The use of eye tracking technologies to the fullest extent started with the study of human visual attention during the reading and information processing experiments (Duchowski, 2002). There remains a long history behind the current eye tracking technologies which dates back to the late 19th century. The first phase of eye tracking studies were conducted between 1879 and 1920 which cemented the foundation of eye movement through discovery of various eye movement patterns such as saccadic suppression, saccade latency, size of perceptual span etc (Duchowski, 2002). The second phase of the development of eye tracking technologies lies between 1930 and 1958 which was a more hands on research period combining the behaviorist movement in experimental psychology (Duchowski, 2002). The third phase remained between 1970 and 1998 which showed improvements in eye movement technologies in terms of accuracy and also became useful as diagnostic and interactive technologies. The diagnostic role consisted of gaining evidence of user’s visual and attentional processes by recording eye movements over some stimulus (Duchowski, 2002). With the advent of new and improved graphical displays, interactive patterns and stronger computational power, eye trackers are also now serving as a powerful means of interaction in an array of devices and applications.

There have been two major focus areas in terms of eye detection technologies: one is eye localization in the image and other is the gaze estimation (Hansen & Ji, 2010). There are three major aspects of eye detection: first is to find out the existence of eyes, second is to determine the location and position of eyes using center of pupil or iris and the third is to track the eyes throughout the video images from frame to frame. Thus detected eyes are used to estimate and track the direct visual scope of a person or the line of sight.

**Some eye tracking algorithms:**

There are two particular types of eye –tracking algorithms: feature – based and model – based(Li, Winfield, & Parkhurst, 2005). Feature – based eye tracking technologies detect the features of the images corresponding to the position of the eyes. They also consist of a threshold value which is a decisive criterion for feature-based technologies to determine the presence or absence of a feature or determinant factor. While model-based approach is an iterative search of a model parameter which is a best fitting model which matches closest to the image. However, this approach can cause significant cost in terms of computational speed and accuracy(Li et al., 2005). There are various types of eye tracking technologies using infrared video cameras and other technologies which require specific calibration and setup as well as are expensive. However, due to new webcam technologies, browser – based eye tracking technologies through simple and common webcams are gaining momentum with increased accuracy.

Starburst is an algorithm which is a combination of feature-based and model-based technologies. The main goal of this algorithm is to find out the location of the pupil center and the corneal reflection to relate vector differences between these measures and the coordinates in the scene images. (Li et al., 2005).

Similar to our eye tracking technology, there have been a few other researches which have used webcams for eye tracking. One of those studies was a study which used video images of mobile cameras to train neural networks (Baluja & Pomerleau, 1994). Another study with self calibrating eye tracking technology which used webcam consisted of pre-recorded gaze patterns (Alnajar, Gevers, Valenti, & Ghebreab, 2013). Another eye tracking technology called PACE could self calibrated using the user interactions (Huang, Kwok, Ngai, Chan, & Leong, 2016). Unlike these previous technologies, the prototype uses a completely browser – based, self-calibrating and real-time technology which uses gaze-interaction relationships. Another important factor is that this prototype technology can be instantly included into any website for performing gaze tracking tasks.

Pupil detection is another aspect of the eye tracking technology. However, depending on pupil detection as a sole technique could cause failure, therefore, mapping from pixels to gaze location can be used (Xu et al., 2015). However, TurkerGaze requires users to stare at calibration points while remaining motionless which can be unrealistic for real time usage. Another eye movement feature which plays an important role in the determination of specific eye movement patterns is the detection of saccade. An algorithm called velocity-based saccade detection algorithm identifies the saccadic velocity peaks. A threshold value is determined such as 75°/s and any motion above the threshold value is considered as a saccade. However, there can be several other factors such as clear speed near the middle of the saccade (Smeets & Hooge, 2003) or the peak saccade velocity cannot be higher than certain threshold (Nyström & Holmqvist, 2010). This algorithm can thus help to distinguish between saccades and fixations. Many commercial software such as BeGaze 2.1 by SMI use this particular algorithm (Holmqvist et al., 2011).

Another commercial software algorithm is ‘Tobii fixation filter’ developed by Olsson (2007) which is used with low speed data using a double window technique(Marple-Horvat, Gilbey, & Hollands, 1996). It uses two sliding windows on opposite sides of the current velocity sample and finds the average velocity within each window. The average values are subtracted to find out if the difference value can exceed a threshold which can be detected as a saccade. Hidden Markov Model (HMM) is another algorithm to classify data samples into saccades and fixations using probabilistic model which is based upon velocity information. The I-HMM model (Salvucci & Goldberg, 2000) uses two states, S1 and S2 which represent the velocity distribution of either fixation samples or saccade samples. I-HMM is found to be accurate and robust however, it is very complex and requires many parameters. It needs eight parameters which are estimated from similar sample data. Besides the two transition parameters for each state, it needs the observation probabilities in the form of velocity distributions. With the model parameters and the sequence of gaze positions which are to be classified, Vertibi algorithm can be used to map gaze points to states such as fixations or saccades. Due to complexity of HMM it is very difficult to use them in commercial applications as it requires various parameters which must be set beforehand of its use.

Alnajar, F., Gevers, T., Valenti, R., & Ghebreab, S. (2013). *Calibration-free gaze estimation using human gaze patterns.* Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.

Baluja, S., & Pomerleau, D. (1994). Non-intrusive gaze tracking using artificial neural networks.

Duchowski, A. T. (2002). A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments, & Computers, 34*(4), 455-470.

Hansen, D. W., & Ji, Q. (2010). In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence, 32*(3), 478-500.

Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., & Van de Weijer, J. (2011). *Eye tracking: A comprehensive guide to methods and measures*: OUP Oxford.

Huang, M. X., Kwok, T. C., Ngai, G., Chan, S. C., & Leong, H. V. (2016). *Building a personalized, auto-calibrating eye tracker from user interactions.* Paper presented at the Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems.

Li, D., Winfield, D., & Parkhurst, D. J. (2005). *Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches.* Paper presented at the Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on.

Marple-Horvat, D. E., Gilbey, S. L., & Hollands, M. A. (1996). A method for automatic identification of saccades from eye movement recordings. *Journal of Neuroscience Methods, 67*(2), 191-195.

Nyström, M., & Holmqvist, K. (2010). An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data. *Behavior research methods, 42*(1), 188-204.

Olsson, P. (2007). Real-time and offline filters for eye tracking.

Salvucci, D. D., & Goldberg, J. H. (2000). *Identifying fixations and saccades in eye-tracking protocols.* Paper presented at the Proceedings of the 2000 symposium on Eye tracking research & applications.

Smeets, J. B., & Hooge, I. T. (2003). Nature of variability in saccades. *Journal of neurophysiology, 90*(1), 12-20.

Xu, P., Ehinger, K. A., Zhang, Y., Finkelstein, A., Kulkarni, S. R., & Xiao, J. (2015). Turkergaze: Crowdsourcing saliency with webcam based eye tracking. *arXiv preprint arXiv:1504.06755*.