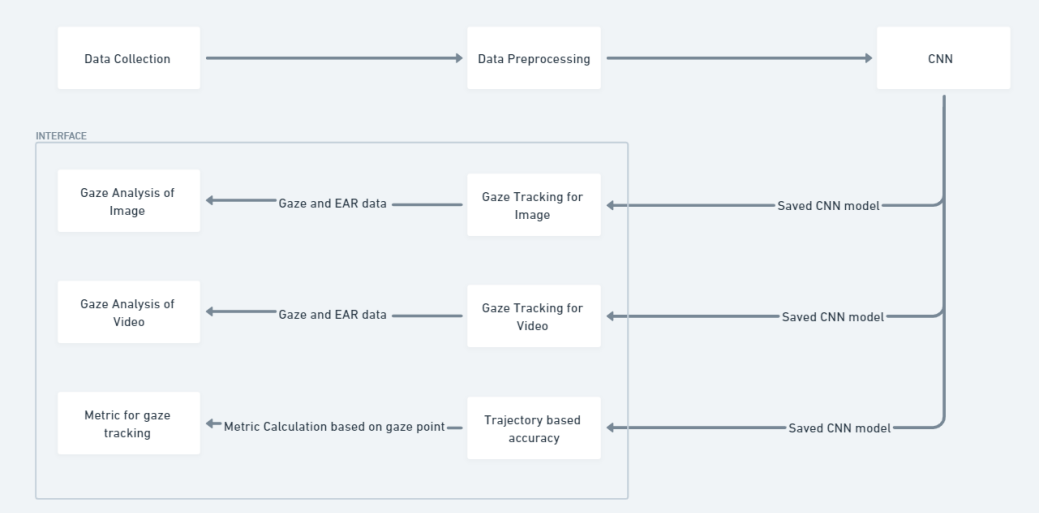

Figure 1. Flowchart of the proposed method.

3.1. Data Collection

This study uses a standard webcam along with the OpenCV library to capture images of the right eye region. Data were collected in a well-lit environment and from nine participants: four males, four females (all over 28 years old), and one eight-year-old child. In accordance with ethical guidelines, all participants were informed about the study's purpose and signed a consent form prior to the experiment. The proposed method employs Mediapipe, a pre-trained model developed by Google, to extract facial landmarks. Specifically, the landmarks corresponding to the right eye region were identified and passed to OpenCV for image capture. The data from all participants will be used to validate the system's accuracy under realistic conditions, such as varying user movements and gaze shifts. The testing phase plays a vital role in assessing the model's generalizability, trajectory-based accuracy, and ability to consistently capture and interpret gaze patterns, making the evaluation comprehensive and robust.

To facilitate data collection, the screen is divided into a grid of 16 cells, each containing a pulsating red dot at its center to draw the user's attention (see Figure 2). The dot appears sequentially in each grid cell, remaining visible for five seconds to ensure accurate capture of the user's gaze images. All participants sat comfortably in front of the screen, approximately 40 cm away from the webcam, focusing on the red dot as it moved across the screen. During this time, images of the right eye region were recorded in grayscale and resized to 256×256 pixels.

During data collection, individual-level discrepancies were noted. Adults displayed stable gaze patterns with minimal head movement, resulting in high-quality images. In contrast, the child participant exhibited rapid gaze shifts and occasional head movement, leading to variability and noise. Additional pre-processing, including filtering blurred frames and using Mediapipe's robust detection, addressed these issues. Variations in eye shape, size, and lighting also affected image quality. Participants wearing glasses experienced occasional glare, mitigated by adjusting the screen brightness and testing angles beforehand.

The data collection process lasted approximately 80s, during which time the required images were systematically captured and saved. These images were organized into a main directory, with each subfolder labeled according to its corresponding grid number. In total, 3387 images were collected, averaging about 211 images per grid cell.

The methodology presented in this study highlights the effectiveness of using Mediapipe and OpenCV for capturing and processing eye-tracking data. Mediapipe provides pre-built models for precise eye detection and tracking, while OpenCV manages image processing tasks, enhancing both the accuracy and efficiency of the eye-tracking process.

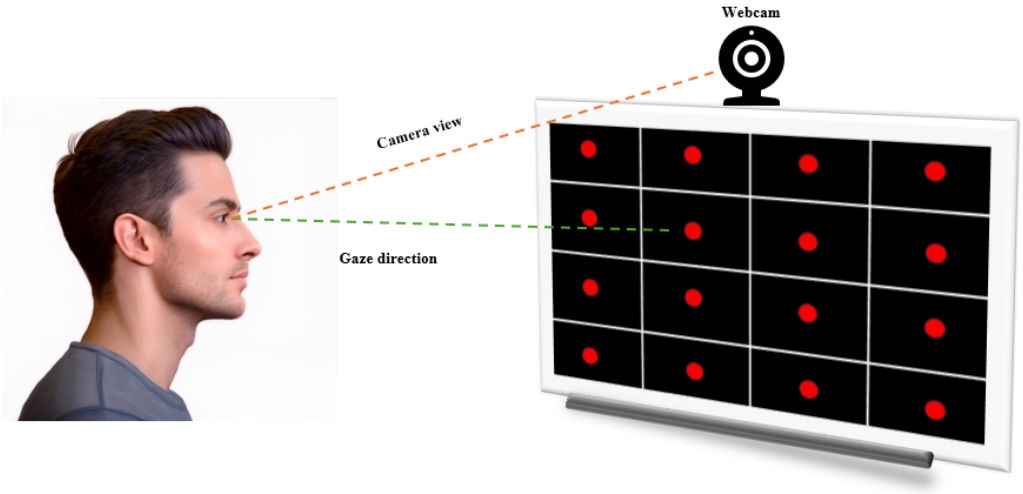

Figure 2. Experimental setup to collect data for gaze tracking. The red dots indicate the center points of each cell within the grid. The grid consists of 16 predefined regions, systematically dividing the screen for spatial reference and analysis.

3.2. Data Pre-Processing

Data pre-processing is a vital step in this methodology, one which encompasses data cleaning, image resizing, normalization, and augmentation to enhance the model's ability to generalize. Effective pre-processing ensures that the neural network can accurately learn and predict gaze directions, which are essential to the model's overall success.

The pre-processing pipeline, illustrated in Figure 3, begins by capturing eye region images using the facial landmarks detected by Mediapipe in conjunction with OpenCV. First, Mediapipe is initialized to identify 468 facial landmarks, with a focus on the right-eye region. The landmarks surrounding the right eye are isolated to precisely capture the area of interest. Once extracted, the image is converted to grayscale using OpenCV and resized to 256 $\times$ 256 pixels. This standardized image size is essential for training the convolutional neural network (CNN), ensuring consistent input data that supports accurate predictions. The second step is data cleaning, which ensures the creation of a high-quality dataset for model training. This process is performed manually, with each image carefully inspected grid by grid. The goal is to remove any images containing blinks or distortions that could

lead to inaccurate predictions. As the red dot moves sequentially across the grid, some early images may not align correctly with the intended gaze direction.

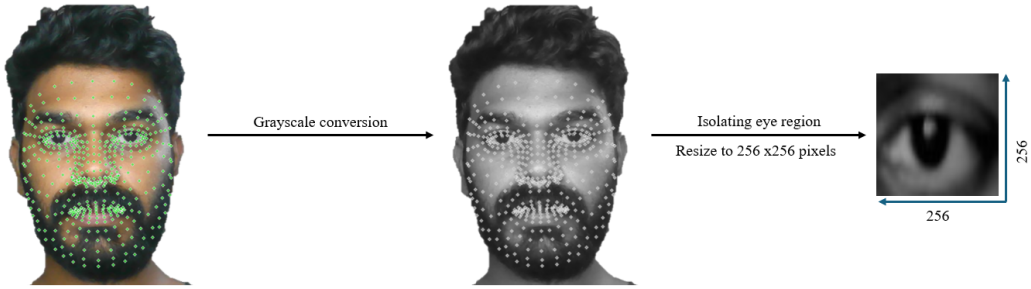

Figure 3. Image Pre-processing: Landmark detection and eye-region segmentation.

These misaligned images, shown in Figure 4, are excluded to prevent them from negatively impacting the neural network's training. Future work aims to automate this detection and correction process for greater efficiency.

The final step of pre-processing involves converting the images to grayscale to ensure consistent image formatting. All images are then resized to the standard 256 × 256 pixel dimensions, and pixel normalization is applied to enhance model stability. The grayscale images have pixel values ranging from 0 to 255, where 0 represents black and 255 represents white. To normalize, each pixel value is divided by 255.0, scaling the values to a range of 0 to 1. This normalization step standardizes the input data, improving the model's stability and enhancing its ability to generalize during training.

Figure 4. Examples of sequence of eye gaze where the blinking eyes and misaligned gaze directions (i.e. within the red box) are removed .

