A Survey on Eye Gaze Estimation Methods

***Abstract*— Gaze estimation becomes an essential tool in many domains since it indicates where a person is looking; hence it is a valuable source for understanding human intention. The recent progress in deep learning algorithms has dramatically improved the performance of many computer vision tasks like gaze estimation. However, there is a lack of proper guidelines for designing deep learning algorithms for gaze estimation. This article presents a comprehensive review of the current state-of-the-art gaze estimation techniques, focusing on CNN and machine learning-based gaze estimation techniques. This study aims to provide valuable insights and empower the research community to help design and develop efficient deep learning-based gaze estimation models. This review article also provides information on various pre-trained models, network architectures, and open-source datasets helpful in training deep learning models. We summarize different techniques from four perspectives: feature extraction, deep learning algorithm architecture design, subject personal calibration, and device and platform used. To compare the performance of various gaze estimation approaches, we characterize all the publicly available gaze estimation datasets and also presented an overview of gaze estimation algorithms in tabular form. This paper not only serves as a reference to develop deep learning-based gaze estimation methods but also a guideline for future gaze estimation research. At last, a review is presented for the research on eye gaze estimation applications across many domains, including human-computer-interaction, psychology, computer vision, Marketing research, Product packaging design, website design, and Product advertisement.**

Keywords*— *Convolutional neural network, gaze estimation, region of interest, accuracy, deep learning, eye movements, computer vision.

I. INTRODUCTION

Eye gaze is considered one of the most important passive forms of communication. Improvements in eye gaze tracking technology have led to the evolution of effective gaze estimation techniques for human-computer interaction over the past few decades. It all started with skin electrodes placed around the eyes for gaze prediction. With improvement and research, head-mounted eye trackers came into the market. With more focus on improving accuracy and reducing constraints for the user, post-2000 due to rapid improvement in processing speed of computers, a number of gaze estimation methods are proposed such as 2D regression model-based method, 3D eye model-based method, appearance-based method. 2D model-based method directly maps the feature vector to the point of gaze (POG) using transformation function. 3D model-based method constructs a geometric model of the eyes to estimate gaze. Both 2D regression model and 3D model requires dedicated complex setup such as infrared cameras. Appearance-based methods are non-PCCR methods that directly learn mapping from input images to the point of gaze. With the recent evolution of deep learning algorithms, Convolutional neural network based architectures for gaze estimation is becoming very popular. These deep learning models can directly map input features to gaze direction without requiring any external calibration or with very few calibration steps compared to other methods available.

Appearance based gaze doesn't need complicated set up, it only uses web cam to capture human eye appearance. To estimate the gaze it requires following set up: 1) An effective feature extractor which extract different eye features from raw facial image. 2) A robust mapping function to learn mapping from appearance to human gaze. 3) A large sample data set needed to train regression function

In recent years, rapid development in Deep learning algorithm has proven to be very effective in estimating good results as compared to conventional appearance based method. It has many advantage over convention methods such as: 1) it can extract high dimensional eye features from high dimensional image data. 2) It can learn highly nonlinear function to estimates the gaze. In conventional method accuracy drop is observed due to variations like head movement , different illumination conditions but deep learning based methods do not get very much affected by these variations. Also improvement in performance was observed for cross subject gaze estimation which makes it more compatible for real world application

In this literature work, we have provided a comprehensive review of appearance based gaze estimation using deep learning algorithms. We have discussed it in four prospective 1) deep feature extraction, 2) deep neural network architecture design, 3) personal calibration, 4) device and platform. In deep feature extraction we divide raw appearance image in to eye image, face image and in videos. We have discussed algorithm used for effective feature extraction in deep neural network architecture. We have also discussed CNN models for different supervision methodologies like supervised, semi-supervised and unsupervised gaze estimation methods .We also reviewed different architectures like multi-task CNNs and recurrent CNNs. In personal calibration we discussed how personal calibration can further improve accuracy of CNN models. In device perspective we reviewed hard ware set ups like RGB cameras, IR cameras and depth cameras, and different platforms, like computer, mobile devices and head-mount device

Rapid improvement in computing, low cost hardware and fast video processing brought eye tracking products more closer to end users with application in various domains like web marketing , virtual reality ,healthcare, product packaging design, gaming. Various user platforms used eye gaze information in different domains like Desktop and mobile based platform uses eye gaze for computer control and communication, text entry, gaze based passwords. Handheld mobile devices like tablet, smartphone uses real time eye tracking information for locking/unlocking phones and different visual interaction .Head mounted real time eye tracking set up with multiple external camera is used extensively for tracking user attention, cognitive studies and neurological domain .Real time eye tracking is also used in

automated systems like to find driver's and pilot's attention level. Remote real time eye tracking is used to activate control function of TV panels.

With different use cases, variability in eye movement, external environment and different individual biological aspect pose challenges in achieving consistency in performance from gaze estimation method

Hence aim of this work is to provide insights in to current gaze estimation research and its accuracy, performance in real world. In this literature review, we present a detailed overview and analysis that includes algorithm, system set up, user conditions, performance and evaluation of various methods discussed in various works. The aim of this work is to highlight a realistic overview currently existing in this field and to identify the different factors that affects the accuracy of real time eye tracking in practical application. It also highlights inaccuracies which arises during gaze tracking due to various factors.

The paper is organized as follows: section II presents basic eye tracking fundamentals. In section III, several gaze tracking algorithms along with feature extraction techniques, calibration techniques, dataset used in various works are discussed. In section IV, some application areas of gaze estimation methods techniques are discussed. In section V, we conclude our discussion and recommend future research directions.

II. EYE GAZE TRACKING FUNDAMENTALS

A. Types of eye movements:

To collect information about the user's intent, cognitive behavior, several types of eye movement are studied [1], these are like:

1. Fixations: It refers to the stationary period between eye movement Fixation related measurement variables include total fixation duration, mean fixation duration, fixation spatial density, number of areas fixated, fixation sequences, and fixation rate.
2. Saccades: These are rapid and involuntary eye movements between fixation points. Parameter for Saccade measurement includes saccade number, amplitude, and fixation-saccade ratio
3. Scanpath: It refers to a number of short fixations and saccades before reaching the final target on the screen.
4. Gaze duration: It is the sum of all fixations in a particular area and proportion of time spent in an area of interest before eyes leave that area of interest
5. Pupil size and blink: These measures are used to examine the cognitive workload of a user.

Table 1 presents significance of different eye movements and applications.

TABLE 1: Characteristics of different eye movements

Eye movement type	**Functionality/Significance**	**Applications in Human Computer interaction**
Fixation	Acquiring information, Cognitive processing, attention	Browsing information, reading, scene perception
Saccades	Moving between targets	Visual search
Scanpath	Path traced by user's eye	Assessing user behavior
Gaze duration	Cognitive processing, conveying intent	Item selection, text/number entry
Blink	Indicates behavioral states, stress	Eye liveliness detection, activate command/control
Pupil size change	Cognitive effort, representing micro emotions	Assessing cognitive workload, user fatigue, command/control

B. Basic setup and method used for eye gaze estimation:

Video based eye tracking mainly requires one or more digital cameras, near-infra-red (NIR) LEDs, and a computer.

Commonly used steps in eye gaze tracking include methods like user calibration, obtaining video frames of the face and eye regions, detecting eyes, and mapping gaze coordinates on the screen. The most commonly used method is Pupil Center Corneal Reflection or PCCR method. In this method, NIR LEDs are used to produce glints on the eye cornea surface, and then images/videos of the eye region are captured[2]. Gaze is then computed from the relative movement between the pupil center and glint positions. External NIR illumination is also used sometimes for better contrast and to avoid variations produced by natural light. Different gaze tracking methods are discussed in section ii.

The user interface for gaze tracking can be active or passive, single or multimodal. In an active user interface, the user's gaze information is used as an input modality and to activate a function. In a passive interface, eye gaze data is consolidated to predict user interest or attention. In a single modal gaze tracking interfaces, only user's gaze is used as an input variable. In contrast, gaze input is combined with a mouse, keyboard, touch, or blink inputs for command in a multimodal interface.

C. Estimation of gaze tracking accuracy

In the literature, gaze tracking accuracy measures are reported in different ways for e.g. angular accuracy in degrees [3],[4],[5],[6],distance accuracy in cm/mm[7], [8],[9]–[14], distance in pixels and gaze estimation accuracy in percentages[15]–[17].

