2023

TANTECHNOVA 2023 Research Document

Project name: HazeAway

Team name: **Codinosaur**

Group Members:

- ***Anantha Prabu***
- ***Shuban***
- ***Vishnu Periyannan***

Cold wave grips north India, 7 dead in fog-related accidents in UP

PTI

Published: January 15, 2023 08:50 PM IST

A headline from the Press Trust of India

"Massive pile-up on Yamuna Expressway: 21 dead in fog-related accidents"

"Deadly fog causes 40-vehicle pile-up on Yamuna Expressway, 16 killed"

"29 dead, many injured in bus collision on foggy Lucknow-Agra Expressway"

"16 killed in collision between two trucks on foggy Gujarat Highway"

Still continuing...

I WONDER IF WE CAN EVER STOP SUCH HORRIBLE AND UNTIMELY ACCIDENTS FROM HAPPENING...

Loading the research on solution...

Table of Contents:

I. Objective of project

Our project has a critical objective: to reduce the number of heavy automobile accidents caused by poor visibility due to fog or smog. This is a serious problem that affects drivers all over the world. In 2021 alone, over 13,000 accidents were attributed to foggy or hazy conditions[1], this is not world-wide data, its specific to India alone! and this number is only increasing. In some areas, such as the million-plus cities in Uttar Pradesh, India, high rates of casualties have been observed during foggy conditions. The consequences of these accidents can be devastating, resulting in injury or even loss of life. To address this issue, our project aims to develop a live video de-hazing system that can be used in cars and other heavy automobiles. By improving visibility in hazy or foggy conditions, we hope to prevent accidents and save lives.

Our project represents an important step forward in the effort to make our roads safer for everyone.

II. Introduction

a) Overview of relevant information related to the project

How fog causes accidents:

- Fog is formed when water in the air *condenses* into *tiny droplets*, which can *scatter light* and form fogbows.
- It can also cause horrific accidents, because it appears opaque in low light and high-density conditions, which blocks light from, say, another vehicle.
- These low light conditions, coupled with the humid environment due to the monsoons, *occur over subtropical and high-altitude regions of India.*
- In places like New Delhi, *unfavourable winds and pollution* give rise to a new form of fog, smog, which is created by smoke, particulate matter and etc. mixed with fog to create an even more opaque cover. Along with causing pollution and respiratory afflictions, it causes accidents to an even higher extent.
- We, with our solution, seek to provide a reduction in these devastating causalities.

The project aims to use computer vision algorithms to remove haze or fog from a video stream in real-time. This will help to improve the visibility of the road ahead for drivers in cars. The project will use relevant information related to computer vision algorithms, image processing techniques, and display technologies. To solve the problem at hand, we identified three key challenges that needed to be addressed:

→ Developing a method to de-haze camera output,
→ Displaying the de-hazed feed to the driver, and
→ Creating a streamlined system that can perform these tasks in a timely manner.

To tackle these challenges, we began by conducting extensive research on existing solutions.

[1] Source: TOI

b) Previous studies related to the project topic

We quickly found many resources regarding our problem. Previous studies related to the project topic have been conducted in the field of computer vision and image processing. Some researchers have explored different algorithms for de-hazing images, such as *Dark Channel Prior* (He, Sun, & Tang, 2011), Colour Attenuation Prior, and the Scattering Model. There are also studies on the use of display *technologies for enhancing visibility*, such as *head-up displays and augmented reality*.

We came across the DCP algorithm (He, Sun, & Tang, 2011), which is a well-known method for de-hazing images, but we found that it is not effective in de-hazing a heterogeneous mixture of fog. We also found a research paper that focuses on a neural network called the AOD-Net (All-in-One De-hazing Network) (Li, Peng, Wang, Xu, & Feng, 2017). This paper demonstrated the lightweight and effective nature of the network in de-hazing images, although *it did not implement a live-feed de-hazing setup*. In addition, we found a corresponding GitHub project[2] that provided some insights on how to implement the network.

Our task then became to develop a portable and compact package that can implement these resources and display the de-hazed feed inside an automobile. This required both hardware and software components that can efficiently take a frame, de-haze it, and display it to the driver. Through our research, we were able *to identify potential solutions* and *determine the best approach to implement them in a streamlined system*.

III. Methodology

When it came to the methodology of implementing our solution, we had to consider several factors, such as which idea would be most practical and efficient. We narrowed down our options to three possibilities:

- Creating an app that would use a phone camera input,
- Developing a standalone system for the car, or
- Creating an app that would integrate with the smart systems in a car.