3.3. Model Development and Training

After pre-processing, the data are ready for the convolutional neural network (CNN) model. Its architecture, shown in Figure 5, comprises convolutional layers, pooling layers, fully connected layers, and an output layer. Convolutional layers extract hierarchical features using image convolution mechanisms, identifying patterns like edges and textures. Pooling layers reduce feature map dimensions, minimizing computational complexity and overfitting. Fully connected layers integrate the extracted features, enabling complex pattern recognition. The output layer predicts gaze coordinates for regression tasks. Model training involves optimizing weights through backpropagation, guided by a loss function, such as the mean squared error (MSE), to minimize prediction errors efficiently.

The algorithm for real-time gaze estimation using CNN is presented in Algorithm 1. Below we describe the CNN layers and parameters used.

Algorithm 1 Real-Time Gaze Estimation using CNN.

Input: Right eye image in grayscale, size 256×256
Output: Gaze coordinates (x, y)
Step 1: Preprocessing
Normalize the grayscale image values to the range [0, 1]
Resize or pad the image to 256×256 if necessary
Step 2: CNN Layers
Convolutional Layer 1:
for each filter $f \in \text{Conv1}(32\,\text{filters}, 3 \times 3)$ **do**
  for each pixel (i, j) in the image **do**
    $Z_{ij}^{(1)} = \text{ReLU}\left(\sum_{m=1}^{3}\sum_{n=1}^{3} W_{mn}^{(1)} X_{(i+m)(j+n)} + b^{(1)}\right)$
  end for
end for
Subsampling Layer 1:
for each 2×2 pooling region (i, j) **do**
  $P_{ij}^{(1)} = \max_{p,q\in[0,1]}\left(Z_{(i+p)(j+q)}^{(1)}\right)$
end for
Convolutional Layer 2:
for each filter $f \in \text{Conv2}(64\,\text{filters}, 3 \times 3)$ **do**
  for each pixel (i, j) in the output from Subsampling Layer 1 **do**
    $Z_{ij}^{(2)} = \text{ReLU}\left(\sum_{m=1}^{3}\sum_{n=1}^{3} W_{mn}^{(2)} P_{(i+m)(j+n)}^{(1)} + b^{(2)}\right)$
  end for
end for
Subsampling Layer 2:
for each 2×2 pooling region (i, j) **do**
  $P_{ij}^{(2)} = \max_{p,q\in[0,1]}\left(Z_{(i+p)(j+q)}^{(2)}\right)$
end for
Convolutional Layer 3:
for each filter $f \in \text{Conv3}(128\,\text{filters}, 3 \times 3)$ **do**
  for each pixel (i, j) in the output from Subsampling Layer 2 **do**
    $Z_{ij}^{(3)} = \text{ReLU}\left(\sum_{m=1}^{3}\sum_{n=1}^{3} W_{mn}^{(3)} P_{(i+m)(j+n)}^{(2)} + b^{(3)}\right)$
  end for
end for
Subsampling Layer 3:
for each 2×2 pooling region (i, j) **do**
  $P_{ij}^{(3)} = \max_{p,q\in[0,1]}\left(Z_{(i+p)(j+q)}^{(3)}\right)$
end for
Step 3: Fully Connected Layers
Flatten the pooled output $P_{ij}^{(3)}$ into a 1D vector
Fully Connected Layer 1:
for each neuron $u \in \text{FC1}(128\,\text{neurons})$ **do**
  $Y_u^{(1)} = \text{ReLU}(W_u^{(1)} \cdot F + b_u^{(1)})$
end for
Fully Connected Layer 2:
for each neuron $v \in \text{FC2}(64\,\text{neurons})$ **do**
  $Y_v^{(2)} = \text{ReLU}(W_v^{(2)} \cdot Y_u^{(1)} + b_v^{(2)})$
end for
Step 4: Output Layer
for each output neuron $o \in \{x, y\}$ **do**
  $Y_o = W_o \cdot Y_v^{(2)} + b_o$
end for
Step 5: Return Gaze Coordinates
return: x, y

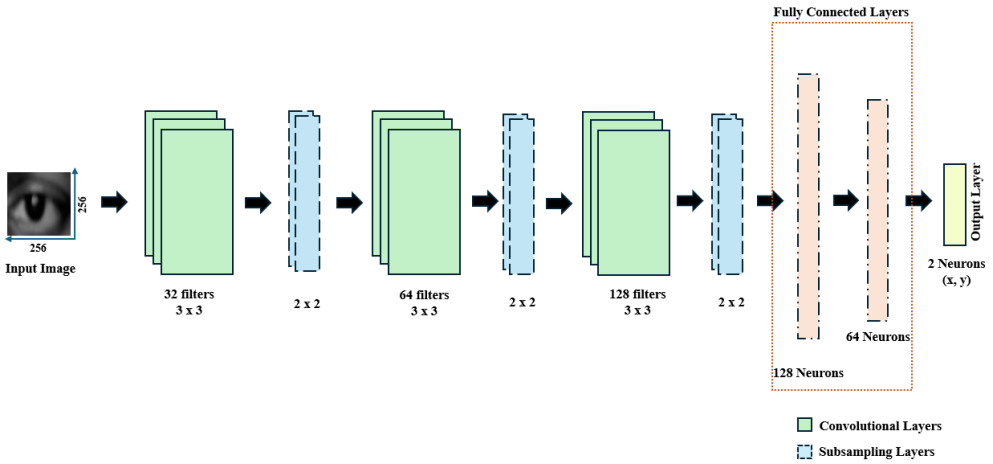

Figure 5. CNN architecture of the proposed method.

Convolutional Layers

The CNN starts with convolutional layers, which are essential for extracting hierarchical features from the input images. The first convolutional layer uses 32 filters, each 3×3 in size, and applies the Rectified Linear Unit (ReLU) activation function. This layer is responsible for detecting basic features, such as edges and textures. The output from this layer is then passed through a subsampling layer, which reduces the spatial dimensions of the feature map. The second convolutional layer, with 64 filters of the same size, captures more complex features by analyzing combinations of those detected in the first layer. This output is again subsampled to retain only the most significant information. The third convolutional layer, using 128 filters, abstracts the feature representations even further, detecting higher-level patterns in the data. The progressive increase in the number of filters allows the model to capture increasingly complex structures, making the feature extraction process more sophisticated. After this third layer, the output undergoes another subsampling operation.

3.3.1. Algorithm for the Proposed Method

Subsampling Layers

Subsampling, also known as max pooling, is applied after each convolutional layer to reduce the spatial dimensions of the feature maps. A 2×2 window slides across the feature map, selecting the maximum value within each window. This operation effectively retains the most important features while discarding less relevant information. By reducing dimensionality, max pooling not only decreases the computational complexity of the model but also helps prevent overfitting. Additionally, it contributes to a more abstract and generalized representation of the input image, focusing on prominent features in each region and helping the model learn more generalized patterns.

Fully Connected Layers

Following feature extraction and dimensionality reduction through the convolutional and pooling layers, the model transitions to fully connected layers. The Flatten layer converts the 2D feature maps into a 1D vector, which is then passed into dense layers. The first dense layer contains 128 neurons with ReLU activation, allowing the model to learn non-linear combinations of the features extracted from previous layers. The subsequent dense layer, which has 64 neurons and also uses ReLU activation, further refines these feature representations. These layers enable the model to learn the complex relationships between features, preparing it for accurate predictions.

Output Layer

The final layer of the model is the output layer, which consists of two neurons representing the x and y coordinates of the predicted gaze direction. Since this model is designed for regression tasks, no activation function is applied to the output layer, which is appropriate for predicting continuous values such as gaze coordinates. During model compilation, the mean squared error (MSE) loss function is used to measure the difference between the predicted gaze coordinates and actual values. The optimization process aims to minimize this error, guiding the model towards more accurate gaze location predictions.

The model is compiled using the Adam Optimizer, a well-regarded optimization method known for its efficiency in training deep neural networks. The learning rate is set to 0.001 to ensure stable and reliable convergence during training. The dataset is divided for comprehensive evaluation and tuning: 80% of the data is used for training, while the remaining 20% is reserved for validation. The training process involves multiple epochs, during which the model weights are adjusted iteratively based on the loss function to minimize prediction errors and improve the model's accuracy in estimating the direction of the user's gaze.

3.4. Data Analysis

After training the CNN model, it is used to track a user's gaze on displayed content, such as marketing visuals, images, and videos. This process involves identifying the areas of the content that capture the user's focus and measuring the duration for which their gaze on specific regions, providing valuable insights into the effectiveness of visual elements in capturing attention.

The content is displayed on a screen divided into 16 grids, each representing different sections of the visuals. The CNN model predicts the user's gaze coordinates on the screen, tracking their focus in real time. These gaze data, including the coordinates and duration of fixation, are recorded in an Excel file, creating a detailed log of where the user looked and for how long. These data are crucial for marketers to determine which parts of the content engage the audience most effectively and to identify areas that may need improvement.

