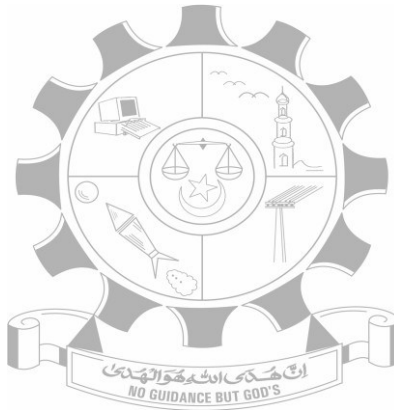


AUTOMATIC NUMBER PLATE RECOGNITION



Project work done by

- | | | |
|----|--------------------------|---------------------|
| 1. | Hari Haran. N | 510617105004 |
| 2. | Pradeep Kumar. B | 510617105020 |
| 3. | Shyam Kumar. S | 510617105026 |
| 4. | VishnuVaradhan. A | 510617105032 |

Guided by

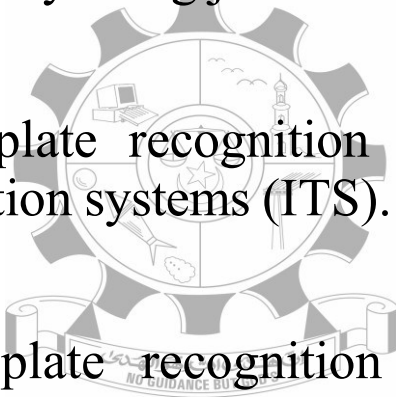
Mohamed Hashim . C, B.E., M.S.,

HEAD OF THE DEPARTMENT

Electrical and Electronics Engineering

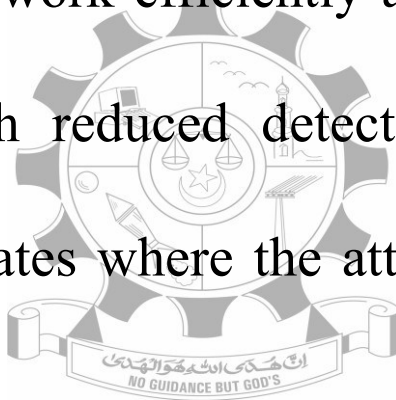
OBJECTIVE

- The aim of the project is to recognize the number plate of vehicle which passes through system automatically by using jetson module and camera.
- Automatic vehicle license plate recognition is an important component of modern intelligent transportation systems (ITS).
- Generally, vehicle license plate recognition is divided into several steps including license plate extraction, image region which contains a license plate, character segmentation, and character recognition.

