FACIAL RECOGNITION FOR STUDENT ENGAGEMENT

Jiachun Guo Jianan Wu, Yifeng Tan and Sasanka Munasinghe

Department of Electrical and Computer Engineering, University of Rochester,
Rochester, New York, United States

ABSTRACT

Active student engagement is pivotal for enhancing learning outcomes, academic performance, and overall satisfaction within educational environments. Traditional assessment methods may lack objectivity or accuracy, such as teacher subjective observations, failing to provide a comprehensive understanding of student engagement. This paper explores the application of facial recognition technology to enhance the student engagement in class, which lead to a better academic performance. Specifically, the technology leverages three key models: facial ID recognition, emotion recognition, and facial orientation recognition. Utilizing convolutional neural networks (CNNS), this technology enables a precise, data-driven approach to determine the students' engagement, which can help educators have a better understanding of their students and change strategies to enhance students' academic performance.

Index Terms— Student Engagement, Face ID Recognition, Emotion Recognition, Facial Orientation Detection

1. INTRODUCTION

In education, maintaining student engagement is critical to effective learning. Educators are finding it increasingly difficult to gauge student attention as online classes have grown in popularity. Traditional methods, such as asking questions or requesting that students turn on their cameras, frequently fall short of providing accurate insights into student engagement.

To address this issue, we're using modern methods like facial recognition, emotion detection, and head tracking. Using deep learning models, our project aims to create an automated system capable of accurately identifying students, detecting their emotions, and tracking their head movements in real-time during in-person and online classes as well.

Through this innovative approach, we hope to provide teachers with valuable insights into student engagement levels, allowing them to adjust their teaching methods accordingly. Educators can optimize online learning experiences and keep students actively engaged in the learning process by understanding when they are focused and when they may require additional assistance. We hope that the results of these models will help educators and students improve their learning experiences and transform the learning process.

2. METHODOLOGY FOR EVALUATING STUDENT ENGAGEMENT

Active student participation plays a crucial role in academic achievement. To achieve a more objective and precise understanding of student engagement, we combine three models to achieve a more accurate assessment of student engagement, as shown in Figure 1.

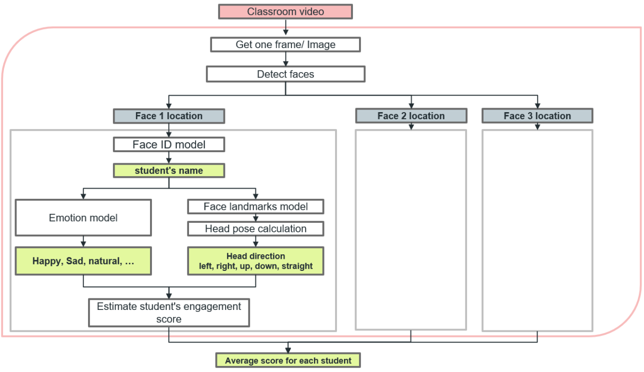


Figure 1 methodology for evaluating student engagement

To identify different students in a video, we first need a camera to record the class. Then we implement a Face ID recognition model. This model can extract different faces and identify different faces so that in the following steps, each student is identified and get their own score.

The emotion model can provide valuable data to help identify whether students are paying attention in class. In a practical classroom setting, it is rare for students to laugh in a disruptive manner. This kind of behavior is a considerable factor in the calculation process.

The facial orientation estimation is the key factor which decides whether students are engaging in class. If students are turning their head around or looking up or down, they are considered not paying attention. In the final calculation process, this is the most influential factor. To successfully estimate students' facial orientation, we implement a face landmark model in relations to a criteria. First, the model extracts landmarks from facial features. Then, the landmarks are evaluated by the criteria to get our final output.

The final step is to combine the output of these data together. On the scale of 10 points, points are deducted based on the output of the three models.

3. MODEL SELECTION AND TRAINING

3.1 Face ID Recognition

To determine the optimal face model, four distinct face ID models were trained using the VGG16, VGG19, mobilenetV2 and a custom CNN architecture. These models were trained on a dataset of celebrity images, which can be found at the following link: https://www.kaggle.com/datasets/vasukipatel/face-recognition-dataset/data. The dataset includes 2562 images, with 31 celebrities. During the training, Keras callback functions were utilized to prevent the issue of overfitting in the dataset. The model checkpoint callback has been used to extract the optimal model during training by monitoring the validation loss. In addition, the ReduceLROnPlateau function was employed to modify the learning rate dynamically once the model performance metrics no longer showed improvement. It will decrease the learning rate according to the metrics.

