Nimble-Programming Challenge

The server.py contains the server code that subscribes to the Webcam in the server machine and starts sending the data to the client.

Instruction

1. Clone the repository:

```
git clone https://github.com/notu97/nibmle-aiortc.git nimble_WS
```

Use the Dockerfile to create a docker image (here ubuntu_image).

```
sudo docker build - < Dockerfile -t ubuntu_image
```

2. Using this image build 2 docker containers (namely Server and Client container) with different port forwarding for each.

Client Container (in one terminal)

```
sudo docker run -it  --env QT_X11_NO_MITSHM=1   -e DISPLAY=$DISPLAY   -v
$XAUTH:/root/.Xauthority   -v /tmp/.X11-unix:/tmp/.X11-unix:rw   --volume
$HOME/PATH/TO/nimble_WS/:/root/   -p 8080:8080  --device
/local/camera/device:/dev/video0 --privileged ubuntu_image bash
```

Server Container (in another terminal)

```
sudo docker run -it  --env QT_X11_NO_MITSHM=1   -e DISPLAY=$DISPLAY   -v
$XAUTH:/root/.Xauthority   -v /tmp/.X11-unix:/tmp/.X11-unix:rw   --volume
$HOME/PATH/TO/nimble_WS/:/root/   -p 8090:8080  --device
/local/camera/device:/dev/video0 --privileged ubuntu_image bash
```

3. Once inside each container go to the root directory by running "cd" and in a 3rd termnal run `xhost +` command (thie terminal isn't needed anymore you may close it).

4. On the server container run `ifconfig` command. Get the ip address of the server container. Change the host address of both server.py and client.py to this new ip address obtained

5. Run `python3 server.py` on the server container, then run `python3 client.py` on the client container. The server container terminal shows the original ball location and the received ball location from the client side.