Some common gaze estimation accuracy is discussed below:

- *Gaze point coordinates in Pixels:*

$$Gaze\_X = mean(\frac{X_{left} + X_{right}}{2})$$

$$Gaze\_Y = mean(\frac{Y_{left} + Y_{right}}{2})$$

Where $(X_{left}, X_{right}, Y_{left}, Y_{right})$ are the measured X and Y coordinate of the left and right eye's point of gaze (POG)

- *Monitor screen pixel size (μ):* It is calculated as:

$$\mu = \frac{dim_m}{dim_p}$$

Where $\dim_m$ = diagonal size of screen in *mm* and $\dim_p$ =diagonal size of screen in pixels as shown below:

$$dim_p = \sqrt[2]{width_p^2 + height_p^2}$$

Where $width_p$and $height_p$ is width and height of screen in pixels.

- *On screen distance(OSD):* It is the distance between origin of gaze coordinate system and a specific gaze point. It is calculated by formula mentioned at the bottom of this page. Where (x_{pixels}, y_{pixels}) = origin of gaze coordinate system and $offset$ = distance between sensor of eye tracker and lower edge of display screen.

- *Gaze angle relative to eye***:** Gaze angle at any point on the screen relative to user's eye can be estimated by using below formula:

$$gaze\_angle(\Theta) = \tan^{-1}(OSD/Z)$$

Where *Z*= distance of eyes from the screen

- *Distance between eyes and gaze point on the screen:* It is given by:

$$EstGP(mm) = \sqrt[2]{((Gaze\_X)^2 + (Gaze\_Y)^2 + Z^2)}$$

- *Pixel Accuracy:* Shift between actual gaze coordinates (GT_x, GT_y), and estimated coordinates$(Gaze_X, Gaze_Y)$can be calculated as:

$$pixel\_shift(pixels) = \sqrt[2]{\left((GT_x - Gaze\_X)^2 + (GT_y - Gaze\_Y)^2\right)}$$

- *Angular accuracy:* Angular accuracy (or prediction error) in degrees can be calculated as:

$$Angular\_accuracy = \left(\mu * pixel\_shift * \cos\left(mean(\Theta)\right)^2\right)/EstGP$$

- *Euclidean distance in cm/mm*: Some studies used Euclidean distance to compute error between predicted and actual gaze estimation. Euclidean distance can be calculated by:predicted and actual gaze estimation. Euclidean distance can be calculated by:

$$ED = \sqrt[2]{(gt\_x_i - e\_x_i)^2 + (gt\_y_i - e\_y_i)^2}$$

Where gt_x_i and gt_y_i is ground truth label for each points and et_x_i and et_x_i are estimated gaze coordinates for each points

Apart from this, some studies simply used distance between predicted gaze and ground truth as error function [14]. Some also uses root mean square error [19] to estimate accuracy.

III. DATASETS

Dataset plays a very crucial role in estimating gaze with good accuracy. Today, number of dataset are available publicly which contains a wide range of images under different environments and constraints. Some of the widely used dataset are: MPIIGaze[20], eyediap[21], Columbia[22], GazeCapture[12], TabletGaze[18] etc. Some of the datasets allows continuous head poses and gaze directions. Gaze360[16] datasets contain over 172,000 images from 238 subjects looking at different gaze directions. Data was collected in 5 indoor (53 subjects) and 2 outdoors (185 subjects) locations over 9 recordings sessions with labelled 3D gaze across a wide range of head poses and distances. MPIIGaze[20] dataset contains 213000 images from 15 subjects collected via laptop under different illumination conditions. This dataset is applicable for both 2D and 3D gaze estimation. Eyediap[21] dataset contains 94 videos from 16 subjects. It contains 3 minutes video sequences of subjects looking at fixed and floating targets taken from RGB-D (standard vision and depth) cameras. This dataset is designed in such a way that it is least affected by head pose variations, changes in ambient and sensing conditions, fixed and floating targets,

---

$$OSD(mm) = \mu \sqrt[2]{\left(\left(Gaze\_X - \frac{x_{pixels}}{2}\right) + \left(y_{pixels} - Gaze\_Y + \frac{offset}{\mu}\right)^2\right)}$$

person specific variations. RT-Gene[23] contains 123,000 images from 15 subjects. It allows automatic annotation of ground truth gaze and head poses labels of subjects under free viewing conditions and large camera to subject distances. However, it is only applicable for 3D gaze estimation. This dataset requires separate eyetracking glasses along with RGB-D cameras and motion capture cameras to collect data. It also makes use of GAN to remove eye tracking glasses after collecting ground truth. The dataset contains RGB images at 1920x1080 resolution and depth images at 512x424 resolution. Deng and Zhu [24] introduced a novel dataset that contains 240,000 images from 200 subjects. This dataset is collected in such a way that it can be used for both 3D and 2D gaze estimation, should be device independent and contains full coverage of head poses and gaze estimation under varying illumination conditions. In the data collection setup, two types of targets (Head pose targets and eyeball targets) are displayed to participants in order to guide their

TABLE 2*:* Summary of Gaze datasets allowing continuous head poses and gaze directions.

References	Dataset	Size	Head Poses	Gaze Direction	Subjects	Description
[16]	Gaze 360	172k images	Continuous	Continuous	238	Data was collected in 5 indoor (53 subjects) and 2 outdoor (185 subjects) locations over 9 recording sessions with labelled 3D gaze across a wide range of head poses and distances
[20]	MPIIGaze	213k images	Continuous	Continuous	15	Collected by laptops for daily life illumination conditions. Applicable for 2D and 3d gaze estimation.
[21]	Eyediap	94 videos	Continuous	Continuous	16	Contains 3 minutes video sequences of subjects looking at fixed and floating targets
[12]	GazeCapture	2.4 M	continuous	13+conti.	1474	Collected by mobile devices via crowdsourcing. Only for 2D gaze.
[23]	RT-GENE	123K	Continuous	Continuous	15	For 3D gaze estimation only. Requires separate eyetracking glasses along with RGB-D camera and motion capture camera
[25]	EVE	~4.2K videos	Continuous	Continuous	54	For both 2D and 3d gaze estimation. Video frames are captured in 1920x1080 pixels.
[24]	(24)	240K images	Continuous	Continuous	200	Dataset is device independent. Can be used for both 2D and 3D gaze estimation.
[17]	[17]	~154K images	Continuous	Continuous	100	Videos were downloaded from YouTube's creative section. Then third frame is considered for data creation
[26]	[26]	165K	Continuous	Continuous	218	Largest RGBD gaze dataset in terms of participants
[27]	[27]	11080 samples	Continuous	Continuous	22	Can be used in Mobile/Laptop/Desktop/Tablet devices.

head pose and eyeball movements respectively. 12 cameras are used to capture wide range of head poses and to provide 3D gaze annotations. GazeCapture[12] dataset contains 2.4 million images from 1474 participants. This dataset is collected using crowdsourcing by using mobile phones and tablets in different orientations. This dataset provides gaze direction as a pixel locations on the screen and distance from the camera. Dubey et al[17] introduced a novel dataset that contains 1,54,251 images of 100 different subjects from YouTube videos. Different types of videos are downloaded from creative common section of YouTube. After that every third frame is considered for dataset creation. Zhang et al. [27] introduced a novel dataset with an objective that it can be used in multiple devices(mobile phone, tablet, laptop, desktop, smart TV). A total of 11080 samples are collected from 22 participants. The camera resolution used for each devices were: 1440 x 2560 in case of mobile phones, 2560 x 1600 pixels for tablets, 1920 x 1080 pixels for laptop and smart TV and 1920 x 1200 pixels for desktop computer. Table 2 summarizes gaze datasets that allows continuous head poses and gaze directions.

Some of the datasets are still limited in gaze direction, head poses and illumination conditions. Columbia [22] dataset contains 6000 images from 58 participants. It consists of only 5 head poses and 21 gaze directions per head poses. Subjects were asked to fix their head on a chin rest while collecting data. The data from each subjects consists of 5 different head poses, 3 eye vertical movements and 7 eye horizontal movements contributing a total of 105 images per subject. The dataset is very diverse in nature as it contains 24 females

and 32 males of different age groups. Rice TabletGaze[18] dataset contains video recordings of 51 subjects consisting of 39 males and 12 females of different age groups. The subjects were looking at 35 points distributed among 5 rows and 7 columns. The device used for data collection is Samsung Tab S 10.5. Four recordings of four postures (Sitting, Slouching, Standing, and Lying) were collected from each participants resulting in a dataset consisting of 16 videos per subject. Unlike Columbia dataset here participants were not asked to remain in a fixed head pose. However, gaze directions were limited to 35 directions only. Wood and Bulling [19] presented a novel dataset consisting of 8 participants aged between 20 to 27. Dataset was collected on a 11-inch tablet with a quad-core 2 GHz processor running on windows 8. Each participant was asked to look at 9 pre-defined locations distributed on a 3 x 3 grid pattern. Although participants were allowed free head movement, distance between the eyes of the participants and device was fixed to 20 cm. Tablet was held in reverse orientation with camera at the bottom. Xia et al. [28] presented a dataset that contains 200 frames from 550

TABLE 3*:* Summary of Gaze datasets allowing fixed head poses and gaze directions.

