Harnessing the Power of Self-Training for Gaze Point Estimation in Dual Camera Transportation Datasets

Hirva Bhagat

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Science

Anuj Karpatne, Co-chair
Lynn Abbott, Co-chair
Abhijit Sarkar
Edward Fox

May 05, 2023
Blacksburg, Virginia

Keywords: Point of gaze, gaze point estimation, self training, semi-supervised learning, driver safety

Harnessing the Power of Self-Training for Gaze Point Estimation in Dual Camera Transportation Datasets

Hirva Bhagat

(ABSTRACT)

This thesis proposes a novel approach for efficiently estimating gaze points in dual camera transportation datasets. Traditional methods for gaze point estimation are dependent on large amounts of labeled data, which can be both expensive and time-consuming to collect. Additionally, alignment and calibration of the two camera views present significant challenges. To overcome these limitations, this thesis investigates the use of self-learning techniques such as semi-supervised learning and self-training, which can reduce the need for labeled data while maintaining high accuracy. The proposed method is evaluated on the DGAZE dataset and achieves a 57.2% improvement in performance compared to the previous methods. This approach can prove to be a valuable tool for studying visual attention in transportation research, leading to more cost-effective and efficient research in this field.

Harnessing the Power of Self-Training for Gaze Point Estimation in Dual Camera Transportation Datasets

Hirva Bhagat

(GENERAL AUDIENCE ABSTRACT)

This thesis presents a new method for efficiently estimating the gaze point of drivers while driving, which is crucial for understanding driver behavior and improving transportation safety. Traditional methods require a lot of labeled data, which can be time-consuming and expensive to obtain. This thesis proposes a self-learning approach that can learn from both labeled and unlabeled data, reducing the need for labeled data while maintaining high accuracy. By training the model on labeled data and using its own estimations on unlabeled data to improve its performance, the proposed approach can adapt to new scenarios and improve its accuracy over time. The proposed method is evaluated on the DGAZE dataset and achieves a 57.2% improvement in performance compared to the previous methods. Overall, this approach offers a more efficient and cost-effective solution that can potentially help improve transportation safety by providing a better understanding of driver behavior. This approach can prove to be a valuable tool for studying visual attention in transportation research, leading to more cost-effective and efficient research in this field.

Dedication

I would like to dedicate this work to my family and to the person who has provided consistent support and encouragement throughout my academic journey - Gitartha.

Acknowledgments

I would like to express my sincere gratitude to my co-advisors, Dr. Lynn Abbott and Dr. Anuj Karpatne, for their invaluable guidance, support, and encouragement throughout my academic journey. Their complementary areas of expertise and perspectives have greatly contributed to the development and improvement of my work. Additionally, I am grateful to my committee member, Dr. Edward Fox, for their insightful feedback and constructive criticism that has further enriched my research. I would also like to extend my sincere thanks to my supervisor, Dr. Abhijit Sarkar, for their exceptional mentorship, which has been critical in shaping both my research and personal growth. Their extensive knowledge, expertise, and willingness to share their insights have been instrumental in the success of this work.

Contents

List of Figures ix

List of Tables xii

1 Introduction 1

1.1 Background and Motivation 1

1.2 Problem Statement and Research Questions 4

1.3 Scope and Objectives 5

2 Review of Literature 7

2.1 Significance of Gaze Point in Transportation Research 7

2.2 Estimation of Gaze Points 8

2.3 Semi-supervised Learning and Self Training 17

2.4 Dual Camera Datasets in Transportation Research 19

2.4.1 Importance of Dual Camera Datasets 19

2.5 Summary and Research Gaps 20

3 Methodology 22

3.1 DGAZE and I-DGAZE 22

3.1.1 DGAZE . . . 22
3.1.2 I-DGAZE Model . . . 24
3.2 PoG Algorithm . . . 26
3.2.1 Overview . . . 26
3.2.2 Evaluation and Use case . . . 29
3.3 Proposed Methodology . . . 31
3.3.1 Semi-Supervised Learning and Self-Training for Gaze Point Estimation 32
3.4 Expected Outcomes and Contributions . . . 35

4 Experiment Setup **36**

4.1 Model Architecture and Training . . . 36
4.1.1 Base CNN Regression Architecture . . . 36
4.1.2 Optimizer and Loss Function . . . 38
4.1.3 Learning Rate Scheduler . . . 39
4.1.4 ST-GP: Self Training Based Gaze Point Estimation . . . 40
4.1.5 ST-GP (NS): Self Training Based Gaze Point Estimation Without Sampling . . . 41
4.1.6 WS-GP: Weighted Semi-supervised Learning Based Gaze Point Estimation Model . . . 42

5 Results **45**

5.1 Experimental Results and Analysis 45
5.1.1 Evaluation on Set 3 Test Dataset 45
5.1.2 Evaluation on Unseen Drivers Dataset 47
5.1.3 Comparative Analysis with Traditional Gaze Point Estimation Techniques 48
5.2 Ablation Studies 49
5.2.1 Training MAE Loss 49
5.2.2 ST-GP Epoch Variation 50
5.3 Qualitative Assessment 51
5.4 Discussion 53
5.5 Analysis of Results 55
5.5.1 Challenges 56

6 Conclusion **57**

6.1 Summary of Findings and Contributions 57
6.2 Future Work 58

Bibliography **61**

List of Figures

1.1 The task of gaze tracking involves determining the 3D gaze direction in the camera coordinate system or the 2D gaze point on the screen from images captured by a camera focused on the user's face region (Figure credit: [1]). . 2

1.2 The figure illustrates the dual camera setup used in transportation research to capture both the driver's perspective (road view) and external view (driver view). 3

2.1 Figure displays an example of an eye image, which highlights both the pupil and two reflections on the cornea, known as glints (Figure credit: [2]). 9

2.2 The figure shows the architecture of Zhang et al., the method takes a facial image as input and performs 2D and 3D gaze estimation using a convolutional neural network (CNN) with spatial weights applied on the feature maps. This approach represents an improvement over previous appearance-based methods, which have only utilized a limited amount of information for gaze estimation (Figure credit [3]). 15

3.1 Showcases DGAZE, a dataset designed to capture driver gaze on the road by incorporating both driver and road view through the use of inexpensive mobile phone cameras (Figure credit: [4]). 22

3.2 I-DGAZE is a two-branch late fusion convolutional neural network architecture aimed at estimating driver gaze on the road. One of the branches receives an eye image as input, while the other branch takes input of facial features such as head pose, face location, and the distance between the driver's face and the mobile phone camera. The overall goal of I-DGAZE is to accurately estimate where the driver is looking while on the road (Figure credit: [4]). . 24

3.3 PoG Algorithm takes the gaze angle extracted from the driver's face image using L2CS-Net and segmented objects to estimate a point and fixated object on the road image. 30

3.4 Proposed self-training approach. 34

4.1 The CNN Regression Architecture is designed to estimate the gaze of drivers while on the road. It is a modified version of I-DGAZE (Figure 3.2) that uses cropped face images to estimate x and y coordinates. In this architecture, the red dot represents the output generated by the CNN (pixel coordinates of the predicted gaze point). The accompanying image, along with its bounding box, is sourced from the DGAZE dataset and is included here solely for illustrative purposes. 36

5.1 Figures compare training loss in proposed models 50

5.2 Figures compare MAE loss on test and unseen drivers dataset for self trained models 51

5.3 Qualitative analysis of estimated gaze points using three top-performing models (ST-GP (50), ST-GP (NS 50), and WS-GP (50)). The figure is organized into columns displaying driver images, road images, and ground truth gaze points (green circles) alongside estimated gaze points (red circles) for each model. The proximity of red and green circles indicates the performance of each model. 52

List of Tables

3.1 Distribution of Fixated Object Events in the Dataset 27

3.2 Number of face images in the DAZE dataset is divided into 4 sets. set 1, 2 & 3 are created randomly while the unseen drivers dataset has 3 drivers strategically separated out to later use for evaluation. 33

4.1 CNN architecture and output shapes 38

5.1 This table shows the comparison of pixel MAE of the proposed models on the test dataset with respect to 10 self training iterations. The models evaluated are ST-GP (25), ST-GP (50), ST-GP (75), ST-GP (NS 50), and WS-GP (50). The key takeaway from the table is that the ST-GP (NS 50) model outperforms all other models in terms of MAE, achieving the lowest MAE of 79.14 pixels at the fourth iteration. 47

5.2 The table presents the comparison of pixel MAE for different proposed models on unseen drivers dataset with respect to 10 self training iterations. The ST-GP model with 25, 50, and 75 epochs and random sampling are compared with ST-GP without random sampling and WS-GP models. The table illustrates that the WS-GP (50) model achieves the lowest pixel MAE on the unseen drivers dataset, outperforming ST-GP models with varying epochs and random sampling as well as the ST-GP (NS 50) model without random sampling. 47

5.3 The table provides a comparison of performance metrics for different gaze estimation methods. The methods included in the table are TurkerGaze, MPIIGaze, iTracker, I-DGAZE, ST-GP (50), ST-GP (NS 50), and WS-GP (50). The table reports training errors, validation errors, and test errors for each method, illustrating that ST-GP (NS 50) achieves a notable performance with a test error of 79.14 pixels, outperforming other methods in this comparison. 48

List of Abbreviations

ALR Adaptive Linear Regression

BL Baseline

BN Batch Normalization

CNC Crash and Near-Crash

CNN Convolutional Neural Network

GANs Generative Adversarial Networks

IR Infrared

MAE Mean Absolute Error

mHoG modified Histogram of Oriented Gradients

NS No Sampling

PoG Point of Gaze

VTTI Virginia Tech Transportation Institute

Chapter 1

Introduction

1.1 Background and Motivation

The use of gaze information in transportation research has experienced a surge in recent years, as gaze offers a non-invasive and objective means of examining driver behavior in automobiles [5, 6]. By tracking a driver's eye movements, researchers can discern patterns of attention and distraction, as well as identify factors that impact visual behavior in automobiles, such as road layout, driving task complexity, and driver experience [7]. Researchers can use eye data to enhance driver support systems, improve car safety features, and guide the development of rules and regulations related to road safety. Moreover, knowing how drivers look around can aid in refining in-car information systems and enhancing automobile interface designs, leading to a safer and more user-friendly driving experience for everyone.

The point of gaze (PoG) or gaze point refers to the specific location on a screen or within an environment where an individual's eyes are fixated [2, 8, 9, 10] (Figure 1.1). Gaze point estimation is crucial for understanding human visual behavior because it reveals where a person's attention is concentrated and how that person processes information in the environment [11, 12]. By analyzing gaze points, researchers can deduce patterns of attention, identify cognitive processes, and understand how individuals interact with their surroundings. In transportation settings, gaze points help researchers comprehend where a driver is looking, offering valuable insights into the driver's attention and intentions [1, 13, 14, 15].

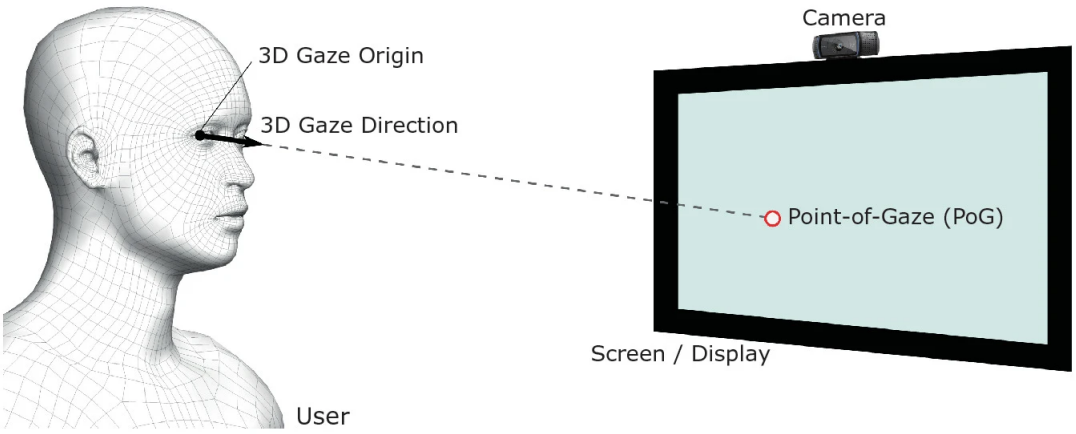


Figure 1.1: The task of gaze tracking involves determining the 3D gaze direction in the camera coordinate system or the 2D gaze point on the screen from images captured by a camera focused on the user's face region (Figure credit: [1]).

In the context of transportation research, the term "driver's perspective" or "road view" typically refers to the view of the road and surrounding environment as seen from the driver's point of view, while "external view" or "driver view" refers to the view of the driver and vehicle from an external point of view (Figure 1.2). Dual camera datasets capture both of these views simultaneously and are commonly used by transportation researchers to gain a more comprehensive understanding of driver behavior, identify risky driving patterns, and develop more effective driver assistance systems [16, 17, 18]. These datasets offer abundant information on the driver's gaze, head movement, and attention, enabling the understanding of how drivers engage with the environment and make decisions while driving. Nevertheless, determining the gaze point accurately in these datasets remains challenging, which necessitates careful alignment and calibration between the two camera views.

Current techniques for estimating gaze points generally depend on datasets labeled with ground truth gaze point information, often gathered using eye-tracking glasses or virtual reality setups [19]. However, the process of collecting labeled data from dual camera datasets can be time-consuming and costly, which makes it difficult to acquire large-scale datasets for

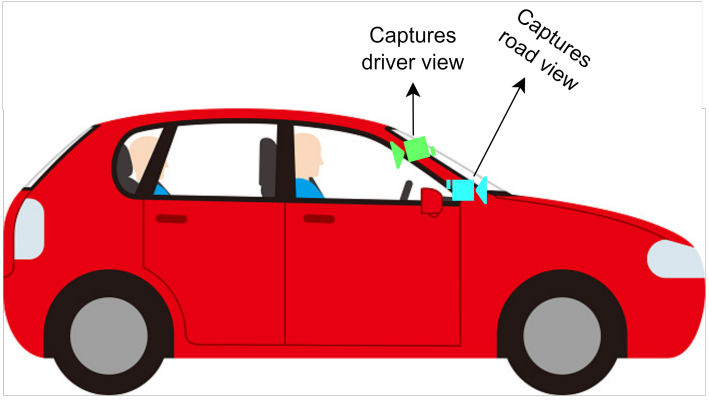


Figure 1.2: The figure illustrates the dual camera setup used in transportation research to capture both the driver's perspective (road view) and external view (driver view).

training and evaluation [20]. Therefore, there is a demand for techniques that can estimate gaze points from unlabeled dual-camera datasets without the need for explicit labeling.