After careful consideration and elimination, we determined that *creating an app* for a *phone camera input* was not ideal, as it would be unreliable and require the driver to fiddle with their phone while driving in foggy conditions, which could be dangerous. We also found *that smart car systems* are notoriously locked down and fickle, so creating an app that would integrate with them would be difficult and limit the number of potential users.

Therefore, we decided to implement our solution on a standalone system that could be used in any car. This system would be designed with a single-minded focus to de-haze camera output and display the resulting feed to the driver with just one touch, making it simple and easy to use in any driving condition. This approach would ensure that our solution is reliable, efficient, and accessible to as many users as possible.

[2] Boyiliee/AOD-Net, MayankSingal/PyTorch-Image-Dehazing

a) Components and equipment needed

The Raspberry Pi was the *clear choice* for our project due to its *flexibility and compatibility with Python*, the preferred programming language for Machine Learning and Neural Networks. Furthermore, its *default Linux system* provided the versatility we needed for implementing our solution. In addition, the *Raspberry Pi's support for USB* simplified the process of integrating a high-quality camera, which was essential for our project.

After testing and evaluation, we determined that the following set of components would be necessary for executing our solution effectively:

- A Raspberry Pi 4 with at least 2 GB RAM to ensure adequate processing power.

- A webcam with a USB connector capable of capturing high-resolution images (at least 1280×720).

- A touchscreen display compatible with the Raspberry Pi to display the de-hazed feed to the driver.

- An external button to initiate the Pi, and de-hazing process with a single touch, providing ease of use for the driver.

By utilizing these components, we could create a compact and efficient system that could de-haze camera output in real-time and display the resulting feed to the driver quickly and easily.

b) Procedure for conducting the project

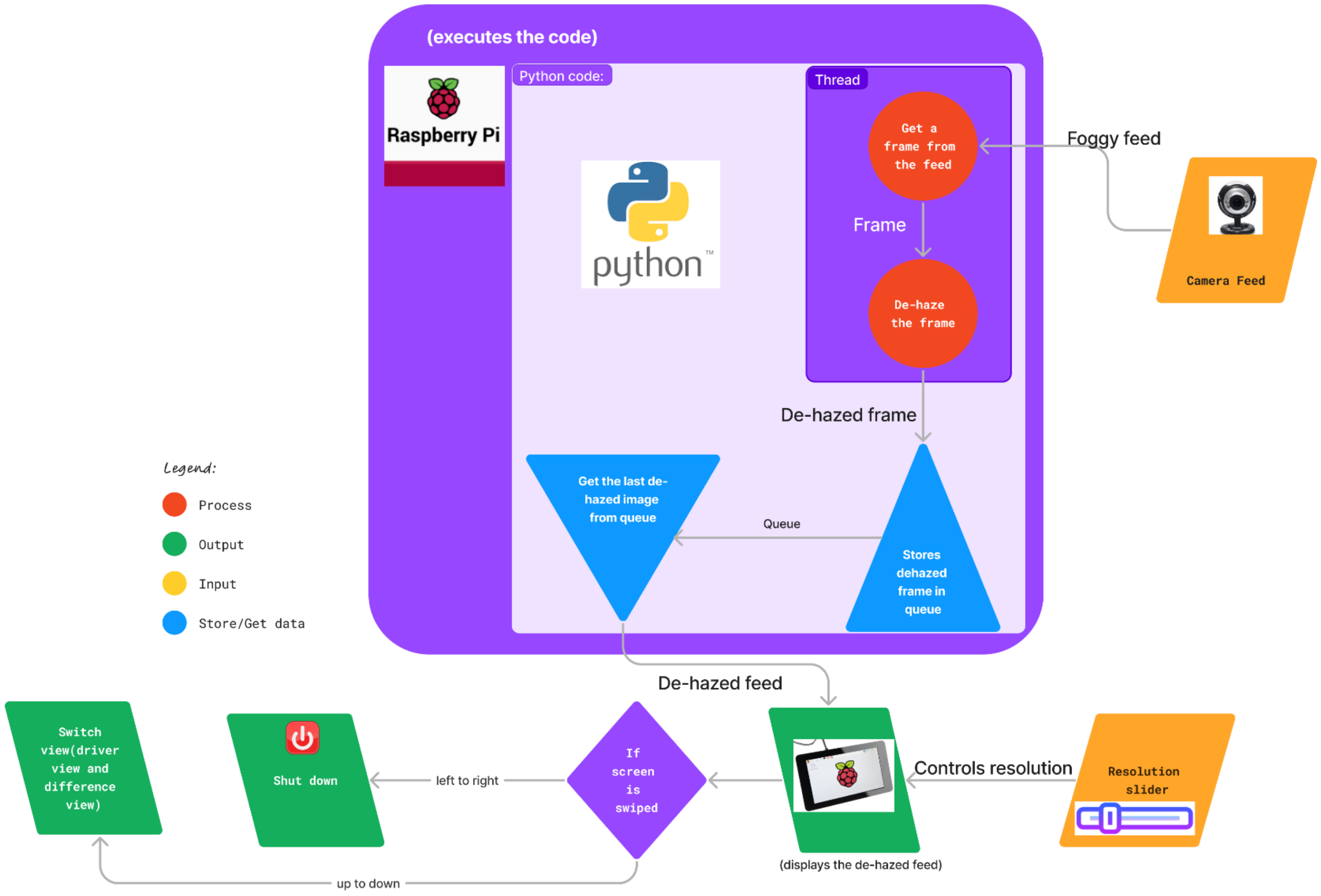
The procedure for conducting the project involves capturing live video using the camera, processing the video using the de-hazing algorithm, and displaying the de-hazed video on the car touch screen. The project will contain some Python code for the software side implemented in the Raspberry Pi, which will automatically start on login.

