Eye Movement Tracking Using Machine Learning Algorithms Report

Supervisor: Dr Benmarrakchi FatimaEzzahra

Learning involves the acquisition of information, knowledge, skills, and attitudes through the interaction between what is taught to the student and their existing ideas or concepts (Posner, G. J., et al., 1982). Many research studies have explored how students' current ideas or concepts evolve in response to new evidence or ideas. Posner et al. (1982) suggested two ways in which students can modify their existing ideas or concepts to incorporate new ones. The first way is through assimilation, where students use their existing concepts or ideas to understand new phenomena. The second approach is accommodation, which occurs when a student's current concepts or ideas are insufficient to grasp new concepts, leading them to replace or reorganize their central concepts.

Traditionally, researchers in the field of education have relied on cognitive interviewing, specifically the think-aloud interviewing method, to explore cognitive activity during learning. This method aims to assess how new concepts or ideas are accommodated and transferred to students (Willis, G. B., 2004). However, the validity of this method has been questioned, prompting educational researchers to explore alternative research methods from various academic domains to gain different perspectives on the learning process (Posner, G. J., et al., 1982).

Eye trackers are tools used to record individuals' eye movements as they interact with various objects such as texts, images, humans, computers, or machines. They have been extensively employed to study cognitive processing during reading and other information-processing tasks (Neuert, C. E., et al., 2016). The study of eye movements during reading has a long and rich history, dating back to the late 19th century (Rayner, K., 2009). Rayner (1998) proposed that there have been three eras of research on eye movements during reading. However, it is evident that we are now in the fourth era, characterized by the dominance of sophisticated computational models of eye movement control (Rayner, K., 2009).

The initial phase (1897-1960) of studying eye movement trackers during reading provided the fundamental investigations necessary for comprehending this phenomenon. In this first era, researchers made significant discoveries regarding the basics of eye movements during reading. They also focused on crucial aspects such as the perceptual span, saccade latency, and saccadic suppression (Rayner, 2009). Woodworth (1899) played a prominent role in conducting one of the pioneering eye movement tracking experiments using a mirror to monitor participants' eye movements while observing various objects during experiments.

The subsequent phase (1960 - 1990) marked the emergence of the first portable eye tracker device, specifically designed to examine eye movement during text reading (Duchowski, 2002). This device enabled researchers to track participants' eye movements as they engaged with texts in various formats. This era was distinguished by a shift towards applied research, aligning with the behaviourist movement in experimental psychology (Duchowski, 2002).

The following phase (1990-2010) experienced the introduction of more advanced eye tracker devices compared to the previous era. Notable advancements were made in eye movement recording systems, along with improvements in computer systems, resulting in enhanced accuracy and ease of measurement (Rayner, 2009). These technological breakthroughs also facilitated the development of innovative techniques, such as dynamically changing the visual display based on eye position (Rayner, 2009). Consequently, a wide range of eye tracker devices became available during this era, broadly classified as diagnostic and interpretive eye tracking devices (Duchowski, 2002).

As we mentioned earlier, the current phase (2010 - present) signifies the fourth era of eye movement research during reading, which commenced in the late 1990s. This era is characterized by the emergence of intricate and sophisticated models, often in the form of implemented computer simulations. A significant portion of the research conducted in the past decade has been dedicated to validating or refuting these models, specifically in relation to eye movements during reading (Rayner, 2009).

Eye-tracking technology has achieved remarkable success in diverse fields such as neuroscience, psychology, and computer science (Duchowski, A. T., 2002). Numerous eye-tracking devices and techniques have been developed, including the following:

1. The pupil-based eye tracker is specifically designed to monitor the movement of the pupil, which is the black dot in the eye, during eye motion. This technique has been applied in various studies, including research examining differences in brain activity and eye movement during active social interaction compared to passive observation (Tylén, K., et al., 2012).
2. Eye-gaze trackers are designed to monitor the eye's position by detecting the direction of focus. They offer higher accuracy compared to pupil-based eye trackers. Hyönä, J. (1995) utilized this technique in their study titled "An Eye Movement Analysis of Topic-Shift Effect During repeated reading." The research aimed to examine how participants' eye movements were affected when they encountered new topics within the same passages (Hyönä, J., 1995).