In addition to gaze tracking, the Eye Aspect Ratio (EAR) is calculated to monitor the user's blink rate throughout the viewing process. The EAR is a measure of eye openness, computed using the following equation:

$$\text{EAR} = \frac{|P_2 - P_6| + |P_3 - P_5|}{2 \times |P_1 - P_4|}. \tag{1}$$

In Equation (1), $|P_2 - P_6|$ and $|P_3 - P_5|$ represent the vertical distances between specific eye landmarks, while $|P_1 - P_4|$ represents the horizontal distance (see Figure 6). A blink is detected if the Eye Aspect Ratio (EAR) falls below a threshold of 0.2 for a specified number of consecutive frames [28]. These blink data, along with timestamps, are also recorded in a separate Excel file, providing insights into user engagement and potential fatigue during image viewing. Monitoring blink rates can help marketers understand when users might lose focus or become fatigued, informing adjustments to content length or pace.

A method has been developed to simulate gaze tracking by drawing a circle at a randomly generated location within a specified grid cell on the screen. The method begins by determining the center of a grid cell based on the user's gaze coordinates. It then computes a random angle and distance within a 200-pixel radius around this center. This randomization introduces variability into the predicted gaze point, making the simulation more dynamic and realistic. After calculating this random location, a circle is drawn at the computed position, and the coordinates are stored for subsequent analysis.This method of generating random coordinates mimics the natural variability found in human gaze

behavior. In reality, the human eye does not focus on a single point but rather on a small area around the target, referred to as the "visual axis". This differs from the "optical axis", which is the straight line passing through the eye's optical centers. As shown in Figure 7, points. the deviation between these two axes is typically about 5 degrees [29], accounting for the natural dispersion of gaze

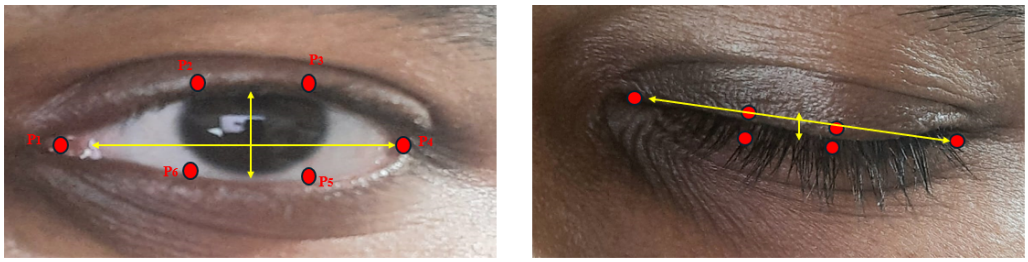

Figure 6. $|P_2 - P_6|$ and $|P_3 - P_5|$ represent the vertical distances between specific eye landmarks (red dots) and $|P_1 - P_4|$ represents the horizontal distance.

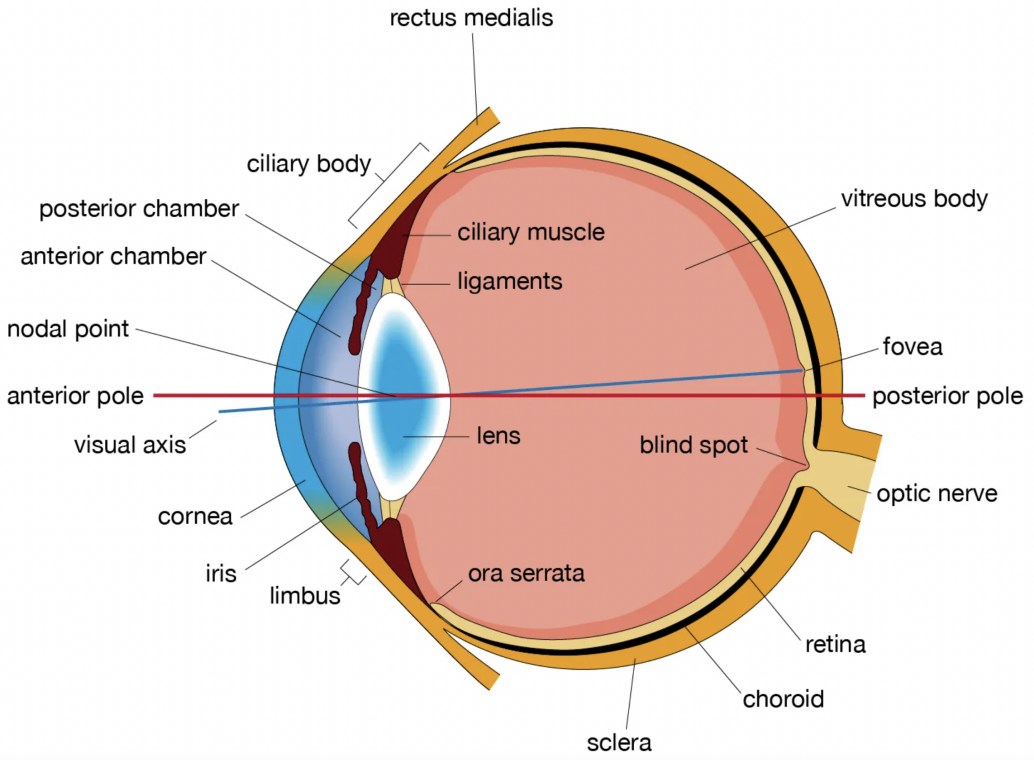

Figure 7. Optical view of the eye, showing that the angle between the visual axis and the optical axis is approximately 5 degrees [29]. Figure adapted from Encyclopædia Britannica [30].

Therefore, using random coordinates within a defined radius effectively simulates this scatter, accurately reflecting the inherent variability and imprecision of human gaze behavior.

To generate heatmaps for the images and videos, the algorithm begins by loading the gaze coordinates and initializing a heatmap matrix corresponding to the dimensions of the image or video frame. These coordinates are used to add values to the heatmap matrix, highlighting areas with a higher gaze concentration. For images, the heatmap is smoothed using a Gaussian blur, followed by normalization and contrast enhancement and then the application of a color map. The heatmap is then overlaid on the original image, and a grid is drawn to segment the image into regions for easier visual reference. For video processing, the algorithm operates frame-by-frame. As the video plays, the gaze coordinates are continuously updated in the heatmap. Each frame undergoes smoothing and contrast enhancement, and the heatmap is overlaid in real time, with a grid drawn on

each frame to highlight regions of interest. This enables the dynamic visualization of gaze patterns and a real-time analysis of user attention (e.g., fixations) throughout the video.

The heatmap can also be interpreted as a probabilistic representation of gaze fixation, which is calculated as follows:

$$\text{Probability for region} = \frac{\text{Fixation on the particular region}}{\text{Total fixation on the content}} \quad (2)$$

By transforming the raw gaze count data into a probability distribution, we generate a heatmap that represents the likelihood of fixation occurring across different regions of the visual content. This transformation is performed using Equation (2), where the total number of gaze fixations across the entire image or video frame is first summed. The probability of fixation in each region is then determined by dividing the count of fixations in that specific region by the total number of fixations. The heatmap not only highlights the regions of interest but also indicates the likelihood of each region attracting attention. This additional layer of analysis provides a more nuanced understanding of gaze patterns, offering deeper insights into how visual focus is distributed across the content. By quantifying the likelihood of fixation, this approach enhances our ability to interpret and optimize visual materials for maximum engagement.

3.5. Trajectory-Based Accuracy

Trajectory-based accuracy provides a novel and insightful approach for evaluating the performance of gaze-tracking systems. Traditional accuracy metrics often focus on point-based comparisons, assessing how closely predicted gaze points align with reference points. While these methods offer valuable information, they fail to capture the dynamic nature of gaze tracking, especially in real-world scenarios where gaze behavior is continuous and fluid. Trajectory-based accuracy addresses this limitation by comparing the predicted gaze path over time with a reference trajectory, providing a more comprehensive measure of system performance.

In this approach, the algorithm tracks gaze points predicted by a convolutional neural network (CNN) model as a user follows a moving object, often represented by a ball, across a grid on the screen. The user follows a predefined reference trajectory, such as a zigzag or circular pattern, which may cover all edges of the screen, as shown in Figure 8. Both the predicted gaze points and the reference trajectory are recorded over time, resulting in two sets of trajectories: one from the model's predictions and one from the predefined path. These trajectories are then converted into arrays to calculate various performance metrics.

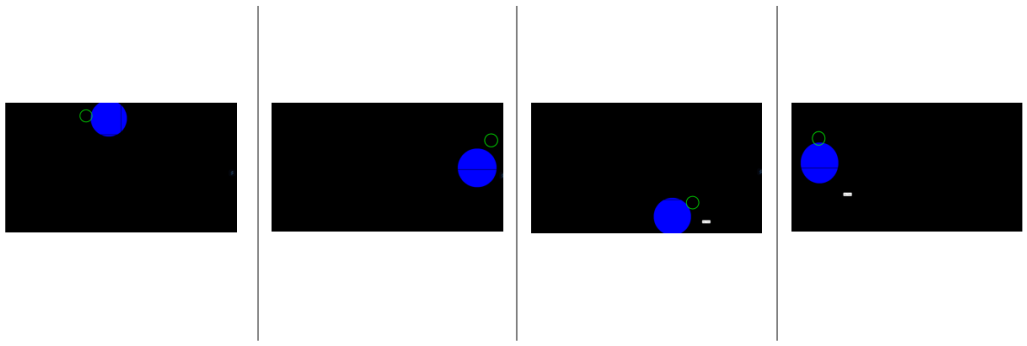

Figure 8. Trajectory-based accuracy method. The idea is to get the user gaze given the new trajectory of the blue ball.