In this thesis, we investigate the potential of semi-supervised learning and self-training techniques to enhance gaze point estimation accuracy from unlabeled datasets, utilizing only a small amount of labeled data. Semi-supervised learning is a machine learning approach that combines a small quantity of labeled data with a larger volume of unlabeled data during training [21, 22]. Self-training, a specific type of semi-supervised learning, involves training an initial model on a small labeled dataset and subsequently using the model to generate pseudo-labels for the unlabeled data [23, 24]. The model is iteratively refined by incorporating the pseudo-labeled data into the training process, effectively leveraging the information contained in the unlabeled data to improve estimation performance.

Semi-supervised learning and self-training for gaze point estimation have several benefits [25, 26, 27]. Firstly, they can substantially reduce the need for large amounts of labeled data. Secondly, they can improve the accuracy of gaze point estimation by utilizing the information present in unlabeled data. Thirdly, they can aid in the generalization of estimated gaze points to new scenarios by learning from a diverse range of unlabeled data.

This thesis aims to address the challenges and limitations associated with existing gaze point estimation techniques in dual camera transportation datasets by investigating the potential of semi-supervised learning, self-training, and deep neural networks, utilizing a small quantity of labeled data. The proposed approach seeks to mitigate the need for large-scale labeled data, improve the accuracy of gaze point estimation, and generalize the estimation to various drivers.

A significant application of this research is to support transportation researchers in integrating gaze information into their driver studies, enabling them to better comprehend the impact of gaze behavior on driver performance and decision-making. By offering a more efficient and accurate method for gaze point estimation, this research can help researchers uncover novel insights into the relationships between gaze, attention, and driving safety. These findings can, in turn, inform the design of driver training programs, road infrastructure, and advanced driver assistance systems, ultimately contributing to safer driving environments and enhanced road safety for all users.

1.2 Problem Statement and Research Questions

The current challenge with estimating gaze points in dual camera transportation datasets is that it heavily relies on labor-intensive and expensive labeled data collection processes. Additionally, careful alignment and calibration between the two camera views can be difficult to achieve in practice, making the process even more challenging. Therefore, a more efficient and accurate approach is needed that can estimate gaze points while reducing the dependency on large-scale labeled data and maintaining accuracy.

This thesis aims to train a CNN model, using driver's face images, to estimate points of fixation that correspond to the driver's gaze directions. The research intends to integrate

self-learning techniques into the CNN, enhancing the accuracy of gaze point estimation and enabling effective estimations with less labeled data.

To address this problem, the following research questions will be investigated:

1. How can semi-supervised learning be utilized to estimate gaze points in dual camera transportation datasets without relying on extensive labeled data?

2. What techniques can be employed to improve the accuracy of gaze point estimation using a self-training approach?

3. How can the proposed approach be generalized to unseen drivers, enabling more robust and reliable 2D gaze estimation across different situations?

4. How can the performance of the proposed approach be evaluated and compared with existing 2D gaze estimation techniques in terms of accuracy?

1.3 Scope and Objectives

This thesis focuses on the development and evaluation of a self-training-based approach for gaze point estimation in dual camera transportation datasets. The research is primarily concerned with transportation research applications, specifically the analysis of driver behavior, attention, and decision-making in various driving contexts. The scope of the study is limited to the utilization of self-training techniques and deep neural networks for gaze point estimation and does not extend to other aspects of driver behavior analysis or general gaze estimation techniques.

The main objectives of this research are as follows:

- Develop a semi-supervised gaze point estimation method to reduce reliance on large-scale labeled data and improve estimation efficiency.
- Design and optimize a deep neural network model capable of estimating gaze point from facial images with high accuracy.
- Evaluate the performance of the proposed self-learning-based gaze point estimation method against existing techniques.
- Investigate the practical implications of the proposed gaze point estimation approach for transportation researchers.

By achieving these objectives, this thesis aims to contribute to the advancement of gaze point estimation techniques in transportation research.

Chapter 2

Review of Literature

2.1 Significance of Gaze Point in Transportation Research

A driver's gaze point serves as a critical aspect of transportation research, offering valuable insights into driver behavior, attention, and decision-making processes. By understanding where drivers focus their attention while operating a vehicle, researchers can identify patterns of attention, distraction, and visual search behavior, as well as the factors that influence these behaviors [28, 29, 30].

Examining gaze point data enables researchers to detect risky driving patterns and develop more effective driver assistance systems [31, 32]. Furthermore, investigating gaze points in various driving scenarios facilitates the study of how drivers allocate their attention and respond to different road layouts, driving task complexities, and driver experience levels [33, 34, 35]. This information is crucial for designing safer vehicles, road infrastructure, and training programs.

Additionally, gaze point data can be utilized to assess the effectiveness of in-vehicle information systems and the potential impact of visual or cognitive distractions on driving performance [36, 37]. This knowledge can guide the design of user interfaces and other in-car technologies to minimize driver distraction and promote safer driving behaviors [38].

In summary, analyzing gaze points in transportation research provides a comprehensive understanding of driver behavior, allowing for the development of more effective interventions, policies, and technologies to promote safer driving and reduce the risk of accidents on the road.

2.2 Estimation of Gaze Points

The role of cameras and their placement in gaze tracking systems is fundamental for accurate data capture. Cameras must be positioned such that they provide a clear, unobstructed view of the user's eye. Typically, the camera is placed directly in front of the user, slightly above or below eye level, or slightly off-center, depending on the specific requirements of the application [19]. The angle and distance between the camera and the user are important factors that can affect the accuracy of gaze tracking. Using multiple cameras can also enhance the accuracy by providing different perspectives of the eye movements, thereby aiding in the detection and estimation of eye positions and gaze direction.

In gaze estimation tasks, the units of measurement employed, such as centimeters (cm), degrees, and pixels, each play a unique role and provide different insights. Centimeters, a unit of linear measurement, often depict physical distances such as the spacing between the user's eyes and the display screen or the size of the viewed object. Degrees, on the other hand, are units of angular measurement that characterize the direction and magnitude of eye movements or the line of sight relative to a reference point, usually the straight-ahead point. Pixels are used as a common unit in digital displays to measure gaze errors or displacements on the screen. The relationship between these units is not one of direct conversion, but they interact in the context of the specific gaze tracking setup, and understanding this interaction is vital for accurate gaze estimation.

In the early years of gaze tracking systems, Ware and Mikaelian [8] pioneered the use of infrared (IR) illumination in eye-tracking technology. The authors found that the use of IR wavelengths significantly improved gaze estimation accuracy by enhancing contrast and reducing the effects of ambient light. They also noted that the use of IR light was particularly useful in situations where eye movements were small and subtle, such as during reading. Many studies since then corroborated the importance of IR light sources in improving gaze estimation accuracy [9]. In 2005, Morimoto and Mimica [10] further advanced the field by developing an eye-tracking system that used multiple cameras and infrared light sources to track gaze in 3D environments. This innovation enabled more accurate gaze estimation, even in the presence of head movements. The authors used a dataset containing images from 10 participants and achieved an accuracy of around 1 degree of visual angle for the eye orientation estimation.

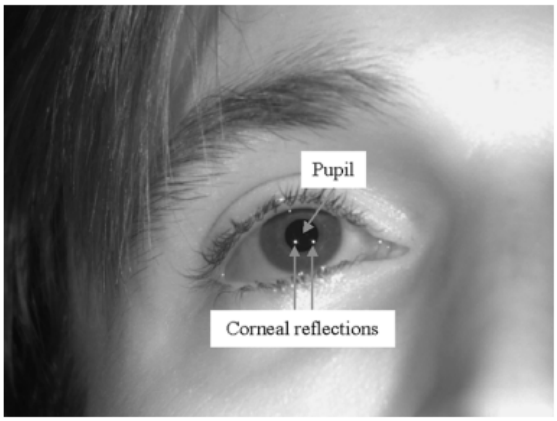


Figure 2.1: Figure displays an example of an eye image, which highlights both the pupil and two reflections on the cornea, known as glints (Figure credit: [2]).

Guestrin and Eizenman's [2] pioneering work on POG estimation using single-glint methods, which relies on the assumption that the corneal surface acts as a perfect mirror, represents a significant contribution to the field. Their system utilizes two near-infrared light sources and a video camera situated beneath a computer monitor to estimate the POG on the

screen. Their method uses a calibration procedure to measure/estimate system and subject-specific eye parameters, including the position of the light sources and camera parameters. Subject-specific parameters are obtained by having the subject fixate on nine points displayed sequentially on the screen. The POG is estimated by determining the coordinates of the centers of the pupil and glints (Figure 2.1) in each video image, transforming them into world coordinates, reconstructing the optic axis of the eye, and finally estimating the POG on the screen. The system's performance was evaluated with data from 4 participants, resulting in an average RMS error of 0.46 degrees of visual angle. Moreover, the system can withstand moderate head movements before the eye features are no longer in the camera's field of view.

Hansen and Ji [39] conducted an extensive survey of models for eyes and gaze, examining various methods, including remote eye-tracking and appearance-based techniques, as well as their respective advantages and disadvantages. They found that remote eye-tracking techniques have high accuracy but require specialized hardware and careful calibration, while appearance-based techniques can be less accurate but are less intrusive and adaptable to different environments. They also highlighted the work of researchers such as Baluja and Pomerleau [40] and Beymer and Flickner [41]. Baluja and Pomerleau [40] developed a non-intrusive gaze tracking method using artificial neural networks. In their study, they used a dataset of 1,500 images of individuals wearing a small camera mounted on the side of their glasses to capture images of the eye. The dataset was recorded under various lighting conditions and environments. The inputs to their model were 30×30 pixel images of the eye, and the model was trained to estimate the x and y coordinates of the point of gaze on the computer screen. Their system achieved a mean accuracy of approximately 1.5 degrees of visual angle.

Beymer and Flickner [41] proposed an appearance-based gaze estimation technique using

gradient-based features. Their method used a dataset of 5,000 images of a person's face captured by a camera mounted on a computer monitor, with the person looking at a target point on the screen. The dataset was recorded under various head poses and gaze directions. The inputs to their model were 40×40 pixel images of the face, and the model estimated gaze direction based on facial features, particularly the eyes. The system achieved an accuracy of 4.2 degrees of visual angle for gaze estimation when the head pose was known, and 5.8 degrees when the head pose was unknown.

Despite the advancements in gaze tracking technology, traditional methods still have significant drawbacks. These methods often require specialized hardware and careful calibration, which can be expensive and time-consuming. Furthermore, their sensitivity to lighting conditions can limit their applicability in real-world scenarios, particularly in transportation research. As a result, researchers have sought alternative approaches, such as appearance-based methods and deep learning techniques, to overcome these limitations and develop more cost-effective and adaptable solutions for gaze estimation in various contexts.

Appearance-based gaze estimation methods are widely used in research due to not requiring explicit eye models. Appearance-based methods estimate gaze direction by analyzing images of the eye or face region, which may include eye corners, iris, and pupil. The performance of appearance-based methods can be influenced by many factors such as illumination, head pose, and gaze direction. To address these challenges, researchers have proposed various techniques such as using multiple cameras, infrared illumination, and machine learning algorithms.

Appearance-based methods for gaze point estimation have gained popularity due to their potential for unobtrusive monitoring, lower cost, and adaptability to different driving environments [42, 43, 44, 45]. These methods estimate gaze points by analyzing facial features, head orientation, and facial landmarks captured by cameras mounted inside the vehicle, without

the need for specialized headsets or eye-tracking devices [46]. Deep learning models, particularly convolutional neural networks (CNNs), have shown great success in appearance-based gaze point estimation tasks, as they can automatically learn hierarchical representations of input data [5, 47, 48, 49].

Lu et al. [50] proposed an adaptive linear regression (ALR) method for gaze estimation from sparsely collected training samples. Their system used a camera to observe a person's eye, extracting low-dimensional features from different resolution eye images. The dataset size is not explicitly mentioned in the paper; however, it mentions the use of sparsely collected training samples. The method performs subpixel alignment to improve accuracy and extends the optimization procedure in ALR to solve the subpixel alignment problem simultaneously for low-resolution test eye images. The paper does not provide a specific claim for accuracy of their proposed method. Instead, they evaluate the performance of their method against various factors such as the number of training samples, feature dimensionality, and eye image resolution to verify its effectiveness. They compare their results with existing methods and show that their ALR method achieves accurate mapping via sparsely collected training samples.

Zhang et al. proposed an appearance-based method for gaze estimation in [43] and later extended their work to the MPIIGaze dataset in [49]. The MPIIGaze dataset, introduced by Zhang et al. [43], is a real-world dataset that provides a large-scale collection of full-face images and corresponding ground-truth gaze positions collected from 15 users during everyday laptop use over several months. The dataset provides an unprecedented level of realism and variation in eye appearance and illumination, which is necessary to improve the generalization of gaze estimation methods. The authors note that while appearance-based gaze estimation methods have shown promise in laboratory settings, they face challenges in the unconstrained daily-life setting. To address this, they propose a spatial weights

CNN method that leverages information from the full face and demonstrates its superior performance in handling facial appearance variation caused by extreme head pose, gaze directions, and illumination compared to current eye-only and multi-region methods. The CNN-based approach proposed in this work achieves the best accuracy on both the proposed MPIIGaze and Eyediap [51] datasets, with a detection rate of 13.9 degrees on MPIIGaze and 10.5 degrees on Eyediap. This represents a significant performance gain over the state-of-the-art RF method, with a 10% improvement on MPIIGaze and a 12% improvement on Eyediap. However, it is worth noting that performance on MPIIGaze is generally worse than on the Eyediap dataset, which indicates the fundamental difficulty of the in-the-wild setting. Additionally, they compare their CNN-based approach to other models such as Random Forest, Support Vector Regression, ALR, and k-Nearest Neighbors, and show that the higher learning flexibility of the CNN contributes to the large performance gain in the cross-dataset case.