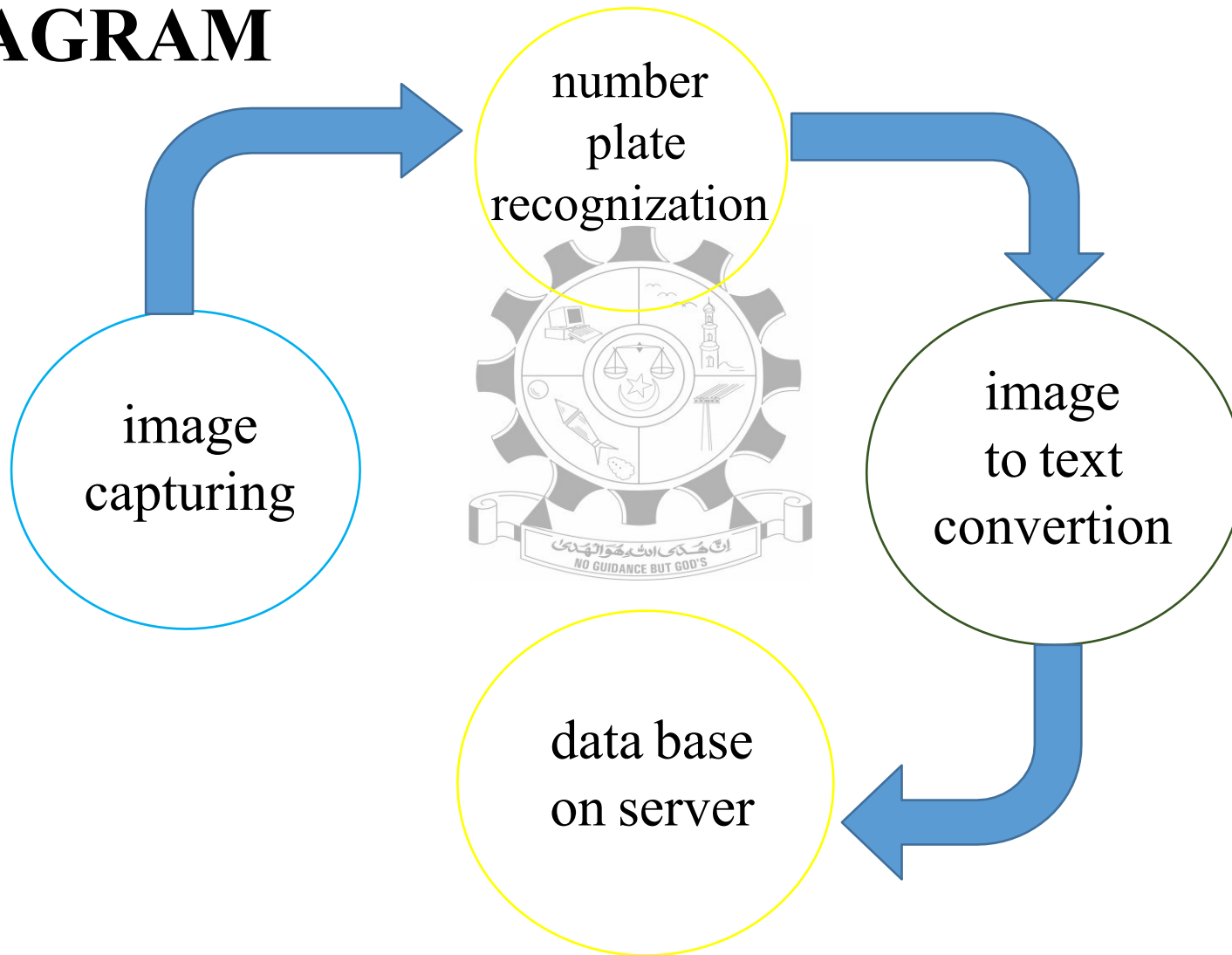


PROJECT DESCRIPTION

To design and develop a real-time detection and license plate recognition system that will work efficiently and adaptive to different traffic environment conditions with reduced detection time. The system should mostly detect the number plates where the attribute of number plate plays a major role.



BLOCK DIAGRAM



HARDWARE REQUIREMENTS

In this chapter, we have discussed about features of each components used in this project.

COMPONENTS USED

The main components used here are,

1. NVIDIA Jetson nano
2. Webcam
3. Memory card
4. Monitor
5. Adapter



SOFTWARE REQUIREMENTS

Software and packages used in this project.

- OS JETPACK 4.4.1(UBUNTU LX11) AN LINUX based operating system
- PYTHON as programming language

Python packages

- JETSON-INFERENCE
- PIL
- PYTESSERACT
- OPENCV
- NUMPY



HARDWARE DESCRIPTION

Component Name	Feature(s)
NVIDIA Jetson nano	2GB RAM
Webcamera	C270
Memory card	64GB
Monitor	HDMI TYPE
Adapter	3A/5V

SOFTWARE DESCRIPTION

INSTALLING JETSON INFERENCE

- Cloning Jetson inference:
`$ git clone --recursive https://github.com/dusty-nv/jetson-inference`
- Open Jetson inference -> `$ cd jetson-inference`
- Then Run the Docker container to download and install the models -> `$ docker/run.sh`
- Then model downloader will appear you have to download **(32. FCN-Resnet18-Pascal-VOC-512*320)** if your network speed is good than it will take 30 or less than 30 minutes.
- Because it will start to download the Docker file called `dustynv/jetson-inference:r32.4.4` (900 MB) for Jetpack 4.4.1 version and L4T R32.4.4 version.

Version based Docker Files:

- Then you can check whether it is installed correctly by this commands:

```
$ cd jetson-inference
```

```
$ docker/run.sh
```

Pic 1

- Then you have to download some other models for Real-time ANPR system:

```
$ cd jetson-inference/tools
```

```
$ ./download.model.sh
```

Download following model packages -> 1. SSD-Mobilenet-v1



- **INSTALLING PIL**

Now we are ready to install PIL. Type the following:

\$ sudo pip install pil

To install Pillow (recommended), type the following:

\$ sudo pip install Pillow

- **INSTALLING PYTESSERACT**

Now we are ready to install pytesseract . Type the following.

