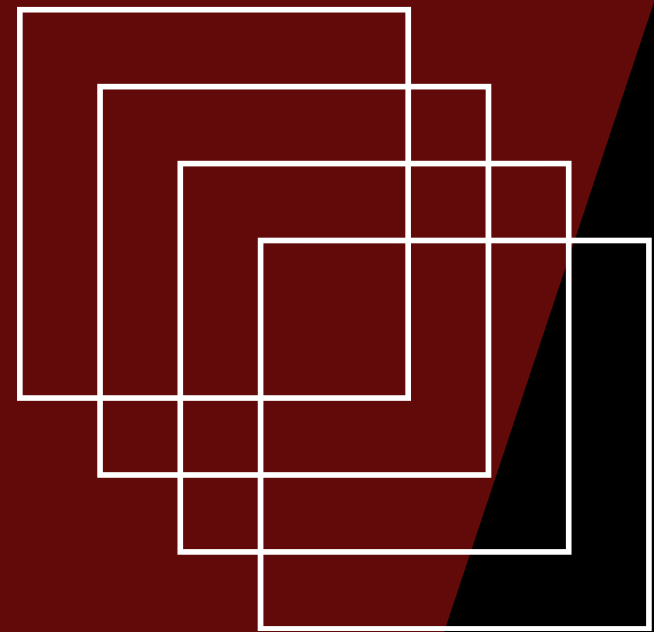
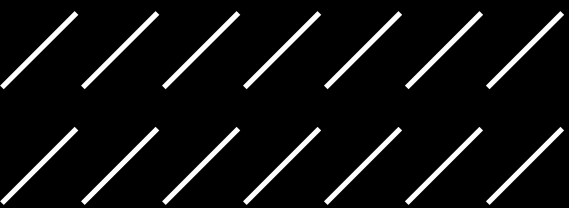




# SYNTH VISION – FOG



Team LogicCrafters

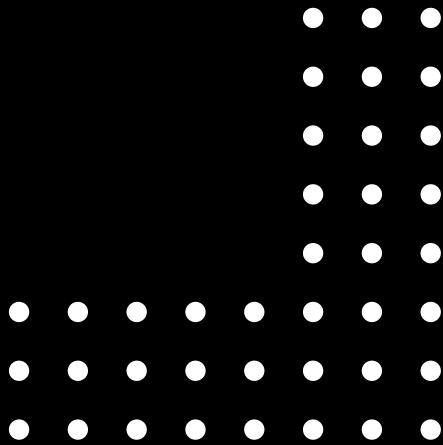


# SOURCE

We have used BDD100k: A large-scale Diverse Driving Image Database, designed for Computer Vision and self driving research.

Resolution: ~1280 x 720

Provides object detection, multiple object classification, and annotation



# PROBLEM WE'RE SOLVING

## INTEGRATING FOG TO THE IMAGES

We are using BDD dataset with open-cv library to create noise data on the top layer of the image as fog

FINAL IMAGE = CLEAR IMAGE (BDD) + FOG LAYER (SYNTHETIC)

## GENERATE OBJECT DETECTION

# PIPELINE

1

## INPUT & PREPROCESSING:

load the clear BDD image and normalize to [0,1].

2

## FOG MAP CONSTRUCTION:

Create depth map + edge map + smooth noise, then combine them into a single fog-intensity map

3

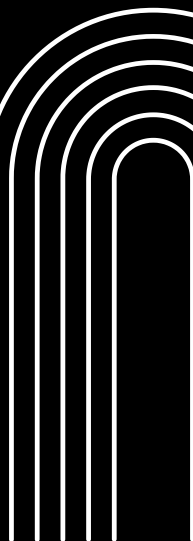
## ATMOSPHERIC SCATTERING MODEL:

Compute transmission map and blend image with atmospheric light A to simulate visibility loss

4

## SPATIAL BLUR & OUTPUT:

Apply distance-based Gaussian blur (and glow for high fog) and save as low/medium/high fog images



