A project report on

GAZE TRACKING FOR OPTIMAL AD RECOMMENDATION IN E-COMMERCE PLATFORM

Submitted in partial fulfillment for the award of the degree of

Bachelor of Technology in Computer Science and Engineering with Specialization in Cyber Physical System

by

PALASH YASH (21BPS1101)

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)
CHENNAI

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

April, 2025

GAZE TRACKING FOR OPTIMAL AD RECOMMENDATION IN E-COMMERCE PLATFORM

Submitted in partial fulfillment for the award of the degree of

Bachelor of Technology in Computer Science and Engineering with Specialization in Cyber Physical System

by

PALASH YASH (21BPS1101)

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)
CHENNAI

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

April, 2025

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CHENNAI

DECLARATION

I hereby declare that the thesis entitled "Gaze Tracking for Optimal Ad Recommendation in E-Commerce Platform" submitted by PALASH YASH (21BPS1101), for the award of the degree of Bachelor of Technology in Computer Science and Engineering, Vellore Institute of Technology, Chennai is a record of Bonafide work carried out by me under the supervision of Dr. Rajakumar Arul

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Chennai

Date: Signature of the Candidate

School of Computer Science and Engineering

CERTIFICATE

This is to certify that the report entitled **"Gaze Tracking for optimal Ad recommendation in e-commerce platform"** is prepared and submitted by **Palash Yash (21BPS1101)** to Vellore Institute of Technology, Chennai, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology in Computer Science and Engineering with Specialization in Cyber Physical Systems** is a bonafide record carried out under my guidance. The project fulfills the requirements as per the regulations of this University and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma and the same is certified.

Signature of the Guide:

Name: Dr. Rajakumar Arul

Date:

Signature of the Examiner	Signature of the Examiner
Name:	Name:
Date:	Date:

Approved by the Head of Department,
Computer Science and Engineering with Specialization in Cyber Physical Systems

Name: Dr. Renuka Devi S

Date:

(Seal of SCOPE)

ABSTRACT

The project seeks to improve the performance of online advertising through the utilization of eye gaze tracking technology. Conventional methods of advertising fail to consider the way audiences respond to visual information on a screen, resulting in inefficient ad placement. The project fills this gap by employing real-time eye gaze tracking to identify which regions on a screen capture the most attention, allowing advertisers to maximize viewer engagement through optimal ad positioning.

Employing computer vision methods, including MediaPipe, OpenCV, and dlib, the project identifies and follows the eyes and pupils of the observer in real-time through an ordinary webcam. The system collects and converts the direction of the gaze based on the positioning of the eye and pupil onto screen coordinates, splitting the screen into smaller zones to determine where observer is more likely to look. To better visualize the aggregated gaze data, heatmaps are created, giving a clearer visualization of attention distribution on the screen. Heatmaps helps in displaying the interest point on the screen, and advertisers can place ads strategically in these areas.

This project also compiles the gaze tracking and the visualization of the heatmap into a wrapper application for easier distribution and usability. The technology is supposed to give advertisers useful information regarding viewer behavior, improving ad visibility and the efficiency of online marketing campaigns. Through the analysis of user eye movements, the system presents a new method of understanding ad consumption patterns and ad engagement optimization. This initiative opens the door to future advances in applying eye-tracking technology to advertising and marketing, creating new opportunities for enhancing ad performance.

ACKNOWLEDGEMENT

It is my pleasure to express with deep sense of gratitude to Dr. Rajakumar Arul, Assistant Professor, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, for his constant guidance, continual encouragement, understanding; more than all, he taught me patience in my endeavor. My association with him is not confined to academics only, but it is a great opportunity on my part of work with an intellectual and expert in the field of Data processing and Python Visualization Library.

It is with gratitude that I would like to extend my thanks to the visionary leader Dr. G. Viswanathan our Honorable Chancellor, Mr. Sankar Viswanathan, Dr. Sekar Viswanathan, Dr. G V Selvam Vice Presidents, Dr. Sandhya Pentareddy, Executive Director, Ms. Kadhambari S. Viswanathan, Assistant Vice-President, Dr. V. S. Kanchana Bhaaskaran Vice-Chancellor, Dr. T. Thyagarajan Pro-Vice Chancellor, VIT Chennai and Dr. P. K. Manoharan, Additional Registrar for providing an exceptional working environment and inspiring all of us during the tenure of the course.

Special mention to Dr. Ganesan R, Dean, Dr. Parvathi R, Associate Dean Academics, Dr. Geetha S, Associate Dean Research, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai for spending their valuable time and efforts in sharing their knowledge and for helping us in every aspect.

In jubilant state, I express ingeniously my whole-hearted thanks to Dr. Renuka Devi S, Head of the Department, B.Tech. Computer Science and Engineering with Specialization in Cyber Physical Systems and the Project Coordinators for their valuable support and encouragement to take up and complete the thesis.

My sincere thanks to all the faculties and staffs at Vellore Institute of Technology, Chennai who helped me acquire the requisite knowledge. I would like to thank my parents for their support. It is indeed a pleasure to thank my friends who encouraged me to take up and complete this task.

Place: Chennai

Date: **PALASH YASH**

CONTENTS

1	CONTENTS	(VII)
2	LIST OF FIGURES	(VIII)
3	LIST OF TABLES	(IX)
4	CHAPTER 1 – INTRODUCTION	1
5	CHAPTER 2 – LITERATURE REVIEW	5
6	CHAPTER 3 – METHODOLOGY	11
7	CHAPTER 4 – VISION TRACKING	17
8	CHAPTER 5 – LINE OF SIGHT TRACKING	20
9	CHAPTER 6 – SCREEN VISULIZATION	21
10	CHAPTER 8 – FUTURE WORK	23
11	CHAPTER 9 – CONCLUSION	25
12	APPENDICES	27
13	REFERENCES	31

LIST OF FIGURES

1	FACE DETECTION USING OPENCV	17
2	HAAR FEATURES USED FOR FACE SEGMENTATION	18
3	HAAR INTEGRAL SLIDING WINDOWS GENERATION	18
4	ISOLATED REGION OF INTEREST	19
5	SAMPLE HEATMAP GENERATED	22
6	SAMPLE SCATTER GENERATED	22
7	CODE – GENERATING DETECTOR AND PREDICTOR	27
8	CODE – DETECTING MOUSE POSITIONING	27
9	CODE – DETECTING EYE BLINKING	28
10	CODE – CALCULATING LINE OF SIGHT	28
11	CODE – CALCULATING VIDEO	29
12	CODE – INTERPRETING HEATMAP DATA	30
13	CODE – DISPLAYING GAZE HEATMAP	30
14	CODE – INTERPRETING SCATTER PLOT DATA	30
15	CODE – DISPLAYING SCATTER PLOT	30

LIST OF TABLES

1	Gaze Tracking Data with Timestamp	20
2	Complete Data with Timestamp	21

Chapter 1
INTRODUCTION

1.1 INTRODUCTION

The "Gaze Tracking for Advertisement Placement Optimization" project aims to apply eye-tracking technology to achieve the maximum efficiency of digital advertising. By understanding how users allocate their attention on a screen, the system will try to optimize where ads are placed in those parts of the screen where naturally most viewer attention is.Advertisers are always seeking means of making their ads more effective and interesting, and gaze tracking offers a new solution by providing users with a clear insight into how to improve ads. Eye-tracking can assist businesses to identify the best place to position advertisements, reach more users, and maximize digital advertising.The initiative involves live tracking of eye motion, translating data on what people are looking at into screen points, and visualizing that data as heatmaps of where users focus their eyes. Data-driven approach gives actionable insights that can result in more strategic ad placement and ultimately increase the efficiency of online campaigns. The aim is to know more about user interaction with ads and to present a new method for advertisers to maximize their approach based on user interaction with graphical content.

1.2. OVERVIEW OF THE PROJECT

1.2.1 VISION TRACKING

The project employs computer vision methods, mostly OpenCV, to track and detect the eyes of the viewer in real-time. OpenCV is a popular library for real-time computer vision applications, and in this project, it is employed to grab live video streams from a webcam. Detecting facial features and concentrating particularly on the eyes, the system continuously monitors eye movement. Eye tracking is done by finding the pupils, which are utilized in order to comprehend the point of focus of the viewer. This is based on eye shape detection and movement, with the key landmarks of the face being detected by Haar Cascade Classifiers. These classifiers learn to spot the particular characteristics such as the eyes and are used on the video stream to constantly follow the user's line of sight with accuracy.