We worked online in this GitHub repo: Codinosaur/AtalMarathon2023, Most of explanation to implement this code is there. There is basically a model pretrained: *dehazer.pth,* The model has weights stored to dehaze image, dehazing occurs using PyTorch library.

To conduct the project the steps are:-

1. In main.py we need to configure resolution value, camera to dehaze.
2. Then run main.py with `python main.py`
3. Add other features like autostart, loading background, etc.
4. There is also `./Test/` folder for testing
5. Further explanation is in `README.md`

c) Design of the project using block diagram



The above block diagram is made through Figma jam, over here.

IV. Results

a) Findings of the experiment

The experiment results show that the live video de-hazing system is effective in removing haze or fog from a video stream in real-time, improving visibility for drivers in hazy or foggy conditions. The PSNR, SSIM values are the highest for output from our dehazing method than any other.

b) Tables and graphs

Average Quality comparison between outputs of several de-hazing algorithm on dataset:

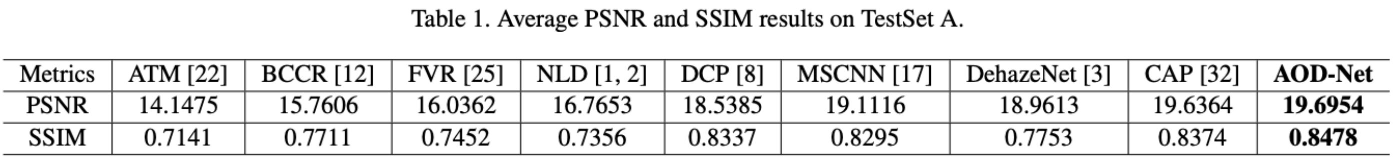


Table 1. Average PSNR and SSIM results on TestSet A.

Metrics	ATM [22]	BCCR [12]	FVR [25]	NLD [1, 2]	DCP [8]	MSCNN [17]	DehazeNet [3]	CAP [32]	**AOD-Net**
PSNR	14.1475	15.7606	16.0362	16.7653	18.5385	19.1116	18.9613	19.6364	**19.6954**
SSIM	0.7141	0.7711	0.7452	0.7356	0.8337	0.8295	0.7753	0.8374	**0.8478**