Zhang et al. [49] further build on the MPIIGaze dataset by introducing a novel approach to appearance-based gaze estimation in the wild. The authors propose a method that uses a CNN to learn the mapping from the eye images to the gaze direction. They also propose a new gaze estimation error metric, which considers the angular difference between the estimated gaze and the ground-truth gaze. The authors show that their approach outperforms the state-of-the-art methods on both the MPIIGaze dataset [43] and the Columbia Gaze dataset [52], demonstrating the effectiveness of the proposed method for unconstrained gaze estimation. They propose GazeNet, It consists of 16 layers, with 13 convolutional layers and 3 fully connected layers. The input to the network is an image patch of size 224×60 pixels, which contains both eyes and the nose region. GazeNet outperforms the previous systems by 22%, reducing mean error from 13.9 degrees to 10.8 degrees.

Xu et al. [53] developed a web-based system for crowdsourcing large-scale gaze data collec-

tion. This system relies on a webcam to track a user's gaze as they view natural images and videos. The dataset they used consisted of 1200 images and 60 videos, and it was recorded by observing the eye movements of 15 participants. The system processed the collected data using meanshift clustering to extract fixations, which were then used to evaluate gaze estimation accuracy and generate saliency maps. The inputs to the model were the webcam-captured images of the participants' eyes, and the resolution of these inputs was not explicitly mentioned in the paper. The study reported a median error of 1.06°, which is comparable to previously reported results from webcam-based algorithms, indicating high accuracy in estimating gaze and fixations.

Huang et al. [54] introduced TabletGaze, a dataset and analysis of unconstrained appearance-based gaze estimation for mobile devices. They used a camera mounted on a tablet to record the dataset, which comprised 51 participants and over 1 million gaze samples. The authors proposed a novel feature, a modified Histogram of Oriented Gradients (mHoG), which demonstrated superior performance compared to other features when used with a random forest regressor. The inputs to the model were images of participants' eyes captured by the tablet's camera, but the resolution of these inputs was not specified in the paper. The study compared the performance of different regressors, including k-Nearest Neighbors, Random Forests, Gaussian Process Regression, and Support Vector Regression. The results showed that the combination of mHoG feature and Random Forest regressor achieved the lowest mean error for gaze estimation. The algorithm fitted an ellipse to eye limbus within the region of interest (ROI) detected by eye detectors and found the optical axis through the ellipse normal vector. The optical axis was directly treated as the gaze direction. An accuracy of 6.88 degrees was claimed in the work. The authors also analyzed the influence of person-dependency on algorithm performance and found that person-dependent training resulted in lower estimation errors.

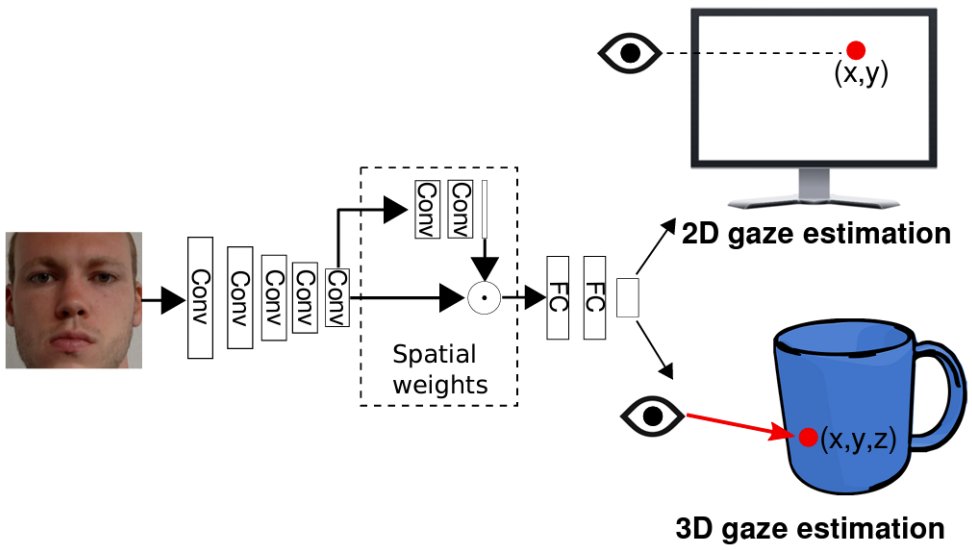


Figure 2.2: The figure shows the architecture of Zhang et al., the method takes a facial image as input and performs 2D and 3D gaze estimation using a convolutional neural network (CNN) with spatial weights applied on the feature maps. This approach represents an improvement over previous appearance-based methods, which have only utilized a limited amount of information for gaze estimation (Figure credit [3]).

Krafka et al. [47] recently proposed an end-to-end eye tracking solution for mobile devices. They introduced GazeCapture, the first large-scale mobile eye tracking dataset, which consists of over 2.5 million frames from 1,474 participants. The dataset was recorded using mobile devices' front-facing cameras, capturing face and eye images along with the ground truth gaze points on the screen. The iTracker model, a deep convolutional neural network, was trained on this dataset and takes as input images of the face, eyes, and screen, as well as the device's orientation. The iTracker achieved an error as low as 1.04 cm and 1.69 cm on mobile phones and tablets, respectively. The paper also evaluated the generalization ability of the iTracker features by applying them to another dataset, TabletGaze [54], and showed that the features significantly outperformed all previous approaches, with an error of 1.63 cm.

Zhang et al. [3] also proposed a novel gaze estimation method that only takes the full face

image as input, unlike traditional computer vision techniques. This is achieved by encoding the face image using a convolutional neural network (CNN) with spatial weights applied to the feature maps (Figure 2.2). These spatial weights enable the model to suppress or enhance information in different facial regions as needed. The method achieved an accuracy of 4.8 degrees and 6.0 degrees for person-independent 3D gaze estimation on the challenging in-the-wild MPIIGaze and EYEDIAP datasets, respectively. The authors conducted extensive evaluations and found that their full-face method significantly outperforms the previous systems in both 2D and 3D gaze estimation, achieving improvements of up to 14.3% on the MPIIGaze dataset and 27.7% on the EYEDIAP dataset for person-independent 3D gaze estimation. Their 2D gaze estimation significantly outperforms ITracker on the EYEDIAP dataset. Additionally, the improvement was consistent across various illumination conditions and gaze directions, and the proposed estimation method was particularly pronounced for the most challenging extreme head poses.

Sugano et al. [55] presented a novel approach for estimating gaze direction based on 3D eye and head pose information. They used a remote eye-tracking system with a stereo camera setup to observe a person's eyes and head pose. The dataset consisted of 50 participants, and the recorded data included high-resolution images of the eyes, head pose, and ground truth gaze points on the screen. The deep neural network took as input the eye images and head pose information and regressed the 3D gaze direction, which was then mapped to a 2D gaze point on the screen. Their method achieved an angular error of about 6 degrees, demonstrating high accuracy on multiple datasets and robustness to head pose variations. The novelty of their approach lies in accurately estimating gaze direction in a person- and head pose-independent manner, which has practical applications in areas such as human-computer interaction and virtual reality.

In earlier work, researchers often relied on specialized hardware such as eye-tracking devices,

infrared cameras, or head-mounted displays to collect gaze point training data. Examples of such devices include the Tobii 1750 Eye Tracker used by Hansen and Ji [39], the SMI RED250mobile eye tracker used by Zhang et al. [43], and the Eye Link 1000 used by Krafka et al. [47]. However, Dua et al. [4] introduced a shift from this norm by presenting DGAZE, a large-scale dataset for driver gaze estimation on the road. They employed a deep neural network architecture, I-DGAZE, trained on this dataset for driver eye gaze point estimation, marking a significant advancement in the field. The model achieved an accuracy of 186.89 pixels on road view images with a resolution of 1920×1080 pixels. Additionally, the authors proposed a calibration procedure for creating driver-specific models. Unlike previous approaches, DGAZE does not require specialized hardware; it employs a single RGB camera to estimate the gaze point, making the system more accessible, cost-effective, and easier to deploy in various environments. The dataset and models can be used to study driver behavior, road conditions that increase driver distraction, and areas of the road that drivers pay attention to.

Dua et al.'s work serves as an excellent benchmark for this thesis, as DGAZE is the only dual-camera transportation dataset with gaze point annotation. This unique dataset offers an opportunity to train and evaluate models for point-level or object-level gaze estimation in driving scenarios. By using DGAZE as a benchmark, we can compare the performance of our models with the I-DGAZE model and explore ways to improve their estimations.

2.3 Semi-supervised Learning and Self Training

Semi-supervised learning is a machine learning approach that combines a small amount of labeled data with a large amount of unlabeled data during training [21]. This approach has shown promise in various computer vision applications, including object recognition [56, 57],

image classification [27, 58, 59], and semantic segmentation [60, 61]. In the context of gaze point estimation, semi-supervised learning has the potential to reduce the need for large amounts of labeled data, which can be time-consuming and expensive to obtain.

Self-training, a specific type of semi-supervised learning, has shown promise in other areas of computer vision and natural language processing [24, 62]. By using an initial model to generate pseudo-labels for unlabeled data and then iteratively refining the model by incorporating the pseudo-labeled data into the training process, self-training can effectively leverage the information contained in the unlabeled data to improve estimation performance.

The task of estimating a 2D gaze point from face images requires CNN regression. There has not been a lot of research done in semi-supervision for CNN regression. A method for monocular depth prediction is introduced in Kuznietsov et al. [63]. The authors propose a novel deep learning approach for monocular depth map prediction using CNN regression. The CNN is used to learn a mapping between input RGB images and continuous-valued depth maps. Long skip connections are added to the CNN architecture to improve spatial information capture and fine details preservation. A semi-supervised learning approach is used to train the CNN, combining supervised and unsupervised learning signals. The proposed method achieves high performance in single image depth map prediction on the KITTI dataset [64], predicting detailed depth maps on thin and distant objects and estimating reasonable depth in areas with no ground truth available. The authors attribute the success of their approach to the combination of supervised, unsupervised, and regularization terms in the loss function, as well as the use of a deep residual network in an encoder-decoder architecture with long skip connections.

Lai et al. [65] proposed a generative adversarial network for learning optical flow in a semi-supervised manner. The method uses a CNN regression model and an adversarial loss to learn the structural patterns of the flow warp error. The proposed approach consistently out-

performs the purely supervised method and achieves competitive performance with previous methods. The experiments validate the effectiveness of the adversarial loss and demonstrate the capability of the proposed method in utilizing unlabeled data.

Semi-supervised learning techniques for gaze point estimation have not been extensively explored. However, the application of semi-supervised learning and self-training techniques to gaze point estimation can offer several benefits. These techniques can significantly reduce the requirement for large amounts of labeled data, enhance estimation accuracy by utilizing information from unlabeled data, and facilitate generalization to new scenarios by training on diverse unlabeled data. Moreover, semi-supervised learning and self-training can be promising solutions to overcome the challenges faced by traditional and deep learning-based gaze point estimation methods.

2.4 Dual Camera Datasets in Transportation Research

Dual camera datasets have become an essential resource in transportation research due to their ability to provide comprehensive and synchronized information about both the driver and the external driving environment. These datasets typically consist of videos captured from two separate cameras: one focused on the driver's face and the other on the road ahead. In this section, we discuss the significance of dual camera datasets in transportation research.

2.4.1 Importance of Dual Camera Datasets

Dual camera datasets play a critical role in transportation research [16, 17, 18], providing comprehensive insights into driver behavior and the external driving environment. These datasets allow for a more in-depth understanding of the complex interplay between drivers

and their surroundings, which is crucial for assessing and improving driving safety. The benefits of using dual camera datasets include:

1. Richer information: Dual camera datasets offer detailed information about both the driver's facial features and the driving scene, enabling researchers to analyze the relationship between the driver's gaze and external factors, such as road layout, traffic conditions, and visual distractions.

2. Synchronized data: The synchronization of the two camera views ensures accurate temporal alignment between the driver's gaze and the driving scene, allowing researchers to study real time driver reactions and decision-making processes.

3. Enhanced context: The combination of driver's facial features and the driving scene provides a more comprehensive context for understanding driver behavior, attention, and decision-making, contributing to the development of more effective driver assistance systems, road infrastructure improvements, and driver training programs.

2.5 Summary and Research Gaps

In summary, gaze point estimation is essential in transportation research for understanding driver behavior, attention, and decision-making processes. Traditional gaze point estimation techniques, such as remote eye-tracking and appearance-based methods, have limitations, including the need for specialized hardware and large amounts of labeled data. Recent advancements in deep learning-based gaze point estimation have shown promise in overcoming some of these limitations, but challenges still remain.

Dual camera datasets provide valuable information about both the driver and the external driving environment, enabling a more comprehensive analysis of driving behavior. However,

most of these datasets are recorded in naturalistic settings and do not have gaze point annotations.

The research gaps identified in this literature review include:

1. The need for semi-supervised learning and self-training techniques that can reduce the reliance on large-scale labeled data for gaze point estimation.
2. Developing a method for robust gaze point estimation models that can generalize well to different drivers.
3. A method for estimating gaze point that can be replicated on unlabelled dual camera datasets in transportation research

By addressing these research gaps, this thesis aims to contribute to the advancement of gaze point estimation techniques in transportation research.

Chapter 3

Methodology

3.1 DGAZE and I-DGAZE

3.1.1 DGAZE

Figure 3.1: Showcases DGAZE, a dataset designed to capture driver gaze on the road by incorporating both driver and road view through the use of inexpensive mobile phone cameras (Figure credit: [4]).