1.2.2 LINE OF SIGHT TRACKING

When eyes are identified and their locations logged, the final step is decoding where the audience is gazing on the monitor. By mapping the eye data into precise coordinates on the monitor, the system can directly allocate the gaze on to different regions of the screen. The information about the gaze is derived through the calculation of the pupil position within the window of the capture video. Next, by utilizing a process of calibration, the data is then converted to corresponding screen coordinates. This process is important since it enables the system to specifically figure out where the user's focus is within an area of the screen that displays a video, website, or any other information. It's this line-of-sight tracking that enables the advertisers to figure out what portions of their adverts are most prominent.

1.2.3 SCREEN VISUALIZATION

After the gaze data is recorded and translated into screen coordinates, the next thing to do is to visualize the data for analysis. The gaze points are accumulated over time and translated into a heatmap that shows regions of high attention. Heatmaps are a powerful method of visualizing spatial data since they employ color gradients to show different levels of attention, with warmer colors representing more attention. This heatmap is superimposed over a screenshot of the content, providing an immediate sense of where the viewer's focus was concentrated during the interaction with the ad. This visualization offers advertisers a concrete representation of user behavior, enabling them to determine which elements of an ad command the most attention and which elements are ignored. It is an effective tool for optimizing ad placement strategies.

1.2.4 WRAPPER APPLICATION

To make it easy for this system to function, the project includes a wrapper application developed with Streamlit. Streamlit is an open-source Python library used to make it easy to build interactive web applications. With Streamlit, all the system, including gaze tracking, data mapping, and the generation of heatmaps, can be combined into one easy-to-use interface. This wrapper application enables the users to engage with the system through a basic web interface, where the users can upload material, execute gaze tracking, and see the results. Streamlit enables easier distribution and deployment of the project, so non-technical users can enjoy the eye-tracking insights without even having to deal with intricate code. This increases the project's usability and makes it more functional for marketers and advertisers who wish to implement the system in order to improve their ad tactics.

1.3. CHALLENGES FACED

1.3.1 SELECTING SUITABLE LIBRARY

One of the most significant issues in this project was choosing which library to employ to do eye tracking. There are many libraries available, and each has its advantages and disadvantages. GazeTracking with OpenCV, MediaPipe, PyGaze, and DLib are all libraries that provide various ways of eye-tracking but with their own set of trade-offs in ease of use, reliability, and computational efficiency. Following rigorous testing, OpenCV with CV2 was selected due to the flexibility it provides, its stability, and the broad available community support. With this choice, the project was able to concentrate on the minimum requirements without having to call upon other, possibly unstable or unsupported libraries.

1.3.2 THRESHOLDING OF GAZE_DATA

Another challenge was controlling the thresholding of the gaze data. The system must gather sufficient information to know where the user was looking, but the data also had to be filtered so as not to record every small eye movement, as this would generate noise and computational overload. Determining the correct threshold was essential in order to make sure that the system only monitored substantial gaze transitions without impacting performance or flooding the entire application with too much data. The system was then able to balance computational efficiency with data accuracy by adjusting the threshold to only process and record useful gaze movements.

1.3.3 COORDINATE MAPPING

A key technical hurdle was mapping the gaze data to screen coordinates correctly. The raw eye position data must be converted to particular regions of the screen. This involved understanding the relationship between the camera view and the screen view of the user in minute detail. Other screen sizes and resolutions also introduced complexity into the mapping. The group employed a calibration procedure to account for these variables and to make sure the gaze data matched well with the screen coordinates. This calibration ensured that the system could consistently calculate where on the screen the viewer was looking.

1.4. SCOPE OF PROJECT

1.4.1 EFFECT OF GAZE PATTERNS ON AD' EFFECTIVENESS

The project seeks to investigate the impact of user gaze patterns on the effectiveness of digital video advertisements. Through monitoring and analyzing where and how long the viewers look, the system facilitates the identification of what parts of an ad engage most. This can result in better ad placement, as ads will be placed in areas that will be strategically positioned for such important features like logos, call-to-action buttons, or product images. Such information can inform advertisers to tailor their content towards higher engagement and performance.

1.4.2 FINDING EFFECTIVE AD PLACEMENT

Another primary objective of the project is to determine which particular ad components—e.g., text, images, or branding—engage and hold the attention of viewers. Through the analysis of the heatmaps derived from the gaze tracking data, the system indicates which elements of an ad receive the most attention. For example, some viewers might be attracted to the visual aspects of an ad, while others might pay more attention to text information or branding. This analysis aids advertisers in knowing what is best when it comes to viewer response and enables them to adjust their ads accordingly.

1.4.3 CORRELATION OF GAZE PATTERNS AND USER OUTCOMES

The project further aims to investigate the relationship between user outcomes like recall accuracy, ad attitudes, and purchase intent, and gaze patterns. By correlating the gaze data to user responses, the system is able to obtain significant insights regarding the impact of visual attention on consumer behavior. This information can be used to optimize ad strategies so that ads are not just interesting but also effective at getting users to perform desired actions, like remembering the ad, developing positive attitudes, or purchasing.

1.4.4 ENHANCING ADVERTISEMENT PERSONALIZATION

The project further aims to investigate the relationship between user outcomes like recall accuracy, ad attitudes, and purchase intent, and gaze patterns. By correlating the gaze data to user responses, the system is able to obtain significant insights regarding the impact of visual attention on consumer behavior. This information can be used to optimize ad strategies so that ads are not just interesting but also effective at getting users to perform desired actions, like remembering the ad, developing positive attitudes, or purchasing.

Chapter 2
BACKGROUND

2.1 INTRODUCTION

The application of eye gaze tracking has attracted much focus as a potential tool for promoting the efficiency of digital marketing activities. Eye gaze tracking technology takes the form of recording and evaluating the movement of an individual's eyes to decide on the visual areas of an image display, like a television advertisement, on which they most focus their attention. By learning where viewers' eyes are concentrated, advertisers are able to determine what viewers' information processing strategies are, which parts of the advertisement appeal to them most, and how these aspects affect the behavior of the viewers. Such information can subsequently be utilized in optimizing advertisement design, location, and content to drive maximum interaction and conversion.

Over the last few years, computer vision and machine learning progress has provided easier and more precise access to eye gaze tracking, enabling real-time tracking with minimal equipment. OpenCV, MediaPipe, and dlib libraries have made it easier to implement more sophisticated systems for detecting gazes. The literature has shown that patterns of viewing play a massive role in advert effectiveness, with some visual elements, including product picture, text, and logos, receiving more attention and affecting recall, attitude, and purchase intention.

The increasing popularity of targeted advertising has all the more highlighted the need to study the pattern of gaze. By personalizing ads for individuals according to where they are looking, advertisers can better design more compelling and effective campaigns. The current project extends such knowledge, leveraging the use of gaze tracking to maximize the placement of advertisements and derive actionable insights into the behavior of viewers, therefore informing the development of digital advertising strategies.

2.2 LITERATURE REVIEW

‘Learning to Find Eye Region Landmarks for Remote Gaze Estimation in Unconstrained Settings’: Seonwook Park, Xucong Zhang, Andreas Bulling, Otmar Hilliges

Recent developments in gaze estimation techniques have shifted from traditional feature-based and model-based to appearance-based techniques, particularly in unconstrained environments. Traditional techniques, although efficient under controlled conditions using specialized cameras, are not efficient in unconstrained environments where illumination changes and visual artifacts are common. This has resulted in the use of appearance-based techniques that are efficient despite these challenges, particularly in person-independent gaze estimation tasks. Nevertheless, such techniques need vast quantities of labeled training data, are computationally intensive, and lack interpretability.

To meet these challenges, Park et al. (2018) presented a new learning-based solution to eye region landmark localization. Using this technique, classical feature-based and model-based methods can achieve performance superior to state-of-the-art appearance-based methods, even when applied to real-world imagery. The innovation of note is the training of a deep convolutional neural network (CNN) using synthetic data to allow for stable landmark detection without the use of real images during training. Not only does this enhance iris localization and eye shape registration, but it also improves the accuracy of gaze estimation in both person-specific and cross-dataset testing. By integrating learned eye landmarks with traditional gaze estimation models, the approach fills the gap between old and new methods, providing a more intuitive and computationally lighter solution for unconstrained gaze tracking.

