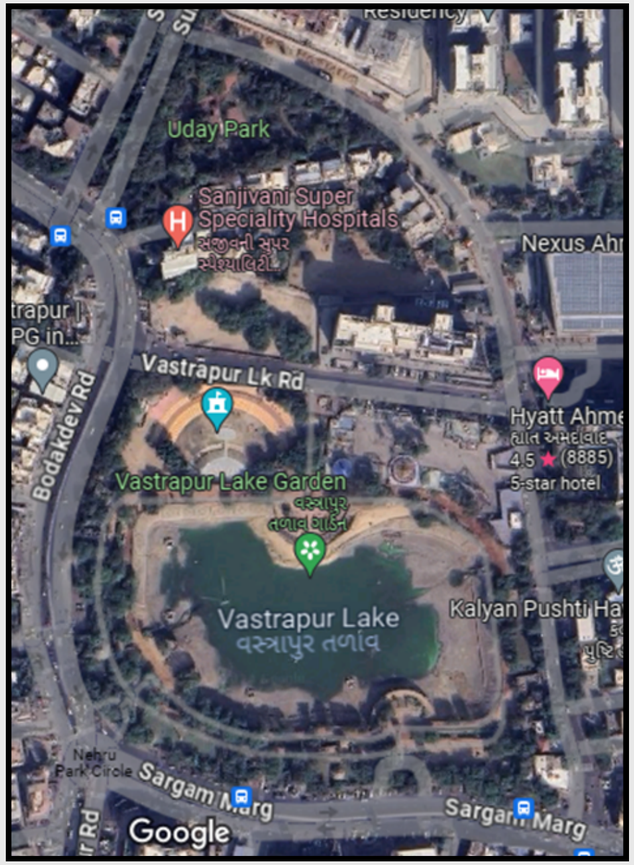
HACK THE FUTURE'24

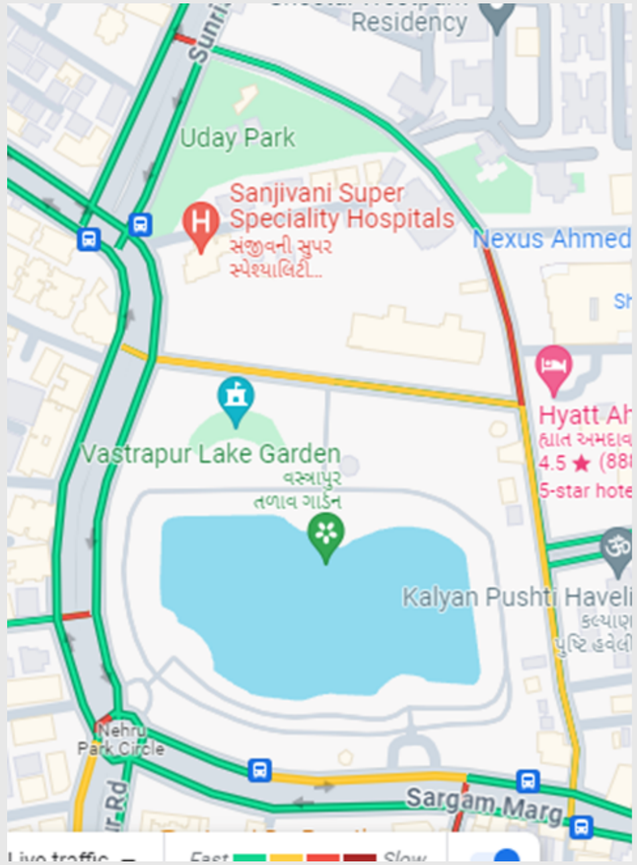
REDUCTION OF TRAFFIC CONGESTION IN AHMEDABAD CITY