References	Dataset	Size	Head Poses	Gaze Direction	Subjects	Description
[29]	Eye Chimera	1170 images	-	7	40	Video was recorded from Canon 600D and Panasonic HDC-TM for 7 gaze directions and then converted to images.
[22]	Columbia	6K images	21	5	58	Collected in laboratory for 2D gaze only.
[18]	Rice TabletGaze	816 videos	Continuous	35	51	Collected on Samsung Galaxy Tab S 10.5 tablet with a screen size of 22.62 × 14.14 cm (8.90 × 5.57 inches) n landscape mode. Four body postures are used.
[30]	UTMultiview	64K images	8+synthesized	160	50	Collected on laboratory via fixed head poses. Data also is synthesized to increase number of samples.
[19]	[19]	-	Continuous	9	8	11-inch (1920×1080 pixels) was used. Tablet was placed 20 cm from the eyes.
[28]	[28]	110K	Continuous	25	550	Dataset contains head poses and body poses of different people using phone in daily life. Distance from participants to screen was not fixed and varies from 25 to 60 cm.

550 participants resulting in a total of approximately 110,000 images. The participants were in the range of 20-35 years of age. The data was collected using a Samsung phone with a screen resolution of 2220 x 1080 pixels. A total of 25 fixed gaze points were presented in screen and participants were asked to look at these points. Participants were allowed free head movements and distance between participants and screen was not fixed and varies from 25 to 60 cm. Table 3 summarizes gaze datasets that does not allows continuous head poses and gaze directions.

IV. EYE GAZE ESTIMATION METHODOLOGY

In general, there are 5 types of eye gaze tracking methods: 2D regression based method, 3D model based method, cross ratio based method, appearance based and shape based methods.

2D regression based methods: The method makes use of IR cameras to detect vector of geometric features such as pupil center, glints and directly maps this vector to the POG on the screen using a polynomial transformation function.

Cross Ratio Based Methods: In this method, four or more infrared lights are positioned on the end of the screen (all these are coplanar). A camera is also placed which captures images of user's eyes. The corneal reflection of each light source on the eyes is called glint. From the captured images, we know the glint of each light sources and also the center of pupil. The cross ratio method consider the surface of cornea as a plane and assumes that all the glints are coplanar. A mapping is then performed between the light sources and glints detected by the camera. Once the mapping is done, pupil center can be projected on the screen to estimate POG [31].

3D Model Based Methods: This method utilizes geometric 3D model of an eye along with eye features such as pupil center [32], corneal reflection [33] and iris contour [34]. As shown in the figure 1, Optical axis is the line joining the center of pupil and center of corneal lens. The line connecting the fovea with the center of corneal lens is called the visual axis. The point of gaze is defined as intersection between visual axis and device screen.

A model based approach for unmodified tablet was presented in [19] to predict gaze direction where 2D ellipse is back projected to 3D to locate eye optical axis. Point of gaze (POG) is then estimated using the intersection of 3D optical axis and display screen. An approach independent of user calibration was presented in [35] which employs multiple

camera and multiple point light sources to estimate gaze direction.

Appearance Based Methods**:** Appearance based methods directly learns mapping from input images to point of gaze. As in this method there are many variability in unconstrained environment, so conventional appearance-based methods cannot handle these variation due to the weak fitting ability. We will use Convolutional neural networks (CNNs) and different deep learning based algorithm to detect gaze as they are capable of handling large datasets.

The different gaze estimation algorithms presented above have distinct characteristics, advantages and disadvantages. The 2D regression based methods utilize the features of the human eye and can be implemented using a single camera and a few NIR LEDs. However, these techniques are very

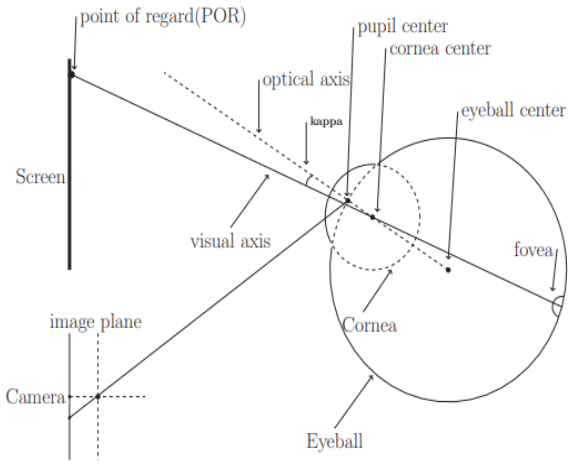

Fig. 1 3D Eye Model [36]

much affected by head movements and often require users to keep their head stationary.

Cross ratio based methods do not need an eye model or hardware calibration and allow free head motion. But distance between the user and screen needs to be kept constant. 3D model based methods allows free head movement but the hardware requirements for implementing 3D methods are high as they need several illumination sources or multiple cameras

Appearance based methods are non-PCCR methods that directly learns mapping from input images to point of gaze. These methods have low hardware requirements but disadvantage is that their gaze estimation accuracy is slightly less than PCCR based methods. However in recent years, with the evolution of deep learning convolutional neural network (CNN) architecture, appearance based methods are widely used in gaze estimation. Accuracy of above methods can be increased by increasing number of calibration points.

We use following structure to review work in this domain: Feature Extraction, Gaze estimation methodology, Calibration, Devices and Platforms.

A. Feature Extraction:

Features extraction from various methods (for e.g. appearance based, model based etc.) is a challenging task due to complex eye appearance. Since accuracy of gaze estimation depends largely on quality eye features hence we have discussed various methods to extract features like extracting features from eye images, from face image, from videos.

Gaze estimation is highly dependent on eye appearance. Rotation of eyeball can change gaze direction. This dependency makes it possible to use eye images to estimate gaze direction.

Features from single eye or both eyes can be extracted. Zang *et al.* [37] proposed LetNet CNN network to extract eye features from grey-scale single eye images and merge these features with an estimated head pose. In [20] he further modified his previous work with GazeNet which is a 13-convolutional layer neural network inherited from a 16-layer VGG network to extract the individual features from single eye and to find gaze. Fischer *et al.* [23] also implemented a two VGG-16 networks to extract individual features from two eye images, and merge these features for regression. Cheng *et al.* [38] used a 4 stream CNN where two streams are used to extract individual features from eye and remaining two streams are used to extract common features from both eyes. Bao *et al.*[39] proposed a self-attention mechanism to merge two eye features. They merged the feature maps of two eyes and used a convolution layer to generate the weights of the feature map. Cheng *et al.*[40] assigned weights to both eye features based on the guidance of facial features . Wood *et al.* [19] extracted eye features using cascade classifier and a shape based approach.

Several works have aimed to extract subject invariant traits from eye images [3], [41]. Wang *et al.* proposed an adversarial learning approach to extract the domain/person invariant feature [42]. Park *et al.* [3] in his paper transform the original eye images into a pictorial presentation of the eyeball, the iris and the pupil. They use an auto encoder to learn the compact representation of gaze, head pose and appearance. Fischer *et al.* [23] used a GAN to remove eyeglasses from images. In addition to above, unannotated eye images can also be used for gaze representation

Various studies have considered face images as input to estimate gaze as they contain information about head poses. Head pose contributes plays a crucial role in gaze estimation to overcome Wollaston effect [43].The extracted features contain facial landmark and head pose [43], [23]. Various studies only uses face images as input and implement a CNN to automatically extract deep facial features [6], [12].

Some works filter unnecessary features from face images as they contain unnecessary information. Zhang *et al.* [6] proposed a spatial weights mechanism to reduce noise and to efficiently extract information about different regions of the face. The spatial weights applied on the feature maps are then fed to a CNN architecture to estimate gaze. Zhang *et al.* [44]proposed an architecture to extract information dynamically based on the image content from input images.

Some works crop the eye images from the input face images and then feed it to network. Cheng *et al.* [40] proposed a coarse-to fine gaze estimation method. They first extracted coarse grain features input face image and then fine grained these features to estimate gaze. Palmero *et al.* [43] combined facial landmark along with face and eye region to detect gaze. Jyoti *et al.*[7] used facial landmarks to extract geometric features like angle between the pupil centers and facial landmarks of the eyes and tip of the nose. Dubey *et al.* [17] collected images from Youtube

videos and then implemented unsupervised learning based method to estimate gaze.

