TEAM NAME - WeAreAI

Implemented solution overview:

- We have tested our solution on 4 streams and also on live feed taken from USB camera of laptop run on GTX1060.

- Used peoplenet model of Nvidia NGC, made comparison with detectnetv2 on performance metrics and arrived at conclusions on why peoplenet model was chosen and its improved performance over other model. (Refer to ppt for metrics, FPS comparison)

- Experimented on various novel ideas like live feed from USB for testing our model, use perspective transformation to improve accuracy(refer to video attached), web server to provide live update on model output, provide heatmap and graphical output for better visualization.

****

Please take a look at our **final video demonstration** which includes, model running, heatmap video generation, graph video generation, live update of webserver for heatmap video, graph video, violation info and Testing on live USB input feed

We request you to take a look at our **ppt** to get details regarding our performance metrics and experimentation

==**Video path - DEMO-VIDEO_AND_PPT/Video_demo.mp4**==

==**ppt path - DEMO-VIDEO_AND_PPT/final_hackathon.pptx**==

****

Instructions on running the code:

* We have installed new libraries like opencv, seaborn etc inside our container. We request you to run the codes inside our container :

1. deepstream_testing21.simg - Present in home dir as well as in final_submission folder using cmd:

  $ singularity shell --nv deepstream_testing21.simg

2. We have provided our solution in jupyter notebook format as well as in python code format.
    a. Jupyter notebook - **final_run_notebook.ipynb** (Recommended)

      We recommended using jupyter notebook to test our solution:

      * Open **final_run_notebook.ipynb** in jupyter notebook and press: Kernal -> Restart & Run All

    b. Python code - final_run.py (Not recommended because of issues with deepstream plugin run inside container)

      [In the case of deepstream plugin errors - run below cmds:

      $ LD_LIBRARY_PATH=/opt/tensorrtserver/lib:/usr/local/nvidia/lib:/usr/local/nvidia/lib64:/.singularity.d/libs

      $ rm ~/.cache/gstreamer-1.0/registry.x86_64.bin

      $ python3 final_run.py

3. Open another terminal and run the below webserver code to start seeing live update on 127.0.0.1:5000 (on web-browser)

  $ python3 server.py

4. Output is available in output folder –

    • output/final_output.mp4 (Video stream with violation details)
    • output/final_heatmap.avi (heatmap)
    • output/final_graph.avi (graph)

In case you wish to test the model performance in live stream (requires webcam), run the following code:

```
$ python3 live_feed.py /dev/video*(Path to video device to be given as
argument)
```