VGG16 and VGG19 are well-known neural network models for image recognition, characterized by their respective 16 and 19 convolutional layers. The implement model was designed to accept images with dimensions of 128x128 pixels and 3 layers for each color. At the end of these models, we implemented three fully connected dense layers. The first layer consists of 1024 neurons, the second layer has 128 neurons, and the third layer has 31 neurons, which represents the number of classes. In addition, the performance of mobilenetV2 was evaluated in a similar manner. The custom CNN model, which was the 4th model, has 5 convolutional layers, each layer was subsequently followed by Batch Normalization and Max Pooling. The activation function used for these layers was Relu. Finally, three dense layers were connected to the bottom using Batch Normalization with ReLU activation function. Next, an additional dense layer was added to the neural network, with the number of neurons equal to the number of prediction classes. This layer was designated as the output layer and utilized the SoftMax activation function.

Various augmentation techniques, such as shear, zoom, flip, shift, and rotation, were used to improve the robustness of the model. In addition, random brightness augmentation was included to account for potential variations in lighting conditions during the real word scenarios. Then the dataset was trained for an appropriate number of epochs until the model reached convergence and the loss flattens, while being careful to not overfit.

Results from the Kaggle celebrity dataset are shown in Table 1. The VGG16 model and the custom CNN achieved the highest performance when applied to the celebrity dataset. Furthermore, we have observed that initializing the VGG16 and VGG19 models with pre-trained weights for a different image recognition task leads to faster convergence of the model and generally higher accuracy. Therefore, it was decided to initialize the weights of the CNN layers with "imagenet" weights and only train the fully connected dense layers. This approach is commonly referred to as transfer learning. However, for the custom CNN, all layers were trained with the model.

Table 1 Comparison of accuracy among different models

Model architecture	**Train accuracy**	**Validation Accuracy**
VGG 16	98.71%	44.29%
VGG 19	78.86%	36.07%
Mobilenet v2	87.96%	35.25%
Custom CNN	99.88%	44.26%

The best performing model was then trained on a real dataset of student faces with five different students, each class with approximately 40 images from various angles before augmentation. The same procedure described above was used for training but with augmentations that had a wider range than the previous dataset.

The figure 2 shows the loss and accuracy curve for the VGG16 architecture model with real student dataset. The best model shows a validation accuracy of 95 %.

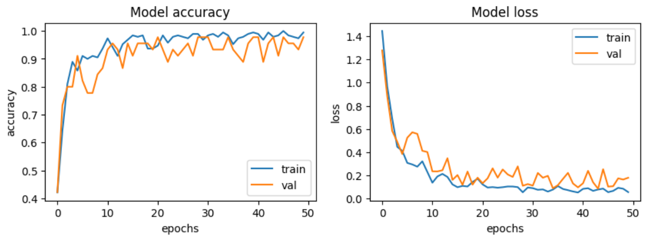


Figure 2 The loss and accuracy curves for VGG16

The difference in Kaggle celebrity between training and validation performance suggests overfitting, which occurs when the model memorizes rather than learns to generalize from the training dataset. This issue can be worsened by insufficient or inconsistent regularization of training data diversity. However, due to the limited number of classes, this issue is not visible in the real student face ID model.

These trained models are only suitable with cropped faces that contain only the face in the input image. Hence, it is necessary to implement a method for detecting faces from images. Previously, we used the OpenCV library Haar cascade detection with a pre-trained classifier. This detects faces by examining the various features that are commonly found on human faces. However, due to the poor detection performance with side faces, it was later decided to detect faces using a pre-trained deep neural network model.

These pre-trained models provide the coordinates of the faces of the given image. Then, the Face Id model was applied to the cropped image, which contained only the student's face, to identify their name.

3.2 Emotion recognition

Our model uses a custom CNN model, designed for efficient image recognition. It consists of four convolutional layers, with filter sizes increased from 64 to 512, supplemented by max pooling, dropout layers, and ReLU activation functions to enhance learning and prevent overfitting. This setup develops to a fully connected layer that divides the image into seven emotional states: anger, disgust, fear, happiness, neutral, sadness, and surprise.