TEAM ETA

TEAM MEMBERS:
HITESH JAISWAL, KATHAN DAVE, LALJI SANKHAT, PARAG SAHU, RIYA MAHERIYA

Physical Planning Interventions

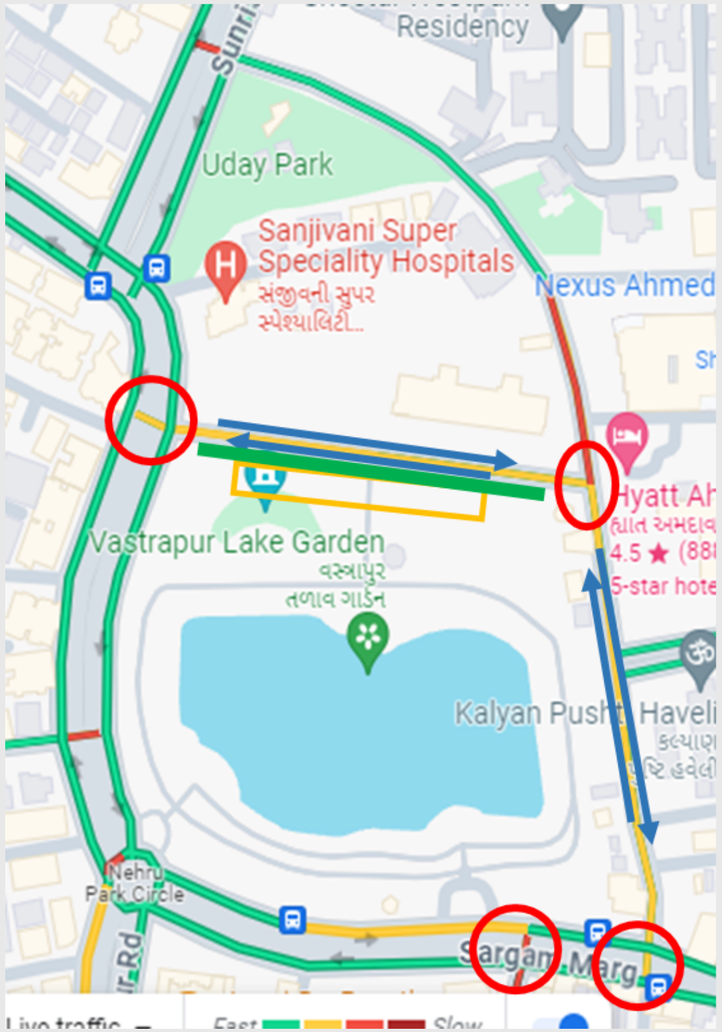




Case Study-1: Periphery around the Vastrapur Lake

https://www.google.com/maps/@23.0390812,72.5275941,16.79z?entry=ttu

Identified Existing Issues



Case Study-1_ Periphery around Vastrapur lake

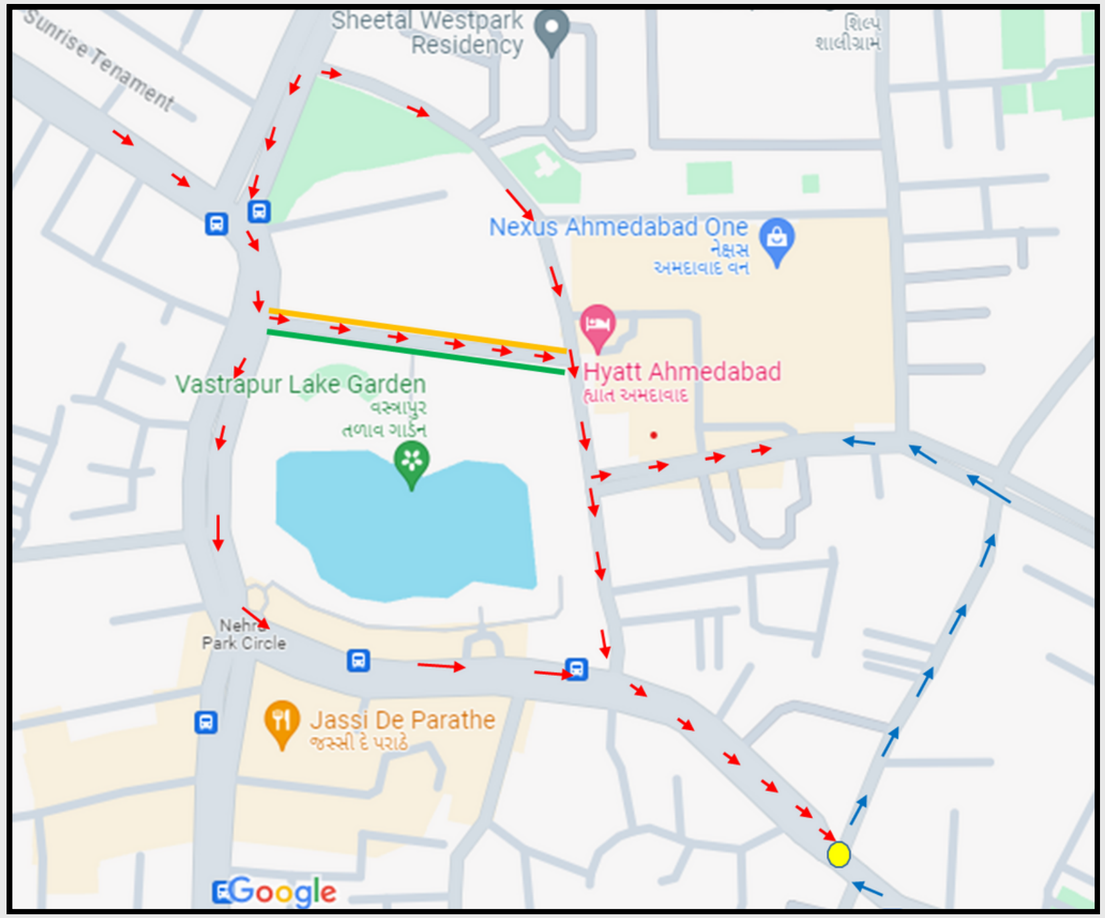
1. Bottle-neck Junctions
2. Two Way Traffic
3. Street Side Vendors
4. Improper on street Parking

