

# Face Obfuscation

Evan Dao  
CPSC 4820  
Clemson University  
Clemson SC, USA  
[edao@g.clemson.edu](mailto:edao@g.clemson.edu)

## Abstract

After being introduced to many computer vision concepts and designing our own facial detection algorithms, I was intrigued with the idea of face anonymization and being able to conceal your identity. So I decided to do my project on face obfuscation. Obfuscation is the action of making something obscure and unclear. So, I have implemented an algorithm to blur out faces and also place a colored bar over the eyes to anonymize an individual.

## KEYWORDS

Anonymization, Obfuscation

## 1 Introduction

Personal data and privacy has been an important topic for a while. Very often businesses and social media posts include photos and videos of people without proper consent which can lead to legal implications. So being able to anonymize faces on images and videos have proven to be very significant in today's society.

So in this project, I utilized the Gaussian blur to blur out faces from a picture, video, and your live webcam. I also implemented alpha blending to overlay a foreground image with transparency over a background image.

## 2. Gaussian Blur

I have implemented three different Gaussian blur programs that are all slightly different from one another. One blurs the faces from an image, another blurs the faces from a video, and the last blurs the face from a live webcam.

```
# Apply Gaussian blur over detected face
blur = cv2.GaussianBlur(roi_color, (101,101), 0)
frame[y:y+h, x:x+w] = blur
```

In order to utilize the Gaussian blur, I first had to implement facial detection using a cascade filter. Once that was done, I applied the Gaussian blur which is a smoothing function provided by OpenCV.

## 3. Alpha Blending

In order to overlay a transparent PNG image over another image, or in my case, a colored bar over a pair of eyes, I discovered a process called alpha blending in OpenCV. At every pixel of an image, you combine the foreground image color ( $F$ ) and the background image color ( $B$ ) using the alpha mask ( $a$ ).

$$I = \alpha F + (1 - \alpha)B$$

As in a similar process before, I first had to implement facial detection before I could apply the Alpha equation to every pixel. In my project, I placed a white bar over an individual's eyes on a live webcam to anonymize the user.

#### 4. Conclusion and Results

Overall, this project successfully implemented and accomplished the goal of face obfuscation. To improve this project further, I would find a more accurate way to detect facial features to make the blurring and image overlay more consistent. As personal data and privacy remain a relevant topic, the ability to hide one's facial features will always be crucial.

#### REFERENCES

- [1] Alpha Blending: <https://learnopencv.com/alpha-blending-using-opencv-cpp-python/>
- [2] Gaussian Blur: <https://www.tutorialkart.com/opencv/python/opencv-python-gaussian-image-smoothing/>