Head pose estimation in the wild using Convolutional Neural Networks and adaptive gradient methods: Massimiliano Patacchiola, Angelo Cangelosi

Head pose estimation has become a highly researched area in the past few years with high potential for applications in different fields, including human-robot interaction, augmented reality, and driver assistance. Classic approaches to head pose estimation have worked well under laboratory conditions but usually fail to cope with the challenges of real-world situations. These constraints have motivated the investigation of deep learning methods, in particular Convolutional Neural Networks (CNNs), to enhance robustness and accuracy in unconstrained settings. The main challenge in such settings involves head pose variations, facial expressions, lighting, and occlusions such as sunglasses or masks. Recent developments have shown that CNNs, in combination with adaptive gradient techniques and dropout strategies, provide encouraging results in head pose estimation. These techniques improve the generalization capability of CNNs, allowing them to work well in "wild" conditions, where face images have high variability in appearance and environmental conditions. The combination of these strategies has resulted in state-of-the-art performance in real-world head pose estimation problems. This piece of work makes an important contribution to the domain by critically analyzing several CNN models and methods with emphasis on how combining CNNs with new deep learning techniques enhances effectiveness. The creation of Deepgaze, a Python package based on TensorFlow, also comes as a real-world utility for real-time head pose estimation in GPUs and on mobile devices and is a useful contribution to academic and commercial environments.

Application of Eye Tracking Technology in Medicine: A Bibliometric Analysis: Gianpalolo Zammarchi, Claudio Conversano

Eye tracking has emerged as an important tool to investigate medical conditions, especially in neurological and psychiatric disorders, because it can supply objective and accurate measurements of the eye movement. Eye tracking is a non-invasive method, which has been used to comprehend cognitive functions in patients with disorders such as autism spectrum disorders (ASD), schizophrenia, bipolar disorder, and other neurodegenerative diseases like Alzheimer's and Parkinson's disease. A bibliometric examination of 2456 articles, carried out using VOSviewer and the Bibliometrix R package, indicates psychiatry, neuroscience, and developmental psychology as the most studied areas in eye tracking. The most prolific scientific production comes from the USA, UK, and Germany, with a significant trend of increased publications over the years, particularly between 2011 and 2017. The research area has witnessed a remarkable 16.52% compound annual growth rate, indicating the widening application of eye tracking in the diagnosis of neurodevelopmental disorders. Eye tracking technology is commonly combined with other neuroimaging methods, including EEG and MRI, to give a better understanding of the neurobiological processes behind

ASD. The combination helps researchers gain more insight into the cognitive and behavioral difficulties of people with ASD. In summary, this review highlights the growing importance of eye tracking in medical research, especially in the area of complex cognitive and behavioral disorders, and its potential for providing useful information to enhance diagnosis and treatment methods.

Assessing Consumer Attention and Arousal Using Eye-Tracking Technology in Virtual Retail Environment: Nayeon Kim, Hyunsoo Lee

The visual experience of consumers in retail spaces is a key factor in determining their emotional and cognitive reactions, which in turn affect their behavior and decision-making. Recent research has investigated how certain design aspects of retail stores influence consumer attention and emotional arousal, with results indicating that the layout, spatial configurations, and store attributes have significant effects on these reactions (Kim, 2020; Kim & Lee, 2020). Traditional methods like self-reporting fall short in capturing the true psychological reaction of consumers and result in the use of more advanced techniques like eye-tracking and biosensors in measuring real-time attention and emotional responses (Ariely & Berns, 2010; Bettiga et al., 2020).Eye-tracking, specifically, offers useful information about consumer gaze patterns, providing an accurate understanding of what attracts attention and how emotional arousal is elicited (Orquin & Wedel, 2020).

This research builds on previous work by combining eye-tracking technology with virtual reality to examine both visual attention and emotional arousal within an immersive shopping environment. By exploring differences between genders in how consumers react to retail environments, the research presents a nuance to past explanations, which detail different patterns with which men and women engage the retail environment. The findings extend the field towards evidence-based merchandising decisions, allowing stores to enhance how they present visual elements and where they locate in-store amenities and products in a way that improves attracting and involving target shoppers.

Accelerating eye movement research via accurate and affordable smartphone eye tracking: Nachiappan Valliappan, Na Dai, Ethan Steinberg, Junfeng He, Kantwon Rogers, Venky Ramachandran, Pingmei Xu, Mina Shojaeizadeh, Li Guo, Kai Kohlhoff & Vidhya Navalpakkam

Eye tracking has been an invaluable instrument in vision science, cognitive psychology, and user studies for many decades. It offers great insight into the way we process and attend to visual information, with applications ranging from visual search to reading to scene perception. Conventional eye tracking systems are based on costly, purpose-built hardware that restricts scalability and availability, particularly to mobile devices. In spite of the presence of smartphones in everyday life, there has been minimal understanding of eye movement behavior on these gadgets because no inexpensive, accurate tracking methods existed.

With recent advances in machine learning (ML), this has now started to change with the ability to perform smartphone-based eye tracking without extra hardware. Such approaches utilize smartphones' front cameras, providing a cost-effective and scalable alternative to existing systems.Experiments revealed that ML-based smartphone tracking is as accurate as state-of-the-art mobile eye trackers, which are much more costly.Most importantly, these approaches have already been able to reproduce outcomes of existing results from past eye movement studies, such as oculomotor tasks and saliency analysis on viewing natural images. They also promise applications involving reading comprehension problem identification, incorporating eye tracking into larger healthcare, accessibility, and vision research contexts. This feat would have the potential to democratize eye tracking research by making large studies with thousands of participants possible, leading to innovations and fresh insights.

Utilising eye-tracking data in retailing field research: A practical guide: Jens Nordfält , Carl-Philip Ahlbom

In recent years, shopper marketing has seen renewed momentum, fueled by shifts such as the revival of shoppers coming back to brick-and-mortar shopping following the pandemic and increased retail digitization (Weinswig, 2023; Grewal et al., 2023). Such shifts in retail behavior have fueled greater interest in sensory marketing, which examines the ways in which sensory stimuli within shops influence shopper behavior.Conventional approaches to studying consumer behavior, including surveys and hand-collected observational studies, have yielded useful insights but also suffer from limitations, especially in obtaining real-time, subtle data. Surveys rely on self-reported subjective information, and hand-observed observations are without minute attentional details and decision-making behaviors (Granbois, 1968; Hoyer, 1984).

Technologies have improved tracking practices, with CCTV, RFID chips, and headsets offering greater details of shopper movement (Gil et al., 2009; Hui et al., 2009; Sorensen, 2008).Even then, these practices are not able to track visual attention, an important driver of consumer choices. Eye-tracking technology has emerged as a solution, providing precise details on where the consumer focuses attention, thus filling the loopholes of traditional tracking methods (Grewal et al., 2020). Eye-tracking provides a precise indication of behavior in the store that can be used to guide retailing strategies and improve the shopping experience.

Using eye-tracking technology in Neuromarketing: Consuela-Mădălina Gheorghe *, Victor Lorin Purcărea *, Iuliana-Raluca Gheorghe *

Eye tracking also became a cornerstone tool of neuromarketing, providing valuable information regarding consumer behavior through the examination of gaze and response to visual stimuli. Scopus, PubMed, Elsevier, Springer, and Science Direct data bases' studies have proven eye tracking to be a crucial tool in monitoring consumers' unconscious decision-making process.Technology is used in neuromarketing in order to get attention, pointing out where consumers direct attention, therefore identifying their unconscious purchase intention and interest in medical products and services.

Eye tracking combined with more traditional methods of marketing research yields stronger signals about the way consumers are interacting with advertising, packaging, and web content. By applying the metric of eye movement, researchers are able to see what visual cues command attention and drive engagement and discover valuable information that can be used to shape marketing practices. Through this hybrid model, marketers are given the ability to optimize strategy and tailor advertising to conform to natural patterns of attention, hence making marketing more efficient.

While it is potentially useful, eye tracking's use in neuromarketing also raises concerns about privacy and manipulation. Proper informed consent is required to ensure proper ethical procedures are followed. However, if used properly, eye tracking can significantly minimize the risk of throwing marketing budgets down the drain for inefficient marketing campaigns, resulting in more effective targeted marketing.

Eye tracking: empirical foundations for a minimal reporting guideline: Kenneth Holmqvist, Saga Lee Örbom, Ignace T. C. Hooge, Diederick C. Niehorster, Robert G. Alexander, Richard Andersson

Eye tracking is a very powerful method to investigate a lot of aspects of human behavior, e.g., perception, attention, memory, consumer behavior. Eye tracking can be used in lots of various domains, from neuroscience and human-computer interaction to ophthalmology and animal behavior (Duchowski, 2002; Kowler, 2011). As technology continues to evolve, eye tracking will continue to be more common, potentially being integrated into consumer devices such as laptops and augmented reality headsets (Chuang et al., 2019; Clay et al., 2019). Despite being widely used, there are limitations to standardizing eye-tracking methods, especially experimental design and reporting data analysis (Hessels & Hooge, 2019; Holmqvist et al., 2012).

