Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 06, June-2020

Solution to Determine Vehicle Density Through Camera System

Hoang Ba Dai Nghia, Tran Hoang Vu
The University of Danang - University of Technology and Education, Vietnam

***Abstract*- Currently, the smart traffic system is one of the top development priorities for Vietnam to build smart cities across the country. Traffic situation in big cities in Vietnam has become a problem for road users during rush hours. Local congestion has occurred on many roads. Reducing congestion in big cities is an urgent issue. Therefore, in this paper, we propose a solution to determine vehicle density, to warn traffic congestion through the city Camera system.**

Key words- ITS; Deep Learning; FCN; Warning of traffic congestion

I. INTRODUCTION

Currently, solving traffic congestion in developing countries is an urgent issue, studies of camera application to determine speed [1], [2].

AI - Artificial Intelligence is the science that makes machines intelligent, with the ultimate goal of allowing robots to possess the human-like capabilities. In fact, AI has had a significant impact on our lives, in ways that improve human health, safety and productivity. For example, face recognition via video [3]

The deployment of AI technologies is important to promote the scope of IoT. AI technologies are highly customized for individual tasks and each application requires specialized research and structure. Deep Learning, a form of machine learning based on trained data sets, has facilitated advanced pattern recognition in images, video and object/ activity recognition. Its algorithms can be widely applied to an array of applications that rely on pattern recognition.

Therefore, in this paper, we apply Deep Learning to analyze images from traffic cameras to determine the vehicle density participating in the traffic. Providing support information about the current traffic situation to road users to know the situation of congestion on roads in big cities in Vietnam, the works we have contributed in the paper include:

- Proposing an algorithm to determine vehicle density through images captured directly from traffic cameras.
- Building a server system to store warning data.

The remainder of the paper is organized as follows: Part 2. Presentation of related works. Part 3. Development of the experimental system and its results. Conclusion and furture development direction in Part 4.

II. RELATED WORKS

A. *Semantic segmentation*

Semantic segmentation is the process of converting digital images into simple images or representations into something more meaningful and easily analyzed. Specific key steps:

1. Finding out the central points of the objects, including making predictions for the input image.
2. The next step is to locate/ detect, provide not only the names of objects at each central point but also provide additional information related to the location of the object.
3. Finally, semantic segmentation provide detailed results from the object name predictions of each pixel, so that each segmentation is assigned the corresponding object name.

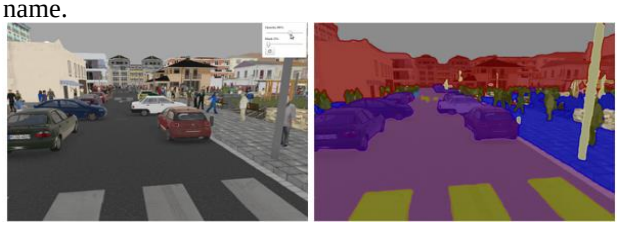

Figure 1. A case of semantic segmentation

The general technique when building the network for this problem is to build a model of 2 components of the encoder and decoder.

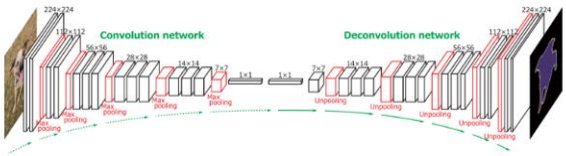

Figure 2. Encoder and decoder

In particular, the encoder is to reduce the length and width of the image using the convolutional and pooling layers. The decoder is used to restore the original image size. The encoder is usually just a regular CNN but removes the last fully conected layers. The available networks can be used in the encoder as VGG16, VGG19, Alexnet, Mobilenet ... and the decoder depending on the network architecture can be built differently. For example in FCN:

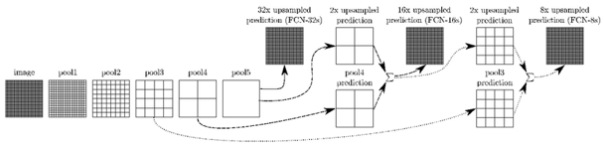

Figure 3. FCN Architecture [4]

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 06, June-2020

In FCN architecture, there are 3 ways to build a decoder to form 3 different types of FCN: FCN32, FCN16, FCN8. For FCN32, after reaching the final pooling layer (the above example is the 5th pooling layer) just upsample to the original size. For FCN16, at the 5th pooling layer we multiply 2 times to get the size equal to the size of the 4th pooling layer, then add 2 layers together and then upsample to the original image size. Similarly with FCN8 we connect to the 3rd pooling layer.

B. Basic concepts in semantic segmentation model

Convolution Layer

In a regular neural network, from the input, go through the hidden layers and then output. For CNN, the Convolutional Layer is also a hidden layer, but, the Convolutional Layer is a set of feature maps and each of these feature maps is a scan of the original input, but extracted to specific features. The scan results depend on the values inside the kernel. This is a matrix that will scan the input data matrix, from left to right, top to bottom, and multiply corresponding values of the input matrix into the kernel matrix and then sum it up, giving via activation funciton (sigmoid, relu, elu, ...), the result will be a specific number, the set of numbers is another matrix, which is the feature map.

