Real-time Bangla License Plate Recognition System for Low Resource Video-based Applications

Alif Ashrafee
alifashrafee@iut-dhaka.edu
Islamic University of Technology

Akib Mohammed Khan
akibmohammed@iut-dhaka.edu
Islamic University of Technology

Mohammad Sabik Irbaz
sabikirbaz@iut-dhaka.edu
Pioneer Alpha Ltd.

MD Abdullah Al Nasim
nasim.abdullah@ieee.org
Pioneer Alpha Ltd.

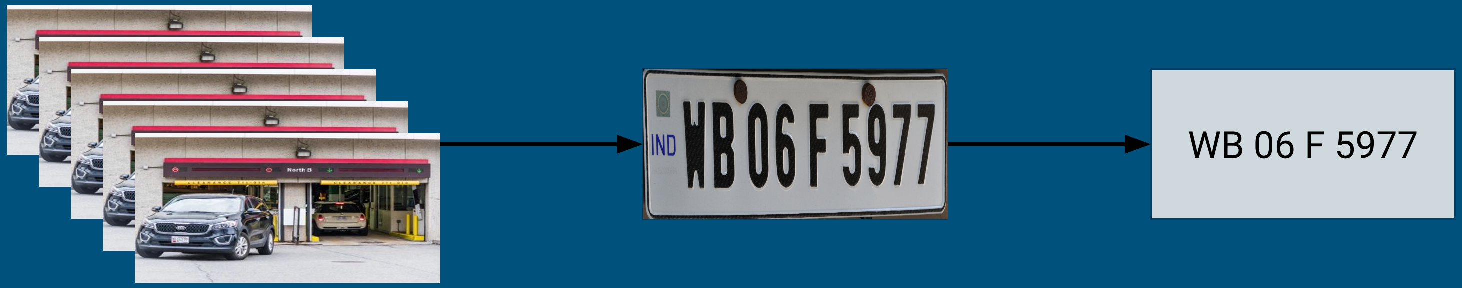
Winter Conference on Applications of Computer Vision 2022

Real-World Surveillance: Applications and Challenges Workshop (RWS @ WACV2022)

Introduction

- Automatic License Plate Recognition (ALPR) systems aim to detect, localize, and recognize license plate characters from vehicles appearing in images or video frames.
- There are no publicly available Bangla datasets for training such systems.
- Our main focus is to provide comprehensive image and video datasets of Bangla license plates in-the-wild.
- We aim to provide a system for real time inference to recognizing license plates from video frames in ***low resource*** environments.

Example of an ALPR system for video feed



Video Frames

Cropped
License Plate

License Plate
Characters

Motivation

- Due to the large volume of vehicles in metropolitan areas of Bangladesh, it is difficult to find a vacant parking slot due to the lack of optimized parking systems.
- There are no publicly available datasets for Bangla License plates to be trained or evaluated on.
- No Bangla ALPR system provides real-time performance using low resource.

Problem Statement

Given a video feed of cars and motorcycles having Bangla license plates, can we teach the machine to identify and recognize license plates?

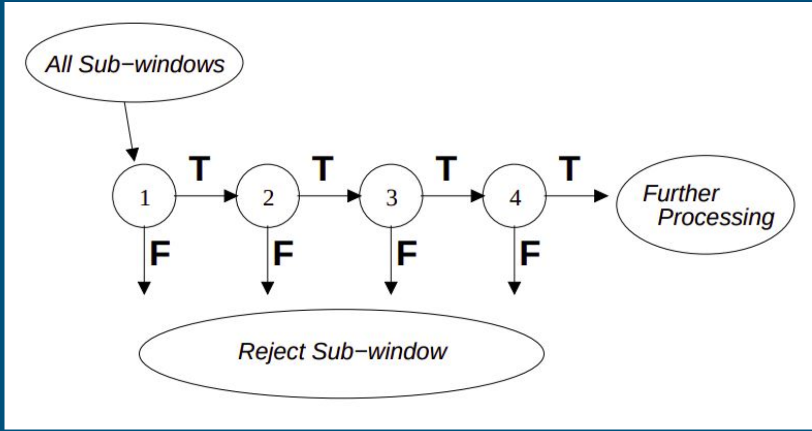
Research Challenges

1. Training for object detection models are computationally expensive
2. No benchmark datasets are available for Bangla license plate videos
3. Separating temporally distinct instances of vehicles from video
4. Running an object detection model in CPU in real-time