The recent literature suggests a lack of reporting consistency in recording data quality of eye-tracking and experimental conditions. Present reporting standards vary significantly between studies and fields, and are not routinely followed (Carter & Luke, 2019; McConkie, 1981). This lack of consistency may avoid replicability and validity of study findings, and thus lead to the replication crisis in psychological science (Open Science Collaboration, 2015). The development of reporting guidelines for evidence-based studies is likely to enhance eye-tracking study design, validation, and reproducibility, and hence clarify communication across disciplines. The guidelines need to be adaptable for use in different fields of research, considering elements of safety, ethics, and study requirements.

Eye-tracking in Marketing Research: Sylwester Andrzej Białowąs, Adrianna Szyszka

Eye-tracking has become a critical methodology in marketing research, delivering high-quality insights into consumer behavior by measuring eye movements and attention. Eye-tracking allows researchers to examine cognitive processes, such as perception and visual attention, that play a significant role in consumer interaction with marketing stimuli.In contrast to conventional methods, eye-tracking provides information about our unconscious responses, responding to where consumers are looking and for how long they are viewing certain features in advertisements or products (Białowąs & Szyszka, 2019). The method works by monitoring eye movements like fixations and saccades, both providing different information about visual attention.

Fixations refer to areas where the eyes fixate on a single point, whereas saccades are utilized to describe rapid eye movement between fixations.Visual attention areas can be ascertained by marketers through observing movements, which will allow them to target advertisements (Garczarek-Bąk, 2017).The information that is gathered can be displayed in various forms, like heat maps or scanpaths, graphically showing the most and least scanned areas (Schall & Bergstrom, 2014). Current research has indicated that combining eye-tracking with other neuromarketing methods, such as EEG or fMRI, can improve insight into consumer decision-making (Białowąs & Szyszka, 2019). Eye-tracking therefore has immense potential to increase marketing effectiveness by learning more about visual attention and consumer engagement.

A study of Eye Tracking Technology and its applications: Pramodini A. Punde, Dr. Mukti E. Jadhav, Dr Ramesh R. Manaza

Eye tracking (ET) technology observes and records eye movement, offering information about where the gaze settles, pupil reaction, and how visual attention is allocated with stimuli. While the idea of eye tracking is straightforward, processing and interpretation is not always easy.ET technology is remote or mobile, monitoring where and how individuals are looking. This data is extremely useful to a wide variety of applications, namely psychology, human-computer interaction, advertising, and medicine. It enables researchers to explore cognitive and affective responses, such as how the subjects react to different stimuli or orientation of space.

In scientific research and experiments, eye tracking is used to investigate cognitive processes, attention, memory, and learning, particularly in psychology and neuroscience. Experiments typically focus on eye movement patterns, attention abilities, and how individuals process visual information. In consumer research as well, ET has gained prominence to allow firms to know where customers are looking, evaluate product shapes, and improve advertisements by finding out where customers are gazing.Eye tracking is also used in neuroscience to study how individuals process faces, emotions, and environments, which is useful information for human cognition. With advancing technology regarding eyes tracking, its usage also changes, and therefore it remains an essential part of research as well as business.

2.3 CONCLUSION AND SCOPE OF WORK

Eye-tracking technology incorporation has vast digital marketing optimization potential via advanced understanding of consumer behavior and attention. Ad copy can be optimized and made effective via areas of focus accessibility based on marketers' analysis of gaze patterns. The work cited in the study mentions some advances in gaze tracking, such as the transition from traditional feature-based methods to more stable appearance-based methods to support accurate estimation of gaze in unconstrained settingsIn addition, the utilization of gaze tracking in retail as well as neuromarketing indicates its use in various domains for studying consumer behavior and choice-making.

The scope of the present work broadens these findings by executing real-time gaze tracking using accessible software like MediaPipe and OpenCV, designed to produce heatmaps to analyze user attention on exposure to ads. The system not only enhances ad placement methods but also delivers valuable insights into consumer engagement to aid in more personalized and efficient marketing campaigns.Subsequent research can delve into further integration with machine learning technology, in-depth examination of individual demographic patterns, and cross-device use in an attempt to gain a broader and more integrated view of consumer patterns in online spaces. The increased access and scalability of eye tracking offer a dizzying array of opportunities for the optimization of marketing tactics and user interfaces in the near future.

Chapter 3
METHODOLOGY

3.1 INTRODUCTION

With the current age of digital technology, knowing the behavior of individuals has become essential to improve online ads and their effectiveness. As online advertising becomes more advanced, marketers are now seeking means to engage users more effectively and to optimize ad positions. The aim of this project is to use eye gaze tracking technology to research where users direct their attention on digital screens.By capturing and processing real-time webcam footage, the system monitors the user's eye movement to produce gaze data. This information is then translated into screen coordinates where the system can recognize particular points of interest and target. By combining powerful libraries like OpenCV for image processing, MediaPipe for robust gaze tracking, and autopy to simulate the mouse, the system is able to achieve precise eye tracking, even in real-time use.

As a result, the system creates visual heatmaps that expose what areas of the screen get most viewed by the viewer. These heatmaps are helpful to advertisers because they provide actionable insights on the effectiveness of ad placement and viewer interaction. Advertisers can utilize these insights to optimize where advertisements appear on a webpage, video, or digital site, thus achieving more effective and engaging advertising campaigns.

Moreover, to further extend the accessibility of this technology, the project is encapsulated in a user interface developed using Streamlit. This makes the system deployable and accessible with lower technical knowledge, thus extending its usability to various types of users such as advertisers and marketers. By painting a sharp picture of attention streams, this strategy not only optimizes ad placements but also unlocks new ways to learn about and shape consumer behavior in the digital era.’

3.2 VISION TRACKING

3.2.1. CAPTURING THE WEBCAM FEED

The initial action in the methodology is to capture a live video stream from the webcam of the user. The fact that it can record video using OpenCV is leveraged so that it can access the webcam's stream, and it is transmitted frame by frame.The feed is the input of preference for the whole system since it allows room for the software to recognize faces and eyes in real-time. OpenCV processing real-time video feeds guarantees the responsiveness of the system to facilitate continuous gaze tracking with minimal lag. The system operates efficiently by processing these frames sequentially, one after the other, for face and eyes detection.

3.2.2. CONVERTING IMAGE TO MATCHING TYPE LIKE GRAY SCALING AND FACE AND EYES DETECTION

After capturing the webcam feed, the image is converted to grayscale. This is achieved in order to enhance processing speed and computational efficiency. Grayscale images simplify the detection process since color information is unnecessary in detecting notable facial features. Pre-trained OpenCV Haar Cascade Classifiers are then applied in detecting faces and eyes in the grayscale image. These classifiers do this by scanning patterns in the image and identifying regions that resemble human faces and eyes. After the face and eyes are recognized, the system isolates the region of interest (ROI) to facilitate better tracking of eye movement in the area.

3.3. LINE OF SIGHT TRACKING

3.3.1. EXTRACTING PUPIL POSITIONING FROM THE FACE

After detecting eyes, the subsequent key action is to extract the position of the pupil. This is done with the Hough Circle Transform method, which is a tool in OpenCV. Hough Circle Transform finds circles in images and is most suited for finding the iris and pupil because they are essentially circular shapes. The pupil center is then determined by finding the center of the identified circle, and it is set as the target point of the gaze tracking system. This enables the software to monitor the eye position and, by proxy, the direction of the user's gaze.

3.3.2. CALCULATING GAZE DIRECTION AND STORING THIS DATA

Once the pupil has been found, the system determines the direction of the gaze by examining the position of the pupil in relation to the eye. Through ongoing monitoring and updating of the pupil position in real-time, the software is able to monitor the direction of the user's gaze on the screen. The information is held as coordinates for the position of the user's attention at any instant. This real-time accumulation of gaze information means the system can build an ongoing record of where the user is directing their attention, which is vital for providing heatmaps for expressing attention distribution on the screen.

3.4. SCREEN VISUALIZATION

3.4.1. LOADING THE DATA AND CONVERTING INTO COORDINATES OF THE SCREEN

After collecting the gaze data, the next step is to project this information onto screen coordinates. This process involves translating the position of the pupil within the video feed to corresponding locations on the screen. The conversion of the gaze data into screen coordinates is necessary for determining the location that the user is looking at.The system also takes into account differences in webcam field of view and screen resolution while accurately mapping.This helps ensure that the correct gaze data corresponds to areas on the screen, thus enabling the generation of a helpful heatmap visualization.