In addition to face and eye images, videos can also be used for feature extraction. From images, we get information about static features only. However temporal information, which can be obtained from videos can be incorporated as an added advantage for better gaze estimation. Recurrent Neural Network (RNN) has been generally used in video processing like long short-term memory (LSTM) [16], [45] as RNN architectures can retain sequence information also. Zhou *at el* [45] applied many to one bidirectional LSTM to fit the temporal information between frames to predict gaze vector for video sequences.

B. Gaze Estimation Methodology

In this section we will discuss about various methodologies adopted in several works. First, we have discussed model based methodologies and then for appearance based gaze estimation with a major focus on Convolutional Neural Network architecture based appearance gaze estimation.

Wood *et al.* [19] presented a model based approach for unmodified tablet to predict gaze direction. This system does not require any external camera or any infrared illumination. However, distance between the user and display was fixed. In this, first rough eye features from the image are extracted using cascade classifiers and eye centers are then located using a shape based approach. Limbus ellipse fitting is then performed along the eye region of interest (ROI) with robust model-fitting approach. The obtained 2D ellipse is then back projected to 3D to locate the eye optical axes. Point of gaze (POG) is then estimated using the intersection of 3D optical axis and display screen. Shih *et al.* [35] presented an approach independent of user calibration which employs multiple camera and multiple point light sources to estimate line of sight. Two light sources and two cameras that are not collinear are used to find the 3D locations of pupil and cornea centers. Gaze estimation is then performed by connecting the pupil and cornea center.

For appearance based gaze estimation, convolutional neural network architectures have been widely used as these networks give better performance. Also they require few input parameters for estimating gaze. Supervised models are most widely used method for appearance based gaze estimation. In these models, CNN network is trained using image samples along with ground truth gaze directions. It is basically learning a mapping function from raw images to gaze directions [37], [46], [4] .Many CNN architectures which are used for computer vision tasks can also be implemented for gaze estimation like LeNet [37], Alex Net [6], VGG [20], ResNet18 [16] and ResNet50 [47]. Zhang et al [37]used LeNet network architecture that comprises of one convolutional layer along with max pool layer, a second convolutional layer along with max pool layer and a final fully connected layer with linear regression at the top to predict gaze vectors. Head vector is added to the output of the fully connected layer. Input to the architecture are gray scale images of size 60 x 36 pixels. The number of feature layer for the two convolutional layers is 20 and 50 for the first and second layer respectively with a feature size of 5 x 5 pixels. In the fully connected layer, the number of hidden layers are 500. The output of the network is a 2D pitch and yaw gaze angles. Zhang et al. [6] used a CNN with spatial weights for 2D and 3D gaze estimation. He used Alex Net CNN architecture that consists of five convolutional layers and two fully connected layers. The input image of size 448 x 448 pixels is passed through this CNN architecture to generate feature vector of size 256 x13 x 13. This feature vector is then passed through spatial weights mechanism which consists of a convolutional layer with filter size of 1 x1 followed by a rectified linear unit to generate a weight map which is again multiplied with the feature vector extracted before using element wise multiplication. Depending on the task whether to find 2D or 3D gaze estimation, the output from the element wise multiplication is then fed to the corresponding fully connected layer to find gaze direction. Zhang et al. [20] developed a new architecture called GazeNet. It is based on a 16 layer VGG architecture consisting of 13 convolutional layers along with two fully connected layers and one classification layer. Grey-scale image of resolution 60 x 36 pixels is used as an input. The output of the network are 2D pitch and yaw gaze angles. Apart from these architectures, some well-designed modules also help to improve the estimation accuracy [48], [40]. Chen and Shi [48] developed an architecture based on dilated convolutions where given a kernel of a particular height and width, we insert a spaces(zeroes) between the weights so that kernel covers a larger region than the given height and width. Input to this architecture are face image of size 96 x 96 pixels and two eye images of size 64 x 96 pixels. The face network consists of four stacked convolutional layers followed by max-pooling layers along with two fully connected layers at the top. The two eye networks shares same weights and consists of four convolutional layers along with max-pool layer in the middle , followed by 1 x 1 convolutional layer and one fully connected layer. Rectified Linear Unit (ReLU) activation function is applied in all the layers. The outputs from the different networks are combined together and fed to another fully connected layer to estimate gaze.

We need large scaled labeled dataset such as MPIIGaze [20] and Gaze Capture [12] to supervise CNN during training. But collection of such a huge amount of data is very difficult and time consuming hence some researcher used synthesized labeled photo-realistic image [30]. These techniques first build an eye-region models and then render new images from these models. Wood *et al.* [49] proposed to synthesize the close-up eye images for head poses, gaze directions and illuminations in large scale to develop a robust gaze estimation algorithm. To make synthesized images more close to the real ones. Fisher *et al.* [23] also implemented a GAN based image inpainting method to remove eye tracking glasses.

Another type of CNN architecture is semi supervised CNN which requires both labeled as well as unlabeled data to optimize the CNN Network. Cheng *et al.* proposed a self-supervised asymmetry regression network for gaze estimation [38]. It contain two network one is regression network to estimate the gaze from two eye images. It also provides ground truth which can be used to train the other

network. Second network is an evaluation network to assess the reliability of two eyes. The proposed network takes two eye images and head pose vector as input. The first network is a four streamed convolutional network consisting of six convolutional layers along with three max pooling layers and a fully connected layer at the end. The second network is a two stream convolutional neural network consisting of six convolutional layers with three max pooling layers along with three fully connected layers at the end. The first network finds asymmetric regression of the two eyes and the second network works on improving gaze estimation accuracy. During training, both networks train each other simultaneously like result of regression network is used to supervise the evaluation network and accuracy of evaluation network is used to optimize the learning rate of regression network. He *et al*. [9] used a person-specific user embedding mechanism to estimate gaze with very few calibration points. For estimate the gaze, they concatenated the user embedding with appearance features .They have developed a teacher - student networking system in which during training teacher network optimize user embedding and student network learn from teacher network. The input to the network are eye landmark features, both eye images and unique id. The network consists of three convolutional layers followed by average pooling layers along with five fully connected layers.

Unsupervised CNN network need only unlabeled image data for training purpose but it takes more time to optimize the CNN network. Dubey *et al.* [17] collect unlabeled facial image from web for gaze representation learning. They approximately annotated the gaze region based on the detected landmarks. Even though these approaches can learn the gaze representation, but then also few labeled samples are required to fine-tune the final gaze estimator. He proposed a novel architecture "Ize-Net". The network takes entire face image of size 128 x128 x3 as a input. It consists of five convolutional layers followed by batch normalization and max pooling. The output is then fed to fully connected layers of size 1024 and 512 with a 'Softmax' activation at the end to estimate gaze.

For improving model generalization we can use multi-task CNN. It contains multiple tasks that provide related domain information which can improve robustness of model [5], [26]. Lian *et al.* proposed a multi-task CNN based learning network to estimate point of gaze by using depth images [26]. They first extracted eyeball features from two single-eye images, head pose features from RGB and depth images with the help of GAN. Then depth values of eye region and original eye coordinates are used to encode 3D eye position. All the features are then concatenated and fed in to a network for gaze estimation. They also collected a large scale RGBD dataset for performance evaluation.

Yu et al. introduced a constrained landmark gaze model (CLGM) for modeling eye landmark locations and gaze directions [5]. They first estimated the coefficients of a joint CLGM landmarks-gaze model as well as the scale and translation parameters that define the eye region. Gaze is then estimated using head poses and CGLM coefficients. Deng *et al*. [24] used two individual CNNs to find head pose and eye ball movement. A gaze transform layer will then combine the results from the above two CNNs for gaze prediction.

