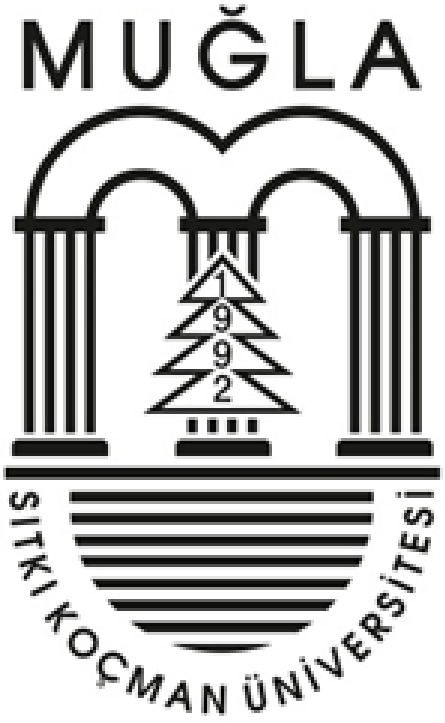


REAL-TIME FACE BLURRING

Final Project of CENG 4524: Fundamental Image Processing Techniques

SUMMARY

This project is about face blurring on images and live videos. It is a lecture project and this lecture is 'Fundamental Image Processing Techniques' which is coded 'CENG 4524'. The instructor is 'Ufuk Bal' and project-makers are in below:

Burak Can Onarım; Rıdvan Çelebi

1. Introduction:

In the image/video processing project(s) as named *'Face recognition'*, the first step is generally face detection. What if we think the opposite? What if we want to anonymize the face by blurring it? Thus, does making it is difficult to identify the face?

Practical applications of face blurring and anonymization include:

- Privacy and identity protection in public or private areas
- Protecting personal information(like face) online
- Photojournalism and news reporting(e.g. blur face of people who did not give a permission)
- Dataset curation and distribution(e.g. anonymize individuals in dataset)

… and more stuffs!

2. Process:

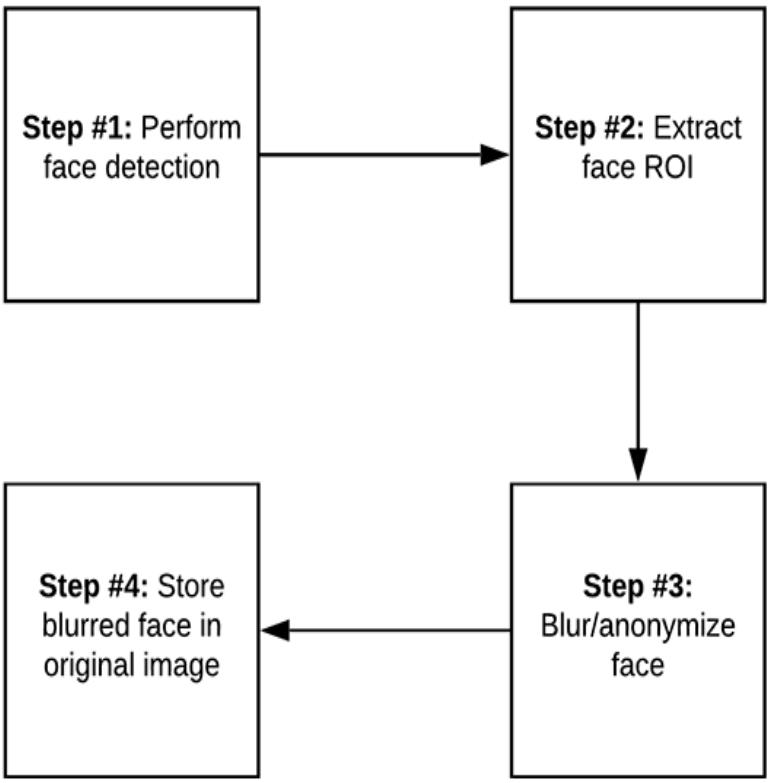
a. Image Blurring:

i. Explanation

In the first part of our project is a method that has four-step to blur faces with 'OpenCV' and 'Python'. We have used Gaussian blur to anonymize faces in images. Also, we will use it for videos in the final project.

ii. How can it be possible?

We use a method that has 4-steps to perform face blurring and anonymization.