3.4.2. GENERATING HEATMAP FROM THE GENERATED DATA

Having transformed the gaze data into screen coordinates, one now proceeds to construct a heatmap.The heatmap visually presents the degree of user attention across the screen. Areas where the user spends the most time focusing are marked by warmer colors (e.g., red), while the less attentive areas are presented with cooler colors (e.g., blue). The heatmap is made using libraries like Seaborn and Matplotlib so that the data will be easy to interpret visually. The heatmap enables advertisers to immediately see the areas of the screen or ad that receive the most attention.

3.4.3. SHOWING THE HEATMAP

The heatmap is shown as an overlay over the original screen or ad. This overlay enables the user to visually see where attention is concentrated during the interaction with the screen. Through the use of this visualization, advertisers can ascertain the success of their ad placements. If specific aspects of the ad (e.g., images, logos, or text) are attracting greater attention, the advertisers can tune their campaign for maximum interaction. This facility gives useful, real-time information on how the digital content is being engaged with by the users

3.5. WRAPPER APPLICATION

3.5.1 DEVELOPMENT OF A WRAPPER APPLICATION USING STREAMLIT TO EXECUTE THE PROJECT

In order to facilitate ease of usage and usability, the project is wrapped into an easy-to-use wrapper application developed using Streamlit. Streamlit is an open-source library that facilitates the process of converting Python code into interactive web applications. Utilizing Streamlit, the application can be seamlessly deployed and run by advertisers using no technical know-how. Users can interact with the gaze tracker and heatmap creation system through a wrapper application in the form of a clean, intuitive interface. This solution allows the advertisers to run the project on any machine with minimal configuration.

3.6. PROPOSED MODEL

The core libraries employed for this project are OpenCV for image processing, MediaPipe for eye-gaze, and autopy for the simulation of the mouse, chosen according to their respective capabilities in delivering a stable and efficient system. OpenCV is amongst the most used libraries in the field of computer vision because it is a high-end library that captures video and does real-time processing of images. Its adaptability makes it optimum for utilization for eye and face detection such that the system is able to select salient features of the face of the user from the video of the webcam. OpenCV's real-time processing support enables the system to detect and track eye movement with minimal lag, which is critical in providing precise eye gaze information.

MediaPipe was utilized for eye tracking because of its advanced, high-accuracy facial landmark detection. This library can especially extract data for eye and facial features at high accuracy levels, which are necessary for the proper identification of the user's pupils' locations and estimation of their point of gaze. The robust handling by MediaPipe for various lighting conditions, angles, and face expressions makes it the ideal library to use in this project since the tracking of gaze will be correct even in harsh conditions.

Autopy, specifically designed to automate mouse movement and action, is used to simulate user interaction with the screen from the gaze data. It allows the system to move the mouse cursor in real time according to the user's eye movements, making it an interactive system. Through the combination of these three robust libraries—OpenCV for image processing, MediaPipe for precise gaze tracking, and Autopy for screen interaction—this project creates a smooth, precise, and scalable process. The coming together of these technologies creates a technologically superior process of tracking user attention patterns, hence being a worthwhile tool in enhancing digital advertisements through the provision of actionable knowledge of where users direct their attention. The convergence makes it possible for the system to function well and with extremely high accuracy, hence its application suitability in modern day advertising.

3.7. POTENTIAL BENEFITS

3.7.1. Real-Time Data: The system incorporates real-time gaze tracking, and advertisers can obtain instant feedback regarding user attention patterns. This helps in making timely adjustments to marketing strategies, enhancing the effectiveness of advertisement placements.

3.7.2. Cost-Effective: By using open-source libraries like OpenCV and MediaPipe, the project is very cost-effective, allowing small businesses and individual developers to use it without having to spend money on proprietary solutions.

3.7.3. High Accuracy: Precise detection of gaze with the assistance of advanced tracking capabilities of MediaPipe enables accurate measurement and analysis of user interest. This implies greater accuracy for the entire system to compute precise visual focus metrics.

3.7.4. Scalability: The platform allows support for simultaneous use by multiple users, which places it at an advantage in situations of large promotions and A/B testing across different target bases, thereby allowing greater usability in business settings.

3.7.5. Actionable Insights: The heat maps generated give advertisers extensive visual representations of viewer behavior. The fact-based process allows precise placement of ads by locating the optimal areas of digital content.

3.8. POTENTIAL LIMITATIONS

3.8.1. Hardware Dependency: The accuracy of gaze tracking relies heavily on the quality of the webcam. Cameras with lower resolutions may struggle to detect pupil movement accurately, thereby making the system less effective overall.

3.8.2. Environmental Factors: Accuracy in eye and face detection can be influenced by lighting variations. Low light or extreme glare can interfere with consistent gaze tracking, which will reduce the performance of the system in low-light environments.

3.8.3. Real-Time Processing Overhead: Real-time processing of video stream and real-time gaze tracking can be system performance intensive, especially on low-end hardware.

3.8.4. User Calibration:

Users can be required to perform a calibration of the system in order to properly track gaze, adding an extra step before the system is used to its benefit, and potentially add errors if done inadequately.

3.8.5. Restricted View Zone:

The system can lose gaze position accuracy when the subject does not direct the gaze onto the camera and more so when s/he displaces the head a long way off from center.

3.9. CONCLUSION

The method outlines a robust system for ad optimization and gaze tracking with the aid of several state-of-the-art libraries like OpenCV and MediaPipe. By tracking eye movements and representing them with heatmaps, the system provides marketers valuable insights to optimize their ads. With the use of a Streamlit wrapper application, the project is made user-friendly and accessible. This approach illustrates an end-to-end approach to enhancing digital advertising using eye gaze tracking.

Chapter 4

VISION TRACKING

4.1 INTRODUCTION

The following subsection addresses following a user's gaze through detection of facial structure and marking where the eyes are. Following gaze is based on following sight, in which it is used to detect the user's eye position, which is then processed to compute the line of sight. This includes webcam video recording, face detection, conversion of the image to grayscale mode, and bounding of the region of interest (ROI), thereby allowing ease and precision in subsequent tracking.

4.2. CAPTURING THE FEED FROM THE WEBCAM – Capturing the feed from the webcam in order to receive an uninterrupted series of frames is vision tracking's initial operation. It is done with OpenCV's VideoCapture, which can handle video processing in real time. The webcam is read continuously by the system, so each frame can be processed for real-time face and eye detection. Dynamic tracking of eye movement direction without delay is allowed by this.

4.3. FACE DETECTION – After capturing the webcam video feed, the face of the user must be detected. Detection is done using OpenCV's pre-trained Haar Cascade classifier.

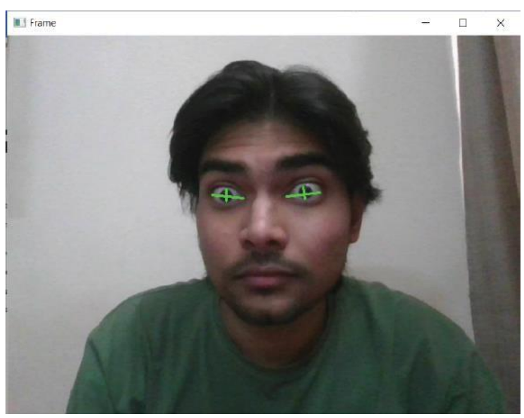

Fig 1– Face Detection Using Opencv

Haar Cascade is a machine learning-based approach for object detection, which detects faces, eyes, and other objects from an image or video. Haar Cascade is a common technique used in computer vision tasks because it is efficient and performs well. Haar Cascade uses patterns of pixel intensity, and not pixel value detection. Haar Cascade uses a fixed pipeline for object detection. The significant steps are:

4.3.1 HAAR FEATURE CALCULATION – The initial step is to gather the Haar features. A Haar feature is basically calculations done on neighboring rectangular areas at a specific point in a detection window. The calculation is done by adding the pixel intensities in each area and computing the differences between the sums.

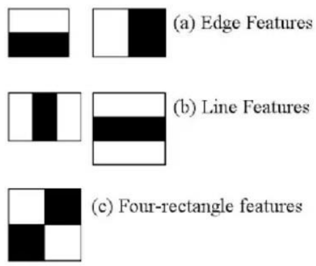

Fig 2– HAAR Features Used For Face Segmentation

4.3.2 CONSTRUCTION OF INTEGRAL IMAGES – Rather than computing at each pixel, Integral constructs sliding windows sub-rectangles and constructs array references for each of the sub-rectangles. These are utilized to calculate the Haar features.