Several key metrics are employed to quantify how closely the predicted gaze trajectory matches the reference trajectory.

Mean Absolute Deviation (MAD): This metric calculates the average absolute difference between the predicted gaze points and the reference trajectory. The formula for MAD is given as follows:

$$\text{MAD} = \frac{1}{n}\sum_{i=1}^{n}|gazedata_i - trajref_i|, \tag{3}$$

where n is the number of points, $gazedata_i$ is the predicted gaze point, and $trajref_i$ is the corresponding point on the reference path. A lower MAD indicates that the predicted gaze closely follows the reference trajectory.

Root Mean Squared Error (RMSE): The RMSE measures the average magnitude of the prediction error, giving more weight to larger errors. It is calculated as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(gazedata_i - trajref_i)^2}. \tag{4}$$

The RMSE is particularly useful for detecting outliers, as it penalizes large deviations more heavily than MAD.

Dynamic Time Warping (DTW): DTW is a method used for measuring the similarity between two sequences that may vary in time or speed.

It calculates the optimal alignment between two trajectories by minimizing the cumulative distance between them, allowing for non-linear alignments. The DTW distance between two trajectories $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_n)$ can be calculated using the following equation:

$$\text{DTW} = \min\left(\sqrt{\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}d(x_i, y_j)}\right), \tag{5}$$

where $d(x_i, y_j)$ is the Euclidean distance between points x_i from the predicted gaze trajectory X and y_j from the reference trajectory Y. The Euclidean distance d between two points is calculated as follows:

$$d(x_i, y_j) = \sqrt{(x_i - y_j)^2}. \tag{6}$$

Accuracy: A unique aspect of the trajectory-based accuracy metric is its ability to calculate accuracy based on a threshold distance around each reference point. In this case, the threshold is set to 150 pixels to cover the grid area. If a predicted gaze point falls within this threshold, it is considered correct. The accuracy is calculated as the ratio of correct points to the total number of points, as shown below:

$$\text{Accuracy} = \frac{\text{Number of correct gaze points}}{\text{Total number of gaze points}}. \tag{7}$$

Trajectory-based accuracy provides a more holistic view of the gaze-tracking system's performance, especially in scenarios where gaze movement is continuous. By considering the entire trajectory rather than isolated points, this method captures the dynamic aspects of gaze behavior, offering a valuable tool for evaluating and refining gaze-tracking systems. It is particularly beneficial in real-world applications where gaze patterns are fluid and constantly evolving.

4. Results

This section presents the results of the proposed CNN-based gaze-tracking model, trained using data collected from participants in diverse locations and environments. The evaluation covers various performance metrics, including model loss, gaze point plotting,

heatmap visualization, a trajectory-based analysis, and an accuracy comparison with other gaze-tracking models.

4.1. Model Evaluation and Case Analysis

Our CNN-based gaze-tracking model was trained over 50 epochs, with continuous monitoring of its validation loss to evaluate its performance on unseen data. By the final epoch, the validation loss was 0.0112, indicating the model's capability to generalize effectively to new data. As depicted in Figure 9, the training process showed a steady decline in loss, indicative of the model's efficient learning.

Initially, the loss decreased sharply, signifying that the model quickly learned fundamental features crucial for gaze prediction. Around the 20th epoch, the loss began to plateau, signaling the model's convergence toward optimal performance. This trend highlights the effectiveness of the proposed CNN architecture in learning underlying patterns from the training data, which is essential for precise gaze tracking.

To comprehensively assess the model's performance, three primary metrics were analyzed (Figure 10).

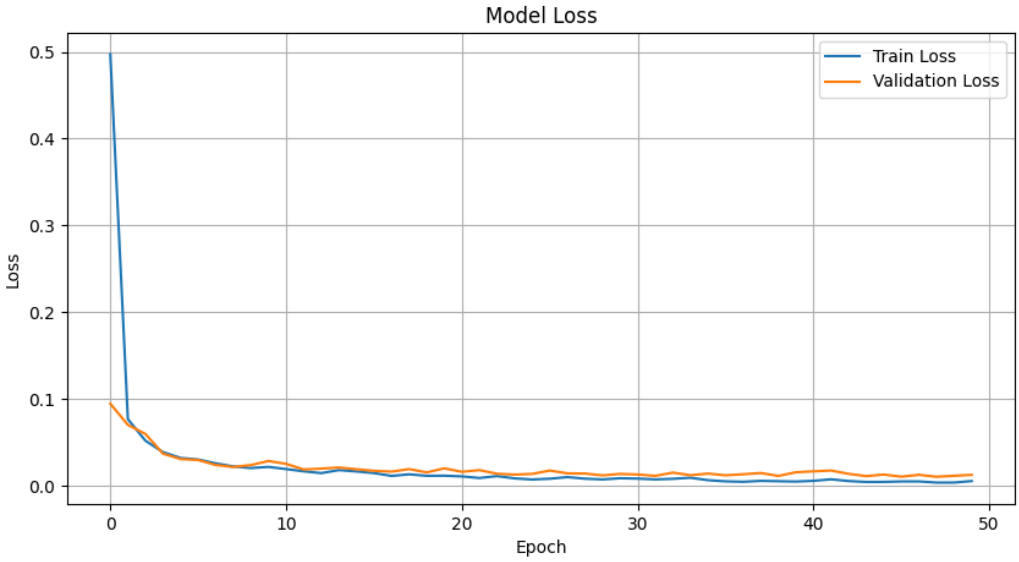

Figure 9. Model loss over 50 epochs, with validation loss of 0.0112.

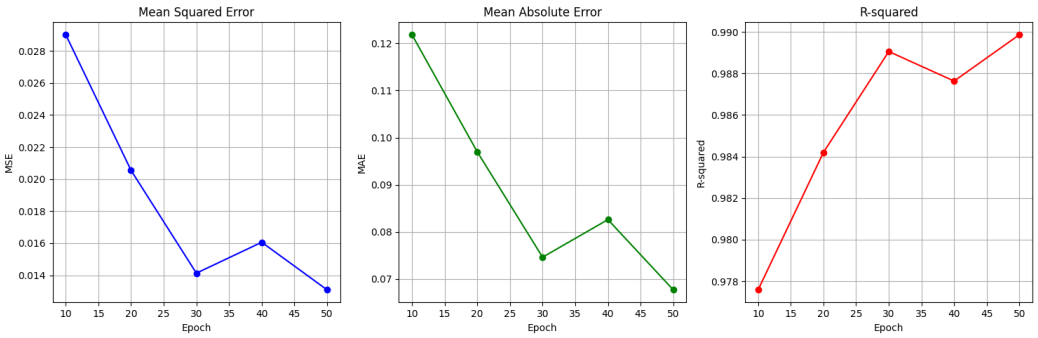

Figure 10. MSE (**left**), MAE (**middle**), and R-squared (**right**) values across epochs.

Mean Squared Error (MSE): The model achieved a low MSE of 0.0112, indicating that the predicted gaze points closely aligned with the actual values.

Mean Absolute Error (MAE): The MAE was 0.0531, suggesting that the average deviation between the predicted and actual gaze points was minimal, affirming the model's precision.

R-squared Value: A high R-squared value of 0.9953 was recorded, signifying that 99.53% of the variance in gaze positions was explained by the model. This underscores its strong predictive power and capacity to establish an accurate relationship between input images and gaze coordinates.

4.1.1. Real-Time Accuracy Assessment

The model's real-time prediction accuracy was validated using a 4×4 grid displayed on a blank screen. Users sequentially focused on each square of the grid while the model's predictions were plotted in real time. The predictions consistently fell within the grid square corresponding to the user's focus, demonstrating the model's reliability in real-time applications. Figure 11 highlights the strong alignment between actual gaze points and the predicted coordinates, with minimal deviation across the grid, showcasing the validity of the model's precision in dynamic scenarios.

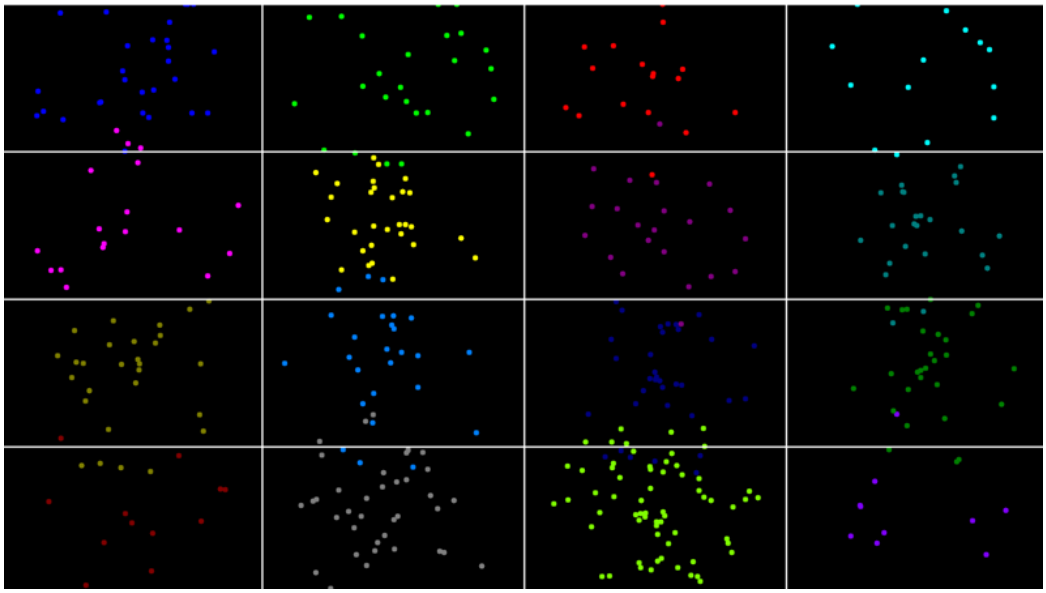

Figure 11. Gaze tracking in different regions of the screen.