https://www.ahmedabadmirror.com/what-centre-told-states-in-may/81810774.html

https://www.google.com/maps

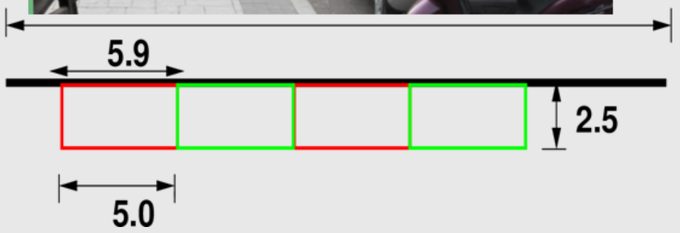
https://www.propertypanther.in/

Proposed Solution



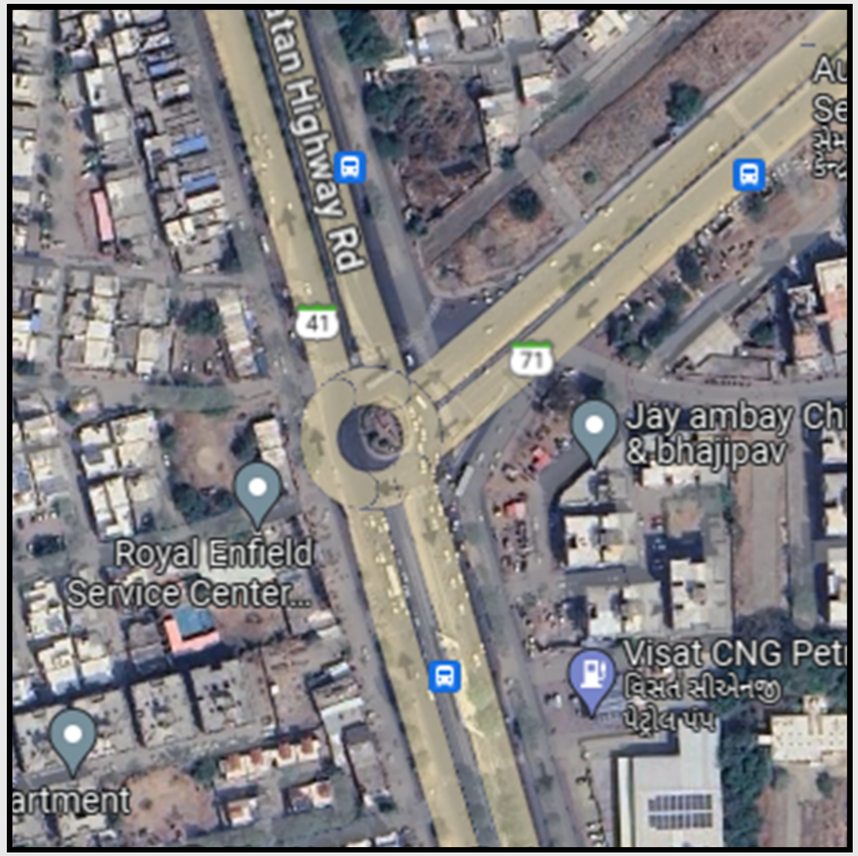
Parking | Street Vendors | Automated Traffic Signal

