Computer Vision Hackathon: Emergency Vehicle Classification

Context
Fatalities due to traffic delays of emergency vehicles such as ambulances and fire brigade is a huge problem. In daily life, we often see that emergency vehicles face difficulty in passing through traffic. So differentiating a vehicle into an emergency and non emergency category can be an important component in traffic monitoring as well as self drive car systems as reaching on time to their destination is critical for these services. The goal of this project is to classify vehicle images as either belonging to the emergency or non-emergency category. Here, emergency vehicles usually include police cars, ambulances, and fire brigades. If successful, these models will be used in traffic monitoring or self-driving car systems.

Problem Statement
What models can be used to classify emergency vehicles accurately?

Criteria for Success
The models will detect the emergency vehicles with an accuracy rate of over 90%.

Constraints
1. We may have an imbalance dataset, the blind data may be under-represented
2. We may have low-quality images that might affect the models and training accuracy
3. We will need a GPU for this project since neural network algorithms require a large computational space. Google Colab in this case will be ideal.

Stakeholder
The primary clients for this project will be judges of the hackathon.

Data
1. The images are provided by the competition host. The data is available in Analytics Vidhya (https://datahack.analyticsvidhya.com/contest/janatahack-computer-vision-hackathon)
2. The training datasets will be used: train/images folders containing over 1600 vehicle images and the csv file containing the information whether each image is classified as emergency or not.
3. The testing datasets will only be used for participation of the contest and will be evaluated using Analytics Vidhya's private datasets.

Approaches
1. First, extract and load the data, update the data so each image id in linked to appropriate images
2. Then, do some data transformation make it easier for image processing
3. Next develop convolutional neural network (CNN) models for image classifications

4. Once we finalize the model, we will predict the results on the testing set

Deliverable

The final report will be a formal written report, with a slide deck for presentation purposes. Detailed steps and codes will be written in Jupyter Notebook.