4.1.2. Heatmap Analysis

The model's gaze-tracking performance was further analyzed using heatmap visualizations of static images and video content displayed on a 4×4 grid, as shown in Figures 12 and 13, which showcase data from one individual. The heatmaps represent the regions receiving the most visual attention, with intense areas indicating prolonged gaze fixation.

Static Image Heatmap: In Figure 12, using an image from Argos's official website, the heatmap revealed that the user's gaze was primarily concentrated on specific grid regions, confirming the model's ability to accurately identify areas of interest.

Video Heatmap: Figure 13 illustrates dynamic gaze tracking conducted on video content sourced from Pexels's official website. The heatmap demonstrated consistent tracking, with the gaze accurately following targeted regions across video frames. The link for the demo heatmap analysis can be found at https://drive.google.com/file/d/1pRTvb9lbpqt_sMOL0DxjfE0QlCOWaSvH/view?usp=sharing (accessed on Feb 2025.).

Figure 14 demonstrates the gaze fixation plots of an individual to showcase the model's capability to accurately record the grid regions with the highest concentration of gaze points—region 3 for the static image and region 5 for the video. This fixation analysis underscores the model's consistent performance in tracking visual focus across both static and dynamic content.

Figure 12. Heatmap visualization computed for a static image. The predefined grid (green lines) segments the image into distinct regions, while the heatmap illustrates the probability distribution of fixations across the screen, highlighting areas of visual attention.

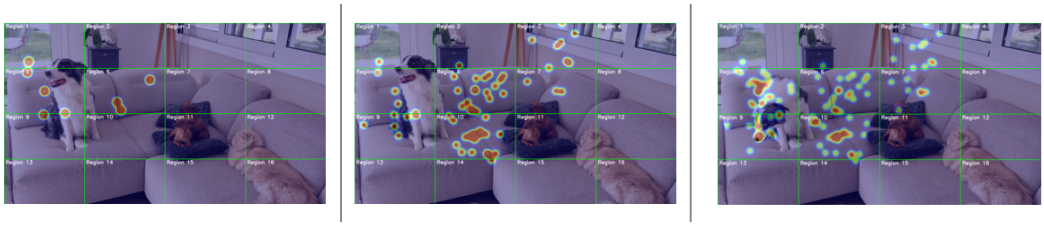

Figure 13. Heatmap interface: heatmap sequence computed given the frames of a video.

Additionally, Figure 15, also based on an individual example, demonstrates the utility of the Eye Aspect Ratio (EAR) graph in analyzing blink behavior during gaze tracking. The graph effectively captured the user's blinks, confirming the model's sensitivity to changes in eye state while maintaining precise gaze predictions. The probability distributions of gaze fixation depicted in Figure 16 and Table 1 similarly use individual examples to provide a detailed view of gaze patterns, offering deeper insights into visual focus and attention. These results demonstrate the model's effectiveness and highlight its robust capabilities.

Figure 17 shows the results of a sample participant for whom the trajectory accuracy was 94.65%, where the orange line represents gaze data, the blue line represents the ball's trajectory, and the green circle represents the threshold limit used to calculate the accuracy. This test involved nine participants and yielded a trajectory accuracy of 90.98%, calculated by averaging the accuracy across all participants. This high accuracy reflects the model's precision in following predefined paths. While accuracy can be influenced by factors such as real-time data processing, system requirements, and environmental conditions, the model consistently tracked gaze trajectories with minimal deviation. This robust performance is critical for applications requiring accurate real-time gaze tracking, including interactive systems, marketing analysis, and behavioral research, where precision and reliability are paramount.

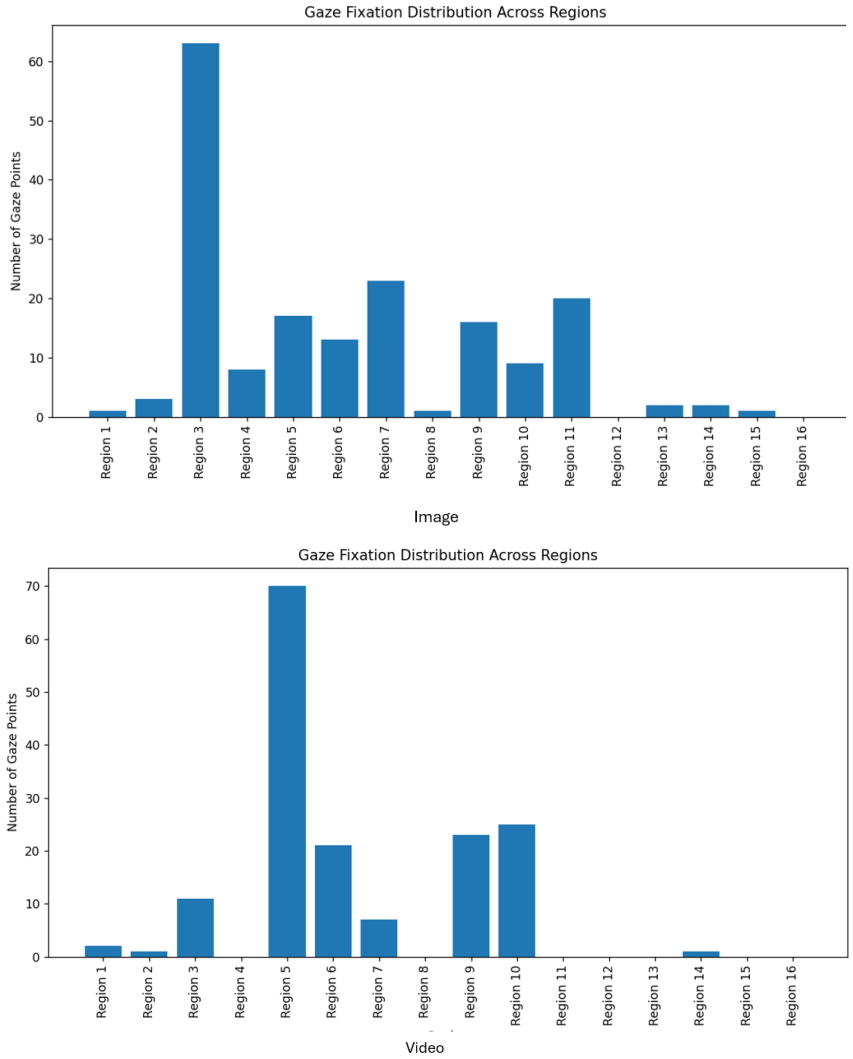

Figure 14. Gaze fixation interface: gaze fixation distribution over different regions on the screen.

Table 1. Percentage of gaze fixation in different regions for image and video.

Image Region	%	Video Region	%
Region 1	0.558659	Region 1	1.24224
Region 2	1.67598	Region 2	0.621118
Region 3	35.1955	Region 3	6.8323
Region 4	4.46927	Region 4	0
Region 5	9.49721	Region 5	43.4783
Region 6	7.26257	Region 6	13.0435
Region 7	12.8492	Region 7	4.34783
Region 8	0.558659	Region 8	0
Region 9	8.93855	Region 9	14.2857
Region 10	5.02793	Region 10	15.528

Table 1. *Cont.*

Image Region	%	Video Region	%
Region 11	11.1732	Region 11	0
Region 12	0	Region 12	0
Region 13	1.11732	Region 13	0
Region 14	1.11732	Region 14	0.621118
Region 15	0.558659	Region 15	0
Region 16	0	Region 16	0