3. Head-mounted eye trackers are wearable devices that utilize mounted cameras on glasses to track eye movements. In a study examining the visual direction of drivers during steering, researchers employed head-mounted eye trackers to monitor the real-time position of the driver's eye (Land, M. F., et al., 1994).
4. Machine learning algorithms offer the potential to enhance the precision and effectiveness of the aforementioned eye-tracking devices. These algorithms enable automated identification of various eye movements such as fixations, saccades, and smooth pursuit, thereby facilitating the recognition of indicators for eye fatigue, including reduced pupil diameter and blink rate (Tang, J., et al., 2010; Khan, M. I., et al., 2018; Zhang, Z., et al., 2016; Devi, M. S., et al., 2008). Several widely employed machine learning algorithms include:

i. Support Vector Machines (SVMs) are effective supervised machine learning models utilized for classifying distinct patterns of eye movements such as fixations, saccades, and smooth pursuits (Memon, Q. 2019).
ii. Random Forest is an ensemble learning algorithm that employs numerous decision trees to classify and predict different eye movements (Zemblys, R., et al., 2016).
iii. Convolutional Neural Networks (CNN) are deep learning algorithms specifically designed to classify various eye movements, including gaze estimation, fixation detection, and object recognition in eye tracking devices (George, A., et al., 2016).

In this experiment, we will utilize a type of deep neural network known as a Convolutional Neural Network (CNN) to implement the eye-tracking algorithm. The unique aspect of CNN lies in its ability to process images in their original structure, taking into account their inherent features. The neurons within CNNs are organized across three dimensions: width, height, and depth. Each neuron within the current layer establishes connections with a small section of the output originating from the previous layer. This process resembles placing an NxN filter over the input image. This is distinct from fully connected layers, where every neuron forms connections with all neurons in the prior layer (Joshi, P. 2017).

When constructing a Convolutional Neural Network (CNN) for our eye-tracking algorithm, it's important to design the network with these specific layers:

- Input Layer: This initial layer takes in the image data.
- Convolutional Layer: After the input layer, this next layer processes the input image. It convolves a filter matrix kernel across the image pixels, generating new pixel values through the dot product of the weights and a portion of the previous input layer.
- Rectified Linear Layer: The output from the convolutional layer enters this layer. It functions as an activation layer, introducing non-linearity to enhance backpropagation and reduce loss.

- Pooling Layer: Between the mentioned layers, we can include a pooling layer. This layer decreases the depth of the previous layer. For instance, a 2x2 Maxpooling layer reduces the preceding layer's dimensions by half.
- Fully Connected Layer: The last layer in this architecture is the fully connected layer, responsible for producing the desired output classes.

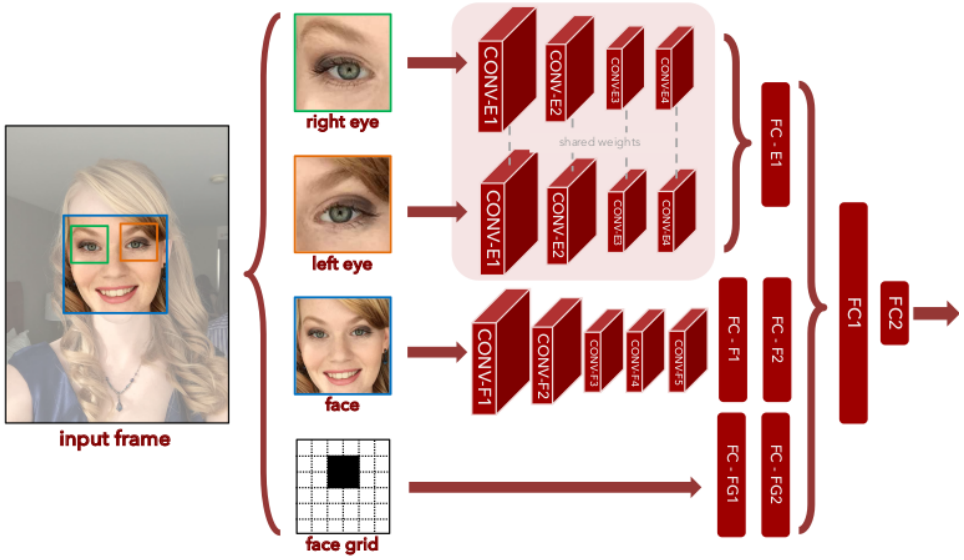

Figure 1: A visual depiction displaying the architectural layers of a convolutional neural network, following the earlier explanation (image credit: Krafka, K., et al., 2016).