The model was trained over 50 epochs using an Adam optimizer with a learning rate of 0.0001. Callbacks for model checkpoints and learning rate adjustments help optimize performance. Finally, the training accuracy of the model is 75.91%, the validation accuracy is 65.60%, and the model has good generalization ability, as shown in Figure 3.

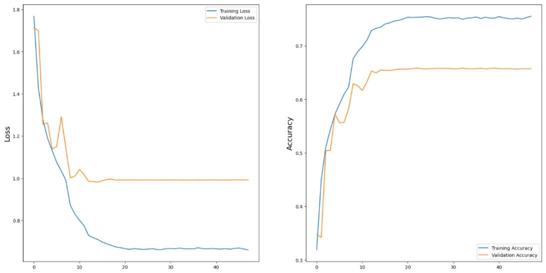


Figure 3 The loss and accuracy curves for emotion recognition model

The performance on test sets demonstrated high accuracy, particularly in the identification of "pleasure" and "surprise." However, the confusion matrix reveals certain misclassifications between similar emotions, like "sad" and "neutral," highlighting areas for potential improvement, as depicted in Figure 4.

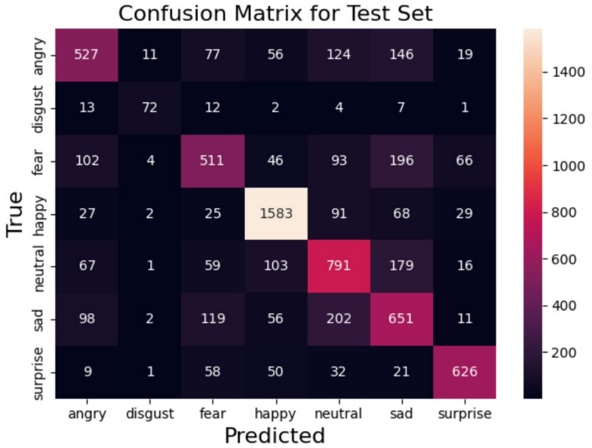


Figure 4 The confusion matrix for test set

3.3 Facial orientation recognition

When predicting student engagement in class, it is important to not only look at their facial expressions, but also determine whether they are looking around or looking down to think. This is where facial marker detection becomes crucial. The data obtained from this detection can be used to determine whether a student is looking around or lowering their head. We can train a model to get the desired facial landmark data.

The architecture chosen for face landmark detection is MobileNetV2, and we tuned the network to output face landmark coordinates of 106 x and y coordinates, which means a total output of 212. Because we need the model to output continuous coordinate values rather than discrete categories, we replace the original classification layer with a new fully connected layer that outputs facial landmark coordinates. This tuning makes the model better suited to handling regression tasks.

Firstly, we split the data into training sets, validation sets, and test sets. Then, we resize the images to 512x512 pixels and normalize them according to the original image size. To enhance data diversity, data augmentation techniques are employed. Subsequently, we convert the images to PIL format and adjust the range of image data from 0-255 to 0-1 to accelerate convergence. The data is processed in batches of 16 samples, and the model is trained for 25 epochs. If the validation loss does not improve for 5 consecutive epochs, training is halted to prevent overfitting, thus conserving computational resources and time. After that, the best model is loaded and executed on the test set. Finally, training and validation loss curves are plotted, and the predicted images are compared with the test data.

Figure 5 illustrates the training and validation loss curves of the landmarks model. The loss consistently decreases as the number of epochs increases.

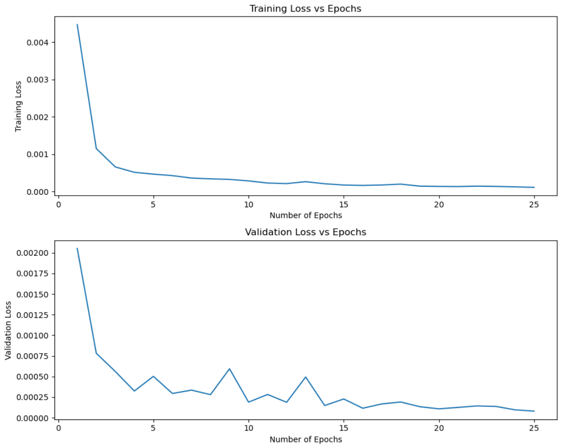


Figure 5 Training and validation loss curves of the landmarks model

In Figure 6, red dots represent original landmarks and blue dots represent predicted landmarks, indicating the model's ability to accurately predict facial landmarks.