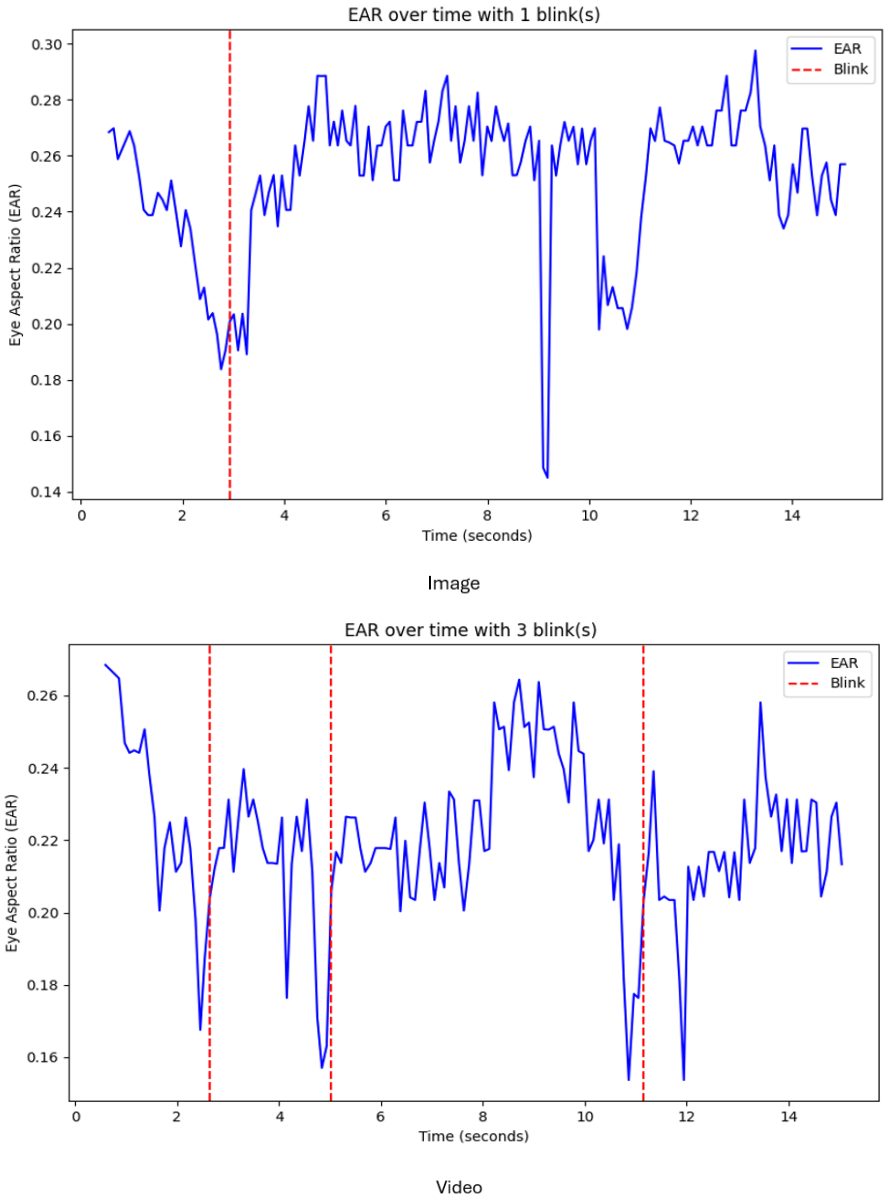

Figure 15. EAR plot displaying the number of blinks.

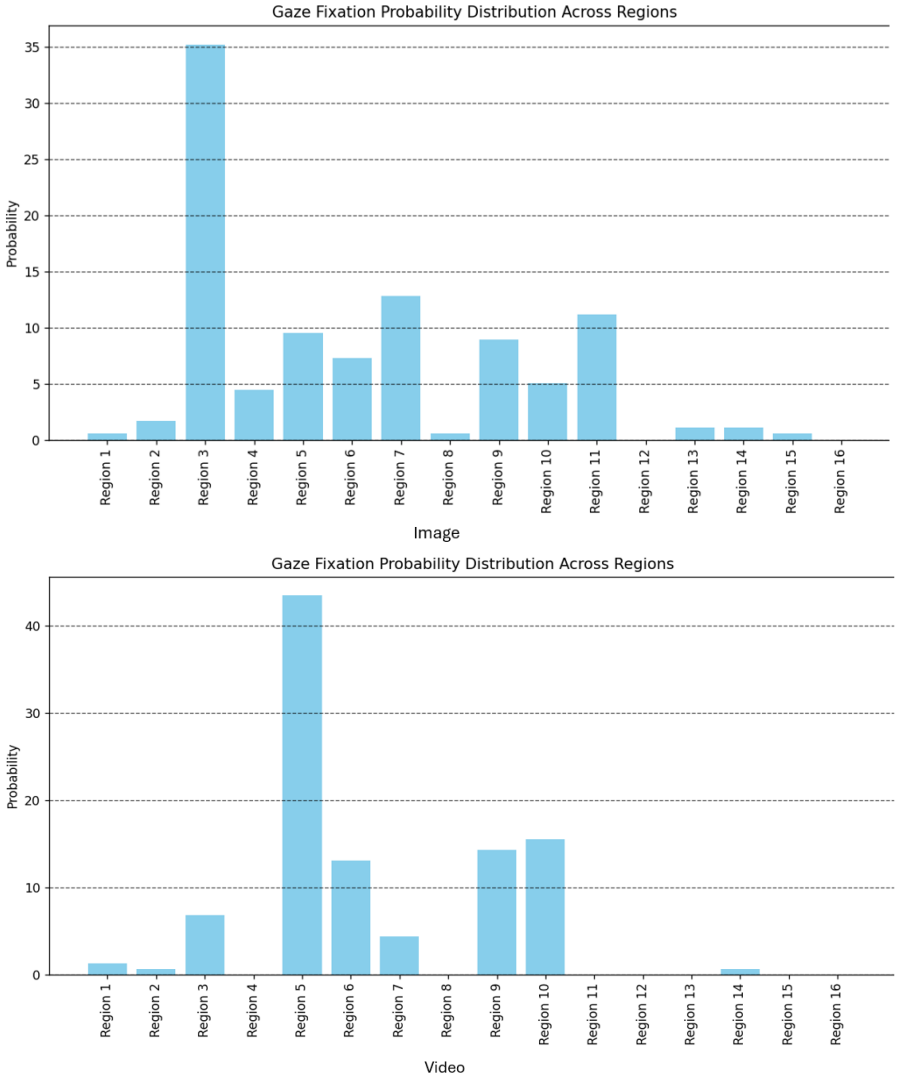

Figure 16. Gaze fixation interface: probability distribution of gaze fixation.

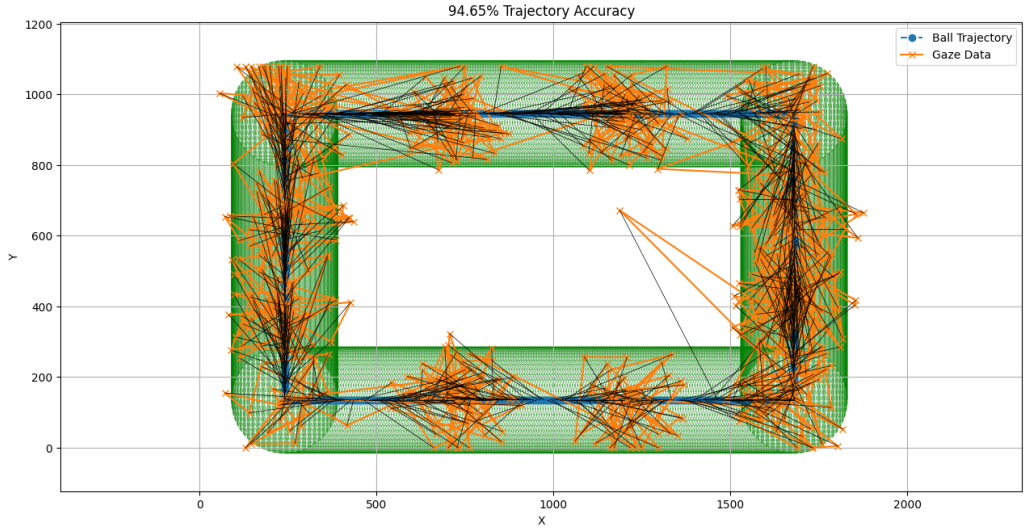

Figure 17. Accuracy of a sample participant using the trajectory-based method.

Mean Absolute Deviation (MAD): The MAD was 156, indicating the average deviation of the predicted gaze points from the reference path.

Root Mean Squared Error (RMSE): The RMSE was 208, representing the standard deviation of the prediction errors.

Dynamic Time Warping (DTW) Distance: The DTW distance was 172,100.09, which is a measure of the similarity between the predicted and reference trajectories, taking into account possible time shifts in the gaze path.

These metrics provide a comprehensive view of the model's trajectory-tracking performance, with the MAD and RMSE indicating the precision of individual predictions and the DTW distance assessing the overall alignment of the gaze trajectory with the predefined path.

4.2. Ablation Study and Performance Analysis

An ablation study was conducted to evaluate the contribution of different components of the proposed CNN model. The results indicate that the convolutional layers play a significant role in extracting meaningful spatial features from the input images, contributing the most to the model's success. With 256 × 256 pixel grayscale images of the right eye from nine participants, the model demonstrated a strong performance under well-lit conditions, leveraging its pre-processing steps and feature extraction capabilities. However, its performance declined in low-light environments due to reduced image quality. These findings highlight the importance of lighting and feature extraction in gaze-tracking accuracy.