Model Implementation

When incorporating a model, it's advantageous to start with an existing predefined model that has been created using a similar dataset of your interest. This process is referred to as Transfer Learning. It involves utilizing an already pretrained model by employing its lower layers as a foundation and adjusting only the outermost layer (fully connected layer) based on the number of outputs or classification you're concerned with. For this model implementation, we will be building upon a pretrained model within the TensorFlow machine learning framework known as Visual Geometry Group 16 (Vgg16). Vgg16 constitutes a deep convolutional neural network architecture tailored for image classification. This model was trained with an extensive 16 million parameters (Simonyan, K., et al., 2014).

Data Collection and Data Preprocessing

Keeping our model's objective in mind — accurately identifying and classifying individuals' faces through a webcam — we opted to construct a dataset of varied facial poses utilizing the OpenCV library and our webcam. While we initially aimed to capture diverse faces, we

eventually chose to focus on a single individual's face and employed image augmentation methods to generate diverse versions of that face. Initially, our training dataset comprised 42 images of this individual's face, while the test and validation datasets each consisted of 9 images of the same individual.

In machine learning, to ensure the model's efficacy, it's essential to partition the dataset into three main segments: training, testing, and validation datasets, distributed in an 80:10:10 ratio as described earlier. This division is executed to guarantee thorough training of the model, enabling its effective generalization in real-world scenarios. The validation dataset plays a crucial role in detecting any overfitting of the model to the training dataset during the training process. On the other hand, the test dataset is utilized to verify the model's accuracy post-training.

Following this, the subsequent strategy involves augmenting each sample's quantity by applying diverse transformations such as scaling, cropping, resizing, rotation, and others to various versions of each sample. This augmentation procedure yields a collective count of 5,040 samples for the training dataset, alongside 1,080 samples for both the test and validation datasets. This augmentation substantially augments our data samples, enhancing the training potential for our model.

Training the Model

As previously mentioned, the model will be constructed using the pre-trained Vgg16 model from the TensorFlow framework. The model will be configured with the following hyperparameters:

- Batch size: 8
- Epochs: 10
- Learning rate: 0.003
- Optimizer: Adam
- Loss function: Binary cross entropy loss

Results and Conclusion

Following the training phase, it's a common practice to assess the model's performance by evaluating it on a previously unseen dataset—this is where the test dataset comes into play. This dataset provides insights into how accurately the model can identify similar objects in real-world scenarios. The summary outlining its performance is presented below.

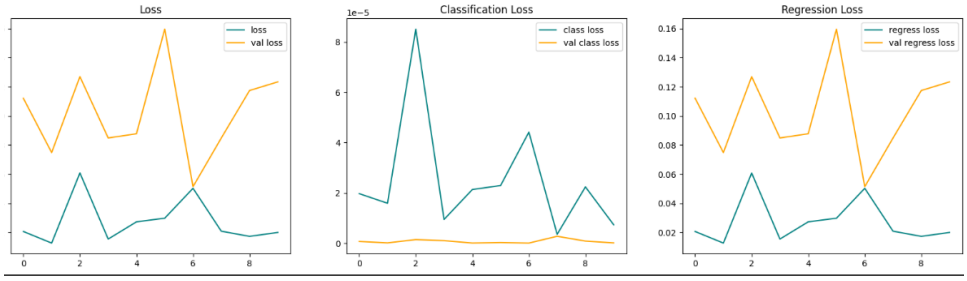

Figure 2: A representation displaying the classification loss performance of the model on the training dataset compared to the validation dataset.

Looking at the graph in Figure 2, it's noticeable that the loss of the training dataset(loss) and the validation dataset (val class loss) aren't converging satisfactorily, particularly in the first and third graphs (loss and regression loss). However, they exhibit better convergence in the classification loss. This indicates that the model's accuracy isn't sufficient to achieve effective convergence. This suggests that further adjustment of the model's hyperparameters is required to attain improved outcomes.

Assessing Model Performance on the Test Dataset

An additional measure to gauge the model's effectiveness involves evaluating it with a fresh set of data that it hasn't encountered before. This assessment was conducted through two methods. Initially, the test dataset partitioned during data preparation was utilized. Subsequently, the model's real-time performance was confirmed by applying it with a webcam.

The first method yielded positive outcomes, as illustrated in the image below.