Related Works

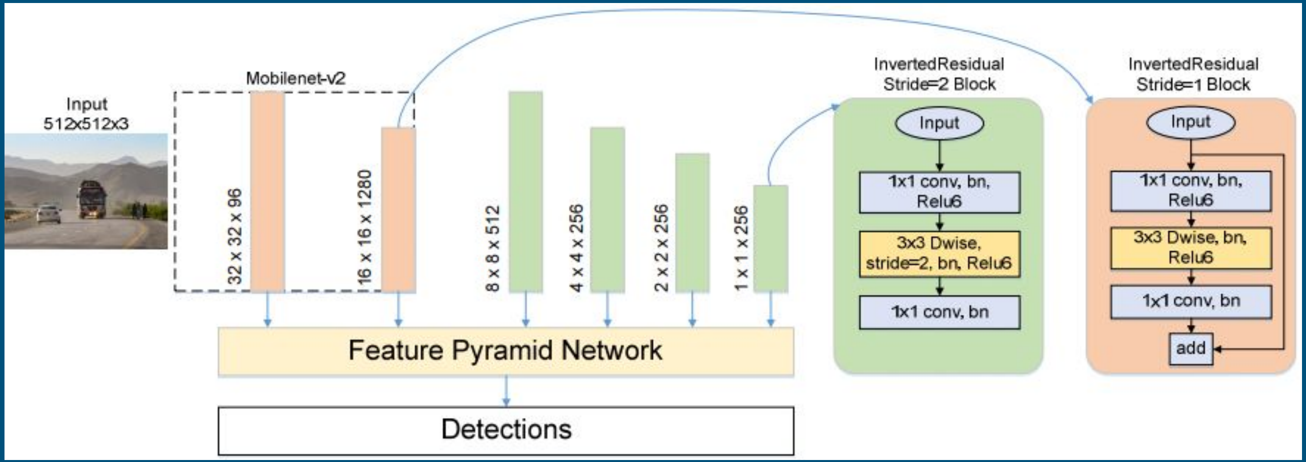
Cascade Classifier [Viola et. al., 2001]

- Trained using positive and negative examples.
- Consists of multiple stages, where each stage is an ensemble of weak classifiers.
- Divide the image into sub-windows.
- The weak classifiers are used to discard the negative subwindows as fast as possible.
- Detect all the positive instances to achieve low false-positive rates.



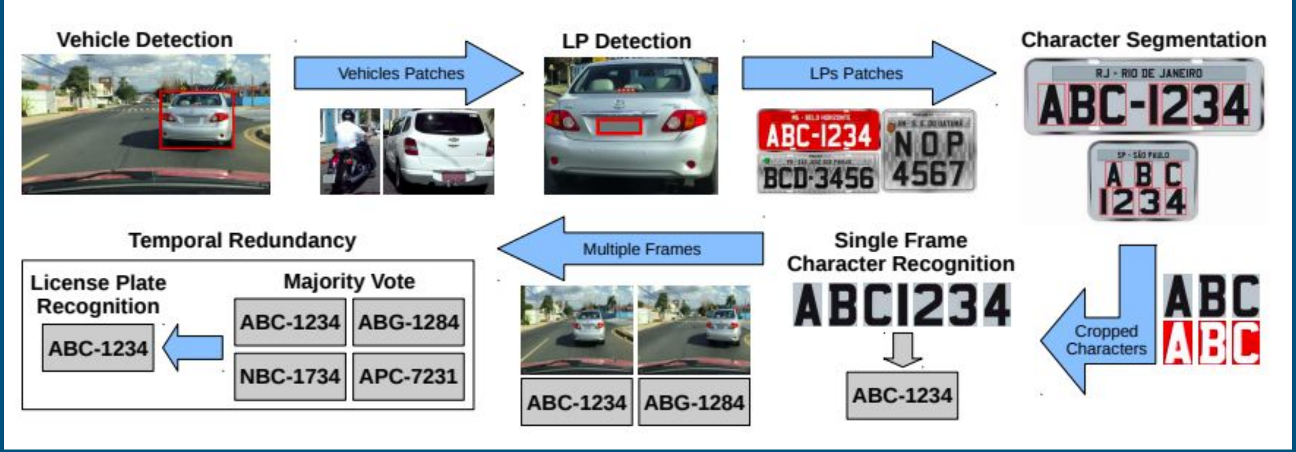
MobileNet SSDv2 [Chiu et. al., 2020]

