**Portfolio (Option 2)**

Alec Meyer

Colorado State University Global Campus

CSC515

Dr. Jack Li

6/8/2025

**Techniques Used**

**Feature Detection**

OpenCV's cascade classifiers were used to identify the subject's face and eyes. More specifically, Haar cascade classifiers are utilized from the OpenCV library, enabling the detection of specific facial features. A Haar cascade classifier works by calculating Haar features that create integral images. Then, an algorithm will determine which of these integral images are the most important for the specific classification task (Mittal, 2020). These Haar cascade classifiers function similarly to image segmentation, where the algorithm looks for varying pixel intensities along edges to define specific features. However, Haar cascade filters are much more complex than a trivial region-based segmentation algorithm.

**Feature Blurring**

To blur the eyes of each human subject, a simple box filter was applied to the specific region of interest. Using the Haar cascade classifier, the location and dimensions of the eyes can be determined. That location and dimension are then used to create a sub-image of just the subject's eye, and then that entire eye image is blurred using the box filter. That blurred image is overlaid on the original image to show the original image with blurred eyes. Depending on the size of the image, a different kernel was needed for blurring, as an image with more pixels per eye would require a relatively larger kernel to blur the eyes effectively.

**Challenges**

Throughoutthe implementation of this project, several major hurdles were faced. The first major challenge faced was having images that were not large enough for the cascade classifiers to function correctly. When examining the original images, they were straightforward to identify. However, when the faces were isolated, the images became very pixelated, making it difficult to discern features. This issue was overcome by using larger images and verifying that when the face was singled out, the eyes remained highly identifiable.

Another challenge faced was handling false positives reported by the cascade classifier. According to Rosebrock, in 2021, Haar cascade classifiers are extremely fast but suffer from reporting many false positives as their accuracy has been traded off for their speed. Resolving this issue required extensive hyperparameter tuning and testing. The image requirements for this project necessitated a different set of hyperparameters per classification, as different lighting conditions and scales affected the classifier. The primary method used to handle most false positives was the minSize(h, w) and maxSize(h, w) functions, as they ignore any classifications outside the specified bounds. Using domain knowledge, the pixel sizes of the eye areas were mainly known, so by using these bounds; many false positives were removed.

**Accuracy**

As previously mentioned, Haar cascade classifiers lose some of their accuracy for the incredible speed of the classifier. However, even with the accuracy tradeoff, the cascade classifiers performed well in detecting eyes for the three types of images. The Haar cascade classifier was able to identify nine out of the ten eyes present in the image, which contained multiple subjects. The reason the classifier failed to identify the tenth eye is because that specific subject was wearing glasses. Out of the five subjects, only one false positive was reported using the specific bounds as it was of similar size to the subject's eyes. The image with poor lighting, where the subject's background is natural light, performed well but did require a much larger and more specific bound for detecting the subject's face. Once the cascade classifier was able to identify the face, identifying the eyes did not require any additional hyperparameter tuning. The final image of a subject standing far away and showing their whole body also performed well with the Haar cascade classifier. The only hyperparameter tuning required so far for full-body images was changing the minSize(h, w), as the subject's head and eyes were much smaller relative to the rest of the image. Once the cascade classifier was able to identify the subject's head, identifying and blurring the eyes worked perfectly.

**Conclusion**

The Haar cascade classifier is a potent and efficient tool for identifying facial features, and, according to Yadav, it is what the most common facial recognition systems use. Even with the downside of reduced accuracy, being able to detect human facial features rapidly is highly beneficial for many use cases, and with simple hyperparameter tuning, the accuracy of the Haar cascade classifier can be significantly improved. Using this technology alongside general image processing in OpenCV enables the solution of complex problems, such as facial privacy, by blurring one's eyes.

**References**

Mittal, A. (2024, January 18). Haar cascades, explained - analytics vidhya - medium. *Medium*. https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d

Rosebrock, A. (2021, April 17). *OpenCV haar cascades - PyImageSearch*. PyImageSearch. https://pyimagesearch.com/2021/04/12/opencv-haar-cascades/

Yadav, A. (2024, August 15). *OpenCV haar cascade explained*. Medium. https://medium.com/@amit25173/opencv-haar-cascade-explained-3b39a390871c