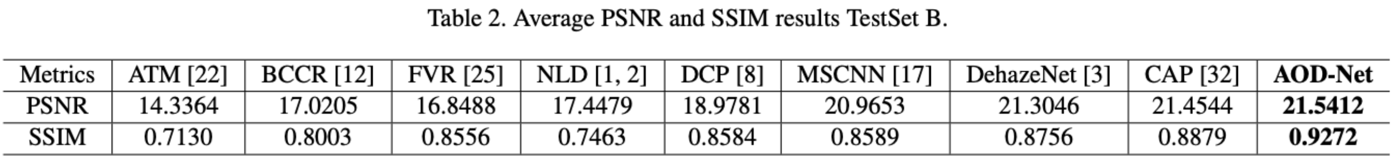
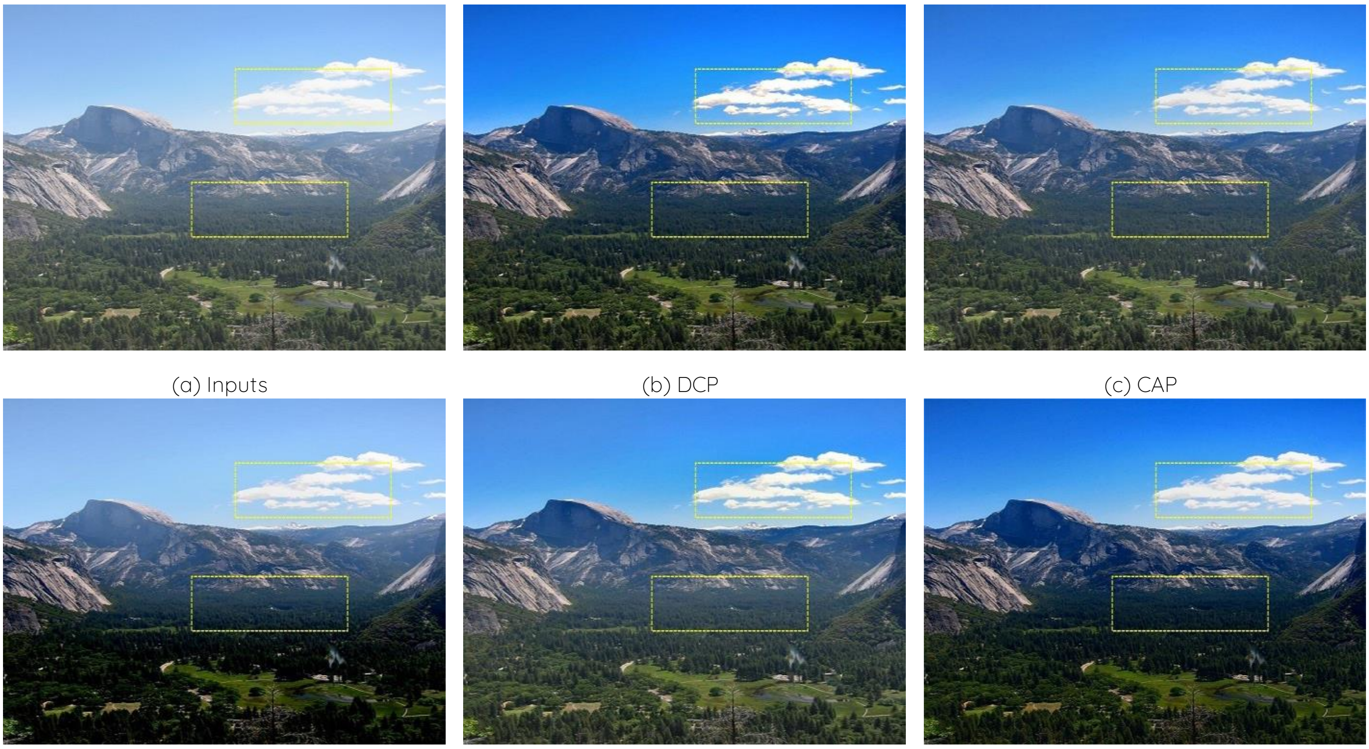


Table 2. Average PSNR and SSIM results TestSet B.

Metrics	ATM [22]	BCCR [12]	FVR [25]	NLD [1, 2]	DCP [8]	MSCNN [17]	DehazeNet [3]	CAP [32]	**AOD-Net**
PSNR	14.3364	17.0205	16.8488	17.4479	18.9781	20.9653	21.3046	21.4544	**21.5412**
SSIM	0.7130	0.8003	0.8556	0.7463	0.8584	0.8589	0.8756	0.8879	**0.9272**

Note: PSNR, SSIM are some methods that can be used to measure the quality of images.

Comparison between other methods to dehaze:



