Final Report

DSC 412

Bryce Yang

1 Abstract

The purpose of this project was to demonstrate how real-time computer vision can be used in academic settings to enhance the productivity of students while studying, while simultaneously displaying how computer vision models operate. While the potential of this project requires more time to fully realize, it successfully demonstrates the inner workings of computer vision and how it can be used in an educational setting.

2 Background

Inefficient studying can have a significant negative impact on a student's life. Wasted hours spent using ineffective methods can lead to frustration and burnout, making learning feel more like a chore than a rewarding process. With AI being used in educational environments in the form of LLMs, it is important to consider the other applications it may have. The purpose of this project is to explore one of those potential areas.

3 Data Analysis

Given that this project is oriented around computer vision models, the data being analyzed is a collection of images. The most difficult part of the process is effectively gathering enough images to create a training, validation, and test set.

To accomplish this, I utilized the iCrawler library, a python-embedded web scraper that can gather images from both Google and Bing. I ran the web scraper to gather images related to human faces, until I had 350 images total. I also imported a Kaggle dataset of 4,000 random non-face images to complete the data.

4 Data Augmentation

Because 350 images of human faces wasn't enough for training, I had to create augmentations to artificially enlarge my dataset. The augmentations I applied changed the tilt and hue of the images, resulting in a dataset of 3,500. This was then combined with the Kaggle dataset, and a final size of 7,500 images.

5 Model Selection

This project orients around Convolutional Neural Networks (CNNs), which are the primary models used for image interpretation. I utilized transfer learning to build on top of a more generalized pre-trained model, which allows for the model to be specified for my use case while reducing training time and resources. For this design, I followed the design example shown on PyTorch's website [1].

6 Training Methodology

I trained my model using a batch size of 50 and 5 epochs. For commercial-level models, a batch size of 30 recommended, but because of the small scale and limited resources I found this to be satisfactory. I experimented with more epochs and smaller batches but found performance to only slightly improve, while taking many times longer to finish running. Averaging a 3 minute runtime per epoch with a batch size of 50, I found this to be optimal for performance and time.

To avoid underfitting and overfitting, I created a validation test set to track the accuracy of the model.

7 Results

With this model, I was able to achieve an accuracy of 100% on my test set. However it must be noted that the accuracy is likely inflated due to the size of the test set only being around 700 images, all of which only vary slightly from images within the training and validation sets.

For my application, I was able to successfully create a real-time video processing app, which allows for video processing, three choices of model, adjustable model confidence, and real-time webcam processing. This application succeeds in presenting the idea of how computer vision can be used in an academic setting, but is not yet developed to the point of being useful.

8 Future Work

I believe this model is limited primarily by the data it was trained in - the web scraping process resulted in a decent proportion of the set being low quality images, some of which were AI generated or faces that were not human. It's possible that instead of recognizing whether an image contains a face, the model recognizes whether an image came from the Kaggle dataset.

In the future, I plan to gather my data myself, which will be possible if I utilize the same image augmentations which I used on this project. This way I can hand-pick every image that enters the dataset, improving the quality of the dataset.

For the application, I plan on furthering the applicable functionalities. I would accomplish this by tracking head positions, then assigning a score for how long the user is focused on the screen.

9 Stakeholder Acknowledgments

This technology could benefit a wide range of people, from high school students to university researchers to industry professionals. Anyone who struggles with staying focused during long study or work sessions could find value in personalized feedback that helps them optimize their time.

10 Conclusion

In conclusion, this project was conducted as a demonstration of how computer vision can be applied in educational environments, just as LLMs have revolutionized the learning process. While this project will require more time to have its potential fully realized, it paints a picture of what this technology is and how it may be applied in the future.

11 Citations

[1] "Transfer Learning for Computer Vision Tutorial," PyTorch Tutorials, PyTorch. [Online]. Available: https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html. [Accessed: Nov. 11, 2024].