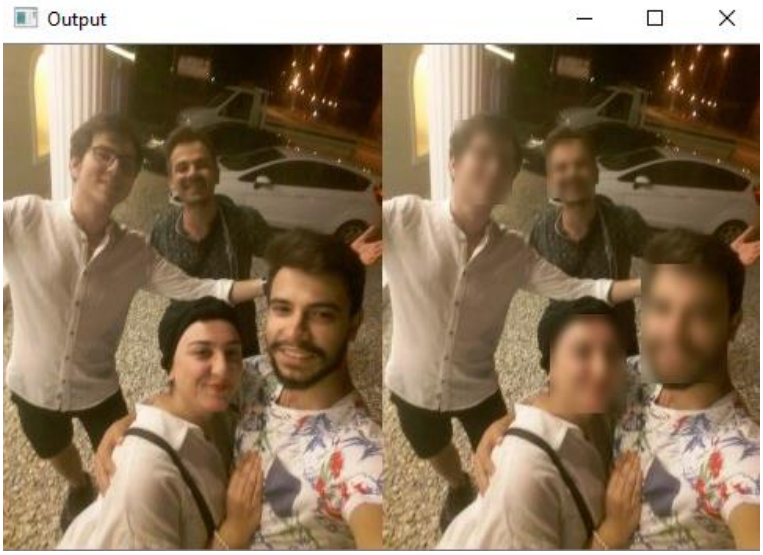


iii. Sample Outputs

In the output, the left part is original, and the right part is an image with blurred face.

- The face does not belong to a real person in this example. This is a portrait drawing. Even so, our algorithm can blur the face.

- This is a sample with multiple faces and the algorithm can blur all of them.



b. Process of Video Blurring:

i. Explanation

In this part, we will blur the face(s) in real-time video and explain simply how it is working.

The video actually consists of a large number of images streaming. First of all, we know processes of face blurring in images as we explained before. Actually, the processes of video blurring are very similar with processes of image blurring except that additional one stuff. And this additional stuff is a loop over for frames. Thus, we have to do all processes of blurring on images over and over again to blur face(s) on video.

ii. Usage

Primarily, the user has to download OpenCV library on local machine and the links how are download are given in below based on operating system. Also, another requirement is extracting rar file, too.

- **Windows**

  Windows users have to go this site to download required library which is called 'OpenCV'. After installation, the next step is running of codes. On Windows PowerShell, user has to run software codes that are provided. For reaching to Microsoft PowerShell, user has to write 'PowerShell' in address bar of extracted folder. Then, user has to write these codes respectively:

```
cd '.\Face Blurring Codes\'
```

```
py -3 blur_video_of_face.py
```

- **Linux**

  Linux users have to go to this site to download required library which is called 'OpenCV'. After installation, the next step is running of codes. On command line (i.e. terminal), user has to run software codes that are provided. For reaching to terminal, user has to right-click to folder which is named 'Face Blurring Codes' in extracted folder from rar file and click 'Open in Terminal'. Then, user has to write this code:

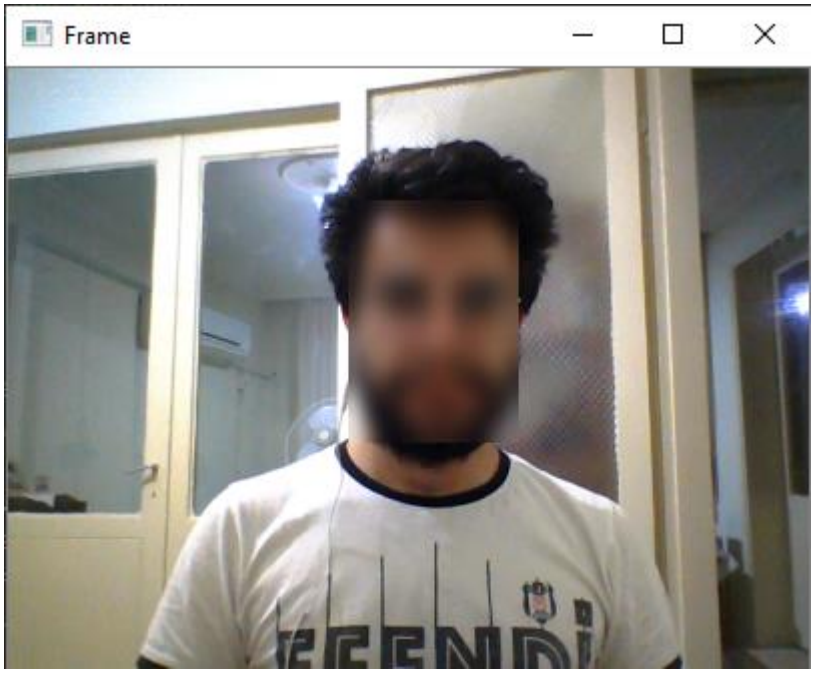
```
python3 blur_video_of_face.py
```

- **Mac**
  Mac users have to go to this site to download required library which is called ‘OpenCV’. After installation, the next step is running of codes. On terminal, user has to run software codes that are provided. For reaching to terminal, user has to open extracted folder from rar file. Then user has to click on the folder where is named 'Face Blurring Codes' to launch a terminal window and afterwards, user has to click on ‘Finder’ followed by ‘Services’ and select ‘New Terminal at Folder’. Finally, user has to write this code:

```
python blur_video_of_face.py
```

➢ User can easily see own face that is become blurred if all processes are done based on operating system of machine. Also, it is enough to pressing **‘q’** by user if the user wants to close output window. This procedure of closing is valid for every operating system.

iii. Sample Outputs



This image is from video stream and this video started when python code is run by Burak Can Onarım. So, this video is a real-time or in other words, live video.