- Single Shot object detection (SSD) model based on MobileNet-v2[1].
- Lightweight and can be applied with limited computational resources.
- Includes feature pyramid network



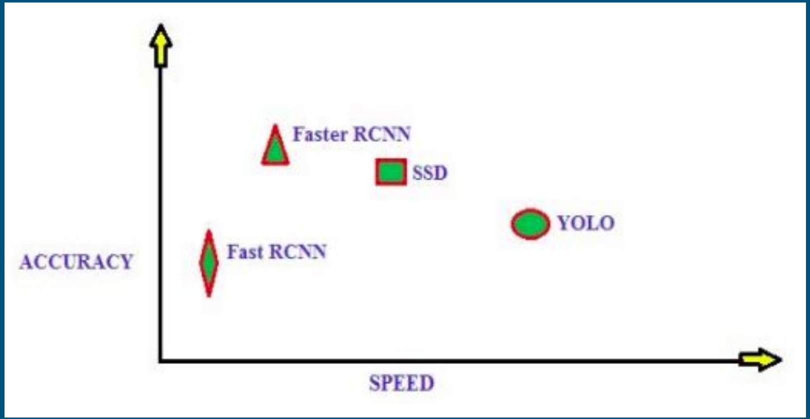
Two stage detection with Temporal Redundancy [Laroca et. al., 2018]

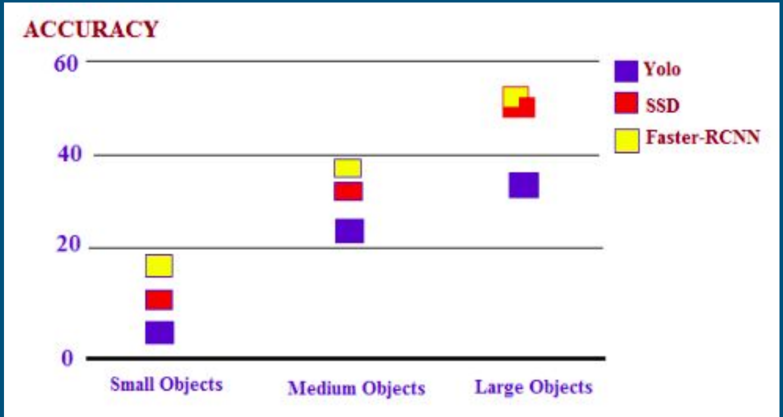
- Detection in two stages, vehicle detection and then LP detection.
- Temporal redundancy information through the union of all frames belonging to the same vehicle.



YOLO vs RCNN vs SSD [Adarsh et. al., 2020]

- SSD provides an optimal solution for both accuracy and speed.
- YOLO is fast but not very accurate, Faster RCNN is more accurate but slower.





Datasets

License Plate Image Dataset

- ➔ We created our own dataset of 1000 images.
- ➔ With different distances, angles and lighting conditions around Dhaka City captured using varying models of smartphones.
- ➔ Converted to 800 x 800 pixels.
- ➔ We manually annotated them by drawing bounding boxes around the license plate regions for training our detection models.