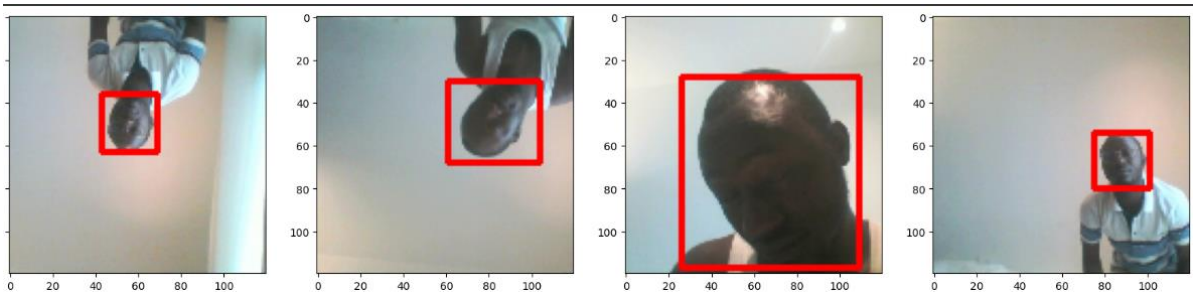

Figure 3: Subset of dataset displaying annotations labeled during the testing phase.

In conclusion, this project provided an opportunity to delve deeper into the realm of deep neural networks and their potential as efficient learning tools for students. Through this model, we are capable of discerning and quantifying students' attention levels within a digital learning setting. This empowers us to identify instances of diminished concentration. Additionally, this understanding can guide the creation of more impactful learning materials, optimizing attentiveness by addressing periods of distraction. Naturally, this model isn't flawless; there's room for improvement to achieve enhanced accuracy. Striving for better results, we aim to refine the model for improved convergence and reduced loss across both the training and validation datasets.

References

Devi, M. S., & Bajaj, P. R. (2008, July). Driver fatigue detection based on eye tracking. In *2008 First International Conference on Emerging Trends in Engineering and Technology* (pp. 649-652). IEEE.

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

George, A., & Routray, A. (2016, June). Real-time eye gaze direction classification using convolutional neural network. In *2016 International Conference on Signal Processing and Communications (SPCOM)* (pp. 1-5). IEEE.

Hyönä, J. (1995). An eye movement analysis of topic-shift effect during repeated reading. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *21*(5), 1365.
Joshi, P. (2017). *Artificial intelligence with python*. Packt Publishing Ltd.

Khan, M. I., & Mansoor, A. B. (2008). Real time eyes tracking and classification for driver fatigue detection. In *Image Analysis and Recognition: 5th International Conference, ICIAR 2008, Póvoa de Varzim, Portugal, June 25-27, 2008. Proceedings 5* (pp. 729-738). Springer Berlin Heidelberg.

Krafka, K., Khosla, A., Kellnhofer, P., Kannan, H., Bhandarkar, S., Matusik, W., & Torralba, A. (2016). Eye tracking for everyone. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2176-2184).

Land, M. F., & Lee, D. N. (1994). Where we look when we steer. *Nature*, *369*(6483), 742-744.

Memon, Q. (2019). On assisted living of paralyzed persons through real-time eye features tracking and classification using support vector machines: array. *Medical Technologies Journal*, *3*(1), 316-333.

Neuert, C. E., & Lenzner, T. (2016). Incorporating eye tracking into cognitive interviewing to pretest survey questions. *International Journal of Social Research Methodology*, *19*(5), 501-519.

Posner, G. J., Strike, K. A., Hewson, P. W., & Gertzog, W. A. (1982). Toward a theory of conceptual change. *Science education*, *66*(2), 211-227.
Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, *124*(3), 372..

Rayner, K. (2009). Eye movements in reading: Models and data. *Journal of eye movement research*, *2*(5), 1.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Tang, J., Fang, Z., Hu, S., & Sun, Y. (2010, August). Driver fatigue detection algorithm based on eye features. In *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery* (Vol. 5, pp. 2308-2311). IEEE.

Tylén, K., Allen, M., Hunter, B. K., & Roepstorff, A. (2012). Interaction vs. observation: distinctive modes of social cognition in human brain and behavior? A combined fMRI and eye-tracking study. *Frontiers in human neuroscience*, *6*, 331.

Willis, G. B. (2004). *Cognitive interviewing: A tool for improving questionnaire design*. sage publications.

Woodworth, R. S. (1899). Accuracy of voluntary movement. *The Psychological Review: Monograph Supplements*, *3*(3), i.

Zemblys, R., Niehorster, D. C., & Holmqvist, K. (2016). Detection of oculomotor events using random forest. In *The Scandinavian Workshop on Applied Eye Tracking 2016*.

Zhang, Z., & Zhang, J. (2006, June). A new real-time eye tracking for driver fatigue detection. In *2006 6th International Conference on ITS Telecommunications* (pp. 8-11). IEEE.