In recent years, Recurrent neural networks is widely used to estimate the gaze in videos as it has been observed that recurrent neural networks have shown good capability in handling the sequential data frame [16], [43], [45]. Since human gaze is continuous, Kellnhofer *et al.* [16] proposed a video based gaze tracking model implemented on bidirectional Long Short-Term Memory capsules (LSTM) where outputs are dependent on both past and future values. Multiple frames of input are fed to a backbone model first to extract high level features. The extracted features are then fed to bidirectional LSTMs with two layers along with a fully connected layer to get gaze prediction.
Palmero *et al.* [43] also implemented a many-to-one recurrent network. The recurrent network extracts sequential information to predict 2D gaze angles. The network is divided in to 3 modules: individual, fusion and temporal module. Input to the individual module are full face images(224 x 224 pixels), eye region (120 x 48 pixels) and facial landmarks. Individual module consists of 13 convolutional layers along with 5 max pooling layers and 1 fully connected layer with a Rectified Linear Unit (ReLU) activation function. The fusion module consists of two fully connected layers along with ReLU activations and two dropout layers as a regularization. All the models are trained using ADAM optimizer with an initial learning rate of 0.0001, batch size of 64 frames. Average Euclidean distance is used as a loss function between the predicted and actual ground truth. The temporal module consists of many-to-one recurrent network and extracts sequential information to predict 2D gaze angles. Zhou *et al.* [45] first extracted features from face and eye images. The extracted features are fed to bidirectional LSTM to secure temporal information between frames for estimating gaze vectors for videos. It consists of two modules, one is static and other is temporal modeules. The static module consists of a two branch convolutional neural network and one fully connected layer. The input to the static module consists of normalized face images and two eye images, both of resolution 224 x224. One branch of convolutional neural network extracts features from the face and other branch from the eye images. The fully connected layer combines these results from two branches which are then fed to many to one bi-directional LSTM. A linear regression is then used to predict gaze in the last time stamp.

C. Calibration

The eye parameters typically required in gaze estimation are pupil center, center of corneal lens, the optical and the visual axes. As shown in figure 1, the back of the eyeball is called retina and a place on the macula of the retina where sharpest image is formed is called as fovea. The line connecting the fovea with the center of corneal lens is called the visual axis. The point of gaze is defined as intersection between visual axis and device screen. Optical axis is the line joining the center of pupil and center of corneal lens. The angle between the visual axis and optical axis is called Kappa angle. Location of fovea is unique for each user. We cannot estimate

visual axis directly, so we need a user specific calibration to find visual axis given optical axis.

In general, Calibration can be performed via active and passive method. In active method, Calibration is performed by showing user a fixed number of points on the screen (see figure 2). Each user is asked to gaze at these points for a certain period. Offset is then calculated between real gaze and estimated gaze. Whereas in passive method, user is asked to do regular device usage and as they are using their devices, there are certain things which require users to fixate on certain locations. Using this passive information, calibration is then performed to improve accuracy.

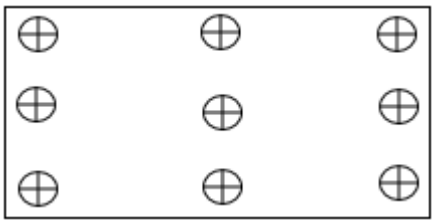

Figure 2: Active calibration method

The common approach adopted in many works is to fine tune the model while testing it on unseen data[12], [27]. Krafka *et al* [12] implemented an SVR in place of a fully connected layer in the last and fine-tuned it to predict the gaze location. With calibration, there error got reduced by 0.40 cm. Zhang *et al* [27] performed both implicit and explicit calibration. The entire CNN network was divided in to three parts: the encoder, the feature extractor, and the decoder. Encoder and decoder was fine tuned in each of the five devices used in the study (mobile devices, tablet, laptop, desktop, smart tv). In external calibration, participants were asked to fixate on a circle and perform a click when circle gets converted into a dot. While in implicit calibration, face videos, timestamp and location of interaction events were recorded to collect ground truth. In another work [13], firstly user was asked to fixate on the circle until it become a dot. The samples obtained are then used to get a third-order polynomial mapping function between the estimated and ground-truth 2D gaze locations.
He *et al.* [9] proposed a supervised calibration method with embedding based few shot learning using only 2-5 calibration points instead of more than 13 points often used in most of the works. They also proposed unsupervised personalization method to improve accuracy based on teacher-student framework using few unlabeled images.
Park *et al.* [41] proposed another algorithm for person specific gaze with very few calibration samples (<9). They used a Meta learning based calibration methodology. Using meta-learning, an adaptable gaze estimation network was trained to implement person specific gaze estimation network. Liu *et al.* [50] proposed a network such that if a subject specific calibration images are given, then the network can predict gaze of any novel sample.

Most of the available calibration methods are supervised and need labeled samples .But to collect such a large amount of data is very cumbersome and time taking so an alternate way is to collect calibration sample in a user unaware manner [8]. Chang *et al*. [8] introduced a framework SalGaze. It makes use of saliency information such that gaze estimation algorithm will automatically be adapted for new user without any explicit calibration.

In many works, calibration is not performed [19], [14]. Even though calibration was not performed, they are able to estimate gaze with very less error. With the advent of better image capturing devices, need of calibration is decreasing day by day and continuous research in going in this area.

D. Results and Discussions

In this section we have discussed about the results obtained in various works. gaze tracking accuracy measures are reported in different ways for e.g. angular accuracy in degrees [3]–[6] , distance accuracy in cm/mm[7]–[14], distance in pixels and gaze estimation accuracy in percentages[15]–[17].

George 2016 [15] Proposed a real time classification framework for eye gaze direction for predicting 7 eyes accessing cues (EAC) classes. He proposed two different methods, first one where landmark detection is carried out using geometrical relations and the second method where ensemble of randomized tree approach (ERT) is used for landmark detection. The first method gave an accuracy of 81.37 percent and the second method accuracy was 86.81 percent. The best result was obtained with the second method when combined with the CNN architecture discussed previously in this work.
Dubey 2019 [17]proposed a method which estimates eye gaze mapping using unsupervised learning methodologies. For training the model, he initialized weights using ‘glorot normal ‘distribution and then trained it using stochastic gradient descent optimizer with a learning rate of 0.001. Categorical cross entropy is used as a loss function to train the network. He achieved an accuracy of 91.5 percent on the proposed dataset.
Some works estimated accuracy in terms of degrees. Park 2018 [3] introduced a novel deep neural network architectures to estimate gaze using single eye input. He trained network using ADAM optimizer with a batch size of 32 and a learning rate of 0.0002. He performed the evaluation on three datasets and achieved an accuracy of 4.5 degrees in MPIIGaze[20] dataset, 10.3 degrees on eyediap[21] dataset and 3.8 degrees in Columbia dataset. Park 2019[41] presented a deep learning gaze estimation methods which can achieve high accuracy requiring very few calibration samples. The entire network was trained using stochastic gradient descent optimization method and it achieved an accuracy of 3.18 degrees in Gazecapture[12] dataset and 3.42 degrees in MPIIGaze[20] dataset. Zhang 2017 [20]proposed a novel method based on multimodal convolutional neural network. From the input image obtained from the RGB camera, facial landmarks and face image are detected. A convolutional model is then used to learn a mapping from features obtained to 3D gaze directions. The proposed method obtained an accuracy of 10.8 degrees improving the state of art by 22 percent from mean error of 13.9 degrees to 10.8 degrees. Zhang 2016 [6]proposed a method that only takes full face image as input. It makes use of spatial weights to suppress and enhance performance in different facial regions. The

proposed method achieved an accuracy of 4.8 degrees in MPIIGaze dataset and 6 degrees in eyediap dataset resulting in improvement of accuracy of up to14.3 percent on MPIIGaze and 27.7 percent on eyediap dataset for 3D gaze estimation. Cheng 2020[40]proposed a coarse to fine strategy to estimate gaze. Two convolutional neural networks are designed to estimate gaze, one to extract coarse features from eye images and predict basic gaze direction. Another convolutional neural network is to extract fine features. Then results from both networks are combined to estimate gaze. The results were compared with iTracker [12], RT-Gene[23]. The prosed method achieved an accuracy of 4.1 degrees in MPIIGaze dataset and 5.3 degrees in eyediap dataset and outperformed other appearance based methods. Palmero 2018 [43]proposed a method to estimate gaze using a multimodal recurrent convolutional neural network. He achieved an accuracy of 5.1 degrees in static head pose conditions and 6.2 degrees in moving head pose conditions achieving an improvement of 14.6 percent over the state of art methodologies like MPIIGaze method[20] on eyediap dataset. Park 2020 [25]proposed a architecture for end to end video based eye tracking. The proposed method achieved a 28 percent improvement in point of gaze estimation resulting in 2.49 degree error. Li 2018 [51] proposed a combined gaze tracking algorithm where a convolutional neural network is utilized to remove blinking images and predict a coarse gaze direction. Next, a geometric model is used for accurate gaze tracking. The proposed method achieved a gaze accuracy of 0.53 degrees. Wood 2014 [19]proposed a model based approach for gaze estimation that runs on unmodified tablets. First, eye region of interest and elliptical outline is obtained using robust model-fitting method. The 2D ellipses are then back projected to 3D to find optical axes. Point of gaze is then estimated using the intersections between optical axes and screen. The proposed method achieved an accuracy of 6.88 degrees.

