**Distributed Systems  
Live Streaming Application  
Report**

By Pádraig Crotty

**Introduction**For my Distributed Systems assignment I chose the live streaming application . I made the base of assignment in python with Flask and OpenCV. I then implemented additional features such as motion detection and censoring. I learned a lot during this assignment about python, GPU’s and various libraries

**Technologies**As previously said the application was programmed in python. A variety of python libraries/packages were used in the creation of the assignment, their name and use in the assignment are as follows:

**Waitress**: Used for the Web server gateway interface(WSGI) for serving the application

**Socket**: Get the IP address that the application is running on

**Webbrowser**: Used to open the application in the default browser after starting

**Pillow**: Used for grabbing the screen

**Numpy**: Used to turn pillow screengrab into an array that OpenCV could process

**Flask**: Used as the Web server for the application

**OpenCV**: Used in all stages of image processing (Flipping image, colour formatting, motion detection, censoring, writing and reading)

**Os**: Used for getting directory of files and in cleaning up the excess images files left behind after the application is stopped

**NudeNet**: A library used for censoring faces and a variety of NSFW content.

**Motion Detection**Using OpenCV I was able to create simple motion tracking, in short this is done by grabbing a frame to be used as the background, grey-scaling it and then blurring. Then for each subsequent frame it grey-scales them, blurs them and then compares them to the background. If a region of the image is too different its max/min x/y co-ordinates are stored and then draw over the frame afterwards.

**Censoring with NudeNet**NudeNet is a python package that is used to censor nsfw content and faces, it was created using machine learning and was trained on 160,000 images,

**Limitations of CPU**NudeNet proved to be an incredibly heavy library to run which cause frame rates of the stream to drop to one frame every 3~8 seconds and in attempt to solve this I tried three things.   
The first I tried to do was reduce how often the censor would detect frames, this did not improve stream frames per second.  
The Second thing I tried filtering it so it only censored faces, which defeats the purpose of NudeNet and also did not improve the frames per second  
The third thing I tried to do was offload the computation to the GPU, there was two additional libraries that I worked with in order to try and offload these heavy computations.   
These libraries were **Numba** and the **Cuda Tool Kit**. I failed to get the programming running by offloading to the GPU as I had a lack of knowledge to how exactly the GPU computes problems. After doing some research I discovered that Numba will only work with numpy array types and that trying to get NudeNet to work with Numba would take a very long time to do so I had to drop the idea.

Diagram

Description automatically generated**UML Diagram**