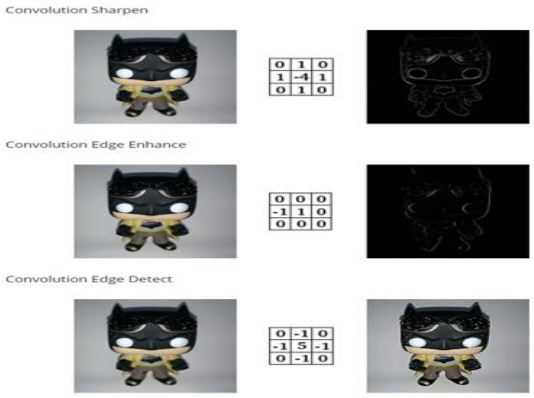

Figure 4. The result of the convolutional layer with different kernels

Stride và Padding

Stride is the distance between the 2 kernels when scanning. With stride = 1, the kernel will scan two adjacent cells, but with stride = 2, the kernel will scan cell 1 and cell 3. Ignore the middle box. This is to avoid duplicate values in the scanned cells.

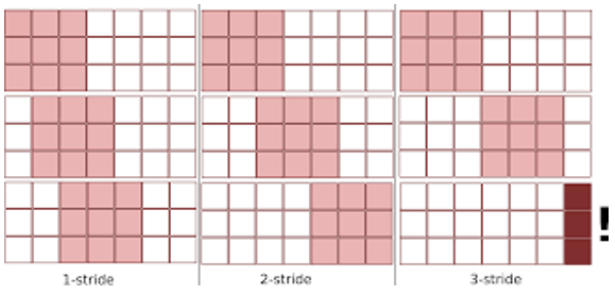

Figure 5. Stride và Padding

The larger stride and kernel size is, the smaller the size of feature map is, partly because the kernel must be completely in the input. There is a way to keep the size of feature map unchanged. This is Padding. When adjusting padding = 1, which means that we have added a cell around the edges of input, the thicker the wrap is, the more padding will be needed.

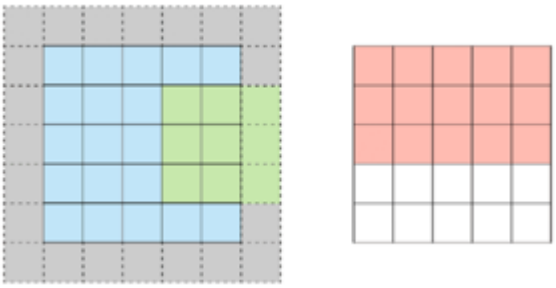

Figure 6. Select stride and kernel size

The gray part is the additional wrap to the input

With stride = 1 and padding = 0, from the initial input image, scan the kernel and form the following cells to map into feature map

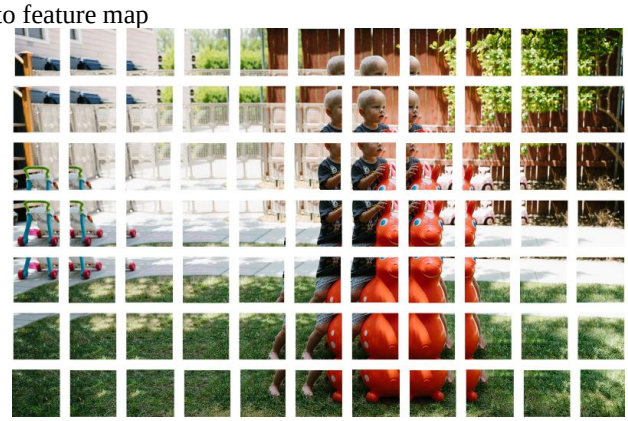

Figure 7. Feature map

Pooling Layer

The purpose of pooling is simple, it reduces the number of hyperparameters that need to be calculated, thereby reducing calculation time, avoiding overfitting. The most common type of pooling is max pooling, which takes the maximum value in a pooling window. Pooling works almost like convolution, there is a sliding window called pooling window, this window slides through each value of the input data matrix (usually the feature maps in the convolutional layer), select a value from the values in the sliding window (with max pooling we will get the maximum value).

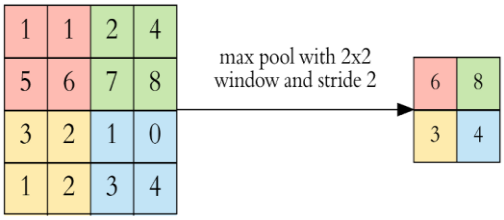

Figure 8. Max pooling

Figure 9.

Transpose convolution Layer

Transposed convolutional layer is a transformation in the opposite direction to convolution, capable of mapping for a larger size result..