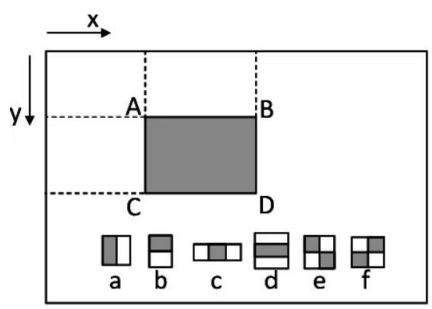

Fig 3 – HAAR Integral Sliding Windows Generation

4.3.3 TRAINING ADABOOST – Adaboost primarily selects the best features and trains the classifiers to employ them. Adaboost applies a mixture of "weak classifiers" in order to generate a "strong classifier" which can be employed by the algorithm for object detection.

4.3.4 CASCADING CLASSIFIERS – The cascade classifier consists of a sequence of stages, with each stage being an ensemble of weak learners. Weak learners are trained with boosting so that a very accurate classifier from the mean prediction of all the weak learners is possible. From this prediction, the classifier either chooses to report an object found (positive) or proceed to the next region (negative).

4.4 IMAGE GRAYSCALE CONVERSION – Grayscale conversion of the image simplifies it for easier processing. The frame captured is converted to grayscale to enhance efficiency in processing. The colored images have three channels (RGB), which contribute to computational complexity. Grayscale conversion also has the benefit of reducing the image into a single channel of intensity to enable algorithms to identify patterns such as eyes and facial features with greater accuracy and lower cost computationally.

4.45 ISOLATING THE REGION OF INTEREST (ROI) – Isolation of eyes as the main region of interest (ROI) is the next step after face detection. Segmentation of the eye area from the face is carried out to consider only those features of importance for gaze tracking. This greatly enhances the speed and accuracy of processing by removing redundant background and superfluous facial features.

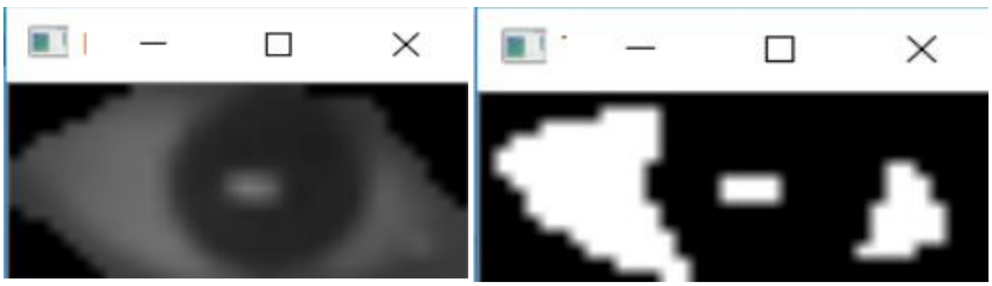

Fig 4– Isolated Region of Interest

Chapter 5

LINE OF SIGHT TRACKING

5.1 INTRODUCTION

The following subsection addresses the path of the user's gaze by converting the pupil position to a point on the screen. According to the movement of the pupil in the area of interest specified, we can approximate the focus of the user. This includes pupil extraction, application of Haar Cascades, computation of the gaze, and data storage.

5.2 EXTRACTING PUPIL LOCATION FROM THE ROI – After segmentation of the eye region, further processing is carried out to find and track the pupil. This is done by image thresholding and edge detection to find the darkest part of the eye, which is generally the pupil. Having knowledge of the pupil's location, we can translate its movement into the corresponding screen coordinates.

5.3 CALCULATING THE GAZE DIRECTION – After the location of the pupil is obtained, it is referenced in relation to the eye region. A reference system of coordinates is set in place such that the center of the eye region is considered to be the zero point. The change in direction of view is any movement of the pupil in the direction of the edge of the eye region. This information is utilized in identifying where the user is looking at on the display.

5.4 STORAGE OF THE DATA TO THE CSV FILE – Since results of gaze tracking can be processed later, gaze coordinates obtained from the feature tracker are also stored in a CSV file. In this way, record of real-time moment pupil positions is maintained, hence helpful in obtaining graphical representations for research purposes.

	A	B	C
1	timestamp	eye_x	eye_y
2	1.71207E+12	320	240
3	1.71207E+12	330	245
4	1.71207E+12	340	250
5	1.71207E+12	335	248
6	1.71207E+12	350	260
7	1.71207E+12	360	270
8	1.71207E+12	370	275
9	1.71207E+12	365	272
10	1.71207E+12	375	280
11	1.71207E+12	380	290
12	1.71207E+12	390	295
13	1.71207E+12	400	300
14	1.71207E+12	410	310
15	1.71207E+12	420	320
16	1.71207E+12	430	325

Table 1 – Gaze Tracking Data with Timestamp

Chapter 6
VISUALIZATION

6.1 INTRODUCTION

Screen visualization section strives to transform raw gaze data into useful visualizations. Some of these include heatmaps and scatter plots that aid in analyzing user interaction with content on the screen.

6.2 SAVING THE MOUSE POSITION DATA INTO CSV FILE – Besides the tracking data of the user's gaze, the position of the cursor is also logged. It is saved into a CSV file so that gaze and cursor movement path comparisons can be made.

	A	B	C	D	E
1	timestamp	eye_x	eye_y	mouse_x	mouse_y
2	1.71207E+12	320	240	500	300
3	1.71207E+12	330	245	510	310
4	1.71207E+12	340	250	520	320
5	1.71207E+12	335	248	515	315
6	1.71207E+12	350	260	530	330
7	1.71207E+12	360	270	540	340
8	1.71207E+12	370	275	550	345
9	1.71207E+12	365	272	545	342
10	1.71207E+12	375	280	555	350
11	1.71207E+12	380	290	560	360
12	1.71207E+12	390	295	570	370
13	1.71207E+12	400	300	580	380
14	1.71207E+12	410	310	590	390
15	1.71207E+12	420	320	600	400
16	1.71207E+12	430	325	610	405

Table 2 – Complete Data with Timestamp

6.3 LOADING THE CSV DATA – Data, after being gathered, is loaded for processing. Pandas and NumPy libraries of Python are used for cleaning and data structure for visualization.

6.4 CREATING THE HEATMAP OF THE GAZE – A heatmap is created in a way that one can visualize how areas of the screen are viewed most. Seaborn library is used to plot gaze points, and the intensity of color represents the most viewed areas.

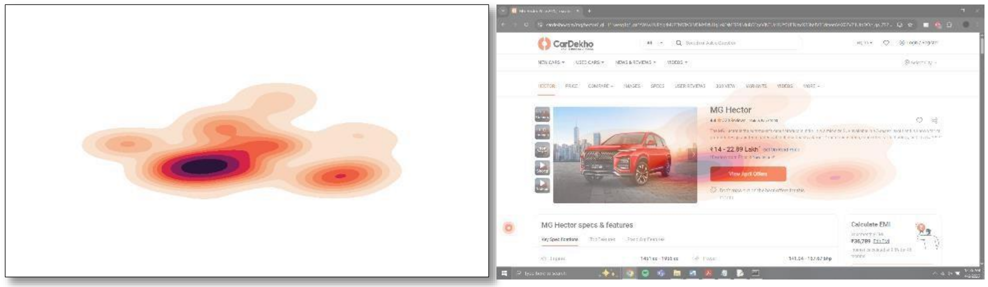

Fig 5 – Sample Heatmap Generated

6.5 DRAWING THE SCATTER FOR THE MOUSE – Apart from heatmaps, scatter plots are drawn in a bid to represent the path of the mouse. It is used to compare cursor interaction with the gaze-tracking information to research user behavior.

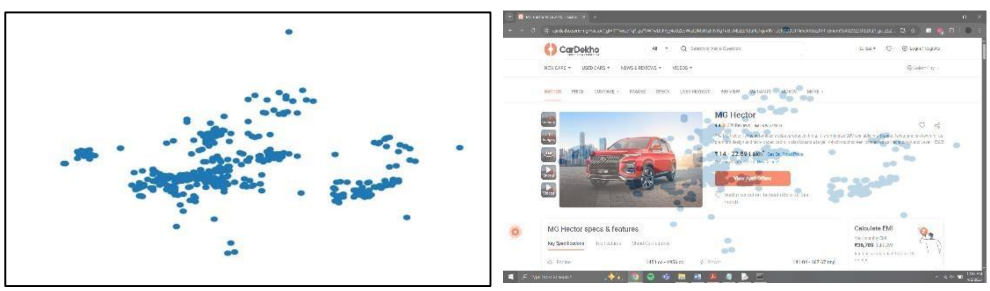

Fig 6 – Sample Scatter Generated

6.6 Creating a Wrapper Application Using Streamlit – Streamlit is used to develop an interactive interface where users can access gaze tracking results. The application allows real-time tracking, visualization of heatmaps, and data export functionalities, making the project accessible for research and commercial use.