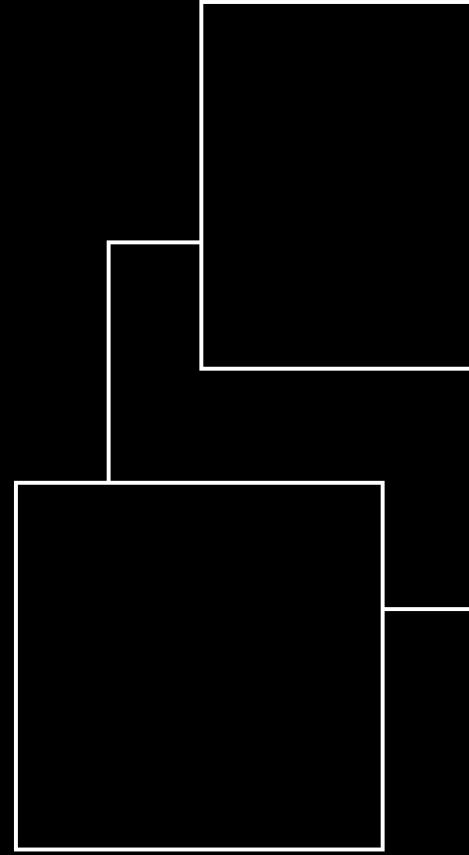
# OUR APPROACH

## Layered Fog Modelling

$$\text{Foggy Image} = I(x) \cdot t(x) + A \cdot (1 - t(x))$$

Where:

- $I(x)$  = original clear image
- $t(x)$  = transmission map (visibility at each pixel)
- $A$  = atmospheric light (sky brightness)



### Depth aware Fog Distribution:

```
y = np.linspace(1, 0, h)
depth = np.tile(y[:, None], (1, w))
```

### Edge-Aware Haze Effect:

```
x = np.linspace(-1, 1, w)
edge = np.abs(x)
```

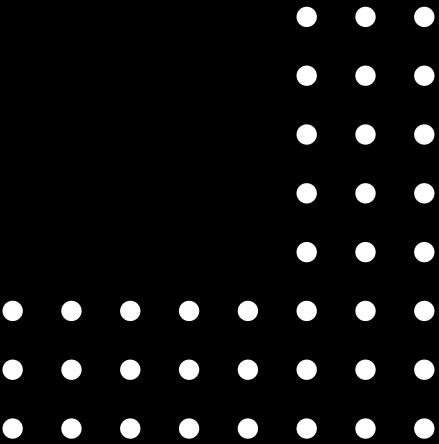
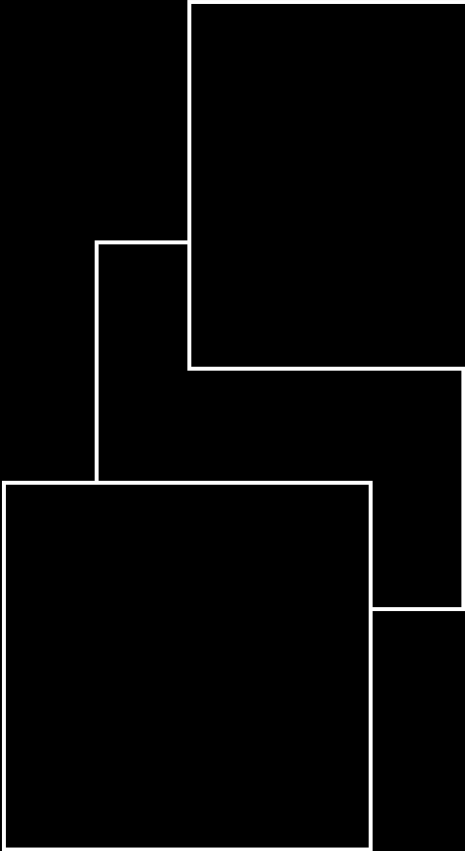
### Stochastic (Random) But Controlled Noise

```
noise =
cv2.GaussianBlur(np.random.rand(h,
w), (101,101), 0)
```



# Intensity Control via Three Modes

Mode	Beta (scattering)	A_val (sky brightness)	Effect
Low	small	lower	mild haze
Medium	moderate	medium	realistic fog
High	large	high	dense fog + glow





# OUR ACHIEVEMENTS

## **Successful FOG simulation on the images:**

- We successfully generated synthetic foggy images from clear BDD100K images using OpenCV.
- The fog is not random noise but follows parameters mentioned in problem statement.