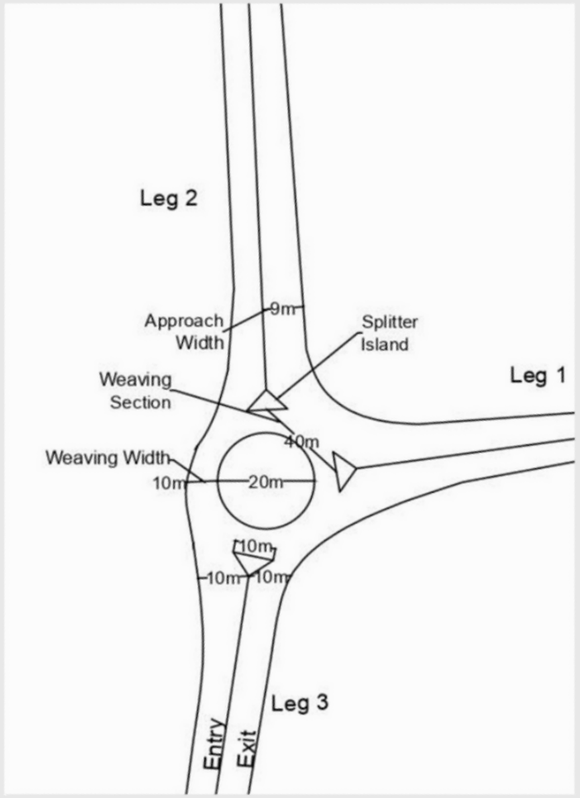
Inflow | Inflow for mall



https://www.google.com/maps/@23.0390812,72.5275941,16.79z?entry=ttu

Physical Planning Interventions

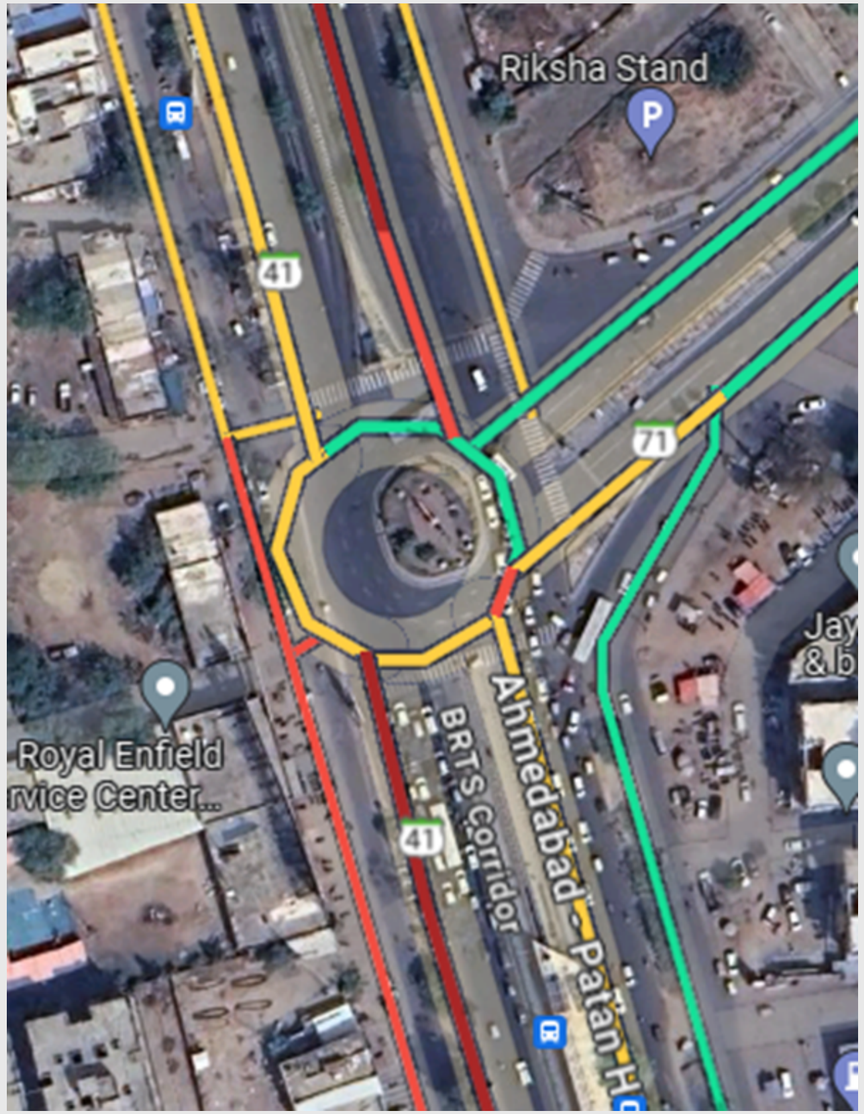




Case Study-1 Roundabout at Visat Circle

https://www.google.com/maps/@23.0390812,72.5275941,16.79z?entry=ttu

Identified issue



Case Study-1 Roundabout at Visat Circle

Delay in traffic flow due to improper time cycle

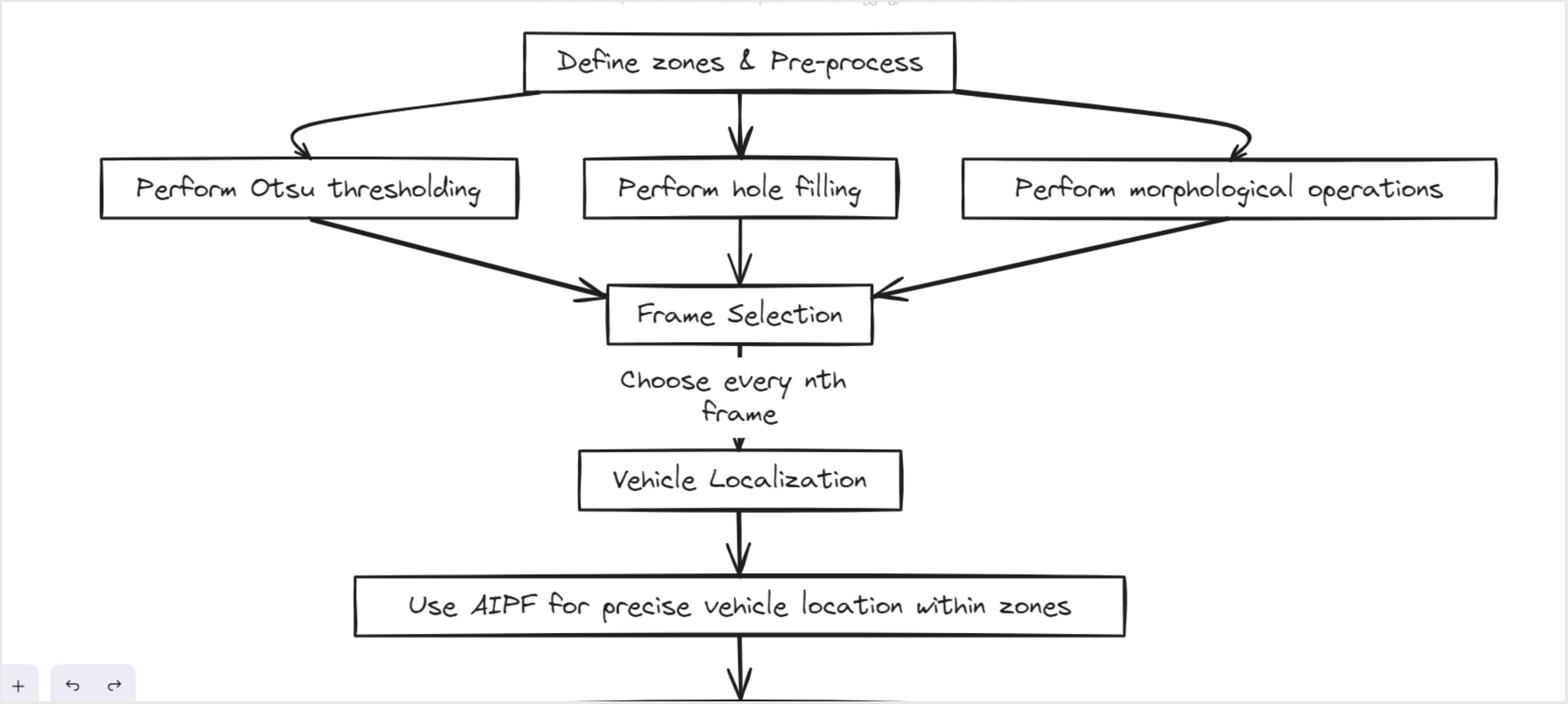
Observed Delay at Visat								
Morning Hours								
Approach	Traffic volume (pcu/hr)	Traffic volume (pcu/sec) q	Saturation Flow (sec) {no. of lane * Saturation Rate) s	Green Time (seconds) g	Amber time(sec)	Cycle Time c	s*g	q*c
leg 1	1284	0.36	1.11	32	3	105	35.52	37.45
leg 2	1144	0.32	1.11	32	3		35.52	33.37
leg 3	1251	0.35	1.11	32	3		35.52	36.49
Noon								
leg 1	1068	0.30	1.17	28	3	93	32.76	27.59
leg 2	1093	0.30	1.17	28	3		32.76	28.24
leg 3	1139	0.32	1.17	28	3		32.76	29.42
Evening								
leg 1	1309	0.36	1.2	32	3	105	38.4	38.18
leg 2	1297	0.36	1.2	32	3		38.4	37.83
leg 3	1339	0.37	1.2	32	3		38.4	39.05