(a) Inputs

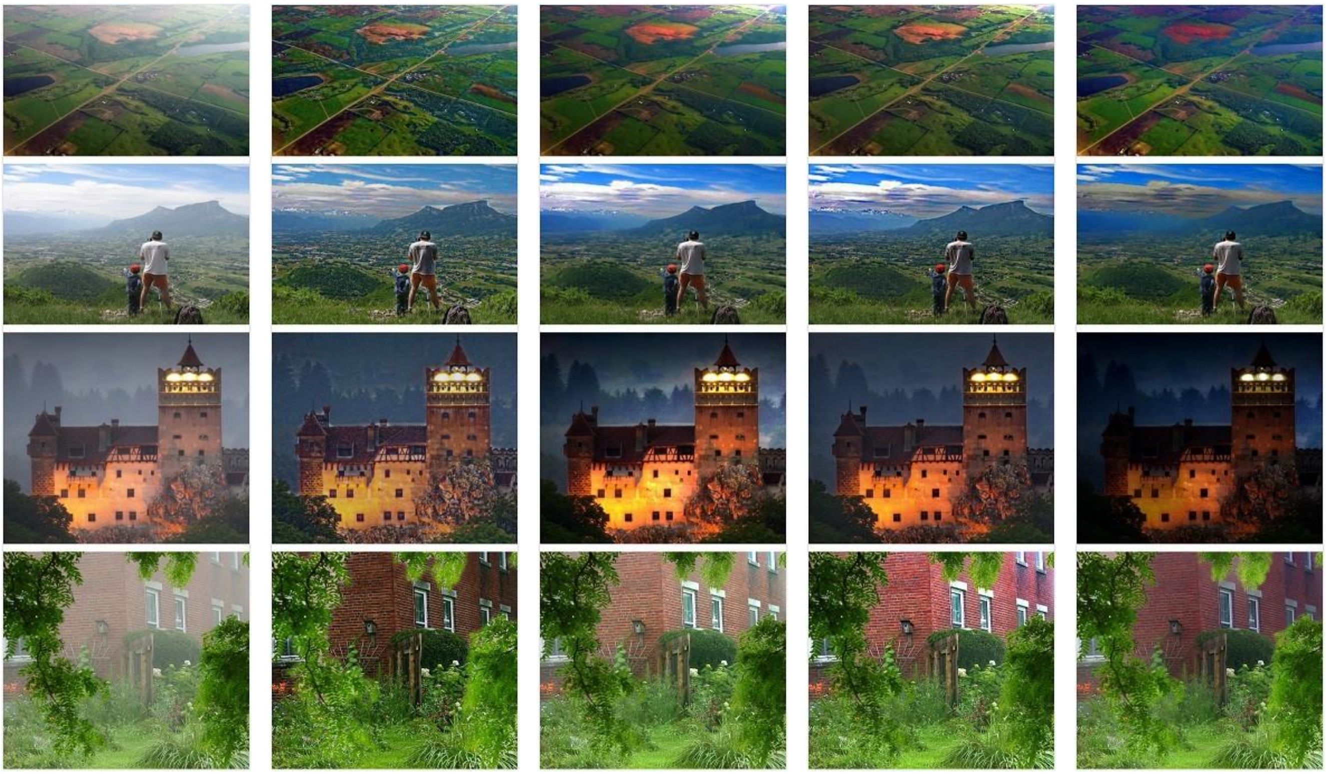
(b) DCP

(c) CAP

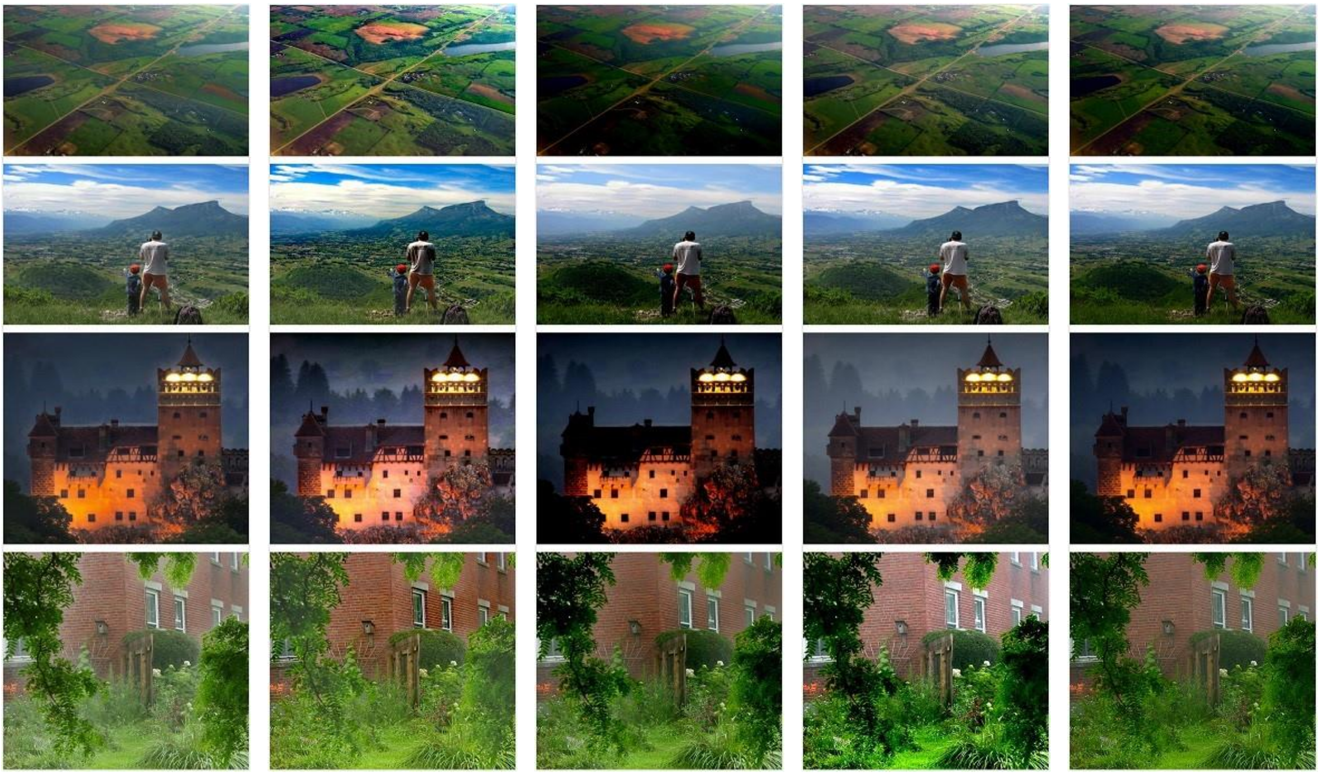
(d) DehazeNet

(e) MSCNN