Determining gaze points is a critical aspect of understanding human visual behavior, particularly in driving scenarios. There is a demand for datasets with precise gaze point annotations to train and evaluate machine learning models specifically tailored to transportation research. The DGAZE dataset (Figure 3.1) is a unique and valuable resource for transportation researchers and machine learning practitioners, offering a large-scale dataset for driver gaze

mapping on the road. In Figure 3.1, the driver is seated on the right side because the dataset was recorded in India, where drivers sit on the right-hand side.

The DGAZE dataset is collected in a lab designed to simulate real driving conditions. Participants, referred to as "drivers", sit in front of a car interior backdrop while a 1920×1080 road video is projected in front of them. The front-facing video is captured using dashboard-mounted cameras in cars driven at various times of the day and under different traffic conditions. A mobile phone mounted on a tripod is placed in front of the driver at a height and distance similar to a dashboard-mounted phone, using both front and rear cameras to simultaneously record the driver and the projected video at the same frame rate. The dataset includes only the second recording of the front-facing video, and not the original, even though both have the same frame rate.

Rather than employing costly eye-gaze trackers, the dataset annotates a single object on each frame of the projected video with a bounding box, marking its center. Drivers are instructed to look at the center of each object, enabling the dataset to be used for estimating gaze at a point or object level.

The projected video has special calibration frames at the beginning and end to synchronize the projected video with the videos taken by the mobile phone. Frame alignment is achieved by carefully dropping frames of the longer video. The objective is to establish a realistic setup for gaze estimation without relying on expensive eye-tracking equipment.

The DGAZE dataset comprises 227,178 images and includes 18-minute videos of 20 drivers aged 20-30, captured using mobile phones mounted on car dashboards in urban environments. These videos were combined to create a single 18-minute video with varied lighting conditions, captured from morning to evening on actual roads.

The dataset is meticulously annotated for the driver's gaze point. For each frame, one of

seven annotated objects (car, bus, motorbike, pedestrian, autorickshaw, traffic signal, and signboard) is marked, along with the center of each bounding box, serving as the ground truth for the gaze point estimation task. The dataset is not considered naturalistic because it may not fully represent real-world, everyday situations that drivers encounter, since it is recorded in a simulated setting (the participants were shown the videos obtained in a real environment). Nonetheless, it remains a valuable resource for evaluating gaze point estimation models, particularly in the absence of large-scale naturalistic datasets with gaze point annotations.

3.1.2 I-DGAZE Model

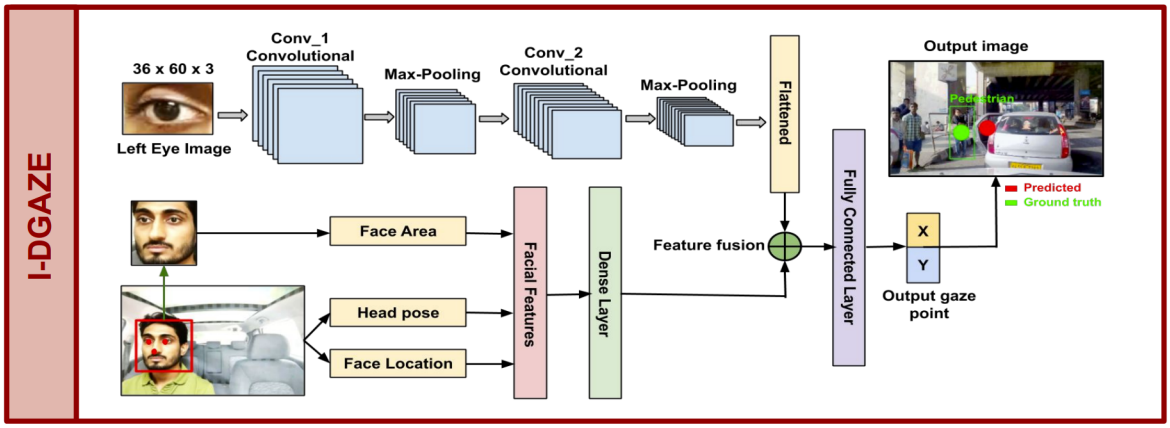


Figure 3.2: I-DGAZE is a two-branch late fusion convolutional neural network architecture aimed at estimating driver gaze on the road. One of the branches receives an eye image as input, while the other branch takes input of facial features such as head pose, face location, and the distance between the driver's face and the mobile phone camera. The overall goal of I-DGAZE is to accurately estimate where the driver is looking while on the road (Figure credit: [4]).

In this section, we discuss the architecture of I-DGAZE, a gaze estimation model that forms the foundation of our research. I-DGAZE leverages the DGAZE dataset to estimate driver eye gaze points considering various factors such as head position, angle, and eye gaze direction. Figure 3.2 provides a visual overview of the I-DGAZE architecture.

The I-DGAZE architecture consists of a two-branch late fusion network: the Feature Branch and the Eye Branch. The Feature Branch focuses on facial features such as face location, head pose, and pupil location. The head pose is detected using the model from Patacchiola et al. [66] and face location and pupil are detected using DLIB [67]. The resulting bounding box, facial landmarks, and pupil locations help encode the face location using the coordinates of the face bounding box and the nose position. The final input vector for the Feature Branch consists of ten elements, including face area, roll, pitch, yaw of the head pose, and the (x, y) coordinates of both pupils and the nose.

Concurrently, the Eye Branch processes a cropped image of the left eye, which is obtained using a facial key landmark detection algorithm DLIB [67]. The algorithm identifies the position of landmarks of the eyes in larger images, which are subsequently cropped and resized to a uniform size of 36×60 pixels.

These two branches merge into a common branch with 4566 dimensions and are processed by a fully connected layer with an output size of 500. The final 2-dimensional output vector represents the driver's gaze point on the road view.

The model is trained using mean absolute error as the loss function and the Adam optimizer [68] with a learning rate of 1e-3 and weight decay of 1e-5. After training for 60 epochs with a batch size of 32, a calibration step is performed for each driver to create driver-specific models, significantly reducing model error.

The significance of various inputs to the I-DGAZE model is explored through an ablation study. The results reveal that facial features provide valuable information regarding the overall face position in the frame, crucial for precise gaze estimation. Furthermore, using both left and right eye images, along with facial features, does not significantly improve gaze estimation, as the movements of both eyes are correlated and the facial features already

include both pupils' position.

To conclude, the I-DGAZE model effectively estimates driver eye gaze points on the road using the DGAZE dataset. It considers the left eye image and additional facial features, achieving an accuracy of 186.89 pixels on the road view with a resolution of 1920×1080 pixels without calibration, and 182.67 pixels with calibration.

3.2 PoG Algorithm

3.2.1 Overview

The PoG Algorithm [69] is a method developed by researchers at Virginia Tech Transportation Institute (VTTI) for estimating the gaze point in the road-facing view by utilizing geometric transformations. The main aim of the algorithm is to determine the driver's gaze location by making specific assumptions and employing geometric transformations.

In this study, the HPV and SHRP2 datasets [70] were used. The HPV dataset was created by VTTI for research purposes with fewer privacy constraints and is available for use by researchers with a valid IRB Training Certificate. The SHRP2 NDS dataset is the largest publicly available dataset of real-world driving data. It includes various information about vehicles and drivers, such as kinematic data, positions, and relative speeds of surrounding vehicles captured by radar, and the drivers' gaze locations, all annotated by humans.

The SHRP2 dataset is divided into two categories: crash and near-crash (CNC) events and baseline (BL) driving events, and all are annotated with information about drivers' gaze locations. The study analyzed 666 CNC and 446 BL events, with the criteria that the driver's gaze was fixed for 2-5 seconds. A manual annotation process was conducted for the selected events and 617 valid CNC and 410 valid BL events were used. Among those relevant

Table 3.1: Distribution of Fixated Object Events in the Dataset

Filtered Fixated Object Classes	Distribution
car (car, SUV, van, and pickup truck)	411
person	62
traffic light	21
truck	18
bicycle	6
bus	6
motorcycle	2
stop sign	1

object classes were filtered and used for evaluation (Table: 3.1).

The PoG Algorithm establishes a connection between the driver-facing and road-facing cameras, with the PoG being defined as a 2D pixel point on the road-facing view, representing the driver's gaze location. The algorithm takes into input gaze angles from the driver's face (the angles are obtained using L2CS-Net pre-trained model [45]) and determines the gaze point in the manner described below.

The algorithm relies on two primary assumptions to estimate the driver's gaze location in the road-facing view. The first assumption is that for the majority of a driving trip, the driver looks straight ahead. This simplifies the problem by providing an initial reference point for the gaze point, based on the observation that drivers typically focus on the road ahead during most of their driving time. The second assumption is that the forward-facing angles provide the necessary information to map the gaze angles from the driver-facing view to the road-facing view. This allows the algorithm to compute the gaze location by rotating the origin point in 3D space, which are represented by the driver's gaze location when looking straight ahead.

To estimate the gaze point, the algorithm requires the direction of the driver's gaze (obtained using L2CS-Net), which is represented by two angles, θ and ϕ. The algorithm goes through a

driving trip video and records (θ_{fc}, ϕ_{fc}) for all the frames. According to the first assumption, the peak of a histogram distribution of the collected gaze angles would give us the forward angles, denoted as (θ_F, ϕ_F).

The algorithm estimates the 3D origin point of the driver's gaze in pixel coordinates (X_0, Y_0, Z_0). Although there exists a z dimension to the rotation, we do not consider it because we are working with a 2D image. This origin point represents the gaze point in 3D when the driver looks straight ahead. To account for variations in the gaze angles, the origin point is rotated according to the gaze angles extracted from the driver-facing video frames. To determine the overall rotation that captures the gaze shift from the origin gaze point, the algorithm generates a rotation matrix that corresponds to the obtained (θ_{fc}, ϕ_{fc}) rotations, denoted as $R_{current}$, and another rotation matrix corresponding to the forward angles (θ_F, ϕ_F), denoted as $R_{forward}$. $R_{current}$ is calculated for each pair of (θ_{fc}, ϕ_{fc}), while $R_{forward}$ is calculated only once.

$$R_{current} = R_x(\theta_{fc}) \cdot R_y(\varphi_{fc}) \tag{3.1}$$

where R_x and R_y are the rotation matrices.

Similarly, $R_{forward}$ is calculated as follows:

$$R_{forward} = R_x(\theta_F) \cdot R_y(\varphi_F) \tag{3.2}$$

$R_{rotated}$ is then obtained by multiplying $R_{current}$ by the transpose of $R_{forward}$:

$$R_{rotated} = R_{current} \cdot (R_{forward})^T \tag{3.3}$$

Once we have $R_{rotated}$, we can estimate the gaze point by rotating the origin gaze point with respect to $R_{rotated}$ to generate a gaze point estimate for a given pair of (θ_{fc}, ϕ_{fc}). The 3D rotated gaze point (X_c, Y_c, Z_c) is obtained using the following equation.

$$(X_c, Y_c, Z_c) = R_{rotated} \cdot (X_0, Y_0, Z_0) \tag{3.4}$$

Since we are projecting on a 2D image, (X_c, Y_c) are the pixel coordinates used to mark gaze point estimation in the road-facing view where the driver's gaze is focused.

In summary, the PoG Algorithm estimates the driver's gaze point by rotating the origin point in 3D space. This rotation allows the algorithm to determine the new location of the gaze point based on the change in gaze angles. The resulting rotated point represents the estimated gaze location for those specific gaze angles. By repeating this process for different gaze angles, the algorithm can estimate the driver's gaze point on the road-facing view.

3.2.2 Evaluation and Use case

The PoG Algorithm [69] takes the input of gaze angles and Detectron2's panoptic segmentation model [71] in order to get gaze point and fixated object estimation (Figure 3.3).

The evaluation process involves cropping events from the annotated videos, determining the forward angle, estimating the fixated object, and evaluating the accuracy of the algorithm. Two forward angle settings are used for the evaluation: Generalized Estimation and Customized Estimation. In the Generalized Estimation setting, all the gaze angles from the dataset of cropped videos are combined, and a histogram plot is generated. The peak of the histogram represents the most frequent angle the driver's gaze refers to and is used as the common forward angle. In the Customized Estimation setting, histogram analysis is done

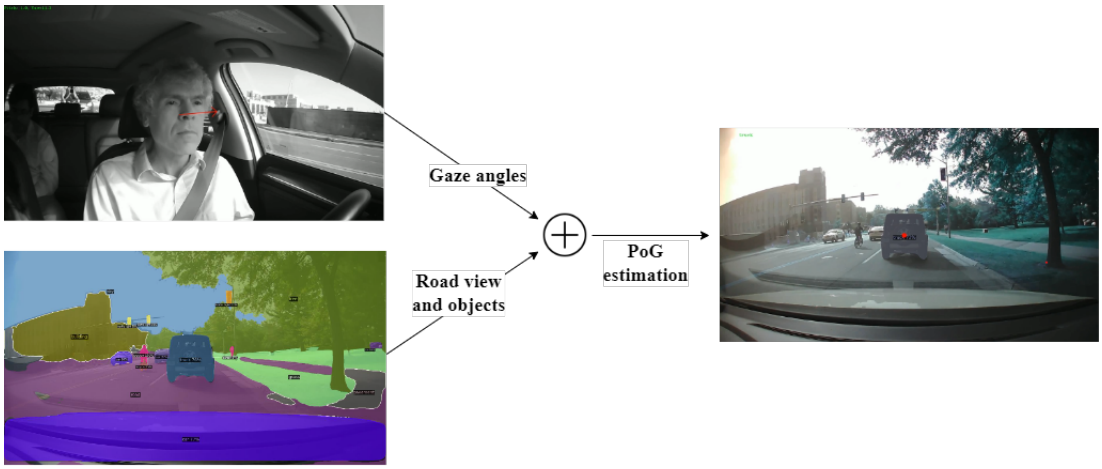


Figure 3.3: PoG Algorithm takes the gaze angle extracted from the driver's face image using L2CS-Net and segmented objects to estimate a point and fixated object on the road image.

separately for each video, and the peak of the histogram generated for a particular video is used as the forward angle for the corresponding video.