https://www.google.com/maps/@23.0390812,72.5275941,16.79z?entry=ttu

Technical Interventions

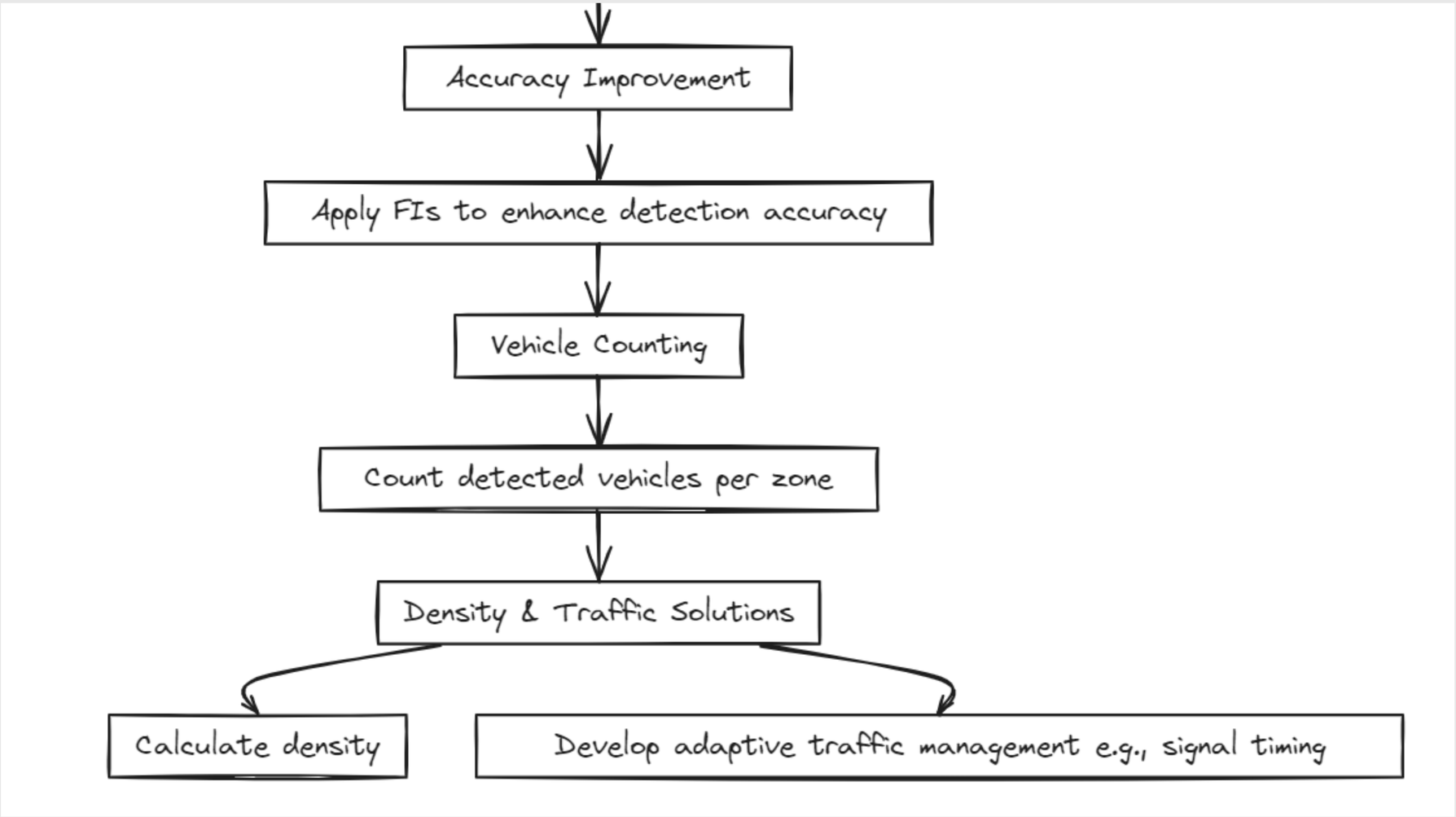
Not every physical planning intervention and land use solution is feasible.

Proposed solution



To cater problem of vehicle counting and congestion estimation and then apply those results to make adaptions on the traffic signals.

Proposed solution



Code

```python
def process_frame(self, target_frame, reference_frame):
    fused_image = self.remove_reference_image(target_frame, reference_frame)

    self.update_background_model(fused_image)
    self.update_foreground_model(fused_image)

    return raw_gradient_result, normalized_gradient_result, fusion_image, otsu_result,
hole_filled_result, morphological_result, aipf_result
```

```python
def calculate_raw_gradient(self, frame):
    # ... (calculates the raw gradient of an image)

def calculate_normalized_gradient(self, raw_gradient):
    # ... (normalizes the raw gradient)

def perform_aipf(self, frame):
    # ... (performs adaptive inverse perspective mapping)
```

```python
def remove_reference_image(self, target_frame, reference_frame):
    result = cv2.absdiff(target_frame, reference_frame)
    return result

def update_background_model(self, frame):
    # ... (creates or updates the background model)

def update_foreground_model(self, frame):
    # ... (creates or updates the foreground model)
```

```python
contours, _ = cv2.findContours(morphological_result, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)
# ... (loop through contours, filter based on criteria, and count vehicles)

traffic_density = vehicle_count / zone_length * (60/frame_rate)  # vehicles per hour
```

Code to process frames for vehicle detection using image fusion and background subtraction