(f) AOD-Net

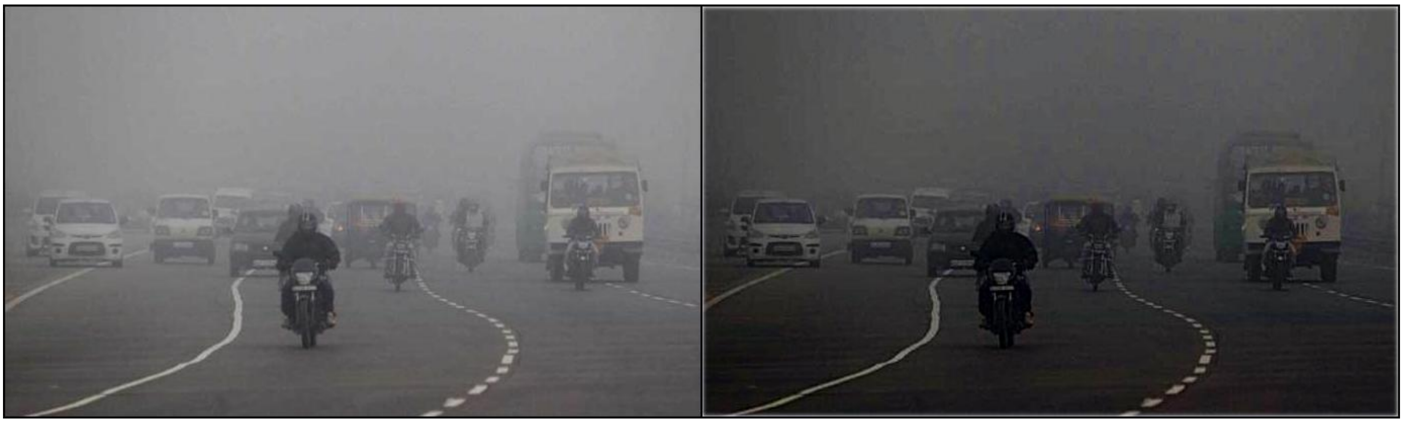


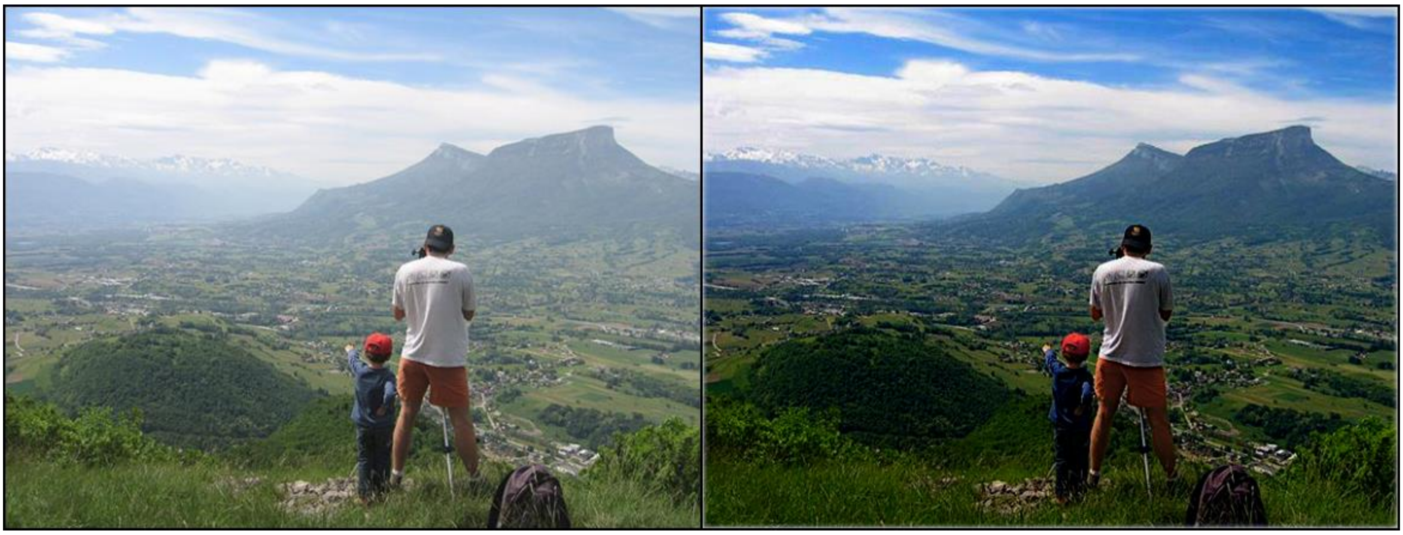
(a) Inputs (b) FVR (c) DCP (d) BCCR (e) ATM