Madina
Gas Stove


SQUARE Hospitals Ltd
HOTLINE
10616

License Plate Video Dataset

- ➔ We collected 79 video clips containing 98 license plates from different types of vehicles using crowdsourcing.
- ➔ Each clip includes single or multiple vehicles from around different districts of Bangladesh.
- ➔ Converted all of the videos to a resolution of 480 x 480 pixels.
- ➔ Each video contains an average of 254 frames and the frame rate of each video was 24 frames per second (FPS).
- ➔ Videos were taken in a variety of locations, such as inside garages, car parks, lanes, and highways.

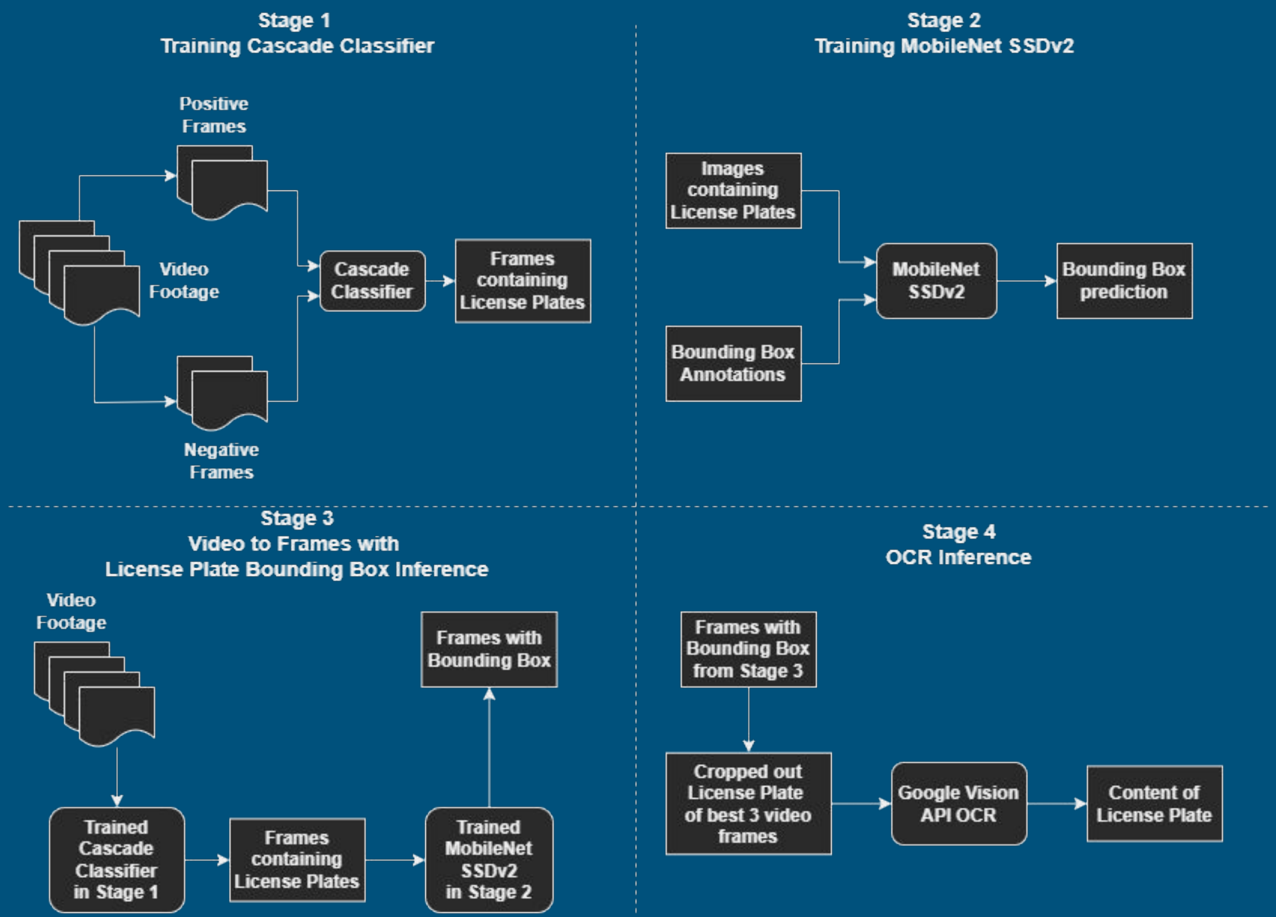
License Plate Video Dataset (cont.)