Original Image

Reference Image

Video frame Image

Processed Image

Fusion Image + AIPF

Otsu thresholding + hole filling

Code

```python
def divide_image_into_grids(image_path, grid_size_mm):
    # ... (rest of the code)
#This defines a function named divide_image_into_grids that takes two arguments:

#image_path: Path to the image file.
#grid_size_mm: Size of each grid cell in millimeters.
```

```python
image = cv2.imread(image_path)
pixels_per_mm = 1
grid_size_pixels = int(grid_size_mm * pixels_per_mm)
height, width, _ = image.shape
num_rows = int(np.ceil(height / grid_size_pixels))
num_cols = int(np.ceil(width / grid_size_pixels))
```

```python
cv2.rectangle(image, (0, 0), (240, height), (0, 0, 0), -1)  # Draws a black rectangle
(optional)
vehicle_count = 0
```

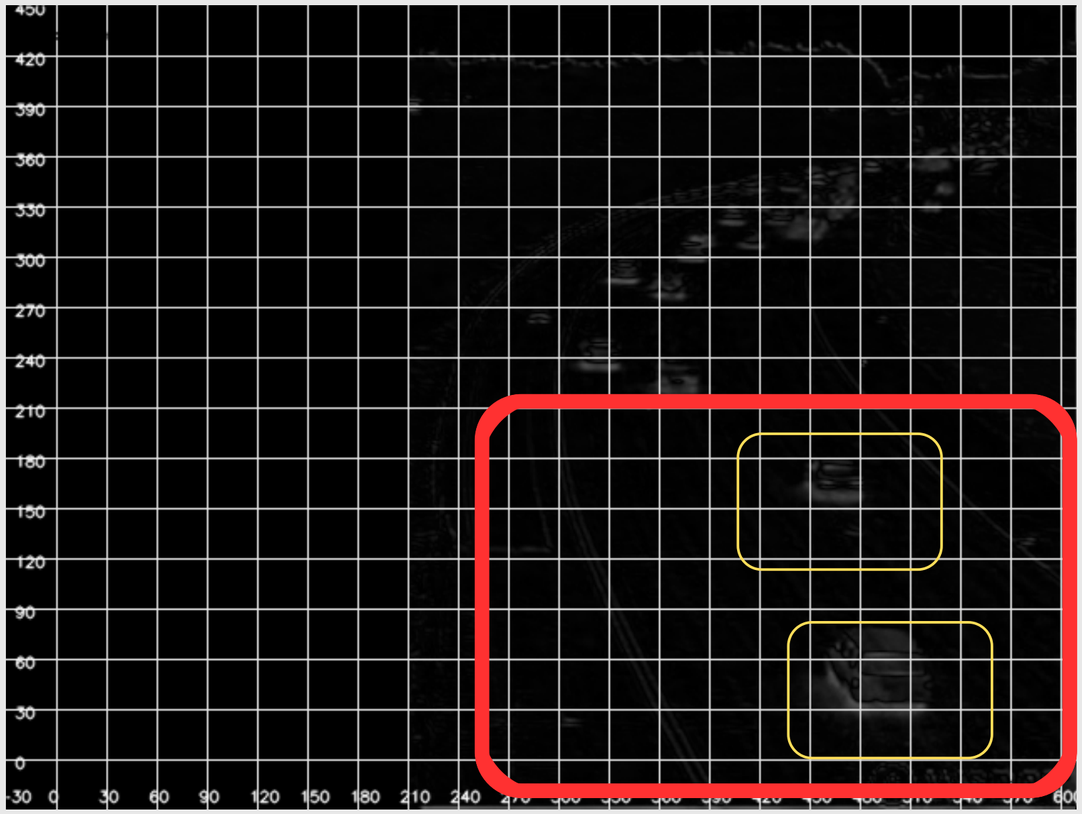
```python
for i in range(num_rows):
    for j in range(num_cols):
        x = i * grid_size_pixels
        y = j * grid_size_pixels
        if 390 <= x <= 600 and 0 <= y <= 210:  # Checks a specific region
            cell = image[y:y+grid_size_pixels, x:x+grid_size_pixels]
            if np.any(cell != [0, 0, 0]):  # Checks if cell contains non-black pixels
                vehicle_count += 1/9
```

```python
vehicle_count = np.floor(vehicle_count)
print(f"Number of vehicles: {vehicle_count}")

# Code for drawing grid lines, labels, and axes
(omitted for brevity)
cv2_imshow(image)
```

This code focuses on dividing an image into a grid and estimating the number of vehicles within a specific region.