\$ pip install pytesseract

- **INSTALLING OPENCV**

OpenCV can be directly downloaded and installed with the use of pip (package manager). To install OpenCV, just go to the terminal and type the following command:

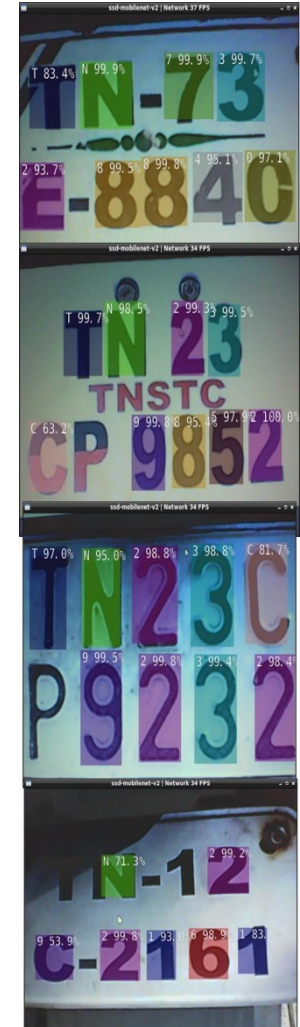
\$ pip3 install opencv-python

- **INSTALLING NUMPY**

The recommended approach is to install the stable Numpy module directly from the Ubuntu repository:

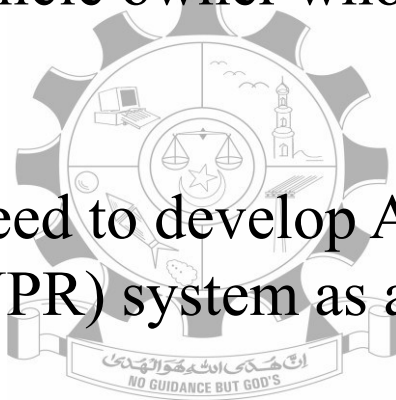
\$ sudo apt install python-numpy

RESULTS / OUTPUT



CONCLUSION

- Traffic control and vehicle owner identification has become major problem in every country. Sometimes it becomes difficult to identify vehicle owner who violates traffic rules and drives too fast.
- Therefore, there is a need to develop Automatic Number Plate Recognition (ANPR) system as a one of the solutions to this problem.
- This ANPR system is secure and accurate one. If this system is employed in our society then crimes can be minimized and records can be maintained. We can expect this system to create a larger impact in our environment



REFERENCES

1. Hamey, L.G.C. and C. Priest, Automatic Number Plate Recognition for Australian Conditions. Proceedings of the Digital Imaging Computing: Techniques and Applications, 2005: p. 8.
2. Chen, Z.-X., et al., Automatic License-Plate Location and Recognition Based on Feature Saliency. IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, 2009. 58(7): p. 3781-3785.
3. Qadri, M.T. and M. Asif, Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition. International Conference on Education Technology and Computer, 2009: p. 335 - 338.
4. Meier, R., Professional Android 2 Application Development 2010, Indianapolis, Indiana: Wiley Publishing, Inc. 399.
5. Maarif, H.A.-Q. and S. Sardy, Plate Number Recognition by Using Artificial Neural Network. Prosiding Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi, 2006: p. 176-182.

6. Badr, A., et al., Automatic Number Plate Recognition System. Annals of the University of Craiova, Mathematics and Computer Science Series, 2011. 38(1): p. 62-71
7. Kim, K.K., et al. Learning-based approach for license plate recognition. in Neural Networks for Signal Processing X, 2000. Proceedings of the 2000 IEEE Signal Processing Society Workshop. 2000.
8. Dai, Y., et al. A high performance license plate recognition system based on the web technique. in Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE. 2001.
9. Khalifa, O., et al., Malaysian Vehicle License Plate Recognition. The International Arab Journal of Information Technology, 2007. 4(4): p. 359-364.
10. Duan, T.D., et al., Building an Automatic Vehicle License-Plate Recognition System. Intl. Conf. in Computer Science – RIVF'05, Vietnam, 2005: p. 59-63.