- Since videos were taken from several different districts, there is a good distribution of the authorized characters in our dataset.
- Videos include plate variations, artifacts on plates, occlusions, illumination changes, and unclear characters.
- We annotated the content of each plate: the ground truth of each character present in the plate, and the number of license plates present in each video.

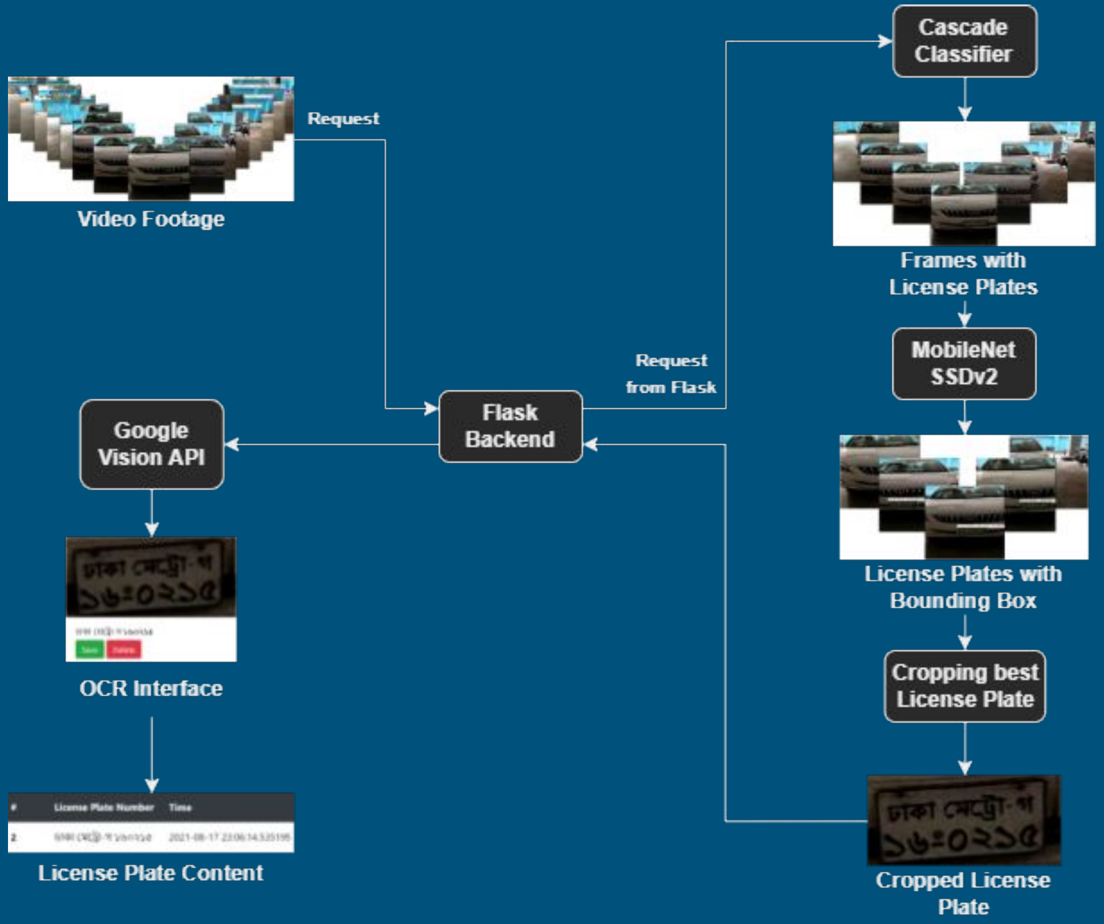

ঢাকা মেট্রো-গ
৩১-৭৮৪০

Proposed Method

Four Stage Pipeline



Overall Pipeline for Real-time interaction

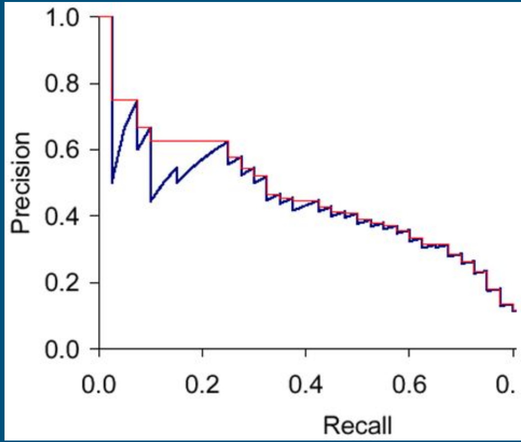


Experimental Analysis

Evaluation Metrics

Mean Average Precision (mAP) for Training detection model

- mAP score is calculated by taking the mean AP over all classes and/or overall IoU thresholds.
- AP is the area under the precision-recall curve.
- Popular metric for evaluating detection models.



Evaluation Metrics (cont.)

Levenshtein Distance for Character Recognition [2]

- Measures the difference between two strings based on the number of insertions, deletions, and substitutions that have to be performed on the target string in order to match it with the reference string.

$$C_{error} = C_{ins} + C_{del} + C_{sub}$$

$$C_{correct} = C_{aln} - C_{error}$$

$$C_{precision} = C_{correct} / C_{H}$$

$$C_{recall} = C_{correct} / C_{R}$$

C_{error} = Total character error

C_{a}□□ = # characters in the aligned sequence

C_{H} = # characters in hypothesis string

C_{R} = # characters in reference string

Experimental Setup

- We trained the MobileNet SSDv2 model using a Tesla T4 GPU on Google Colaboratory.

Image size	300 * 300
Batch Size	24
IOU Threshold	0.35
Learning Rate	0.004
Epochs Run	50

- For testing, all the experiments were conducted on a single thread of an Intel Core i5-7200U mobile CPU with 8GB of RAM and video resized to 480*480 pixels.

Result Analysis (Quantitative)