4.3. Model Comparison

To evaluate the performance of the proposed CNN-based gaze-tracking model, its accuracy was compared with that of existing gaze-tracking models. Table 2 provides a detailed comparison, showcasing the accuracy metrics of each model.

Table 2. Comparison of the proposed model with existing gaze-tracking models.

Study	System Setup	Accuracy	Method Used
Wu et al. (2012) [31]	Webcam	88%	Support vector machine
Meng and Zhao (2017) [32]	Two cameras	88%	CNN
Sattar et al. (2020) [33]	Tobii eye tracker	80%	CNN
Ou et al. (2021) [34]	Wearable eye tracker	80%	CNN
Singh and Modi (2022) [16]	Webcam	84%	CNN
The proposed method	Webcam	90.98%	CNN

The results demonstrate the superior effectiveness of the proposed method, with notable improvements in both precision and reliability seen. This comparative analysis highlights the model's advanced capability to accurately predict gaze points, establishing it as a robust solution for gaze-tracking applications.

Singh and Modi [16] conducted a comprehensive literature review analyzing various gaze estimation models, evaluating them based on parameters such as accuracy, system setup, and the datasets used. Building on their findings, the present study introduces a novel gaze-tracking method that achieves a notable accuracy of 90.98%, outperforming most existing models, particularly those that rely solely on standard webcam setups. This final accuracy represents the average performance of nine participants who took part in the testing phase. The proposed model demonstrates precise gaze-tracking capabilities, making it a cost-effective and efficient alternative to complex systems that typically involve infrared light sources or multiple cameras.

Unlike earlier models that often required expensive equipment and frequent recalibration, this approach combines robustness with simplicity, eliminating the need for constant adjustments. Its practicality makes it particularly suitable for scenarios where both cost efficiency and reliable performance are crucial. By offering high accuracy without relying on advanced or costly hardware, the proposed method addresses the challenges of accessibility and affordability. This positions it as a strong candidate for real-world applications, especially in settings where deploying more intricate and expensive setups is impractical. Its versatility makes it suitable for a wide range of use cases, from marketing and user behavior analysis to interactive system design.

4.4. Gaze-Tracking Interface

Figure 18a presents a user-friendly graphical interface developed using Python's Tkinter library, designed to simplify the process of gaze tracking and analysis. This interface serves as a clean, visually appealing platform that facilitates the running of various scripts related to eye-tracking research and analysis tasks.

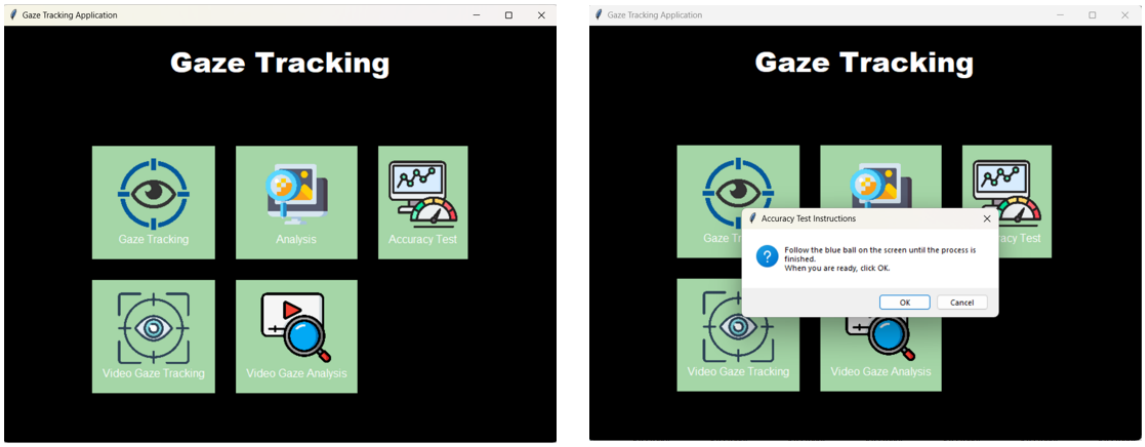

Figure 18. Main gaze-tracking interface (**a**) and all available options (tracking, calibration, test, and analysis). Figure (**b**) shows the accuracy test initialization.

4.4.1. Algorithm and Functionality

The interface consists of five primary buttons, each linked to specific scripts that perform key functions in gaze tracking and analysis.

Gaze-Tracking Button: Activates the image gaze-tracking script, which records and stores the user's gaze data for specific areas of interest in the image. It also tracks blink data, which is valuable for marketing research.

Video Gaze-Tracking Button: Runs the video gaze-tracking script, extending the model's gaze-tracking capabilities to video content. This function collects gaze and Eye Aspect Ratio (EAR) data, enabling more dynamic analyses.

Analysis Button: Executes the heatmap script for the image, providing additional insights into gaze fixation areas and blink counts. This script processes the stored gaze data from image tracking, offering a visual representation of where the user focused on the screen most.

Video Gaze Analysis Button: Runs the script for generating heatmaps for video content, creating frame-by-frame heatmaps and metrics based on the collected gaze and EAR data.

Accuracy Test Button: Before running the trajectory-based accuracy script, this button prompts the user to follow a blue ball on the screen, ensuring they understand the

task, as shown in Figure 18b. Once confirmed, the script calculates the accuracy of the gaze tracking.

4.4.2. Visuals and Usability

The interface features a sleek black background, contributing to its modern and professional look. The buttons are styled in pastel green and light walnut hues, offering easy-to-read contrast. Each button includes an icon above its descriptive text, adding a contemporary touch and enhancing the overall look and feel. The buttons have a 3D effect, with hover states providing immediate visual feedback, enhancing its interactivity.

The layout is organized into a 2×3 grid, ensuring the buttons are evenly spaced and easily clickable and improving overall usability. This design not only makes the interface aesthetically pleasing but also ensures it is intuitive and easy to navigate. The combination of its modern design and functional elements allows users to effortlessly manage and execute complex gaze-tracking tasks.

Overall, this interface provides a robust tool for researchers and professionals, streamlining the process of conducting gaze tracking and analysis while maintaining a balance between style and practicality.

5. Conclusions and Future Work

This research successfully developed a cost-effective and accurate gaze-tracking system by leveraging a convolutional neural network (CNN) and a standard camera. Our approach provides a practical alternative to traditional gaze-tracking systems, which typically rely on expensive infrared (IR) cameras and controlled environments. By using a conventional camera, this method significantly broadens the accessibility of gaze-tracking technology, making it feasible for a wider range of applications. The primary objective of this project was to design and implement a CNN-based eye-tracking system capable of delivering precise gaze estimation across various real-world scenarios. To achieve this, the project focused on developing a robust calibration method for collecting eye data, creating a CNN model specifically tailored to gaze estimation and designing a user-friendly interface that integrates gaze tracking, data storage, and analysis functionalities. The findings of this research underscore the effectiveness of the proposed approach, with the CNN model achieving an accuracy of 90.98%. This represents the average performance of nine participants during the test phase, assessed using a trajectory-based evaluation system. The model's performance exceeds that of many existing methods, confirming the viability of this webcam-based approach for real-time gaze tracking across diverse environments. Key performance metrics, including a low mean squared error (MSE), a high R-squared value, and a minimal mean absolute error (MAE), further highlight the model's precision and robustness in predicting gaze points with exceptional reliability. Moreover, the real-time validation and trajectory-based tests revealed that the model maintains high accuracy during dynamic visual interactions, which is crucial for applications such as human–computer interactions, marketing, and behavioral research.

The integration of advanced features like heatmaps, gaze fixation plots, and blink rate analyses, all housed within an intuitive user interface, enhances the system's usability and broadens its range of potential applications. The proposed methodology stands out compared to other gaze-tracking techniques due to its combination of high precision and affordability. These attributes make it especially valuable in scenarios where both cost efficiency and optimal performance are essential. The inclusion of Google's Mediapipe model for facial landmark detection further strengthens the system's reliability, ensuring consistent gaze estimation even under varying environmental conditions. Therefore, this study has successfully addressed the limitations of traditional gaze-tracking systems by

introducing a CNN-based approach that is both user-friendly and efficient. The developed system not only delivers high accuracy but also provides a versatile tool suitable for a wide range of applications, from academic research to commercial use in areas such as marketing and user experience design. The comprehensive evaluation metrics and the intuitive interface enhance the practical value of the system, positioning it as a significant advancement in gaze-tracking technology. Future work could refine the model and interface, adding more features or improving its adaptability to diverse environments and use cases.

Future Work

Looking ahead, there are numerous opportunities for further development. Future iterations could involve designing more advanced CNN architectures to enhance both accuracy and robustness. Allowing users to choose their preferred eye for tracking, expanding the dataset, and implementing novel methods to clean outlier images would reduce manual data sorting and further improve system performance. Additionally, increasing the number and diversity of participants in future studies to include a wider range of age groups, genders, and ethnicities would enhance the model's generalizability and accuracy across different populations. Furthermore, ensuring consistent model performance across different devices is crucial. As this study was conducted on a 15-inch laptop, transitioning to varying screen sizes will require additional data collection and model retraining. A broader dataset covering resolutions from 240p to 1080p will be incorporated, alongside an advanced system to map gaze points accurately across different screen sizes. These advancements would build on the foundation laid in this project and push the boundaries of webcam-based gaze tracking.