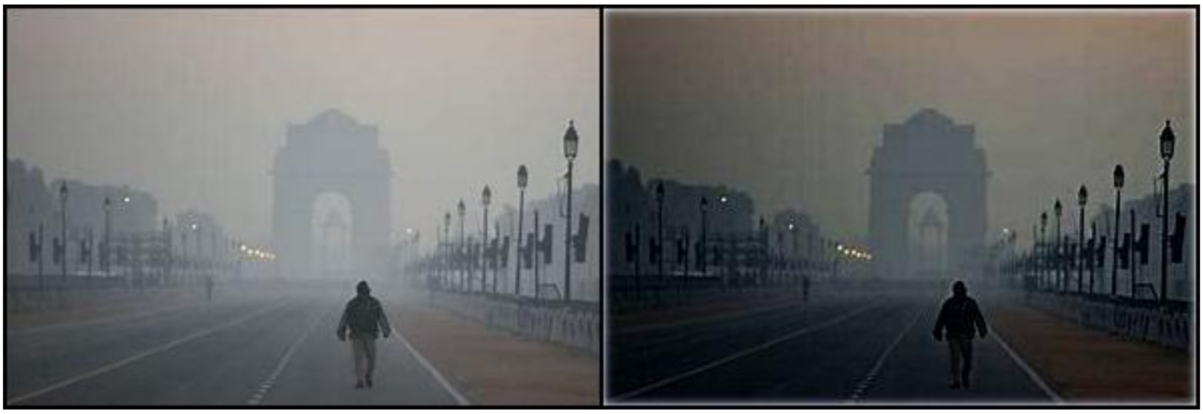


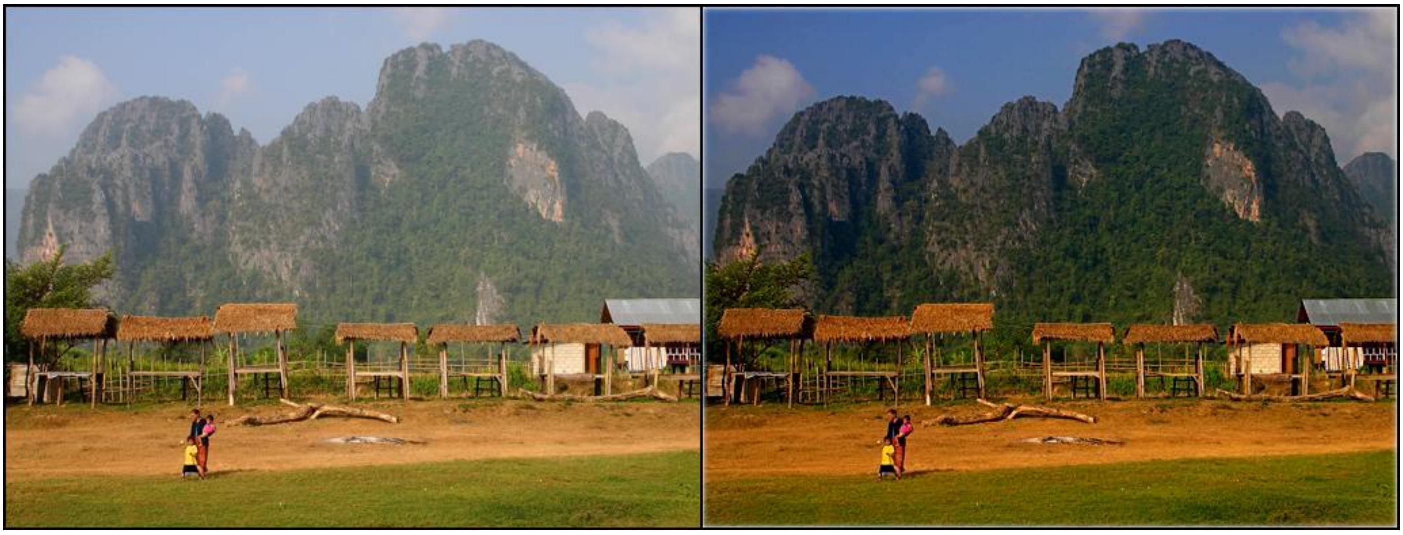
(f) CAP (g) NLD (h) DehazeNet (i) MSCNN (j) AOD-Net