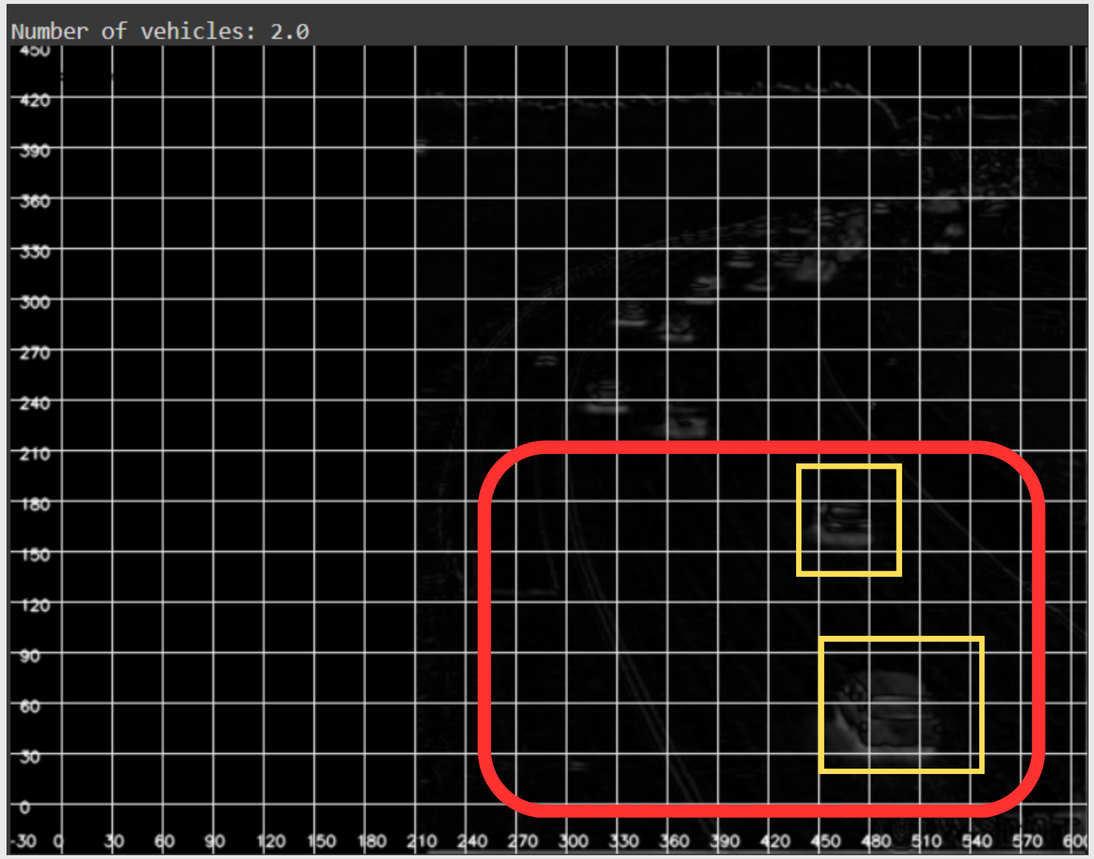
Our Approach



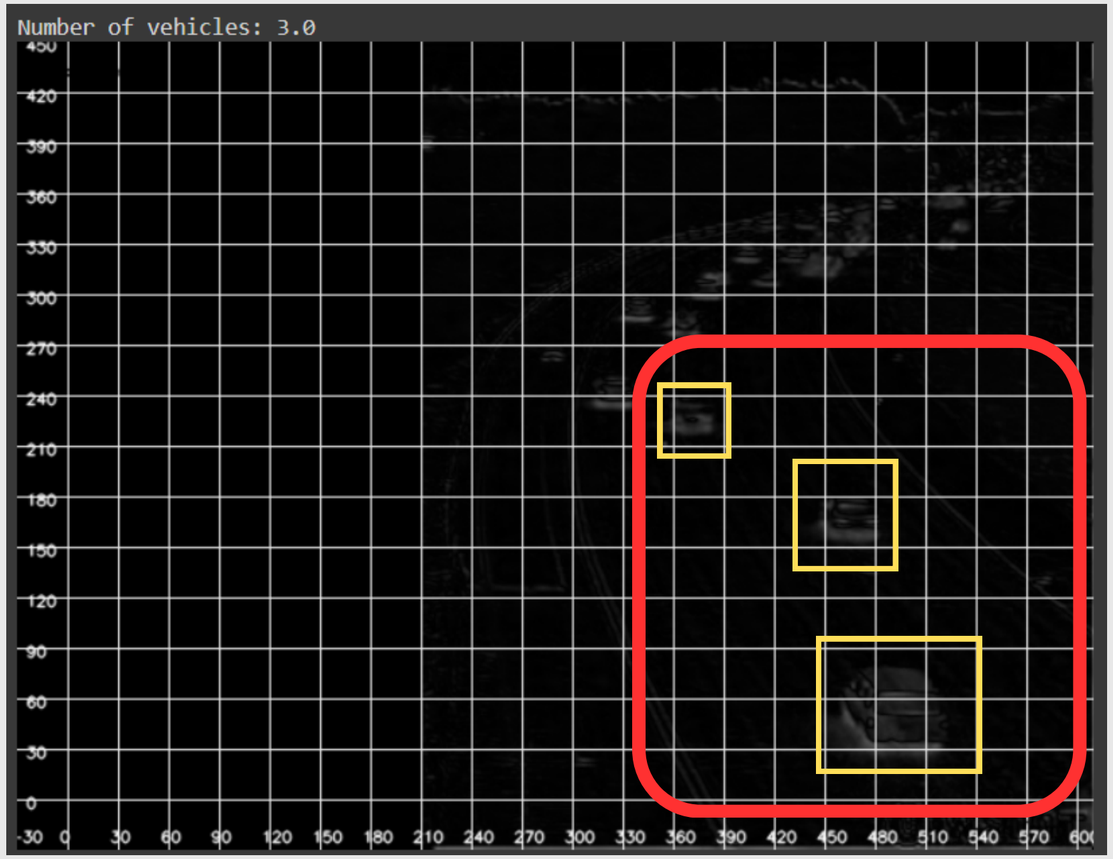
Processed Image

Original Image Frame

Results

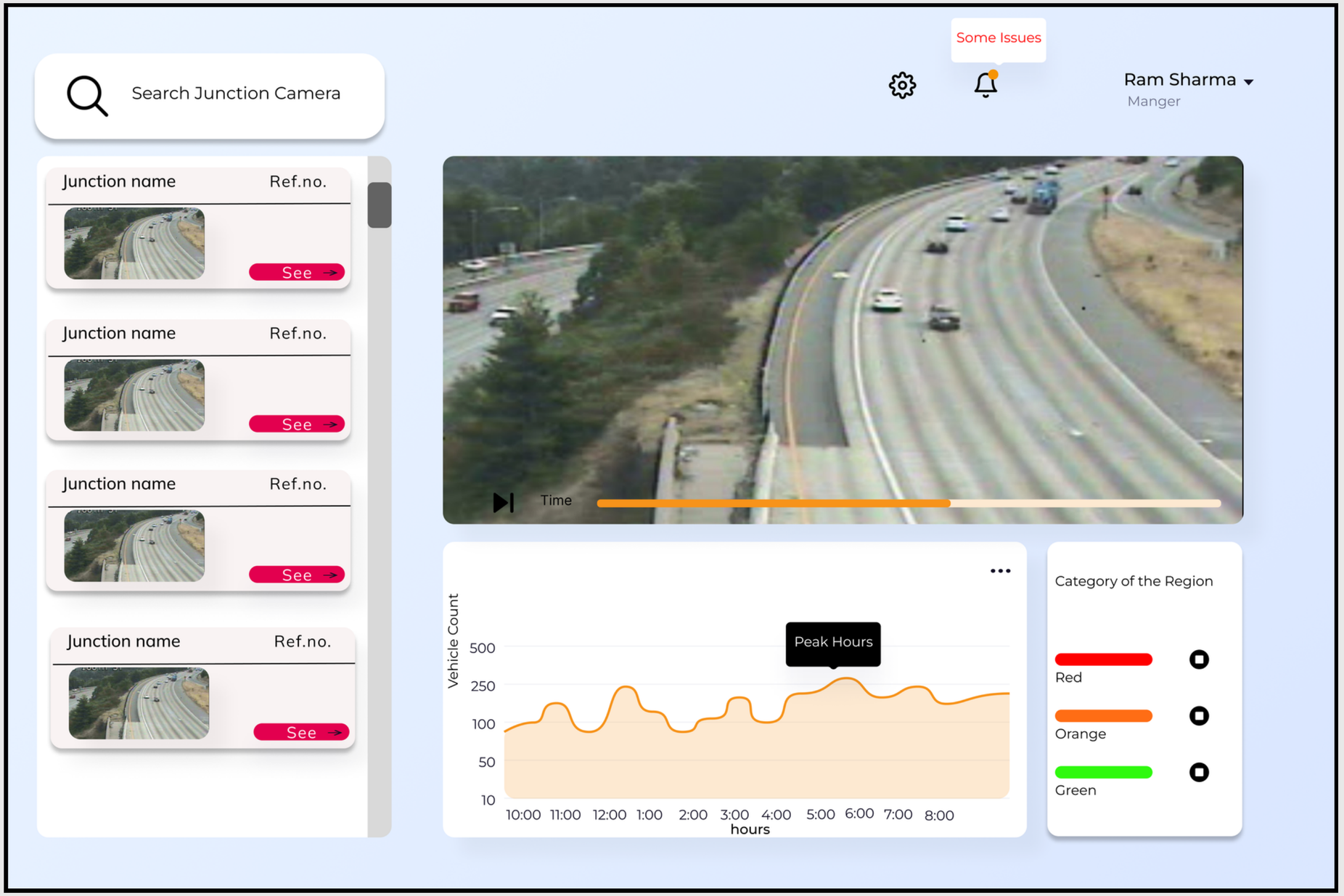


Processed Image



Original Image Frame

Prototype of the Data Analysis Tool/Software



Implementation of Adaptive Traffic Signal

1. Centralised system
2. Q-learning algorithm to create a q-valued table that tells about congestion
3. **Back pressure** increases as cameras detect more vehicles. When back pressure values increase, that shows high congestion and longer waiting times
4. Evaluation of give-back pressure value Q(B) based on previous data
5. High q value implies high congestion density
6. This will indicate that reaching the current signal will take longer
7. Predicted time = Current time + Q(B)
   [Q(B) - predicted waiting time for current back-pressure]
8. Err = Actual arrival time - Predicted time
9. Green signal given to another signal at that junction using the predicted time

Future Work

- Use of ML Model for the vehicle density detection
- Optimization of the Grid Method
- Use of a sensor system (for eg. by insurance companies) in real-life traffic flow detection
- Ideating and Implementing ideas, that can make small changes in Human behavior

References

[1] *A. Cevolini, E. Morotti, E. Esposito, L. Romanelli, R. Tisseur, and C. Misani, "Can Telematics Improve Driving Style? The Use of Behavioural Data in Motor Insurance," arXiv.org, Sep. 06, 2023. LINK (Accessed: 11 March 2024)*