The accuracy of the algorithm is evaluated by dividing correct estimations by total estimations using the following formula.

$$accuracy\,(y, \overline{y}) = \frac{1}{n}\sum_{i=0}^{n-1} 1\,(\overline{y_i} \;=\; y_i) \tag{3.5}$$

The results show that the algorithm performs well, achieving an accuracy of 93.58% in the Generalized Estimation setting and 95.37% in the Customized Estimation setting with an improvement of 1.79%. The results of the CNC+baseline model evaluation show positive results for several class categories, with successful detection of car and person with accuracy rates of over 100% and 80%, respectively.

In our thesis, we rely solely on gaze point annotations rather than object-level annotations. The driver's face images are cropped from the DGAZE dataset. Then L2CS-Net is used to obtain gaze angles. These gaze angles are used as input to the PoG Algorithm and are then

utilized to estimate the gaze point, which is annotated for experiments where necessary (for seed labels or sampling).

3.3 Proposed Methodology

In this study, we center our efforts on the task of gaze point estimation, which is essential to transportation research for understanding driver behavior, attention, and decision-making. We focus on this problem in order to illuminate the areas where the driver is looking at any given time.

To accomplish this task, our system is designed to process the face image of the driver, the input for our model. We utilize DLIB [67], a software library that provides tools for identifying and locating human faces in digital images. It allows us to detect the driver's face, generate a bounding box around it, locate key facial landmarks, and pinpoint pupil locations. However, we only utilize the bounding box in order to get the face crop image.

We then utilize a regression-based Convolutional Neural Network (CNN) to process these inputs. The input to the CNN is a cropped face image from the driver-facing image and the output of the CNN is the (x, y) coordinates of the gaze point location on the forward-facing image. Our goal is to derive highly accurate gaze point estimations, taking advantage of the minimal amount of labeled data available. We further optimize our model's performance by implementing self-training and weighted semi-supervised learning techniques.

Ultimately, our system strives to address the identified gaps in the literature, including the need for a robust gaze point estimation model that can generalize well across different drivers and can be effectively deployed on unlabeled dual camera datasets.

To meet these research objectives and address the gaps identified in the literature, we propose

the following methodology.

3.3.1 Semi-Supervised Learning and Self-Training for Gaze Point Estimation

We have developed a semi-supervised learning and self training framework for gaze point estimation in transportation research. The framework is described in this subsection.

1) Data Preprocessing and Feature Extraction

SHRP2 primarily focuses on improving highway safety, renewal, reliability, and capacity by gathering detailed naturalistic driving data, which aids researchers in developing innovative safety countermeasures and enhancing transportation infrastructure. On the other hand, the DGAZE dataset is a large-scale dataset specifically designed for gaze estimation.

In our thesis, we utilize DGAZE for training and evaluation. However, both the SHRP2 and DGAZE datasets incorporate dual camera views and fixation annotations. We would like to note that the SHRP2 dataset is not open source and its usage is restricted to researchers with the appropriate license agreement. In contrast, the DGAZE dataset is openly available for contribution to the development of gaze estimation and human attention technologies. This thesis aims to develop an effective way to get gaze point estimations of dual camera transportation datasets like SHRP2 and DGAZE (without using a large amount of labeled data) which could aid transportation researchers to incorporate gaze information when developing innovative safety countermeasures and enhancing transportation infrastructure.

For our data preprocessing step, we start by extracting all the frames from the videos in the DGAZE dataset. Then, we use RetinaFace [72] to perform face recognition and save the

cropped faces. The dataset is divided into four parts, as shown in Table 3.2.

Table 3.2: Number of face images in the DAZE dataset is divided into 4 sets. set 1, 2 & 3 are created randomly while the unseen drivers dataset has 3 drivers strategically separated out to later use for evaluation.

Dataset Name	No. of the driver face images
Set 1	83,118
Set 2	83,118
Set 3	34,000
Unseen drivers dataset	26,942
DGAZE (total)	227,178

Sets 1 and 2 serve as unlabeled datasets and are employed for pseudo-labeling or sampling purposes. Set 3 comprises a smaller ground truth dataset, which includes labeled data points. The DGAZE dataset consists of 20 drivers, and to assess the model's generalizability, we completely separate three drivers for the unseen driver's dataset. This allows us to evaluate the model's performance on previously unencountered drivers.

To prepare the cropped face images for our deep learning model, we apply a standard method of normalization by dividing the pixel values by 255, resulting in pixel values ranging from 0 to 1. This is a common technique in deep learning. The input of the model is cropped face images, resized to 300×300 pixels and the output is the x and y coordinates of the gaze points in the forward looking image.

2) Self Training Approach

Set 1 is used to train a small ground truth model (GT model). The GT model is only used to generate seed labels (pixel coordinate of the gaze point) for the self training task; it takes a face image and generates a 2D gaze point estimation. We use the GT model to generate pseudo labels for the unlabeled data. In this manner, we can use the unlabeled data to train a better model that has access to more data points and information. The self-trained

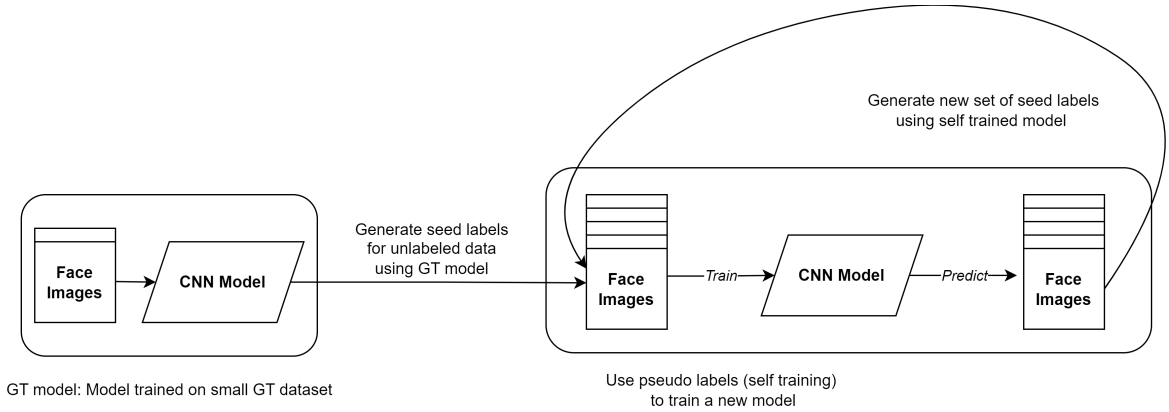


Figure 3.4: Proposed self-training approach.

model is used iteratively to generate pseudo labels for the unlabeled data (as observed from Figure 3.4). As CNN regression does not generate confidence scores, there are techniques like Monte Carlo [73] uncertainty estimation that can be used for refining estimations for regression tasks, but they do not work effectively for gaze point estimation. Therefore, we iteratively self-train for 10 epochs and observe the performance of each self-trained model separately.

This experiment is essential as it provides insight into the performance without refinement and allows us to estimate the future scope of gaze point estimation using self training.

3) Weighted Semi-Supervised Learning Approach

To further enhance the performance of our gaze point estimation model, we implement a weighted semi-supervised learning approach. In this approach, we use both labeled and unlabeled data to train our model. The labeled data is used with their ground truth labels, while the unlabeled data is assigned pseudo-labels generated by the GT model.

During training, we assign weights to the loss function based on whether the data is labeled or unlabeled. The labeled data is given a higher weight, while the unlabeled data is given

a lower weight. This allows the model to learn from both the labeled and unlabeled data, while giving more importance to the ground truth labels. The weights can be adjusted based on the confidence of the pseudo-labels or the desired trade-off between using labeled and unlabeled data.

3.4 Expected Outcomes and Contributions

By addressing the research gaps and implementing the proposed methodology, we expect the following outcomes and contributions:

1. A semi-supervised learning and self training framework for gaze point estimation that reduces the reliance on large-scale labeled data and improves estimation performance.
2. A robust gaze point estimation model that generalizes well to different drivers and driving conditions.
3. A replicable method for estimating gaze points on unlabelled dual camera datasets in transportation research, facilitating the analysis of driver behavior, attention, and decision-making processes in various driving scenarios.
4. Enhanced understanding of the relationship between driver gaze and external driving factors, contributing to the development of more effective driver assistance systems, road infrastructure improvements, and driver training programs.

Chapter 4

Experiment Setup

4.1 Model Architecture and Training

This section provides a detailed overview of the training process for three types of gaze estimation models: ST-GP, ST-GP (NS), and WS-GP. All three models employ a base CNN regression architecture as the foundation for training.

4.1.1 Base CNN Regression Architecture

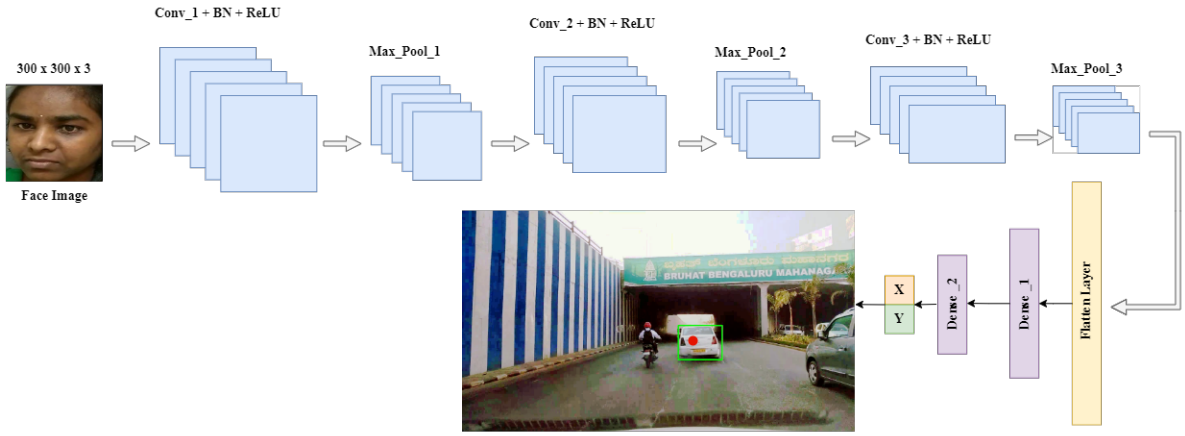


Figure 4.1: The CNN Regression Architecture is designed to estimate the gaze of drivers while on the road. It is a modified version of I-DGAZE (Figure 3.2) that uses cropped face images to estimate x and y coordinates. In this architecture, the red dot represents the output generated by the CNN (pixel coordinates of the predicted gaze point). The accompanying image, along with its bounding box, is sourced from the DGAZE dataset and is included here solely for illustrative purposes.

This section focuses on the base gaze estimation model, which is inspired by the I-DGAZE architecture and adapted to meet our specific requirements. The model performs the task of regression using CNN architecture, this module is where the training will occur (Figure 4.1).

The proposed model does not take into account any segmentation inputs or features of different modalities. It is specifically designed to analyze single images, as opposed to sequences of images. The system's output is an (x, y) location in pixel units, relative to a 1920×1080 resolution display. The display's placement in DGAZE mimicked real-world driving conditions, the exact measurements are not known. Likewise, the relationship between pixels and angular distances was not determined due to variable factors like viewing distance and display size. Hence, while the model predicts gaze points within the visual scene, it doesn't provide precise physical or angular gaze locations.

We do not analyze the distribution of (x, y) gaze points in the dataset as it may not directly provide insight into the model's performance. This is because the model learns the estimation of (x, y) gaze directions separately and optimizes its loss function using both dimensions together. Therefore, the focus is on minimizing the discrepancy between the predicted gaze directions and the ground truth values, rather than analyzing the distribution of gaze points in the dataset.

The analysis is conducted through multiple convolutional layers, combined with batch normalization, L2 regularization, and dropout layers, processing each input image individually. The detailed architecture can be found in Table 4.1.

The modifications from I-DGAZE are as follows:

1. Input data: Our model uses cropped face images as input, while I-DGAZE uses left eye images and facial features.

Table 4.1: CNN architecture and output shapes

Layer	Kernel Size	Output Size	Layer	Kernel Size	Output Size
Conv2d_1 (BN+ReLu)	(3, 3)	(300, 300, 64)	Dropout_3	-	(37, 37, 256)
Max_Pooling2d_1	(2, 2)	(150, 150, 64)	Flatten_1	-	(350464,)
Dropout_1	-	(150, 150, 64)	Dense_1 (BN+ReLu)	-	(512,)
Conv2d_2 (BN+ReLu)	(3, 3)	(150, 150, 128)	Dropout_4	-	(512,)
Max_Pooling2d_2	(2, 2)	(75, 75, 128)	Dense_2 (BN+ReLu)	-	(256,)
Dropout_2	-	(75, 75, 128)	Dropout_5	-	(256,)
Conv2d_3 (BN+ReLu)	(3, 3)	(75, 75, 256)	Dense_3	-	(2,)
Max_Pooling2d_3	(2, 2)	(37, 37, 256)			

2. CNN architecture: Our model utilizes a distinct architecture, featuring extra convolutional layers and larger fully connected layers (512 and 256 units) in contrast to the I-DGAZE model, which consists of two convolution layers and a single fully connected layer (500 units). This design enables our model to efficiently process the increased resolution of facial images.

3. Dropout rates: Our model uses a 35% dropout rate while I-DGAZE does not use dropout in the fully connected layers.

4.1.2 Optimizer and Loss Function

The model is compiled using the Adam optimizer [68], with a mean absolute error (MAE) loss function. The MAE loss is calculated separately for x and y coordinates and averaged per batch of n data points. The equation for MAE calculation are given below.

MAE for x-coordinate:

$$MAE_x = \frac{1}{n}\sum_{i=1}^{n}|x_i - \hat{x}_i| \tag{4.1}$$

MAE for y-coordinate:

$$MAE_y = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4.2}$$

Overall loss (MAE for both coordinates):

$$Loss = \frac{1}{n}\sum_{i=1}^{n}(|x_i - \hat{x}_i| + |y_i - \hat{y}_i|) \tag{4.3}$$

where:

n is the total number of data points.

x_i is the true value of the x-coordinate for the i^{th} data point.