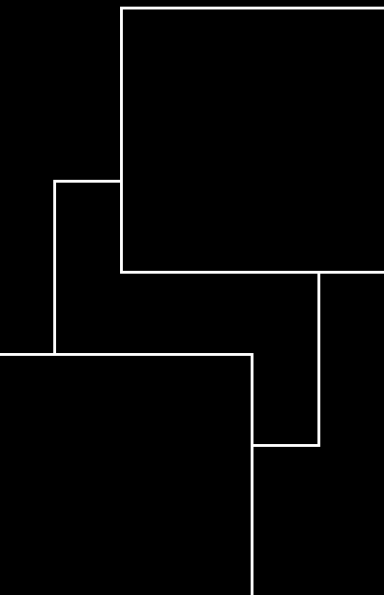
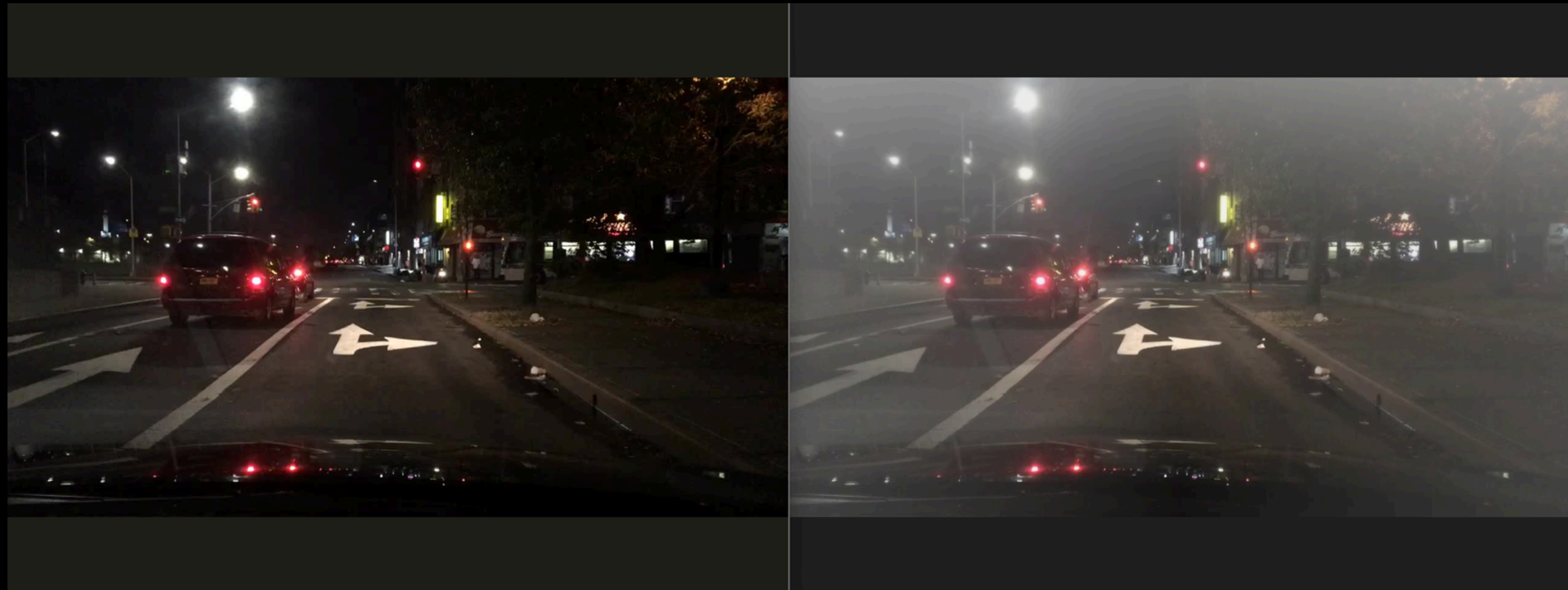
## **Generation of Multiple Fog Intensities:**

We created three controlled weather conditions from the same dataset:

- Low fog → mild visibility reduction
- Medium fog → balanced degradation
- High fog → heavy haze with glow and blur



# CLEAR IMAGE VS SYNTHETIC FOG IMAGE





# ANNOTATED FOG OUTPUT



**THANK YOU**