Training Results

Model	AP	AP50	AP75	APs	APm	APl
MobileNet SSDv2	0.47	0.86	0.47	0.13	0.46	0.55

AP: Average Precision

AP50: AP at IOU thresh = 0.5

AP75: AP at IOU thresh = 0.75

APs: AP for small objects

APm: AP for medium objects

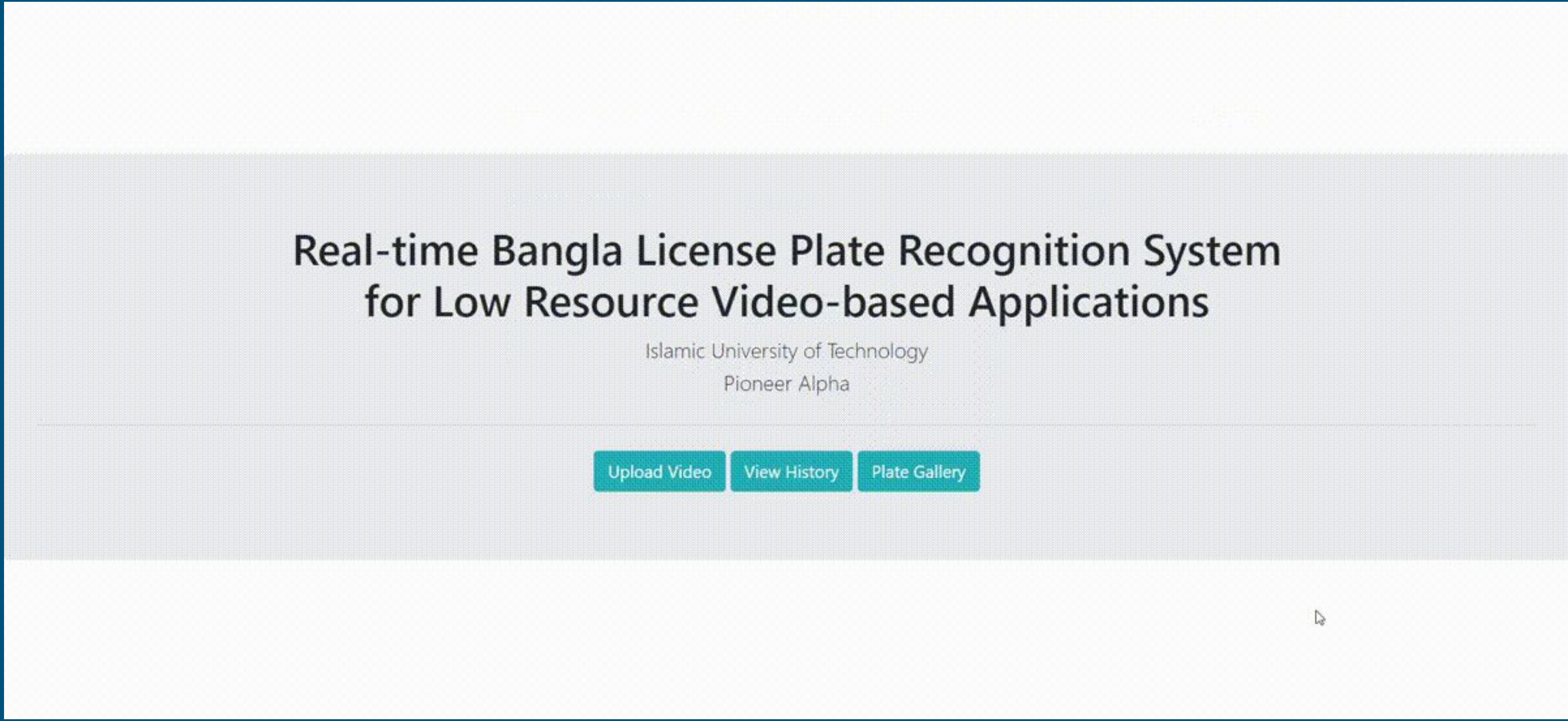
APl: AP for large objects

Result Analysis (Quantitative) (cont.)

Ablation Study

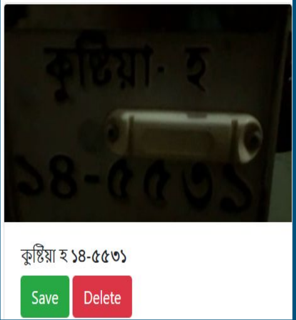
Pipeline	Precision(%)	Recall(%)	F1 Score (%)	Detection Rate(%)	FPS
YoloV3 Tiny	60.2	55.6	57.1	75.5	11.1
Cascade + YoloV3 Tiny	56.3	51.6	53.1	69.4	27.1
SSDv2	**66.1**	58.4	60.5	**98.0**	17.7
Cascade + SSDv2	63.6	**59.3**	**60.8**	82.7	**27.2**

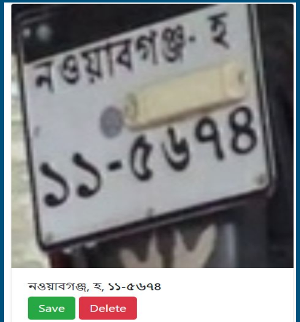
Result Analysis (Qualitative)



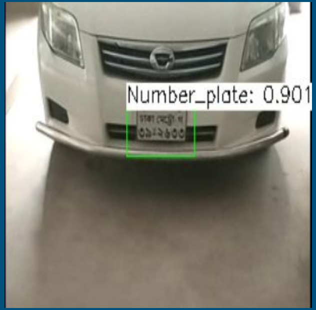
License plates that are **missed by Cascade + SSDv2** but successfully **detected by standalone SSDv2**