$\hat{x}_i$ is the estimated value of the x-coordinate for the i^{th} data point.

y_i is the true value of the y-coordinate for the i^{th} data point.

$\hat{y}_i$ is the estimated value of the y-coordinate for the i^{th} data point.

4.1.3 Learning Rate Scheduler

The learning rate is initialized at 0.001 and scheduled to decrease by a factor of 0.5 every five epochs. The model is trained for a different set of epochs and later compared on basis of impact on accuracy.

$$LR(e) = LR_0 \times d^{\left\lfloor \frac{e+1}{E_d} \right\rfloor} \tag{4.4}$$

Let $LR(e)$ represent the learning rate at epoch e, LR_0 be the initial learning rate, d be the drop factor, and E_d be the number of epochs after which the learning rate drops. Initial

values to calculate $LR(e)$: $LR_0 = 0.001$, $d = 0.5$, and $E_d = 5$.

We adopt a step decay learning rate scheduler because it helps the model adapt its learning rate during training. Initially, the model starts with a higher learning rate to enable faster convergence. As training progresses and the model gets closer to the optimal solution, the learning rate is gradually reduced, allowing the model to fine-tune its parameters with smaller updates. This approach helps prevent overshooting the optimal solution and can lead to better model performance.

4.1.4 ST-GP: Self Training Based Gaze Point Estimation

We propose a self-learning approach, named ST-GP, to train a model for gaze point estimation. In this approach, we utilize three datasets, Set 1, Set 2, and Set 3. Set 3 is used to train an initial ground truth model (GT model), which is subsequently employed to pseudo-label Set 1. We also incorporate random sampling to make the model more robust to overfitting. The overall architecture and training process of ST-GP is illustrated in the algorithm presented in Algorithm 1.

Random Sampling

In order to enhance the diversity of the training data and prevent overfitting, we employ random sampling in the training process of the ST-GP model. Specifically, we label Set 2 using the PoG Algorithm discussed in the proposed methodology (Section 3.2). During each iteration, we randomly sample 30% of Set 2 and add it to Set 1 before training commences. This sampling strategy ensures that our model remains robust even when trained on a relatively small dataset.

Different Epoch Lengths

We train the ST-GP model for 10 iterations using three different epoch lengths per model training: 25, 50, and 75. This experiment helps us observe the impact of varying convergence rates on the model's performance.

Algorithm 1 Self training for gaze point estimation with sampling

```
Load dataset Set 1, Set 2, and Set 3
Update Set 2 with estimations from PoG Algorithm
for itr = 1 to 10 do
    if itr == 1 then
        Replace Set 1 with Set 3
    end if
    if itr > 1 then
        Update Set 1 with estimations from the previous model
        Sample 30% of Set 2 and add it to Set 1
    end if
    Perform data preprocessing on Set 1
    Define and compile the CNN model
    Define the learning rate scheduler with step decay
    Train the model on Set 1 using the learning rate scheduler
end for
```

4.1.5 ST-GP (NS): Self Training Based Gaze Point Estimation Without Sampling

To investigate the effect of sampling on the model's performance, we train another version of the ST-GP model, called ST-GP (NS), where NS refers to "no sampling". In this approach, we combine Set 2 and Set 1 and pseudo-label all unlabeled data before training. The architecture of ST-GP (NS) is depicted in the detailed algorithm presented in Algorithm 2. By comparing the performance of ST-GP (NS) with the original ST-GP, we can evaluate the impact of the sampling strategy on the overall performance of the model.

Algorithm 2 Self training for gaze point estimation without sampling

```
Load dataset Set 1, Set 2, and Set 3
for itr = 1 to 10 do
    if itr == 1 then
        Update Set 1 with Set 3 data
    end if
    if itr > 1 then
        Add Set 2 to Set 1
        Update Set 1 with estimations from the previous model
    end if
    Perform data preprocessing on Set 1
    Define and compile the CNN model
    Define the learning rate scheduler with step decay
    Train the model on Set 1 using the learning rate scheduler
end for
```

4.1.6 WS-GP: Weighted Semi-supervised Learning Based Gaze Point Estimation Model

In this section, we propose a semi-supervised learning approach for gaze point estimation, referred to as WS-GP. The model combines labeled and unlabeled data with assigned weights to each type of data point during training. In our experiments, we assign a weight of 0.9 to the labeled data ($w_{labeled} = 0.9$) and a weight of 0.1 to the unlabeled data ($w_{unlabeled} = 0.1$). The high-level architecture is shown in the detailed algorithm presented in Algorithm 3.

Set 1 contains unlabeled data, and Set 3 contains labeled data. Set 2 is first used to update the unlabeled data in Set 1 using the PoG Algorithm (Section 3.2) estimations. Then, Set 3 is combined with the updated Set 1 to create a new dataset for training the model.

1. Custom Data Generator:

The custom data generator is used to preprocess and generate batches of training data for the WS-GP model. It allows for data augmentation and assigns weights to labeled and unlabeled samples, ensuring that the model learns from both types of data. This approach

Algorithm 3 Weighted semi-supervised learning for gaze point estimation

1: Load Set 1, Set 2 and Set 3
2: Add Set 2 to Set 1
3: Update Set 1 with PoG Algorithm estimations
4: Set Set 1 samples to unlabeled and Set 3 samples to labeled
5: Add Set 3 to Set 1
6: Set $w_{labeled}$ and $w_{unlabeled}$
7: Create a custom data generator for training and validation with $w_{labeled}$ and $w_{unlabeled}$
8: Define weighted MAE loss function and loss wrapper
9: Define learning rate scheduler
10: Train the model using the custom generator and weighted loss function.

helps improve model performance and prevent overfitting.

The custom data generator is implemented as a function named *custom_generator*, which takes in several parameters:

- `dataframe`: The dataframe containing information about the images, labels, and a flag to indicate if the data point is labeled or not.
- `image_data_generator`: An instance of Keras ImageDataGenerator for data augmentation.
- `labeled_weight`: The weight assigned to labeled samples.
- `unlabeled_weight`: The weight assigned to unlabeled samples.
- `sample_weight_holder`: A class instance to store sample weights.
- `batch_size`: The number of samples to generate per batch.
- `target_size`: The target dimensions (width, height) of the images.

The generator function first randomly samples a batch of data from the input dataframe. It then preprocesses the images, extracts the labels, and assigns appropriate weights to each

sample in the batch depending on whether the sample is labeled or not. Finally, the generator yields the preprocessed images, labels, and sample weights.

2. Custom Loss Function:

The custom loss function is a weighted MAE loss function, which takes into account the sample weights. It is implemented as a nested function called *weighted_mae_loss*:

$$L(x, \hat{x}, y, \hat{y}, w) = \frac{1}{n} \sum_{i=1}^{n} w_i \left(|x_i - \hat{x}_i| + |y_i - \hat{y}_i| \right) \tag{4.5}$$

Where:

- L is the custom loss function.
- x_i and y_i are the ground truth gaze coordinates in the x and y directions, respectively.
- $\hat{x}_i$ and $\hat{y}_i$ are the estimatedted gaze coordinates in the x and y directions, respectively.
- w_i is the sample weight for the i-th sample.
- n is the number of samples in the batch.

The custom loss function is then wrapped in another function, *weighted_mae_loss_wrapper*, to incorporate the `sample_weight_holder` instance.

The model is then compiled using the custom loss function and trained using the custom data generator. By incorporating different weights for the labeled and unlabeled data, the WS-GP approach effectively leverages the information from both types of data.

Chapter 5

Results

This section evaluates the performance of the proposed models on the Set 3 test dataset and the unseen drivers dataset, and compares them with traditional gaze point estimation techniques. Ablation studies are conducted to gain insights into the training MAE loss and epoch variation for the ST-GP model. To balance the trade-off between model performance and computational cost, 20 self training iterations were initially considered. However, increasing the number of iterations could lead to better model performance but would also increase the time and resources required for training. After careful consideration, 10 iterations were selected as a reasonable number to achieve good results without incurring excessive computational overhead. It was observed that as the number of self training iterations increased, the MAE also increased, leading us to decide not to pursue a higher number of iterations.

5.1 Experimental Results and Analysis

5.1.1 Evaluation on Set 3 Test Dataset

Since set 3 is not used directly for training any of our models (we only use the GT model for pseudo-labeling), we use it as our test dataset.

The evaluation results of the models on the Set 3 test dataset are presented in Table 5.1. ST-GP (NS 50) outperforms the other models in terms of MAE across all iterations with

79.14 pixels MAE. In comparison to ST-GP (25) and ST-GP (75), ST-GP (50) demonstrates a more stable performance with less variation in MAE across different iterations. The WS-GP (50) model has the highest MAE, indicating that it may not generalize well to the test dataset.

The higher MAE exhibited by the WS-GP (50) model, as compared to self training techniques, can be attributed to a combination of factors that lead to poorer generalization:

1. **Data utilization**: Semi-supervised learning may not use labeled and unlabeled data as effectively as self training techniques, which iteratively refine the model using its own estimations as additional labeled data for better generalization.

2. **Robustness**: The WS-GP (50) model lacks the benefits of random sampling and varying epoch lengths found in self training techniques, which improve robustness by exposing the model to diverse data samples and aiding generalization.

3. **Noisy seed labels**: Seed labels from the PoG Algorithm might impede the WS-GP (50) model's learning from the GT data. In contrast, self training techniques like ST-GP iteratively enhance the model's performance by updating estimations, generating more accurate and less noisy pseudo-labels.

A comparative analysis between self training and semi-supervised learning models reveals that semi-supervised learning does not utilize labeled and unlabeled data as effectively as self training techniques. Additionally, the WS-GP model lacks the benefits of random sampling and varying epoch lengths found in self training techniques, which improve robustness by exposing the model to diverse data samples and aiding generalization. The use of noisy seed labels from the PoG algorithm can also impede the WS-GP model's learning from the ground truth data.

Table 5.1: This table shows the comparison of pixel MAE of the proposed models on the test dataset with respect to 10 self training iterations. The models evaluated are ST-GP (25), ST-GP (50), ST-GP (75), ST-GP (NS 50), and WS-GP (50). The key takeaway from the table is that the ST-GP (NS 50) model outperforms all other models in terms of MAE, achieving the lowest MAE of 79.14 pixels at the fourth iteration.

Model	ITR1	ITR2	ITR3	ITR4	ITR5	ITR6	ITR7	ITR8	ITR9	ITR10
ST-GP (25)	124.18	123.61	115.66	109.35	131.32	138.05	141.98	138.141	141.98	145.92
ST-GP (50)	92.86	91.75	102.49	82.98	110.75	117.33	125.10	130.59	136.51	142.08
ST-GP (75)	99.51	98.33	97.40	89.27	105.37	105.37	108.14	118.38	118.41	127.62
ST-GP (NS 50)	97.87	91.29	94.62	**79.14**	94.62	98.10	104.10	104.30	110.56	107.30
WS-GP (50)	154.80	-	-	-	-	-	-	-	-	-

5.1.2 Evaluation on Unseen Drivers Dataset

Table 5.2 presents the evaluation results of the models on the unseen drivers dataset. In contrast to the Set 3 test dataset results, the WS-GP (50) demonstrates the best performance among all models, with the lowest MAE value of 149.79 pixels. ST-GP (75), ST-GP (25), and ST-GP (50) show competitive performance but have slightly higher MAE values. The ST-GP (NS 50) model performs better on the unseen drivers dataset compared to the test dataset but still falls short compared to the WS-GP (50).

Table 5.2: The table presents the comparison of pixel MAE for different proposed models on unseen drivers dataset with respect to 10 self training iterations. The ST-GP model with 25, 50, and 75 epochs and random sampling are compared with ST-GP without random sampling and WS-GP models. The table illustrates that the WS-GP (50) model achieves the lowest pixel MAE on the unseen drivers dataset, outperforming ST-GP models with varying epochs and random sampling as well as the ST-GP (NS 50) model without random sampling.

Model	ITR1	ITR2	ITR3	ITR4	ITR5	ITR6	ITR7	ITR8	ITR9	ITR10
ST-GP (25)	225.91	175.01	178.12	168.09	171.52	175.30	174.65	174.73	173.20	171.44
ST-GP (50)	193.85	176.65	169.30	160.09	172.17	171.03	177.47	178.01	179.72	175.01
ST-GP (75)	193.67	184.42	167.32	166.12	175.78	171.07.	169.86	173.52	178.77	177.78
ST-GP (NS 50)	166.73	165.31	162.35	162.13	162.21	163.79	164.69	168.10	171.02	169.11
WS-GP (50)	**149.79**	-	-	-	-	-	-	-	-	-

5.1.3 Comparative Analysis with Traditional Gaze Point Estimation Techniques

Table 5.3 compares the performance of the proposed ST-GP model with traditional gaze point estimation techniques. The ST-GP model achieves lower error values on training and test datasets compared to TurkerGaze [53] and I-DGAZE [4], both with and without calibration. This indicates that the ST-GP model provides better gaze estimation accuracy. The ST-GP model also outperforms MPIIGaze [49] and iTracker [47] in terms of validation and test errors, demonstrating its robustness and effectiveness in gaze estimation tasks. It is important to note that the table presented shows the reported results from Dua et al. [4], rather than results computed by us.

Table 5.3: The table provides a comparison of performance metrics for different gaze estimation methods. The methods included in the table are TurkerGaze, MPIIGaze, iTracker, I-DGAZE, ST-GP (50), ST-GP (NS 50), and WS-GP (50). The table reports training errors, validation errors, and test errors for each method, illustrating that ST-GP (NS 50) achieves a notable performance with a test error of 79.14 pixels, outperforming other methods in this comparison.