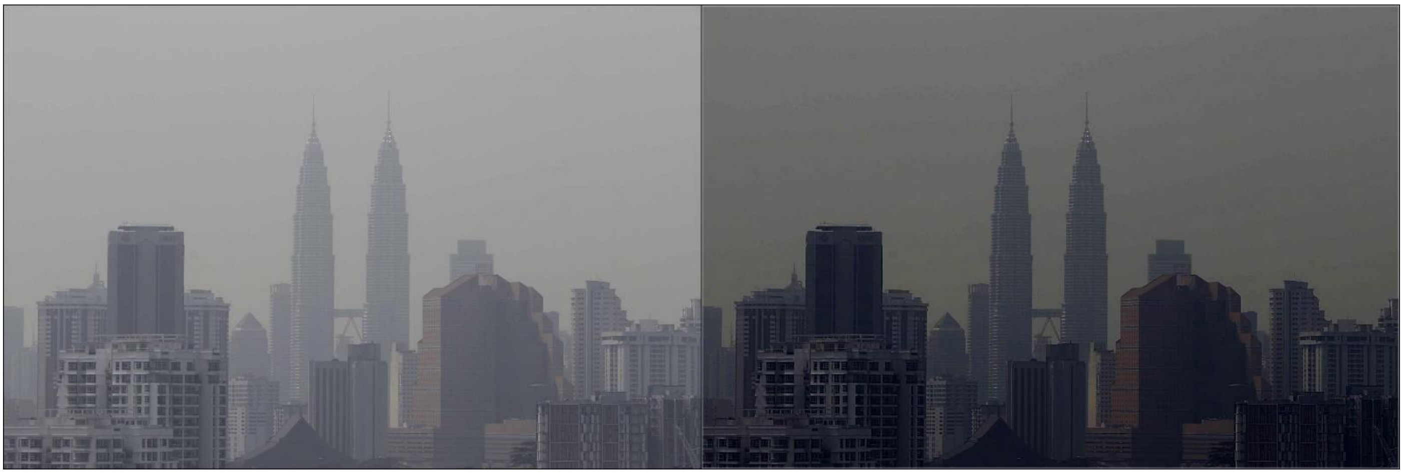
The below images have input on left output by our AOD-Net on right











V. Future scope of the project

The system could be expanded to include other vehicles, such as airplanes and boats, where visibility is also critical for safe operation. We need to find something to do about two wheelers where there can't be a touch screen, or even if it is, the drivers won't be able to see it properly.

We also mainly want make the model an actual, real device so that people's lives are really saved! We will need entrepreneurial support for it, as making a compact Raspberry Pi device is out of the hands of amateur tinkerers like us.

VI. Credits

We would like to thank the following people for helping us complete this project:-

- Our school ATL staff for encouraging us to execute this idea.
- The research paper An All-in-One Network for Dehazing and Beyond (Li, Peng, Wang, Xu, & Feng, 2017) for giving us the inspiration for the central mechanism of our project.
- The GitHub Repository for providing us with an example of how to implement the AOD-Net.

VII. References

He, K., Sun, J., & Tang, X. (2011). Single Image Haze Removal Using Dark Channel Prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(12), 13. doi:10.1109/tpami.2010.168

Li, B., Peng, X., Wang, Z., Xu, J., & Feng, D. (2017). An All-in-One Network for Dehazing and Beyond. *International Conference on Computer Vision*. doi:10.48550/arxiv.1707.06543

VIII. Project links

- ➢ GitHub Project Source: Codinosaur/tantechnova
- ➢ Figma: Block diagram

THANK

YOU!