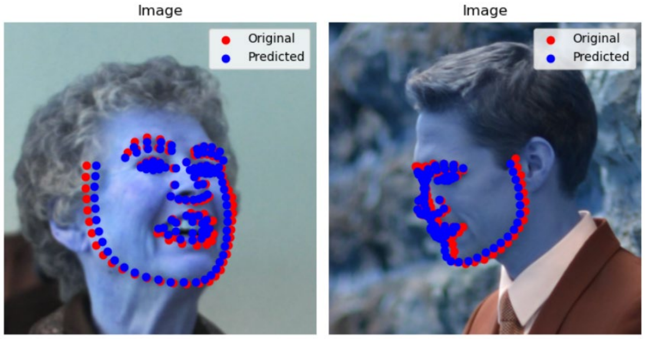


Figure 6 Thecomparation between original and predicted landmarks

Next, we select four landmarks from the predicted facial landmarks, as shown in Figure 7: point a (left eyebrow center), point b (right eyebrow center), point c (nose tip), and point d (chin).

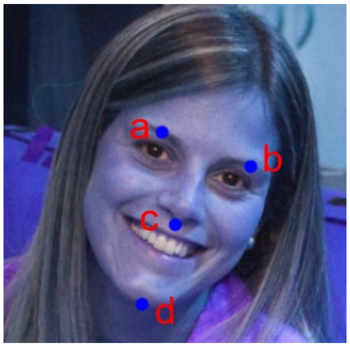


Figure7 The four points used to estimate facial oritation

Generally, if the face is directly facing the camera, the distance between points a and c should equal the distance between points b and c. If the face is turned to the left or right, these distances will no longer be equal; if turned left, ac will be smaller than bc, and if turned right, ac will be larger than bc. Similarly, we assume that when the face is facing the camera directly, the distance between points c and d equals the distance between point c and the midpoint of line ab. Tilting the head up or down will alter this equality relationship. Based on these assumptions, we design the criteria outlined in Table 2 to evaluate facial orientation, where m is the midpoint of ab.

Table 2 criteria of evaluating facial orientation

Criteria	Facial Orientation
ac /bc < 0.95	Left
ac/bc>1.05	Right
Otherwise	Forward
cd/cm>1.15	Upward
cd/cm<0.85	Downward
Otherwise	Forward

There are two features to describe facial orientation: the first one indicates the direction the face is facing (left, forward, right), and the second one indicates whether the face is facing upward, forward, or downward. As indicated by the annotations above Figure 8, our designed criteria can provide a relatively accurate estimation of facial orientation.

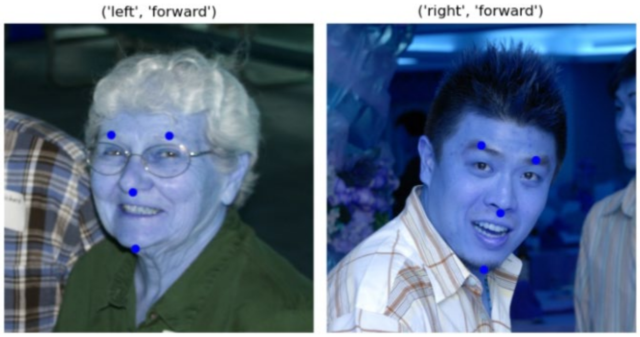
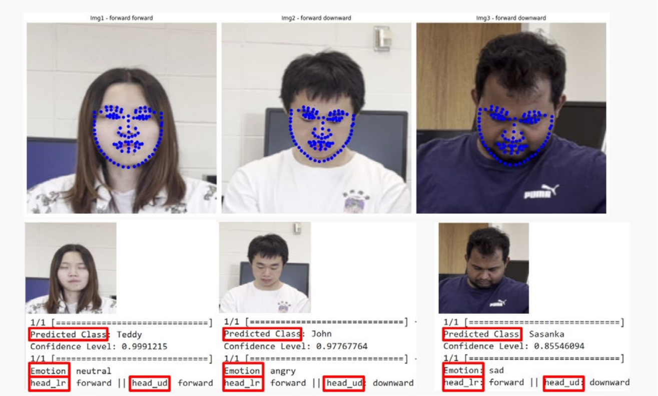