Chapter 8
FUTURE WORK

The current project has laid a sound foundation for heatmap generation and gaze tracking with better advertisement placement. However, there are some scopes where the project can further strengthen its relevance and usefulness in greater ways. One of the most logical scopes for expansion is leveraging more sophisticated machine learning models in order to even better improve gaze tracking accuracy. Although the system is OpenCV and MediaPipe-based, the latter two do not provide significant gaze tracking. With deep learning capabilities that allow for detecting lesser changes like shape change in eyes and slight movement, the system can be better optimized. The inclusion of user-trained models that are part of it would provide for having a more adaptive form of gaze tracking that can cope with more dissimilar facial patterns and surroundings and thus make the system even more powerful.

A further potential addition is the inclusion of a multi-user tracking function. The current implementation of the system is used for single-user gaze tracking, but most genuine applications—especially advertising applications—must be able to track multiple users simultaneously. Further implementations of the system can include functions for processing multiple individuals simultaneously, e.g., through facial identification and separation of gaze. This would facilitate easier monitoring of group activity, user reaction to the content, and interaction between various viewers. An augmentation of this type would immensely enhance the usability of the system upon use in consumer behavior research as well as in research on the market.

Other interactive features for interaction could be added to ensure that the project is enhanced. The heatmap visualisation is currently static and points where the users predominantly glance at whenever there is a screen interaction. Future heatmaps may be interactive too, whereby researchers or advertisers can click areas of interest and get more specifics on user activity within such areas. This interaction would have the impact of displaying detailed analytics on the screen, i.e., gaze time or over-time trends in behaviour. Additionally, it would be interesting to merge eye-tracking data with other user-related data, such as eye-movements per stroke or head movements. This would provide an idea about the overall pattern of engagement by the users. A combination of some biometric indicators would be capable of having the maximum possible accuracy in behavioral prediction and advertising.

Further research could involve incorporating the software to be utilized in a variety of environments in an effort to further enhance user experience as well as increase uses of the system. This is possible through adaptations of gaze tracking to be effective across a variety of light conditions or different screen display formats. For instance, it can be designed to support gaze tracking algorithms that can operate effectively in low-light conditions, thereby making the system stable across environments. This would specifically be useful in applications like interactive shelf signs for physical retail stores or public-scale screens, where lighting will be unpredictable. The system can also be designed to support mobiles and tablets, which would make it even more platform-independent in the sense of portability so that gaze tracking is done uniformly on smartphones, tablets, and laptops.

Lastly, there may be more work on the project in the direction of employing real-time adaptive advertisement placement based on the gaze data. Even though the current system supported adding generation of heatmap into it, it is also a dynamic real-time system that has the capability of dynamically altering the placement of advertisements based on where the viewer's current location is. For example, if the user's focus is shifted from one part of the screen to another, then the content of the advertisement itself can be adjusted based on their interest. This would introduce a level of interactivity and personalization that can revolutionize digital advertising by showing viewers content relevant to their actual real-time attention, increasing engagement, and conversion. The integration of these sorts of real-time, gaze-based advertising systems can help to further optimize an increasingly user-centered, immersive digital media experience.

Lastly, further future advancement of this system of one's eye tracking will be capable of further enhancing placement and ad-viewing. Adding machine learning, multi-user support, dynamic user interaction, environment awareness, and real-time ad adaptation, this project also can act as an effective solution for learning about the user experience as well as adapting ad-targeting strategies to digital media.

Chapter 9
CONCLUSION

This project shows the possibilities of eye gaze tracking technology improving digital ad placement optimization with a sense of where people place their visual attention on a screen. OpenCV, MediaPipe, and autopy can be used to develop a system not only that can trace eye movement but also convert it into actionable information for advertisers. This approach allows advertisers to make informed decisions regarding the placement of the advert, optimizing user interaction and campaign efficacy. Since feedback can be captured in real time with the use of real-time gaze tracking, advertisers are able to optimize strategy in real time as a function of sequences of user attention.

The ability of the project to become cost-effective and implementable with the use of open-source libraries such as OpenCV and MediaPipe as an added benefit. The modularity of such libraries allows for the construction of a strong and efficient gaze tracking system without the need to invest in proprietary software, thus allowing small companies and individual developers to use such technology, thus democrazing its use. High-precision accuracy provided by MediaPipe facial landmark tracking also increases the robustness of the system and allows for the accurate tracking of user attention. This quantity of data offers marketers an overview of the depth of user action and how they must position their ads so that they belong in the most captivating parts of their content. Scalability is one of the principal advantages of this project.

Multiple users can be supported by the system at the same time, which makes it ideal for A/B testing or large ad campaigns. It is this scalability that also makes the technology accessible for application in any type of business environment, ranging from the small tests to large campaigns on enormous sets of audiences. The heatmaps provided by the system also provide a sound visualization tool which the advertisers can utilize to get tangible insight into the attention of the viewers. To certain areas of the screen, proportionate advertisements can be placed so that their visibility and responsiveness are optimized. While there are several benefits to the project, there is a constraint.

Webcam quality is key to the effectiveness of the eye tracking. Subpar or low-quality cameras would result in poor pupil detection, cascading to the entire system. Environmental conditions, especially lighting, also play an important role in the performance of the system. Poor lighting or excessive glare would lower the performance of the system in recognizing facial features, thereby impacting the accuracy of the gaze tracking. These limitations form a strong argument for the use of the highest quality equipment and controlled environments in a bid to provide consistent performance. Despite such constraints, the system is a gigantic leap forward for internet advertising. With the provision of accurate real-time gaze tracking and the visualization of user attention in the form of heatmaps, the project provides valuable insights regarding the consumption of digital content by users.

Data provided by this can be used for the designing of ad positioning strategies in a manner that ads reach their audience optimally. With the evolving technology, the system is even capable of optimizing the ad, providing advertisers a valuable resource to best optimize their campaigns. Overall, this project shows the future of eye gaze tracking in digital ads and lays the groundwork for future user experience breakthroughs research and improved marketing campaigns. With current technology and by putting them together in another manner, the project provides a cost-efficient and scalable solution which can be an innovative game-breaker in perception and consumption of advertising. Through the capability of measuring eye movements and creating heatmaps, more is known about the viewer activity, which carries potential for a more engaging and effective advertisement to come.

APPENDICES

GENERATING FACE DETECTOR AND LANDMARK PREDICTOR

```
import cv2
import dlib

detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
```

Fig 7 – Code – Generating Detector And Predictor

MOUSE POSITIONING

```
def transform_video_coordinates_to_screen(eye_x_pos, eye_y_pos):
    if not video_resolution:
        return (eye_x_pos, eye_y_pos)
    return (
        eye_x_pos / video_resolution[0] * screen_resolution[0],
        eye_y_pos / video_resolution[1] * screen_resolution[1],
    )
def update_mouse_position(hough_circles, eye_x_pos, eye_y_pos, roi_color2):
    try:
        for circle in hough_circles[0, :]:
            circle_center = (circle[0], circle[1])
            cv2.circle(
                img=roi_color2,
                center=circle_center,
                radius=circle[2],
                color=WHITE,
                thickness=2
            )
            cv2.circle(
                img=roi_color2,
                center=circle_center,
                radius=2,
                color=WHITE,
                thickness=3
            )
            x_pos = int(eye_x_pos)
            y_pos = int(eye_y_pos)
            (x_pos, y_pos) = transform_video_coordinates_to_screen(eye_x_pos, eye_y_pos)
            autopy.mouse.move(x_pos, y_pos)
    except Exception as e:
        print('Exception:', e)
```

Fig 8 – Code – Detecting Mouse Positioning

EYE BLINKING

```python
# Calculate eye blinking ratio
def get_blinking_ratio(eye_points, facial_landmarks):
    left_point = (facial_landmarks.part(eye_points[0]).x, facial_landmarks.part(eye_points[0]).y)
    right_point = (facial_landmarks.part(eye_points[3]).x, facial_landmarks.part(eye_points[3]).y)
    center_top = midpoint(facial_landmarks.part(eye_points[1]), facial_landmarks.part(eye_points[2]))
    center_bottom = midpoint(facial_landmarks.part(eye_points[5]), facial_landmarks.part(eye_points[4]))

    hor_line_length = hypot((left_point[0] - right_point[0]), (left_point[1] - right_point[1]))
    ver_line_length = hypot((center_top[0] - center_bottom[0]), (center_top[1] - center_bottom[1]))
    ratio = hor_line_length / ver_line_length
    return ratio
```