Some works estimated accuracy in terms of distance in cm/mm between the actual and predicted gaze direction. Chang 2019 [8]proposed a method which utilizes saliency information to estimate gaze direction without explicit user calibration. The proposed method achieved an error of 3.3 cm resulting an accuracy improvement of about 24 percent over existing methods like iTracker[12] by 24 percent. He 2019 [9] proposed a on device few shot personalization method for 2D gaze estimation. The proposed method can achieve better accuracy using very few calibration points and achieved 24.26 percent better accuracy compared to other existing methods. The method achieved an accuracy (measured by mean error) of 1.37 cm on mobile devices and 2.1 cm on tablets. Krafka 2016 [12]proposed a deep convolutional neural network for estimating gaze, achieving an error of 1.04 cm on mobile devices and 1.69 cm on tablets respectively. The proposed method achieved a significant reduction in error as compared to other approaches like MPIIGaze[20]-3.63 cm, Tabletgaze[18]-3.17 cm and Alexnet[6]-3.09 cm respectively.

Lack of homogeneity can be observed in performance evaluation among several approaches. While some works estimated accuracy in percentage, others have measured accuracy in terms of distance or degrees. This variation makes inter-comparisons between different works improbable. There is need of development of standard methodologies for evaluating performances of different methods.

.

E. Devices and Platforms

In this section, we have discussed about the devices which are used to capture data. Also user platforms where eye gaze tracking is incorporated are discussed. Mainly three platforms are used: computers, handheld devices, and head-mounted devices.

The majority of gaze estimation systems makes use of a single RGB camera to gather data, while some systems makes use of different camera settings, e.g., using multiple cameras to gather multi-view images[16], using infrared (IR) cameras to tackle low illumination condition[51] and using RGBD cameras to collect the depth information[23], [26].

Kellnhofer *et al*. collected data using setup built on a Ladybug5 360° panoramic camera placed on tripod [16]. To build the dataset, they have used AlphaPose [52]to detect head key point positons and participant's feet from each camera unit independently. Wang et al. collected data through 4 different cameras in different perspective to have dataset more close to real-world scenarios [53]. Deng et al. used two TV screens (60 inches, screen size: 1345 mm * 780 mm) for target presentation and array of 12 cameras for presenting data.

Lian *et al.* used an Intel RealSense SR300 as RGBD camera to capture data using an Apple iMac machine as display. Depth images were used to provide head pose and 3D eye position information[26]. Zhang et al. used five different devices (mobile phone, tablet, laptop, desktop, smart tv) to capture data [27]. Built-in cameras of tablets, mobile devices and laptop was used. Logitech C910 and Logitech 930e camera was mounted on desktop and smart TV respectively to gather data.

Li *et al*. used an infrared camera to collect eye data since infrared camera is insensitive to external light changes [51]. Kassner *et al.* used an eye tracking glasses equipped with a

TABLE 4: Summary of Gaze Estimation Methods applicable to Desktops

Reference	Accuracy	Architecture	Dataset	Image Resolution	Input
[15]	89.81%	Own CNN architecture	Eye Chimera	42x50	Two eyes
[16]	51%	Bidirectional LSTM	Gaze360,Own Dataset	-	Video
[3]	4.5 degree 10.3 degree 3.8 degree	fully convolutional (Hourglass) and regressive (DenseNet) architecture	MPIIGaze, EYEDIAP, Columbia	150×90	Single Eye image

[41]	3.18 degrees 3.42 degrees	Disentangling Transforming Encoder-Decoder (DT-ED)	GazeCapture MPIIGaze	-	Eye image
[20]	10.8 degrees	Own CNN architecture-GazeNet	Own Dataset-MPIIGaze	60x36	Eye Image
[6]	4.8 degrees, 6 degrees	AlexNet	MPIIGaze,Eyediap	448X448	Full Face
[48]	4.8 degrees	Own architecture-DilatedNet	MPIIGaze	Eye-64x96 face-96x96	Eye and Face
[23]	7.7 degrees	Own architecture	Own dataset-RT-GENE		Face image
[38]	5 degrees	Own architecture-ARE-Net	Modified MPIIGaze	36x60	Eye Image
[40]	4.1 degrees,5.3 degrees	Own architecture-CA-Net	MPIIGaze Eyediap	Eye-36x60 Face- 224x224x3	Eye and Face Image
[44]	6.6,4.5,3.3 degrees	Own architecture	Eyediap, MPIIGaze GazeCapture	224x224	Face Image
[53]	1.79 degrees	RESNET-34	Own dataset	224x224	Face Image
[24]	4.3 degrees	AlexNet	Own Dataset	224x224	Full face and One eye
[43]	5.1,6.2 degrees	VGG-16	Eyediap	224x224 120x48	Full face, two eyes
[7]	2.22 degree, 2.08 cm	Own CNN architecture	Columbia Eye Gaze TabletGaze	-	Face Image
[17]	91.5%	Own architecture-Ize-Net	Own dataset	128x128x3	Full Face collected from Youtube creative common section
[45]	4.18,5.84 degrees	Own Architecture	MPII Gaze Eyediap	233x224	Face image, Eye image
[46]	2.84,10.04 degrees	Own architecture-Gaze-net	MPII Gaze Columbia Eye Gaze	36x36x1	Eye Images
[4]	4.918 degrees	Own architecture	MPII Gaze	60x36	Eye images
[5]	5.7,5.4 degrees	Own CNN architecture	UTMultiview Eyediap	36x60	Eye images
[26]	4.8 degrees	Own architecture	Own dataset	224x224	Face image
[54]	6.4 degrees	Own architecture	GazeFollow+Eyediap+SynHead	227x227	Entire image, face image
[27]	Mobile-2.3 Desktop-3.5 Tablet-2.8	Own architecture	Own Dataset	448x448	Face Image
[50]	3,3.8,3.77 degrees	Own architecture	Eyediap, MPIIGaze, UT-Multiview	48x72x3	Eye images
[25]	2.49 degrees	Own architecture-GazeRefineNet	Own dataset-EVE	128x128	both eyes images
[37]	13.9 degrees	Own architecture	Own dataset-MPIIGaze	36x60	eye image
[6]	4.8 degrees, 6 degrees	Spatial weights CNN architecture	MPIIGaze Eyediap	538x448	Face image

scene camera and one infrared spectrum eye camera for dark pupil detection [55]. Fischer at el. implemented eight motion capture cameras, one RGBD camera and a mobile eye tracking glasses to capture data [23].

The computer is the most common platform for gaze estimation. The cameras are usually installed below/above the computer screen [3], [15], [35]. Some works focus on using deeper neural networks or extra modules[15], [35] to improve gaze performance, while the other works make use of custom devices for gaze estimation, such as multi-cameras and RGBD cameras [23], [26]. Table 4 provides a summary of various gaze estimation techniques for desktops only.

Head-mounted gaze trackers are portable platforms with applications ranging from computer input, interactions in virtual environments, gaming controls, augmented reality, and neuro/psychological research. The general setup includes two cameras; one camera pointing at the wearer's eye to detect the pupil; and the scene camera capturing the user's point of gaze, with sometimes additional components like NIR light sources and hot mirrors. Head-mounted gaze trackers have been implemented as attachment-free, mobile, cost-efficient, lightweight devices with simple hardware and software. Also, they are known to provide high-accuracy gaze information in unconstrained settings. Kassner *et al.* used eye tracking glasses with two cameras: Eye camera and a scene camera[55]. In this, first eye images are converted into a grayscale image, and the initial region of interest is generated. Then ellipse fitting is done to locate the darkest pupil in the IR illuminated eye camera image. Gaze mapping is then done using a transfer function consisting of two bivariate polynomials of adjustable degrees.

Smartphones and tablets provide a unique paradigm for gaze tracking applications. Gaze tracking on handheld devices is done using the device front camera, one or more IR light sources and various computer vision algorithm. Table 5 provides a summary of various gaze estimation methods applicable to mobile devices and tablets.

V. SOME APPLICATION AREAS

Real-time gaze estimation can be implemented in different domains. Naples[56] in his research work studied psychological behavior with the help of real-time gaze estimation and proposed his study on "How the fear of COVID-19 changed the way we look at human faces".He used a regular laptop with a good webcam, an online server to collect data in his experiment. Fifty-four participants (31 females; mean age = 26.46 years, SD = 5.82) with self-reported normal or corrected-to-normal vision were enrolled in the experiment. Fifty-four participants (31 females; mean age=26.46 years, SD = 5.82) with self-reported normal or corrected-to-normal vision were enrolled in the experiment.