Author Contributions: Methodology, V.V. and D.R.F.; Software, V.V.; Validation, V.V.; Investigation, V.V.; Writing—original draft, V.V.; Writing—review and editing, D.R.F.; Supervision, D.R.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The source code and dataset utilized in this research are publicly available and can be accessed at the following link: https://drive.google.com/drive/folders/16RAPKtnFTm7Tmrs9i3h9Z9y7IzBJ6G1p?usp=sharing. This repository contains the necessary files for reproducing the experiments and analyses presented in this study.

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

References

1. Clay, V.; König, P.; König, S.U. Eye tracking in virtual reality. *J. Eye Mov. Res.* **2019**, *12*. [CrossRef]
2. Suzuki, Y.; Shirahada, K.; Kosaka, M.; Maki, A. A new marketing methodology by integrating brain measurement, eye tracking, and questionnaire analysis. In Proceedings of the ICSSSM12, Shanghai, China, 2–4 July 2012.
3. Rotariu, C.; Costin, H.; Bozomitu, R.G.; Petroiu-Andruseac, G.; Ursache, T.I.; Cojocaru, C.D. New assistive technology for communicating with disabled people based on gaze interaction. In Proceedings of the 2019 E-Health and Bioengineering Conference (EHB), Iasi, Romania, 21–23 November 2019.
4. De Silva, S.; Dayarathna, S.; Ariyarathne, G.; Meedeniya, D.; Jayarathna, S.; Michalek, A.M.P.; Jayawardena, G. A Rule-Based System for ADHD Identification using Eye Movement Data. In Proceedings of the 2019 Moratuwa Engineering Research Conference (MERCon), Moratuwa, Sri Lanka, 3–5 July 2019.
5. Outram, B.; Pai, Y.S.; Person, T.; Minamizawa, K.; Kunze, K. Anyorbit. In Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications, Warsaw, Poland, 14–17 July 2018.
6. Chen-Sankey, J.; Elhabashy, M.; Gratale, S.; Geller, J.; Mercincavage, M.; Strasser, A.A.; Delnevo, C.D.; Jeong, M.; Wackowski, O.A. Examining Visual Attention to Tobacco Marketing Materials Among Young Adult Smokers: Protocol for a Remote Webcam-Based Eye-Tracking Experiment. *JMIR Res. Protoc.* **2023**, *12*, e34512. [CrossRef] [PubMed]
7. Zhu, Z.; Ji, Q. Eye Gaze Tracking under Natural Head Movements. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005.

8. Macinnes, J.J.; Iqbal, S.; Pearson, J.; Johnson, E.N. Wearable Eye-tracking for Research: Automated dynamic gaze mapping and accuracy/precision comparisons across devices. *bioRxiv* **2018**. [CrossRef]
9. Wood, E.; Baltrusaitis, T.; Morency, L.-P.; Robinson, P.N.; 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.
10. Krafka, K.; Khosla, A.; Kellnhofer, P.; Kannan, H.; Bhandarkar, S.; Matusik, W.; Torralba, A. Eye Tracking for Everyone. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
11. Deng, H.; Zhu, W. Monocular Free-Head 3D Gaze Tracking with Deep Learning and Geometry Constraints. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
12. Jigang, L.; Lee, S.; Rajan, D. Free-Head Appearance-Based Eye Gaze Estimation on Mobile Devices. In Proceedings of the 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Okinawa, Japan, 11–13 February 2019.
13. Chen, H.-H.; Hwang, B.-J.; Wu, J.-S.; Liu, P.-T. The Effect of Different Deep Network Architectures upon CNN-Based Gaze Tracking. *Algorithms* **2020**, *13*, 127. [CrossRef]
14. Kanade, P.; David, F.; Kanade, S. Convolutional Neural Networks(CNN) based Eye-Gaze Tracking System using Machine Learning Algorithm. *Eur. J. Electr. Eng. Comput. Sci.* **2021**, *5*, 36–40. [CrossRef]
15. Ansari, M.F.; Kasprowski, P.; Obetkal, M. Gaze Tracking Using an Unmodified Web Camera and Convolutional Neural Network. *Appl. Sci.* **2021**, *11*, 9068. [CrossRef]
16. Singh, J.; Modi, N. A robust, real-time camera-based eye gaze tracking system to analyze users' visual attention using deep learning. *Interact. Learn. Environ.* **2022**, *32*, 409–430. [CrossRef]
17. Darapaneni, N.; Prakash, M.D.; Sau, B.; Madineni, M.; Jangwan, R.; Paduri, A.R.; Jairajan, K.P.; Belsare, M.; Madhavankutty, P. Eye Tracking Analysis Using Convolutional Neural Network. In Proceedings of the 2022 Interdisciplinary Research in Technology and Management (IRTM), Kolkata, India, 24–26 February 2022.
18. Donuk, K.; Ari, A.; Hanbay, D. A CNN based real-time eye tracker for web mining applications. *Multimed. Tools Appl.* **2022**, *81*, 39103–39120. [CrossRef]
19. Zhang, X.; Fan, C.-T.; Yuan, S.M.; Peng, Z.-Y. An Advertisement Video Analysis System Based on Eye-Tracking. In Proceedings of the 2015 IEEE International Conference on Smart City, Chengdu, China, 19–21 December 2015.
20. Lee, I.; Cha, J.; Seo, J.; Kwon, O. User interest visualizing and analysing system using eye gaze. In Proceedings of the 2015 17th International Conference on Advanced Communication Technology (ICACT), PyeongChang, Republic of Korea, 1–3 July 2015.
21. Okano, M.; Asakawa, M. Eye tracking analysis of consumer's attention to the product message of web advertisements and TV commercials. In Proceedings of the 2017 5th International Conference on Cyber and IT Service Management (CITSM), Denpasar, Indonesia, 8–10 August 2017.
22. Zhang, X.; Yuan, S.-M. An Eye Tracking Analysis for Video Advertising: Relationship Between Advertisement Elements and Effectiveness. *IEEE Access* **2018**, *6*, 10699–10707. [CrossRef]
23. Muñoz Leiva, F.; Rodríguez López, M.E.; García, Martí, B. Discovering prominent themes of the application of eye tracking technology in marketing research. *Cuad. Gestión* **2022**, *22*, 97–113. [CrossRef]
24. Modi, N.; Singh, J. Real-time camera-based eye gaze tracking using convolutional neural network: A case study on social media website. *Virtual Real.* **2022**, *26*, 1489–1506. [CrossRef]
25. Onwuegbusi, T.; Roberts, A.; Sharman, S.; Hogue, T. An Eye Tracking Investigation of Young People's Gaze Behaviour to Gambling and Non-Gambling Moving Adverts. *Eur. Addict. Res.* **2023**, *29*, 109–118. [CrossRef]
26. Xie, W.; Lee, M.H.; Chen, M.; Han, Z. Understanding Consumers' Visual Attention in Mobile Advertisements: An Ambulatory Eye-Tracking Study with Machine Learning Techniques. *J. Advert.* **2023**, *53*, 397–415. [CrossRef]
27. Tsubouchi, K.; Taoka, K.; Ikematsu, K.; Yamanaka, S.; Narumi, K.; Kawahara, Y. Eye-tracking AD: Cutting-Edge Web Advertising on Smartphone Aligned with User's Gaze. In Proceedings of the 2024 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Biarritz, France, 11–15 March 2024.
28. Soukupova, T.; Cech, J. Eye blink detection using facial landmarks. In Proceedings of the 21st Computer Vision Winter Workshop, Rimske Toplice, Slovenia, 3–5 February 2016; Volume 2.
29. Atchison, D.A. *Optics of the Human Eye*; CRC Press: Boca Raton, FL, USA, 2023.
30. Encyclopedia Britannica. Human Eye—Extraocular Muscles. Available online: https://www.britannica.com/science/human-eye/Extraocular-muscles#/media/1/1688997/3421 (accessed on 1 February 2025).
31. Wu, Y.-L.; Yeh, C.-T.; Hung, W.-C.; Tang, C.-Y. Gaze direction estimation using support vector machine with active appearance model. *Multimed. Tools Appl.* **2012**, *70*, 2037–2062. [CrossRef]
32. Meng, C.; Zhao, X. Webcam-Based Eye Movement Analysis Using CNN. *IEEE Access* **2017**, *5*, 19581–19587. [CrossRef]

33. Sattar, H.; Fritz, M.; Bulling, A. Deep Gaze Pooling: Inferring and Visually Decoding Search Intents From Human Gaze Fixations. *Neurocomputing* **2020**, *387*, 369–382. [CrossRef]
34. Ou, W.-L.; Kuo, T.-L.; Chang, C.-C.; Fan, C.-P. Deep-Learning-Based Pupil Center Detection and Tracking Technology for Visible-Light Wearable Gaze Tracking Devices. *Appl. Sci.* **2021**, *11*, 851. [CrossRef]