[2] *A. Maipradit, J. Gao, T. Kawakami, and M. Ito, "Adaptive Traffic Control Algorithm Based on Back-Pressure and Q-Learning," IEEE Xplore, Oct. 01, 2019. LINK (Accessed: 11 March 2024)*

[3] *Mathew, T.V. (2023) Area Traffic Control, Area traffic control. Available at: LINK (Accessed: 11 March 2024)*

[4] *K. S. P, Prabu Mohandas, and S. C. S, "Smart junction: advanced zone-based traffic control system with integrated anomaly detector," Jun. 2023, doi: LINK (Accessed: 11 March 2024)*

[5] *M. A. Hossain, V. Nguyen, and E. Huh, "The trade-off between accuracy and the complexity of real-time background subtraction," IET Image Processing, Dec. 2020, doi: LINK (Accessed: 11 March 2024)*

[6] *P., K.S., Mohandas, P. and S., S.C. (2023) Smart junction: Advanced zone-based traffic control system with integrated anomaly detector - annals of operations research. Available at: LINK (Accessed: 11 March 2024).*

CONTACT US

Hitesh Jaiswal
IIT Gandhinagar
jaiswalhitesh@iitgn.ac.in
+91-9029159600

Parag Sarvoday Sahu
IIT Gandhinagar
parag.sahu@iitgn.ac.in
+91-8462901727

Lalji Sankhat
LD College Of Engineering
sankhatlalji0306@gmail.com
+91-9904847911

Kathan Dave
Ahmedabad University
kathan.d@ahduni.edu.in
+91-6351121594

Riya Maheriya
CEPT University
riya.maheriya17@gmail.com
+91-7041069790

THANK YOU!