TABLE 5: Summary of Gaze estimation Methods for Mobile/Tablets

Reference	Accuracy	Architecture	Dataset	Image Resolution	Input
[27]	Mobile-2.3 Desktop-3.5 Tablet-2.8	Own architecture	Own Dataset	448x448	Face Image
[8]	3.3 cm	SalGaze- CNN based architecture	Own Dataset	64x64	Both eyes, Face, Face grid
[39]	1.62 cm – mobile 2.3 cm -tablet	AFF-Net- a CNN based architecture	GazeCapture	Face-224x224x3 Eyes-112x112x3	Face image, Both eyes, top left corner and bottom right corner of face and eye bounding boxes
[9]	mobile-1.37 cm tablet-2.1 cm	SAGE- CNN architecture	GazeCapture	64x64	Eye images, eye landmark features
[10]	1.97 cm	Own architecture	GazeCapture	64x64	Face image, let and right eye
[11]	4.85 cm	iTracker	GazeCapture	144x144	Face and eyes
[12]	~2 cm	iTracker	GazeCapture	224x224 face grid-25x25	left and right eyes, face images, face grid
[28]	1.96 cm	Own CNN architecture	Own dataset	227x227	left and right eyes, face image
[51]	0.54 Degrees	ResNet-101	Own Dataset	224x224x3	Eye images
[19]	6.88 degrees	Uses cascade classifiers and shape-based approaches to determine the eye region and centers. Elliptical model-fitting and 3D back-projections are then used to determine the eye optical axes and point-of-gaze	Own Dataset	1920x1080	Eye regions and centres.
[14]	1.25 cm	Gaze Estimator- CNN architecture	GazeCapture	224x224 50x50	Eye image Eye Grid
(53)	3.17 cm	Own Algorithmn	Own Dataset-Rice TabletGaze	1280 x 720	Full face image

In another work, Marco Fazio [57] proposed how a real-time Eye-Tracking Experiment can be used to know customer preference in implementing new Labels and Packaging designs. In this paper, he described human attention statistically with the help of different eye movement and visualization tools like heatmap, Scanpath, Gaze duration, etc. Vlad Georgescu [58] introduced iFish An Interactive Web-Based Eye Tracking Visualization Tool in which data visualization and statistical analysis are performed with different techniques like scan path, spatial heat map, bubble chart, Box plot, etc.

Carl Alexanderos Laundberg [64] in his work with the help of eye-tracking methodologies, examined the effectiveness of embedded brand placement within Esports.

Raphael Menges [59] in his research work developed a visualization tool for eye-tracking data analysis in the web with visualization features like fixation, scan path and heatmap. It also offers detailed analyses like data clustering and demographic correlation. With this study, he proposed how usability analysis can be utilized in optimizing web interaction by understanding the behavior of end user.

VI. CONCLUSIONS AND FUTURE WORK DIRECTIONS

Over the last few decades, eye gaze estimation has received quite a lot of interest from several industrial, academic, and other areas. In this paper, a detailed study of gaze estimation methods is discussed to highlight diversity in various aspects such as gaze estimation basics, feature extraction, architecture implemented to estimate gaze, calibration, datasets, and performance measures implemented in various works.

Lack of homogeneity can be observed in performance evaluation among several works. Some works estimated accuracy in percentage; others have measured accuracy in terms of distance or degrees. This variation makes inter-comparisons between different works improbable. Even

though CNN-based deep learning architectures are very effective in estimating gaze, these methods also have some limitations that can become the basis of future works. CNNs are very time-consuming and computationally very expensive. Future research can focus on developing hardware-friendly, computationally inexpensive architectures that do not require any external GPUs or multi-core CPUs for smooth implementation.

REFERENCES

[1] R. J. K. Jacob and K. S. Karn, "Eye Tracking in Human-Computer Interaction and Usability Research," in *The Mind's Eye*, Elsevier, 2003, pp. 573–605.

[2] E. D. Guestrin and M. Eizenman, "General Theory of Remote Gaze Estimation Using the Pupil Center and Corneal Reflections," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 6, pp. 1124–1133, Jun. 2006, doi: 10.1109/TBME.2005.863952.

[3] S. Park, A. Spurr, and O. Hilliges, "Deep Pictorial Gaze Estimation," 2018, pp. 741–757.

[4] J. Lemley, A. Kar, A. Drimbarean, and P. Corcoran, "Convolutional Neural Network Implementation for Eye-Gaze Estimation on Low-Quality Consumer Imaging Systems," *IEEE Trans. Consum. Electron.*, vol. 65, no. 2, pp. 179–187, May 2019, doi: 10.1109/TCE.2019.2899869.

[5] Y. Yu, G. Liu, and J.-M. Odobez, "Deep Multitask Gaze Estimation with a Constrained Landmark-Gaze Model," 2019, pp. 456–474.

[6] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "It's Written All Over Your Face: Full-Face Appearance-Based Gaze Estimation," Nov. 2016, [Online]. Available: http://arxiv.org/abs/1611.08860.

[7] S. Jyoti and A. Dhall, "Automatic Eye Gaze Estimation using Geometric & Texture-based Networks," in *2018 24th International Conference on Pattern Recognition (ICPR)*, Aug. 2018, pp. 2474–2479, doi: 10.1109/ICPR.2018.8545162.

[8] Z. Chang, M. Di Martino, Q. Qiu, S. Espinosa, and G. Sapiro, "SalGaze: Personalizing Gaze Estimation Using Visual Saliency," Oct. 2019, [Online]. Available: http://arxiv.org/abs/1910.10603.

[9] J. He *et al.*, "On-device few-shot personalization for real-time gaze estimation," *Proc. - 2019 Int. Conf. Comput. Vis. Work. ICCVW 2019*, pp. 1149–1158, 2019, doi: 10.1109/ICCVW.2019.00146.

[10] T. Guo *et al.*, "A Generalized and Robust Method Towards Practical Gaze Estimation on Smart Phone," Oct. 2019, [Online]. Available: http://arxiv.org/abs/1910.07331.

[11] M. Kim, O. Wang, and N. Ng, "Convolutional Neural Network Architectures for Gaze Estimation on Mobile Devices," p. 231, 2017.

[12] K. Krafka *et al.*, "Eye Tracking for Everyone," Jun. 2016, [Online]. Available: http://arxiv.org/abs/1606.05814.

[13] X. Zhang, Y. Sugano, and A. Bulling, "Evaluation of Appearance-Based Methods and Implications for Gaze-Based Applications," Jan. 2019, doi: 10.1145/3290605.3300646.

[14] L. Jigang, B. S. L. Francis, and D. Rajan, "Free-Head Appearance-Based Eye Gaze Estimation on Mobile Devices," *1st Int. Conf. Artif. Intell. Inf. Commun. ICAIIC 2019*, no. May 2021, pp. 232–237, 2019, doi: 10.1109/ICAIIC.2019.8669057.

[15] A. George and A. Routray, "Real-time Eye Gaze Direction Classification Using Convolutional Neural Network," May 2016, [Online]. Available: http://arxiv.org/abs/1605.05258.

[16] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically Unconstrained Gaze Estimation in the Wild," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 6911–6920, doi: 10.1109/ICCV.2019.00701.

[17] N. Dubey, S. Ghosh, and A. Dhall, "Unsupervised Learning of Eye Gaze Representation from the Web," Apr. 2019, [Online]. Available: http://arxiv.org/abs/1904.02459.

[18] Q. Huang, A. Veeraraghavan, and A. Sabharwal, "TabletGaze: Unconstrained Appearance-based Gaze Estimation in Mobile Tablets," Aug. 2015, [Online]. Available: http://arxiv.org/abs/1508.01244.

[19] E. Wood and A. Bulling, "EyeTab," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, Mar. 2014, pp. 207–210, doi: 10.1145/2578153.2578185.

[20] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "MPIIGaze: Real-World Dataset and Deep Appearance-Based Gaze Estimation," Nov. 2017, [Online]. Available: http://arxiv.org/abs/1711.09017.

[21] K. A. Funes Mora, F. Monay, and J.-M. Odobez, "EYEDIAP," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, Mar. 2014, pp. 255–258, doi: 10.1145/2578153.2578190.

[22] B. A. Smith, Q. Yin, S. K. Feiner, and S. K. Nayar, "Gaze locking," in *Proceedings of the 26th annual ACM symposium on User interface software and technology*, Oct. 2013, pp. 271–280, doi: 10.1145/2501988.2501994.

[23] T. Fischer, H. J. Chang, and Y. Demiris, "RT-GENE: Real-Time Eye Gaze Estimation in Natural Environments," 2018, pp. 339–357.