Figure 8 The results of predicted facial orientation

ANALYSIS OF STUDENT ENGAGEMENT

After running through the three model, we get multiple output. In Figure 9, the blue dot in the image represents the predicted face landmarks. The output of the face ID model is printed above the image, while the outputs of the emotion model and facial orientation calculation are printed below the images.



```
1/1 [==============================]
Predicted Class: Teddy
Confidence Level: 0.9991215
1/1 [==============================]
Emotion: neutral
head_lr: forward || head_ud: forward
```

```
1/1 [==============================]
Predicted Class: John
Confidence Level: 0.97767764
1/1 [==============================]
Emotion: angry
head_lr: forward || head_ud: downward
```

```
1/1 [==============================]
Predicted Class: Sasanka
Confidence Level: 0.85546094
1/1 [==============================]
Emotion: sad
head_lr: forward || head_ud: downward
```

Figure 9 The predicted results of ID, emotion and facial orientation

Next, we will convert the emotion outputs from the model into numerical data. We merge expressions of anger, fear, and disgust into "angry", happiness and surprise into "happy", and neutral and sadness into "neutral", because most of the emotions will not be useful to determine the student engagement and may not happen in a real classroom scenario, as shown in Table 3.

Table 3 emotion combination

Emotion	Combined Emotion
angry, fear, disgust	angry
happy, surprise	happy
neutral, sad	neutral

Initially, students' engagement scores are set at 10 per frame. Points will be deducted if a student displays any inappropriate emotion for the classroom setting, as shown in Table 4.

Table 4 Engagement Rating Criteria

Condition	Score Adjustment
Emotion is 'angry'	-1
Emotion is 'happy'	-1.5
Emotion is 'neutral'	0
Head_lr is 'left' or 'right'	-2
Head_lr is 'forward'	0
Head_ud is 'upward' or 'downward'	-2
Head_ud is 'forward'	0

To provide a more intuitive reflection of students' engagement throughout an entire class, we calculate the average engagement score over time by selecting one frame per second from a video. Use the equation: Average Score = Current Cumulative Score / (10 * Current Frame Number) to derive a final score, which is the average score over time for each student across the entire video.

Figure 10 shows the predicted results of student engagement score in a certain frame.

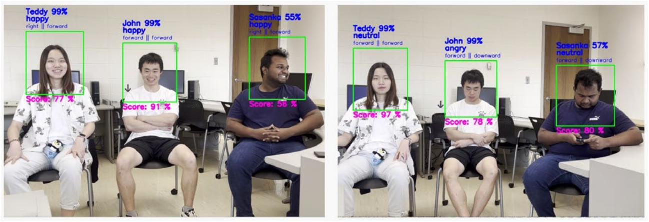


Figure 10 Engagement Score Prediction

Figure 11 illustrate the changes in average scores over time for three students in two different videos.

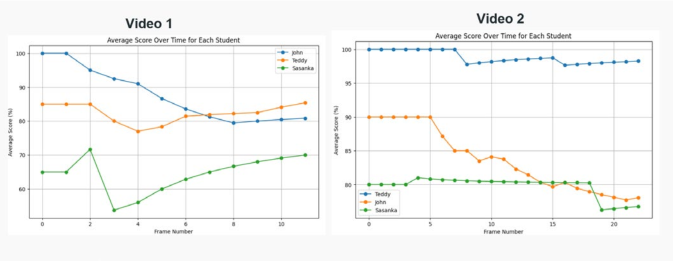


Figure 11 Average Score Over Time

CONCLUSION

In conclusion, our comprehensive approach, which combines face recognition techniques with emotion analysis and face direction detection, provides a comprehensive solution for measuring student engagement in educational settings. By leveraging the latest machine learning algorithms and innovative methodologies, we hope to provide educators with valuable insights into student behavior and learning experiences, allowing for more personalized and effective educational approaches.

In the future, we need to find a model capable of identifying subtle changes in facial expressions. This capability will enable us to utilize real classroom student expressions as datasets, indicating the practical applicability of our project.

REFERENCES

[1] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444.

[2] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. Journal of Big Data, 6(1), 60.

[3] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929-1958.

CONTRIBUTION

Report: Jiachun Guo & Sasanka Munasinghe & Yifeng Tan &Jianan Wu

Code: Jiachun Guo & Sasanka Munasinghe Jianan Wu &Yifeng Tan

Slide: Jianan Wu