Showing what matters!

03.08.2019

—

Team: T3st3rs

Naveen Kumar

Nitikesh Bhad

Overview

The idea is to remove the background from the video content and use the entire space for more useful content. Especially, in the case of Unacademy, most of the users are mobile-based or small screen users, for them, even the slightest space to show more content will improve the way they understand the content.

In our approach, rather than going with the traditional way of removing the background using OpenCV, which will fail in uncontrolled environments, we went with deeplearning based approach, where we do semantic segmentation to determine the exact position of person's body with clear contour at the pixel level. Once we know the exact shape, we mask the interested region and make the rest of the pixels transparent. Through this, we get the output stream, which contains the foreground only with a person without any other noise.

Goals

1. Make it as accurate as possible, it should look visually realistic. In terms of number, we are looking at 0.9 IoU

2. Make it run at almost realtime. The best case is around 24FPS, the minimum, we are planning to achieve is more than 18FPS

3. Memory consumption: The idea is to fit things, under 100-200mb of RAM during processing, so that it won't affect the other applications user might be using while streaming.
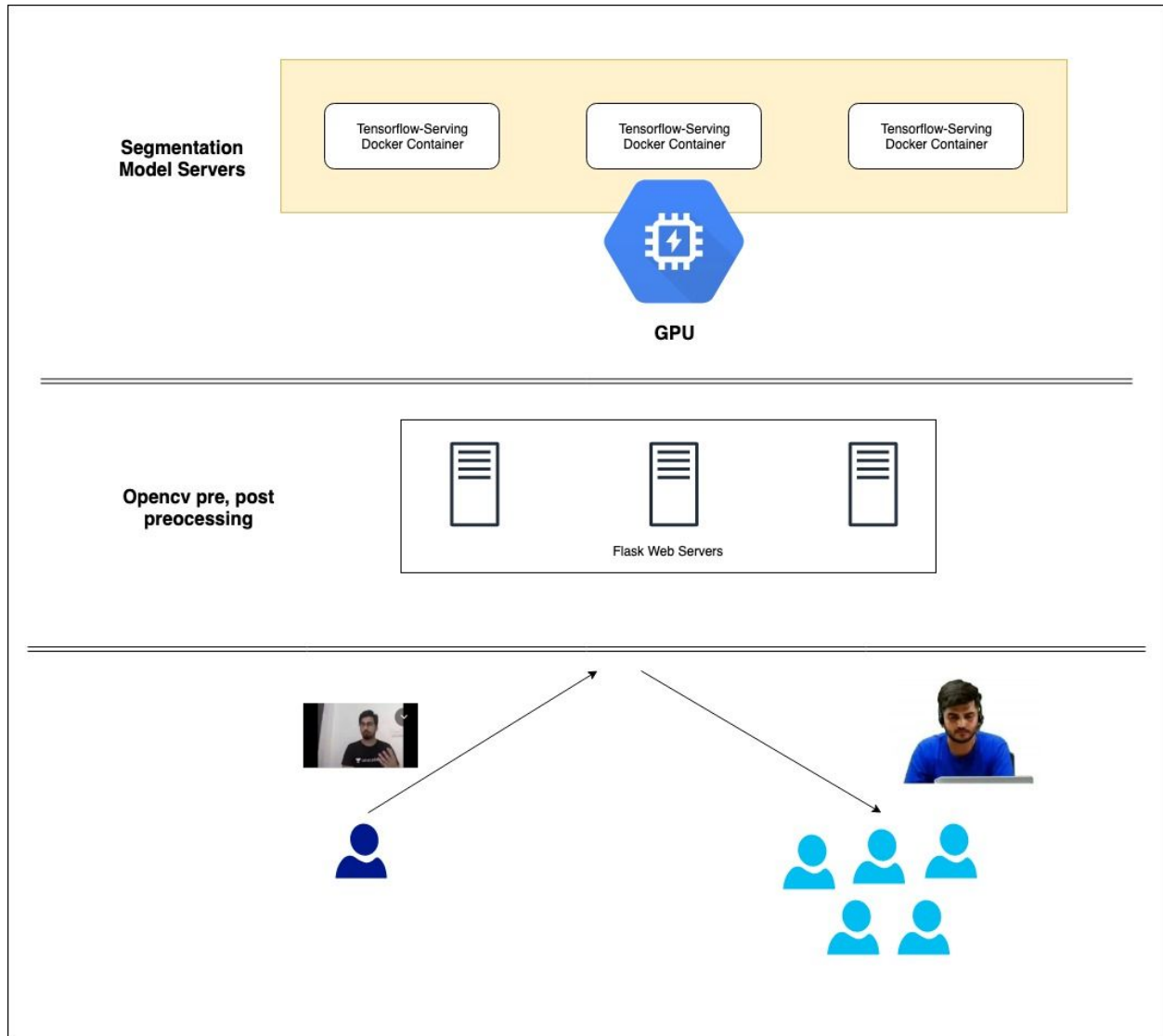
Tools Used

Language: Python, Javascript, Html

Libraries: Tensorflow, Opencv, Flask

ML stack: Semantic Segmentation, Gaussian post processing

Architecture

Conclusion:

In the entire approach, because of time constraint, we just picked up existing segmentation networks which were already trained on datasets like Cityscapes, through that, we filtered

the person class to determine the region of interest. With this approach, the challenge arises when more than one person sits side by side without any gap in between them, as our application expects the foreground to be only the tutor, not anyone else.

This is were, we need a more Instance semantic segmentation model which should separate the linked users. From the timings point of view, we can always achieve near realtime speeds with lightweight custom models.