Method	Train Error	Val Error	Test Error	description
TurkerGaze[38]	171.30	176.37	190.71	(Pixelfeatures+RidgeRegression)
MPIIGaze[7]	144.32	229.0	189.63	(CNN+headpose)
iTracker[8]	140.10	205.65	190.50	(fc1of iTracker[8]+SVR)
I-DGAZE	133.34	204.77	186.89	(CNN+FacialFeatures)
I-DGAZE (calibration)	133.34	187.18	182.67	(CNN+FacialFeatures)
ST-GP (50)	79.22	-	82.96 & 160.09	(Self Training + Sampling)
ST-GP (NS 50)	56.39	-	**79.14** & 162.13	(Self Training)
WS-GP (50)	**23.70**	-	154.80 & **149.79**	(Weighted Semi-Supervision)

A comparative analysis between self training and semi-supervised learning models reveals that semi-supervised learning does not utilize labeled and unlabeled data as effectively as self training techniques. Additionally, the WS-GP model lacks the benefits of random sampling and varying epoch lengths found in self training techniques, which improve robustness by

exposing the model to diverse data samples and aiding generalization. The use of noisy seed labels from the PoG algorithm can also impede the WS-GP model's learning from the ground truth data.

5.2 Ablation Studies

In this section, we perform ablation studies to gain insights into the impact of different factors on the performance of the proposed ST-GP model. Specifically, we analyze the effect of training loss and the number of epochs on the model's performance. The aim of this analysis is to provide a better understanding of the model's behavior and identify areas for improvement in gaze point estimation.

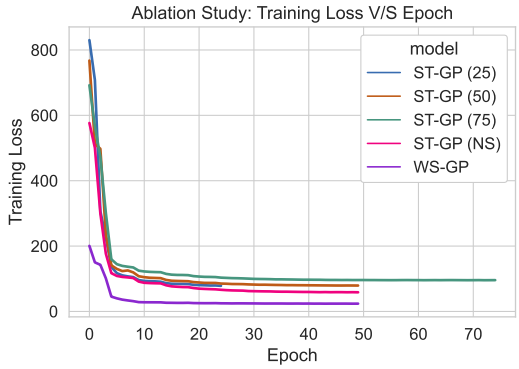
5.2.1 Training MAE Loss

Figure 5.1a displays the training MAE loss for different models, and Figure 5.1b presents the training MAE loss for various iterations. The training loss decreases as the number of iterations increases, signifying that the models are effectively learning from the data. Notably, the WS-GP model exhibits the lowest training loss among all models, consistent with its superior performance in the evaluation results. Interestingly, the training loss is lower for the 25-epoch model compared to the 50 and 75-epoch models, which is unusual as it suggests that the model learns better at 50 epochs than 75 epochs.

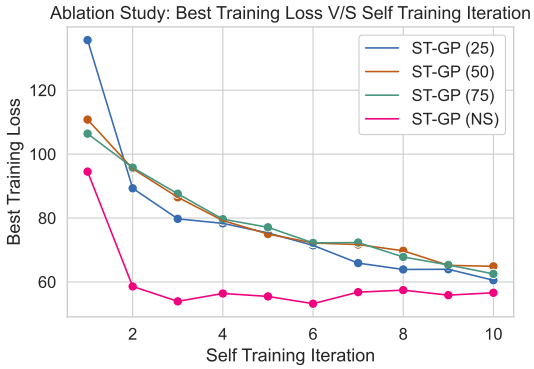
When comparing the loss of the best model for each self training iteration, as shown in Figure 5.1b, the training loss decreases with each self training iteration but not consistently. However, this observation contradicts the performance of models on the test dataset, where models with lower training loss did not necessarily perform better. Self training iteration

4 demonstrates the best performance across all self training models, suggesting that self training ceases to learn new information after this iteration.

This discrepancy between training loss and test performance highlights the importance of evaluating models on unseen data, as lower training loss does not always translate to better generalization. It also underscores the need for further analysis and potential modifications (like confidence selection) to self training techniques to ensure continued learning and improved performance on test datasets for gaze point estimation.



(a) Figure shows a line plot of MAE loss for different models V/S epochs (for the 4th self training iteration)



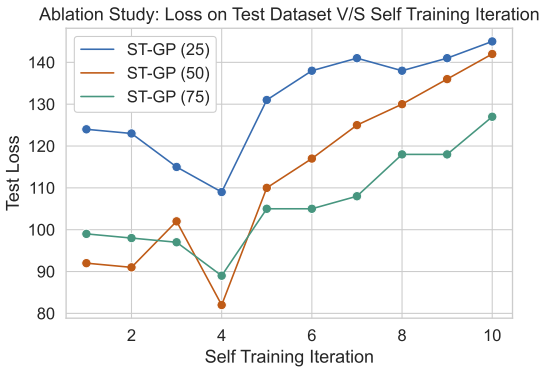
(b) Figure shows a line plot of MAE loss for self trained models V/S self training iteration

Figure 5.1: Figures compare training loss in proposed models

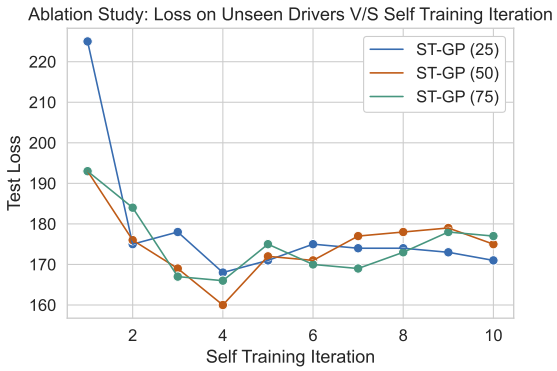
5.2.2 ST-GP Epoch Variation

Figure 5.2a and Figure 5.2b illustrate the test and unseen drivers dataset MAE loss, respectively, for different epoch variations of the ST-GP model. It can be observed that, as the number of training epochs increases, the performance of the ST-GP model generally improves. The performance at 50 epochs is superior to the performance at 25 epochs. However, at 75 epochs, the performance is worse than at 50 epochs, suggesting that the model overfits after a certain number of epochs.

This pattern highlights a trade-off between the improvement in performance and the increased computational cost associated with longer training times. It emphasizes the need for careful selection of the optimal number of training epochs to achieve a balance between model performance and computational efficiency. Finding the right balance can help ensure that the model generalizes well to new data without incurring unnecessary computational overhead.



(a) Figure shows a line plot of MAE loss for self trained models V/S self training iteration on test dataset



(b) Figure shows a line plot of MAE loss for self trained models V/S self training iteration on unseen drivers dataset

Figure 5.2: Figures compare MAE loss on test and unseen drivers dataset for self trained models

5.3 Qualitative Assessment

Figure 5.3 provides a qualitative assessment of the estimated gaze points using three of our top-performing models: ST-GP (50), ST-GP (NS 50), and WS-GP (50). The figure is organized into columns: the first column displays the driver image, the second column presents the road image, and columns 3, 4, 5, and 6 exhibit the ground truth gaze point (represented by a green circle) alongside the estimated gaze point (represented by a red circle) from the I-DGAZE, ST-GP, ST-GP (NS), and WS-GP models, respectively. The

Figure 5.3: Qualitative analysis of estimated gaze points using three top-performing models (ST-GP (50), ST-GP (NS 50), and WS-GP (50)). The figure is organized into columns displaying driver images, road images, and ground truth gaze points (green circles) alongside estimated gaze points (red circles) for each model. The proximity of red and green circles indicates the performance of each model.

circles in the visual representation vary in size to indicate the disparity between the error rates of I-DGAZE and our proposed methods. The intention behind using different circle sizes is to visually represent and compare the performance of the two models: I-DGAZE and our proposed model. By showcasing the error rates in this way, it becomes easier to appreciate the comparative effectiveness and accuracy of the two different approaches. The smaller circles in our model indicate a reduced overall error rate. This visualization method enables us to demonstrate that our model, despite specific instances where I-DGAZE might

appear more accurate, achieves a lower overall error rate, signifying improved performance.

Upon examining the proximity between the red and green circles, we can observe that the error is reduced in self-learning methods as compared to the I-DGAZE model. Furthermore, the ST-GP (NS) model appears to perform marginally better than the ST-GP model, particularly in the second and third rows of the figure. It is essential to note that the drivers in rows 5 and 6 are entirely new and have not been encountered by the models before.

By analyzing the error of the WS-GP model, we can determine that it outperforms all other models when dealing with unseen drivers. In summary, this assessment supports the conclusion that the weighted semi-supervised method excels at estimating gaze points for completely unfamiliar data. However, it is somewhat inferior to self-learning methods when applied to familiar data.

5.4 Discussion

Why does 4th iteration always perform the best?

The consistent observation that the 4th iteration of the self-trained models performed the best could be attributed to several factors. Here are a few possible explanations:

- Early iterations as warm-up: The initial iterations of the self-training process may serve as a warm-up phase, where the model gradually adapts to the dataset and learns the basic patterns. By the 4th iteration, the model might have reached a sufficient level of understanding and proficiency in capturing relevant features.
- Balance between underfitting and overfitting: The 4th iteration may strike a balance between underfitting and overfitting. In earlier iterations, the model might underfit

the data, not capturing all the complex relationships. As the iterations progress, there is a risk of overfitting, where the model starts to memorize the training data. The 4th iteration could represent a point where the model finds a good balance between the two.

- Lack of refinement module and validation feedback: The absence of a refinement module and lack of validation feedback beyond the 4th iteration may contribute to the subsequent increase in error. A refinement module or feedback mechanism could help fine-tune the model, incorporate validation data, and prevent the error from increasing after a certain iteration.

How would results differ if tried with different datasets?

If we were to try our models with different datasets, the results could vary depending on the characteristics and challenges of those datasets. While the DGAZE dataset is known to be challenging, our system performs well when trained on it. Therefore, there is a possibility that our system could adapt and perform well on other datasets as well.

However, it is important to consider that different datasets may have variations in data quality, distribution, and challenges specific to their domain. As a result, the performance of our models could be influenced by these factors. It would be necessary to carefully evaluate and fine-tune the models on the new datasets to ensure optimal performance.

Suppose a driver needs to fixate 2 or 3 different objects in quick succession. How would this need to relate to the proposed system?

The proposed system is designed to predict gaze fixation based on a single image, the temporal aspect of fixations in quick succession would not be directly captured by the model.

The system would provide information about the predicted gaze fixation for each individual image, but it would not explicitly account for the sequence or timing of fixations across multiple images.

In order to capture the temporal dynamics of fixations, it would require a video recording with a sufficient frame rate to accurately capture the rapid succession of fixations. By analyzing the gaze positions across consecutive frames, it would be possible to infer the sequence and timing of fixations on different objects.

5.5 Analysis of Results

The proposed ST-GP model with random sampling was demonstrated to be robust and accurate for gaze point estimation, outperforming the ST-GP model without random sampling on the unseen drivers dataset by a 2.04 MAE pixel difference. This suggests that random sampling enhances training data diversity, mitigating overfitting and leading to better model performance. The WS-GP model showed some adaptability to new contexts but had limitations compared to other self training techniques.

Training the ST-GP model for a larger number of epochs resulted in better performance, but there is a trade-off with increased computational cost. The evaluation of the unseen drivers dataset showed that the WS-GP model had the best performance, while the ST-GP models also performed well, especially ST-GP (NS 50).

Comparative analysis with traditional gaze point estimation techniques revealed that the ST-GP model outperformed TurkerGaze, I-DGAZE, MPIIGaze, and iTracker on training, validation, and test datasets, highlighting its superiority in gaze estimation tasks. Ablation studies on training loss and ST-GP epoch variation provided insights into the impact of these

factors on model performance, highlighting the need for a refinement method for continued improvement in gaze point estimation.

It is also important to note that the PoG Algorithm, used to generate seed labels, played a crucial role in the experiments by providing accurate initial estimates, which further influenced the effectiveness of the self training approach. We have observed that the presence of eyeglasses in the input images may influence the seed labeling process as they could potentially obstruct or distort the visibility of the eyes. However, a comprehensive study is necessary to determine the extent to which they affect the overall accuracy of the model. This would involve analyzing the model's performance on images with and without glasses and possibly refining the model to better handle such variations in input data.

5.5.1 Challenges

Despite the promising results, there are some limitations and challenges associated with the ST-GP model. One limitation is the computational cost associated with longer training times. As the number of training epochs increases, the time and resources required for training the model also increase. This could be a constraint for some applications, especially when real-time processing is needed.

Another challenge is the potential for dataset bias. While the random sampling strategy helps improve the model's generalization capabilities, it is essential to ensure that the dataset used for training and evaluation is representative of the target population and application scenarios.

Chapter 6

Conclusion

6.1 Summary of Findings and Contributions

In conclusion, this research demonstrated that semi-supervised learning can be effectively utilized to estimate gaze points in dual-camera transportation datasets without extensive labeled data by employing a self training approach. The proposed ST-GP model improved the accuracy of gaze point estimation by leveraging self-learning techniques and random sampling. This approach enabled the model to generalize to unseen drivers, ensuring more robust and reliable 2D gaze estimation across different situations. Especially, the WS-GP model generalized to unseen drivers with higher accuracy compared to train data. The performance of the proposed approach was evaluated and compared with existing 2D gaze estimation techniques in terms of accuracy, demonstrating its superior performance and potential applicability in various real-world scenarios.

In this study, we conducted an extensive quantitative comparison of gaze estimation models by evaluating their pixel MAE performance. Our findings revealed that the self training ST-GP (NS) model with random sampling achieved the lowest MAE of 79.14 pixels on the Set 3 test dataset, outperforming the other variation of the ST-GP model and the WS-GP model, which exhibited the highest MAE of 154.80 pixels. However, the WS-GP model demonstrated better performance on the unseen drivers dataset with the lowest MAE of 149.79 pixels. In comparison to the I-DGAZE method ST-GP (NS) performs with an MAE

reduction of approximately 107 pixels (from 186.89 pixels for I-DGAZE to 79.14 pixels for ST-GP (NS 50) on the Set 3 test dataset), the ST-GP (NS 50) model's notable improvement in both accuracy and resilience for gaze estimation tasks is clearly demonstrated. This represents a 57.2% improvement in performance compared to I-DGAZE. This quantitative comparison highlights the advantages of the ST-GP (NS 50) model in addressing the challenges associated with gaze estimation, paving the way for its potential application in various real-world scenarios.