Fig 9 – Code – Detecting Eye Blinking

LINE OF SIGHT CALCULATION

```python
# Get gaze ratio (horizontal)
def get_gaze_ratio(eye_points, facial_landmarks, frame, gray):
    left_eye_region = np.array([(facial_landmarks.part(eye_points[i]).x, facial_landmarks.part(eye_points[i]).y) for i in range(6)], np.int32)
    height, width, _ = frame.shape
    mask = np.zeros((height, width), np.uint8)
    cv2.fillPoly(mask, [left_eye_region], 255)
    eye = cv2.bitwise_and(gray, gray, mask=mask)

    min_x = np.min(left_eye_region[:, 0])
    max_x = np.max(left_eye_region[:, 0])
    min_y = np.min(left_eye_region[:, 1])
    max_y = np.max(left_eye_region[:, 1])

    gray_eye = eye[min_y: max_y, min_x: max_x]
    _, threshold_eye = cv2.threshold(gray_eye, 70, 255, cv2.THRESH_BINARY)
    left_side_threshold = threshold_eye[:, :threshold_eye.shape[1] // 2]
    right_side_threshold = threshold_eye[:, threshold_eye.shape[1] // 2:]
    left_side_white = cv2.countNonZero(left_side_threshold)
    right_side_white = cv2.countNonZero(right_side_threshold)

    if left_side_white == 0:
        return 0  # Looking Left
    elif right_side_white == 0:
        return 1  # Looking Right
    else:
        return left_side_white / right_side_white

# Get gaze ratio (vertical)
def get_vertical_gaze_ratio(eye_points, facial_landmarks, frame, gray):
    left_eye_region = np.array([(facial_landmarks.part(eye_points[i]).x, facial_landmarks.part(eye_points[i]).y) for i in range(6)], np.int32)
    height, width, _ = frame.shape
    mask = np.zeros((height, width), np.uint8)
    cv2.fillPoly(mask, [left_eye_region], 255)
    eye = cv2.bitwise_and(gray, gray, mask=mask)

    min_x = np.min(left_eye_region[:, 0])
    max_x = np.max(left_eye_region[:, 0])
    min_y = np.min(left_eye_region[:, 1])
    max_y = np.max(left_eye_region[:, 1])

    gray_eye = eye[min_y: max_y, min_x: max_x]
    _, threshold_eye = cv2.threshold(gray_eye, 70, 255, cv2.THRESH_BINARY)
    upper_half = threshold_eye[:threshold_eye.shape[0] // 2, :]
    lower_half = threshold_eye[threshold_eye.shape[0] // 2:, :]
    upper_white = cv2.countNonZero(upper_half)
    lower_white = cv2.countNonZero(lower_half)

    if upper_white == 0:
        return 1  # Looking Down
    elif lower_white == 0:
        return 0  # Looking Up
    else:
        return upper_white / lower_white
```

Fig 10 – Code – Calculating Line Of Sight

VIDEO CAPTURING

```python
video_capture = cv2.VideoCapture(0)
eye_x_positions = list()
eye_y_positions = list()
screen_resolution = autopy.screen.size()
print("screen resolution is")
print(screen_resolution)
if video_capture.isOpened():
    video_resolution = (
        video_capture.get(cv2.CAP_PROP_FRAME_WIDTH),
        video_capture.get(cv2.CAP_PROP_FRAME_HEIGHT),
    )
else:
    video_resolution = None
while 1:
    success, image = video_capture.read()
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    #faces = face_cascade.detectMultiScale(gray, 1.3, 5)
    eyes = eye_cascade.detectMultiScale(gray)
    for (eye_x, eye_y, eye_width, eye_height) in eyes:
        cv2.rectangle(
            img=image,
            pt1=(eye_x, eye_y),
            pt2=(eye_x + eye_width, eye_y + eye_height),
            color=GREEN,
            thickness=2
        )
        roi_gray2 = gray[eye_y: eye_y + eye_height, eye_x: eye_x + eye_width]
        roi_color2 = image[eye_y: eye_y + eye_height, eye_x: eye_x + eye_width]
        hough_circles = cv2.HoughCircles(
            roi_gray2,
            cv2.HOUGH_GRADIENT,
            1,
            200,
            param1=200,
            param2=1,
            minRadius=0,
            maxRadius=0
        )
        eye_x_pos = (eye_x + eye_width) / 2
        eye_y_pos = (eye_y + eye_height) / 2
        print(eye_x_pos, eye_y_pos)
        eye_x_positions.append(eye_x_pos)
        eye_y_positions.append(eye_y_pos)
        update_mouse_position(hough_circles, eye_x_pos, eye_y_pos, roi_color2)
    cv2.imshow('img', image)
    key_pressed = cv2.waitKey(30) & 0xff
    if key_pressed == ESCAPE_KEY:
        break
video_capture.release()
```

Fig 11 – Code – Calculating Video

HEATMAP DATA INTERPRETATION

```
# Load the CSV file
df = pd.read_csv(csv_filename)
# Get the screen resolution
screen_width, screen_height = pyautogui.size()
# Generate the gaze heatmap data
gaze_data = []
# Calculate the position based on the ratios and screen resolution
for index, row in df.iterrows():
    vertical_ratio = row["Vertical"]
    horizontal_ratio = row["Horizontal"]
    # Map gaze ratios to screen coordinates
    gaze_x = int(horizontal_ratio * screen_width)
    gaze_y = int(vertical_ratio * screen_height)
    gaze_data.append([gaze_x, gaze_y])
```

Fig 12 – Code – Interpreting Heatmap Data

GAZE HEATMAP DISPLAY

```
# Window 1: Screenshot with Gaze Heatmap
plt.figure(figsize=(8, 8))
plt.imshow(screenshot_np)
plt.scatter(gaze_df["Gaze X"], gaze_df["Gaze Y"], color='red', s=10)
plt.title("Screenshot with Gaze Heatmap")
plt.show()
```

Fig 13 – Code – Displaying Gaze Heatmap

SCATTER PLOT DATA INTERPRETATION

```
# Convert the gaze data to a DataFrame for easy manipulation
gaze_df = pd.DataFrame(gaze_data, columns=["Gaze X", "Gaze Y"])
# Scatter plot for the mouse movement (we'll use Mouse X, Mouse Y from the CSV)
mouse_data = df[["Mouse X", "Mouse Y"]]
# Take a screenshot of the current screen
screenshot = pyautogui.screenshot()
screenshot_np = np.array(screenshot)
screenshot_bgr = cv2.cvtColor(screenshot_np, cv2.COLOR_RGB2BGR)
```

Fig 14 – Interpreting Scatter Plot Data

SCATTER PLOT DISPLAY

```
# Window 2: Screenshot with Mouse Movement Scatter
plt.figure(figsize=(8, 8))
plt.imshow(screenshot_np)
plt.scatter(mouse_data["Mouse X"], mouse_data["Mouse Y"], color='blue', s=10)
plt.title("Screenshot with Mouse Movement")
plt.show()
```

Fig 15 – Displaying Scatter Plot

REFERENCES

[1] 'Learning to Find Eye Region Landmarks for Remote Gaze Estimation in Unconstrained Settings': Seonwook Park, Xucong Zhang, Andreas Bulling, Otmar Hilliges

[2] Head pose estimation in the wild using Convolutional Neural Networks and adaptive gradient methods: Massimiliano Patacchiola, Angelo Cangelosi

[3] Application of Eye Tracking Technology in Medicine: A Bibliometric Analysis: Gianpalolo Zammarchi, Claudio Conversano

[4] Assessing Consumer Attention and Arousal Using Eye-Tracking Technology in Virtual Retail Environment: Nayeon Kim, Hyunsoo Lee

[5] Accelerating eye movement research via accurate and affordable smartphone eye tracking: Nachiappan Valliappan, Na Dai, Ethan Steinberg, Junfeng He, Kantwon Rogers, Venky Ramachandran, Pingmei Xu, Mina Shojaeizadeh, Li Guo, Kai Kohlhoff & Vidhya Navalpakkam

[6] Utilising eye-tracking data in retailing field research: A practical guide: Jens Nordfält , Carl-Philip Ahlbom

[7] Using eye-tracking technology in Neuromarketing: Consuela-Mădălina Gheorghe *, Victor Lorin Purcărea *, Iuliana-Raluca Gheorghe *

[8] Eye tracking: empirical foundations for a minimal reporting guideline: Kenneth Holmqvist, Saga Lee Örbom, Ignace T. C. Hooge, Diederick C. Niehorster, Robert G. Alexander, Richard Andersson

[9] Eye-tracking in Marketing Research: Sylwester Andrzej Białowąs, Adrianna Szyszka

[10] A study of Eye Tracking Technology and its applications: Pramodini A. Punde, Dr. Mukti E. Jadhav, Dr Ramesh R. Manaza