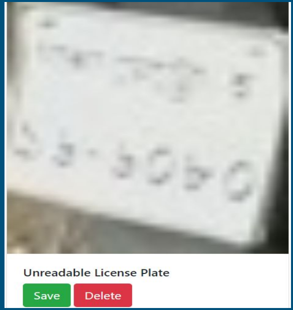
Favorable for Vision OCR

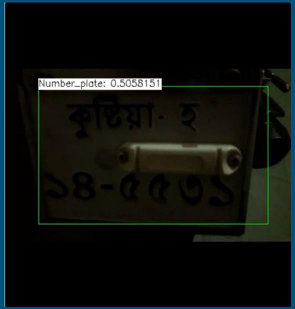


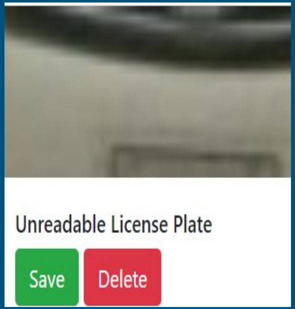


Unfavorable for Vision OCR

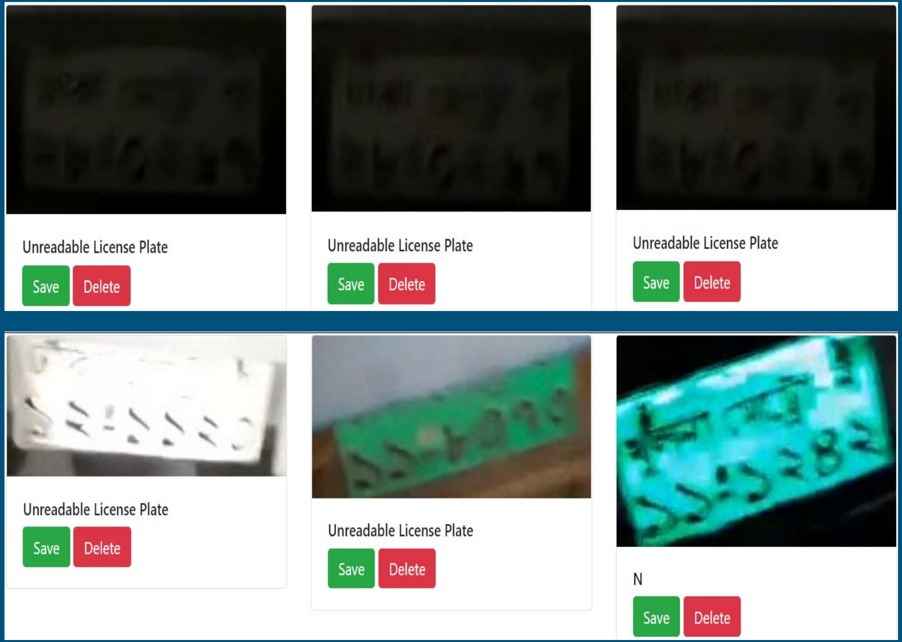








Standalone SSDv2 OCR Predictions that are not favorable for Vision API



Cascade + SSDv2 OCR predictions that are favorable for Vision API

Conclusion and Future Work

- We provide two Bangla license plate image and video datasets.
- We propose a robust system that can detect license plates from video footage in low resource environment.
- Consists of a method for individually storing temporally separate instances of different vehicles appearing in the same video.
- We plan to deploy our system in various real-life environments and verify its day-to-day performance.
- We also plan to build our own Bangla license plate OCR system that can improve the recognition performance.

References

[1] Sandler, Mark, et al. "Mobilenetv2: Inverted residuals and linear bottlenecks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[2] Karpinski, Romain, Devashish Lohani, and Abdel Belaid. "Metrics for Complete Evaluation of OCR Performance." IPCV'18-The 22nd Int'l Conf on Image Processing, Computer Vision, & Pattern Recognition. 2018.

Thank you!