The ST-GP (NS) model demonstrated superior performance in gaze point estimation tasks, outperforming ST-GP and WS-GP in terms of accuracy. However, the ST-GP model that used random sampling seemed to be a bit more robust when evaluated on unseen data. The ability of semi-supervised learning to adapt to unseen data was demonstrated by WS-GP and contributes to the field of gaze point estimation. Additionally, the insights gained from this research on random sampling and varying epoch lengths can guide the development of future gaze estimation models and techniques.

6.2 Future Work

Several areas of future work can build on the contributions of this thesis. One possibility is exploring confidence selection to further improve the self training process by identifying high-confidence and accurate data samples. Another area for future research is investigating the transfer learning capabilities of the ST-GP model, which can improve its performance on new and unseen datasets and reduce the time and resources required for training.

While we have not specifically created driver-tuned models in this study, it's certainly feasible to do so. By tailoring the model to the individual characteristics of each driver, such as their unique head height or specific patterns in head and eye movements, we may be able to

enhance the model's performance for each driver individually.

Furthermore, the potential of the ST-GP model in other gaze-related applications, such as eye-tracking research and assistive technologies, should be explored. This investigation can help identify areas of improvement and optimization, enabling the model to achieve even greater accuracy and robustness. Future research can also focus on alternative sampling strategies, model architectures, and loss functions, as well as integrating the ST-GP model with other modalities to enhance its capabilities in more challenging scenarios.

In addition to the previously mentioned areas of future work, two more noteworthy aspects to consider are the integration of Generative Adversarial Networks (GANs) and the exploration of Explainable AI techniques.

Incorporating GANs can help to generate additional training data for gaze point estimation systems. By training a GAN on existing labeled data, it can learn to generate synthetic samples that resemble real gaze patterns. This augmented dataset can then be combined with the original dataset, providing more diverse examples for training the gaze point estimation model.

On the other hand, Explainable AI techniques can aid in unraveling the inner workings of the self-training process. Self-training, often involving iterative refinements and feedback loops, can be seen as a black box in terms of how the model evolves and improves over iterations. Explainable AI techniques, such as attention maps or saliency analysis, can provide insights into which regions of the input contribute the most to the model's predictions. By interpreting and visualizing these attention mechanisms, it becomes possible to gain a better understanding of how the self-training process operates and what cues or features are important for accurate gaze estimation.

An interesting application of the system is in the training of human drivers. While the

system cannot directly train drivers, it can be used as a valuable tool to provide feedback and assistance. By accurately estimating the driver's gaze direction, the system can offer insights into their visual attention and identify areas for improvement in their gaze behavior. This information can be leveraged in driver training programs, simulators, or advanced driver assistance systems to enhance situational awareness and promote safe driving habits.

Bibliography

[1] X. Zhang, S. Park, and A. M. Feit, "Eye gaze estimation and its applications," *Artificial Intelligence for Human-Computer Interaction: A Modern Approach*, pp. 99–130, 2021.

[2] E. D. Guestrin and M. Eizenman, "General theory of remote gaze estimation using the pupil center and corneal reflections," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 6, pp. 1124–1133, 2006.

[3] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "It's written all over your face: Full-face appearance-based gaze estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 51–60, 2017.

[4] I. Dua, T. A. John, R. Gupta, and C. Jawahar, "DGAZE: Driver gaze mapping on road," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5946–5953, IEEE, 2020.

[5] A. A. Akinyelu and P. Blignaut, "Convolutional neural network-based methods for eye gaze estimation: A survey," *IEEE Access*, vol. 8, pp. 142581–142605, 2020.

[6] C. Ahlström, M. Nyström, K. Kircher, M. Nyström, M. Nyström, and B. Wolfe, "Eye tracking in driver attention research—how gaze data interpretations influence what we learn," *Frontiers in Neuroergonomics*, 2021.

[7] M. Q. Khan, M. S. Khan, and S. Lee, "Gaze and eye tracking: Techniques and applications in ADAS," *Sensors*, 2019.

[8] C. Ware, "An evaluation of an eye tracker as a device for computer input; in human factors in computing systems," in *CHI+ GI'87 Conference Proceedings*, 1987.

[9] A. T. Duchowski, "A breadth-first survey of eye-tracking applications," *Behavior Research Methods, Instruments, & Computers*, vol. 34, no. 4, pp. 455–470, 2002.

[10] C. H. Morimoto and M. R. Mimica, "Eye gaze tracking techniques for interactive applications," *Computer Vision and Image Understanding*, vol. 98, no. 1, pp. 4–24, 2005.

[11] A. L. Yarbus, *Eye Movements and Vision*. 1967.

[12] D. D. Salvucci and J. H. Goldberg, "Identifying fixations and saccades in eye-tracking protocols," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, pp. 71–78, 2000.

[13] M. F. Land and D. N. Lee, "Where we look when we steer," *Nature*, vol. 369, no. 6483, pp. 742–744, 1994.

[14] S. Tuhkanen, J. Pekkanen, P. Rinkkala, C. Mole, R. M. Wilkie, and O. Lappi, "Humans use predictive gaze strategies to target waypoints for steering," *Scientific Reports*, vol. 9, no. 1, p. 8344, 2019.

[15] Y. Okafuji and T. Fukao, "Theoretical interpretation of drivers' gaze strategy influenced by optical flow," *Scientific Reports*, 2021.

[16] T. Victor, M. Dozza, J. Bärgman, C.-N. Boda, J. Engström, C. Flannagan, J. D. Lee, and G. Markkula, "Analysis of naturalistic driving study data: Safer glances, driver inattention, and crash risk," tech. rep., Transportation Research Board, Washington, DC, 2015.

[17] T. Seacrist, E. C. Douglas, E. Huang, J. Megariotis, A. Prabahar, A. Kashem, A. Elzarka, L. Haber, T. MacKinney, and H. Loeb, "Analysis of near crashes among teen, young adult, and experienced adult drivers using the SHRP2 naturalistic driving study," *Traffic Injury Prevention*, vol. 19, pp. S89–S96, 2018.

[18] C. Carney, D. McGehee, K. Harland, M. Weiss, and M. Raby, "Using naturalistic driving data to assess the prevalence of environmental factors and driver behaviors in teen driver crashes," 2015.

[19] A. T. Duchowski, *Eye tracking methodology: Theory and practice*. 2017.

[20] A. Papoutsaki, "Scalable webcam eye tracking by learning from user interactions," in *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 219–222, 2015.

[21] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 3, no. 1, pp. 1–130, 2009.

[22] W. Wang, J. Shen, and F. Porikli, "Saliency-aware geodesic video object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3395–3402, 2015.

[23] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd Annual Meeting of the Association for Computational Linguistics*, pp. 189–196, 1995.

[24] I. Triguero, S. García, and F. Herrera, "Self-labeled techniques for semi-supervised learning: Taxonomy, software, and empirical study," *Knowledge and Information Systems*, vol. 42, pp. 245–284, 2015.

[25] X. Li and Y. Guo, "Adaptive active learning for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 859–866, 2013.

[26] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[27] D. P. Kingma, S. Mohamed, D. Jimenez Rezende, and M. Welling, "Semi-supervised learning with deep generative models," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[28] G. Underwood, P. Chapman, K. Bowden, and D. Crundall, "Visual search while driving: Skill and awareness during inspection of the scene," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 5, no. 2, pp. 87–97, 2002.

[29] D. D. Salvucci and R. Gray, "A two-point visual control model of steering," *Perception*, vol. 33, no. 10, pp. 1233–1248, 2004.

[30] M. A. Recarte and L. M. Nunes, "Effects of verbal and spatial-imagery tasks on eye fixations while driving," *Journal of Experimental Psychology: Applied*, vol. 6 1, pp. 31–43, 2000.

[31] T. W. Victor, J. L. Harbluk, and J. A. Engström, "Sensitivity of eye-movement measures to in-vehicle task difficulty," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 8, no. 2, pp. 167–190, 2005.

[32] S. Martin and M. M. Trivedi, "Gaze fixations and dynamics for behavior modeling and prediction of on-road driving maneuvers," in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1541–1545, IEEE, 2017.

[33] D. Crundall, G. Underwood, and P. Chapman, "Driving experience and the functional field of view," *Perception*, vol. 28, no. 9, pp. 1075–1087, 1999.

[34] D. R. Large and G. E. Burnett, "The effect of different navigation voices on trust and attention while using in-vehicle navigation systems," vol. 49, pp. 69–75, 2014.

[35] S. G. Klauer, T. A. Dingus, V. L. Neale, J. D. Sudweeks, and D. J. Ramsey, "The impact of driver inattention on near-crash/crash risk: An analysis using the 100-car naturalistic

driving study data," tech. rep., United States Department of Transportation, National Highway Traffic Safety Administration, 2006.

[36] W. J. Horrey and C. D. Wickens, "Examining the impact of cell phone conversations on driving using meta-analytic techniques," *Human Factors*, vol. 48, no. 1, pp. 196–205, 2006.

[37] J. K. Caird, C. R. Willness, P. Steel, and C. Scialfa, "A meta-analysis of the effects of cell phones on driver performance," *Accident Analysis & Prevention*, vol. 40, no. 4, pp. 1282–1293, 2008.

[38] P. Green, "Visual and task demands of driver information systems," tech. rep., The University of Michigan Transportation Research Institute, 1999.

[39] D. W. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 478–500, 2009.

[40] S. Baluja and D. Pomerleau, "Non-intrusive gaze tracking using artificial neural networks," *Advances in Neural Information Processing Systems*, vol. 6, 1993.

[41] D. Beymer and M. Flickner, "Eye gaze tracking using an active stereo head," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 451–458, 2003.

[42] M. Mansouryar, J. Steil, Y. Sugano, and A. Bulling, "3D gaze estimation from 2D pupil positions on monocular head-mounted eye trackers," in *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*, pp. 197–200, 2016.

[43] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Appearance-based gaze estimation

in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4511–4520, 2015.

[44] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically unconstrained gaze estimation in the wild," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6912–6921, 2019.

[45] A. A. Abdelrahman, T. Hempel, A. Khalifa, and A. Al-Hamadi, "L2CS-Net: Fine-grained gaze estimation in unconstrained environments," *arXiv preprint arXiv:2203.03339*, 2022.

[46] D. Pomerleau and S. Baluja, "Non-intrusive gaze tracking using artificial neural networks," in *AAAI Fall Symposium on Machine Learning in Computer Vision*, pp. 153–156, 1993.

[47] K. Krafka, A. Khosla, P. Kellnhofer, H. Kannan, S. Bhandarkar, W. Matusik, and A. Torralba, "Eye tracking for everyone," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2176–2184, 2016.

[48] A. Recasens, A. Khosla, C. Vondrick, and A. Torralba, "Where are they looking?," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[49] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Mpiigaze: Real-world dataset and deep appearance-based gaze estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 162–175, 2017.

[50] F. Lu, Y. Sugano, T. Okabe, and Y. Sato, "Inferring human gaze from appearance via adaptive linear regression," in *International Conference on Computer Vision*, pp. 153–160, IEEE, 2011.

[51] K. A. Funes Mora, F. Monay, and J.-M. Odobez, "Eyediap: A database for the development and evaluation of gaze estimation algorithms from RGB and RGB-D cameras," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, pp. 255–258, 2014.

[52] B. A. Smith, Q. Yin, S. K. Feiner, and S. K. Nayar, "Gaze locking: Passive eye contact detection for human-object interaction," in *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, pp. 271–280, 2013.

[53] P. Xu, K. A. Ehinger, Y. Zhang, A. Finkelstein, S. R. Kulkarni, and J. Xiao, "Turkergaze: Crowdsourcing saliency with webcam-based eye tracking," *arXiv preprint arXiv:1504.06755*, 2015.

[54] Q. Huang, A. Veeraraghavan, and A. Sabharwal, "Tabletgaze: Dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets," *Machine Vision and Applications*, vol. 28, pp. 445–461, 2017.

[55] Y. Sugano, Y. Matsushita, and Y. Sato, "Learning-by-synthesis for appearance-based 3D gaze estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1821–1828, 2014.

[56] R. Fergus, P. Perona, and A. Zisserman, "A visual category filter for google images," in *8th European Conference on Computer Vision,*, pp. 242–256, 2004.

[57] O. T. Nartey, G. Yang, S. K. Asare, J. Wu, and L. N. Frempong, "Robust semi-supervised traffic sign recognition via self-training and weakly-supervised learning," *Sensors*, vol. 20, no. 9, pp. 2684–2708, 2020.

[58] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged

consistency targets improve semi-supervised deep learning results," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[59] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[60] G. French, M. Mackiewicz, and M. Fisher, "Self-ensembling for visual domain adaptation," *arXiv preprint arXiv:1706.05208*, 2017.

[61] S. Mittal, M. Tatarchenko, and T. Brox, "Semi-supervised semantic segmentation with high-and low-level consistency," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 4, pp. 1369–1379, 2019.

[62] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-training with noisy student improves ImageNet classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10687–10698, 2020.

[63] Y. Kuznietsov, J. Stuckler, and B. Leibe, "Semi-supervised deep learning for monocular depth map prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6647–6655, 2017.

[64] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, 2012.

[65] W.-S. Lai, J.-B. Huang, and M.-H. Yang, "Semi-supervised learning for optical flow with generative adversarial networks," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[66] M. Patacchiola and A. Cangelosi, "Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods," *Pattern Recognition*, vol. 71, pp. 132–143, 2017.

[67] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1867–1874, 2014.

[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[69] H. Bhagat, S. Jain, L. Abbott, A. Sonth, and A. Sarkar, "Driver gaze fixation and pattern analysis in safety critical events," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, 2023.

[70] K. L. Campbell, "The SHRP2 naturalistic driving study: Addressing driver performance and behavior in traffic safety," *TR News*, no. 282, 2012.

[71] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2." https://github.com/facebookresearch/detectron2, 2019.

[72] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, "Retinaface: Single-shot multi-level face localisation in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5203–5212, 2020.

[73] J. M. Wolterink, A. M. Dinkla, M. H. Savenije, P. R. Seevinck, C. A. van den Berg, and I. Išgum, "Deep MR to CT synthesis using unpaired data," in *Simulation and Synthesis in Medical Imaging: Second International Workshop*, pp. 14–23, Springer, 2017.