[24] H. Deng and W. Zhu, "Monocular Free-Head 3D Gaze Tracking with Deep Learning and Geometry Constraints," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 3162–3171, doi: 10.1109/ICCV.2017.341.

[25] S. Park, E. Aksan, X. Zhang, and O. Hilliges, "Towards End-to-end Video-based Eye-Tracking," Jul. 2020, [Online]. Available: http://arxiv.org/abs/2007.13120.

[26] D. Lian *et al.*, "RGBD Based Gaze Estimation via Multi-Task CNN," *Proc. AAAI Conf. Artif. Intell.*, vol. 33, pp. 2488–2495, Jul. 2019, doi: 10.1609/aaai.v33i01.33012488.

[27] X. Zhang, M. X. Huang, Y. Sugano, and A. Bulling, "Training Person-Specific Gaze Estimators from User Interactions with Multiple Devices," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, Apr. 2018, pp. 1–12, doi: 10.1145/3173574.3174198.

[28] Y. Xia, B. Liang, Z. Li, and S. Gao, "Gaze Estimation Using Neural Network And Logistic Regression," *Comput. J.*, May 2021, doi: 10.1093/comjnl/bxab043.

[29] L. Florea, C. Florea, R. Vrânceanu, and C. Vertan, "Can your eyes tell me how you think? a gaze directed estimation of the mental activity," *BMVC 2013 - Electron. Proc. Br. Mach. Vis. Conf. 2013*, pp. 1–11, 2013, doi: 10.5244/C.27.60.

[30] Y. Sugano, Y. Matsushita, and Y. Sato, "Learning-by-Synthesis for Appearance-Based 3D Gaze Estimation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1821–1828, doi: 10.1109/CVPR.2014.235.

[31] J.-B. Huang, Q. Cai, Z. Liu, N. Ahuja, and Z. Zhang, "Towards accurate and robust cross-ratio based gaze trackers through learning from simulation," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, Mar. 2014, pp. 75–82, doi: 10.1145/2578153.2578162.

[32] R. Valenti, N. Sebe, and T. Gevers, "Combining Head Pose and Eye Location Information for Gaze Estimation," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 802–815, Feb. 2012, doi: 10.1109/TIP.2011.2162740.

[33] Zhiwei Zhu and Qiang Ji, "Novel Eye Gaze Tracking Techniques Under Natural Head Movement," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 12, pp. 2246–2260, Dec. 2007, doi: 10.1109/TBME.2007.895750.

[34] K. A. Funes Mora and J.-M. Odobez, "Geometric Generative Gaze Estimation (G 3 E) for Remote RGB-D Cameras," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1773–1780, doi: 10.1109/CVPR.2014.229.

[35] Sheng-Wen Shih, Yu-Te Wu, and Jin Liu, "A calibration-free gaze tracking technique," in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, vol. 4, pp. 201–204, doi: 10.1109/ICPR.2000.902895.

[36] K. Wang, S. Wang, and Q. Ji, "Deep eye fixation map learning for calibration-free eye gaze tracking," in *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*, Mar. 2016, pp. 47–55, doi: 10.1145/2857491.2857515.

[37] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Appearance-based gaze estimation in the wild," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 4511–4520, doi: 10.1109/CVPR.2015.7299081.

[38] Y. Cheng, F. Lu, and X. Zhang, "Appearance-Based Gaze Estimation via Evaluation-Guided Asymmetric Regression," 2018, pp. 105–121.

[39] Y. Bao, Y. Cheng, Y. Liu, and F. Lu, "Adaptive Feature Fusion Network for Gaze Tracking in Mobile Tablets," Mar. 2021, [Online]. Available: http://arxiv.org/abs/2103.11119.

[40] Y. Cheng, S. Huang, F. Wang, C. Qian, and F. Lu, "A Coarse-to-Fine Adaptive Network for Appearance-Based Gaze Estimation,"

Jan. 2020, [Online]. Available: http://arxiv.org/abs/2001.00187.

[41] S. Park, S. De Mello, P. Molchanov, U. Iqbal, O. Hilliges, and J. Kautz, “Few-Shot Adaptive Gaze Estimation,” May 2019, [Online]. Available: http://arxiv.org/abs/1905.01941.

[42] K. Wang, R. Zhao, H. Su, and Q. Ji, “Generalizing Eye Tracking With Bayesian Adversarial Learning,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 11899–11908, doi: 10.1109/CVPR.2019.01218.

[43] C. Palmero, J. Selva, M. A. Bagheri, and S. Escalera, “Recurrent CNN for 3D Gaze Estimation using Appearance and Shape Cues,” May 2018, [Online]. Available: http://arxiv.org/abs/1805.03064.

[44] X. Zhang, Y. Sugano, A. Bulling, and O. Hilliges, “Learning-based Region Selection for End-to-End Gaze Estimation,” *Bmvc*, pp. 1–13, 2020, [Online]. Available: https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/455196/2/0086.pdf.

[45] X. Zhou, J. Lin, J. Jiang, and S. Chen, “Learning A 3D Gaze Estimator with Improved Itracker Combined with Bidirectional LSTM,” in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, Jul. 2019, pp. 850–855, doi: 10.1109/ICME.2019.00151.

[46] B. Mahanama, Y. Jayawardana, and S. Jayarathna, “Gaze-Net: Appearance-Based Gaze Estimation using Capsule Networks,” Apr. 2020, doi: 2004.07777.

[47] X. Zhang, S. Park, T. Beeler, D. Bradley, S. Tang, and O. Hilliges, “ETH-XGaze: A Large Scale Dataset for Gaze Estimation under Extreme Head Pose and Gaze Variation,” Jul. 2020, [Online]. Available: http://arxiv.org/abs/2007.15837.

[48] Z. Chen and B. E. Shi, “Appearance-Based Gaze Estimation Using Dilated-Convolutions,” Mar. 2019, [Online]. Available: http://arxiv.org/abs/1903.07296.

[49] E. Wood, T. Baltrusaitis, X. Zhang, Y. Sugano, P. Robinson, and A. Bulling, “Rendering of Eyes for Eye-Shape Registration and Gaze Estimation,” May 2015, [Online]. Available: http://arxiv.org/abs/1505.05916.

[50] G. Liu, Y. Yu, K. A. F. Mora, and J.-M. Odobez, “A Differential Approach for Gaze Estimation,” Apr. 2019, doi: 10.1109/TPAMI.2019.2957373.

[51] B. Li, H. Fu, D. Wen, and W. LO, “Etracker: A Mobile Gaze-Tracking System with Near-Eye Display Based on a Combined Gaze-Tracking Algorithm,” *Sensors*, vol. 18, no. 5, p. 1626, May 2018, doi: 10.3390/s18051626.

[52] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, “RMPE: Regional Multi-person Pose Estimation,” Nov. 2016, [Online]. Available: http://arxiv.org/abs/1612.00137.

[53] Z. Wang *et al.*, “Learning to Detect Head Movement in Unconstrained Remote Gaze Estimation in the Wild,” Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.03737.

[54] E. Chong, N. Ruiz, Y. Wang, Y. Zhang, A. Rozga, and J. Rehg, “Connecting Gaze, Scene, and Attention: Generalized Attention Estimation via Joint Modeling of Gaze and Scene Saliency,” Jul. 2018, doi: 1807.10437.

[55] M. Kassner, W. Patera, and A. Bulling, “Pupil: An Open Source Platform for Pervasive Eye Tracking and Mobile Gaze-based Interaction,” Apr. 2014, [Online]. Available: http://arxiv.org/abs/1405.0006.

[56] G. Federico, D. Ferrante, F. Marcatto, and M. A. Brandimonte, “How the fear of COVID-19 changed the way we look at human faces,” *PeerJ*, vol. 9, p. e11380, Apr. 2021, doi: 10.7717/peerj.11380.

[57] M. Fazio, A. Reitano, and M. R. Loizzo, “Consumer Preferences for New Products: Eye Tracking Experiment on Labels and Packaging for Olive Oil Based Dressing,” *Proceedings*, vol. 70, no. 1, p. 59, Nov. 2020, doi: 10.3390/foods_2020-08124.

[58] V. Georgescu *et al.*, “iFish : An Interactive Web-Based Eye Tracking Visualization Tool iFish : An Interactive Web-Based Eye Tracking Visualization Tool,” no. May, 2020.

[59] R. Menges, S. Kramer, S. Hill, M. Nisslmueller, C. Kumar, and S. Staab, “A Visualization Tool for Eye Tracking Data Analysis in the Web,” in *ACM Symposium on Eye Tracking Research and Applications*, Jun. 2020, pp. 1–5, doi: 10.1145/3379156.3391831.