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 06, June-2020

Suppose that we want to increase the denominator of the 2x2 input matrix into a 4x4 matrix that transforms through a 3x3 kernel as follows:

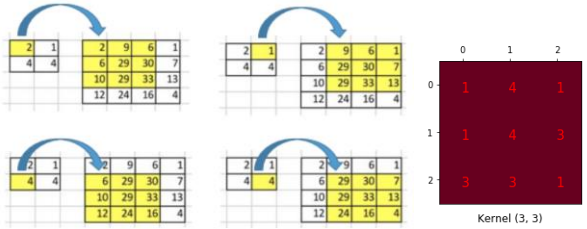

Figure 10. Transpose convolution

Arranging the values of 3x3 kernel into 16x4 matrix and 2x2 input matrix into 4x1 matrix. Rearranging the values of matrix multiplication results, we obtain a 4x4 matrix.

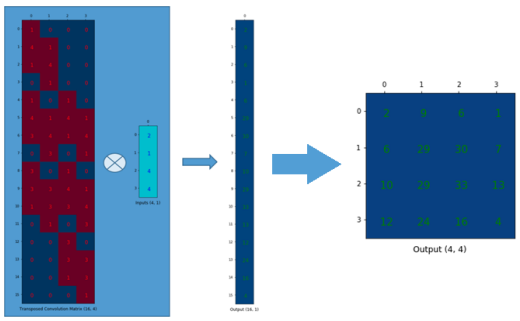

Figure 11. Rearranged Matrix

III. SYSTEM DEPLOYMENT

A. *Statistical storage block*

Creating mysql database with the following structure:

Density Table (Store density values and traffic status at a given time)

Column name	Type	Description
ID	Bigint	Number of each storage
IDCamera	Bigint	Camera number
CreateDatetime	Datetime	Storage time
Density	float	Road density
DensityLevel	Int	Traffic status by number

tmDensityType Table (Store names and limits of traffic statuses)

Column name	Type	Description
IDDensity	Bigint	Number of each status
DensityName	Nvarchar	Traffic status name
DensityPercent	Int	Lower limit value of road density

tmCameras Tablet (Store information and Camera configuration)

Column name	Type	Description
ID	bigint	Camera number
CameraName	nvarchar(100)	Camera name
ConnectString	nvarchar(100)	Connection string
Description	nvarchar(MAX)	Description Camera
Setting	nvarchar(MAX)	Configuration of the program
TimeReport-Second	int	Lamp cycle time (in seconds)
Lat	float	Latitude index of camera position on the map
Long	float	Longitude index of camera position on map

B. *DETERMINING THE DENSITY*

In general, the status of the intersections is reflected through the functional area of the branch leading to the intersection including the situation of traffic congestion. A congested intersection will result in congestion at the inlet branches. Therefore, studying the status of the functional area on the inlet branch can give us necessary warnings about traffic conditions at the intersection.

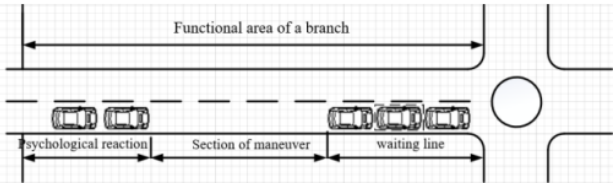

Figure 12. Traffic situation at an intersection [5]

Determining the road segmentation is the remaining part of the road / lane segmentation model.

The percentage encroaching on the road of the vehicles will be saved the smallest and largest value in a light cycle from which to determine the difference value.

C. *System training results*

The model is trained with the Kitti Road database [6] for the road / lane detection problem with over 500 images with corresponding segmentation images.

Figure 13. Training with Kitti Road

Optimizing the model using cross entropy to find loss function and optimizing by Adam algorithm [7] to obtain the results in both dropout cases of 0.5 and 0.75

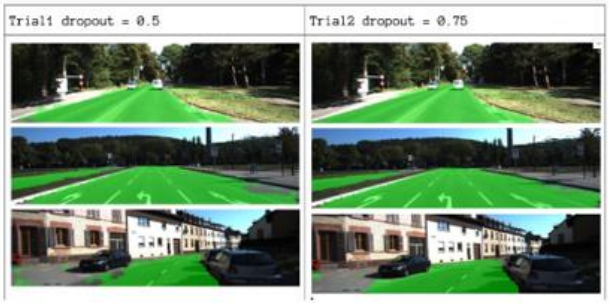

Figure 14. Optimizing with the Adam algorithm

D. *Experimental results*

After the image data obtained from the Traffic Camera was applied through the software system that our team had developed, the result of the covering density is analyzed as Figure 14.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 06, June-2020

Figure 15. Determining the density of coverage

Based on the result of the percentage of density covering the vehicles on the road over time, the system will issue a warning to road users.

IV. CONCLUSION AND DIRECTION FOR PROSPECTIVE DEVELOPMENT

In this paper, we have built a system for warning congestion through the traffic camera system.

In the coming time, we propose a solution to guide the road users to avoid congested intersections..

REFERENCES

[1] Adi Nurhadiyatna, Benny Hardjono, Ari Wibisono,Wisnu Jatmiko, and Petrus Mursanto “ITS Information Source: Vehicle Speed Measurement Using Camera as Sensor” ICACSIS 2012, ISBN: 978-979-1421-15-7, pp.179-184

[2] Asif Khan, Imran Ansari, Dr.Mohammad Shakowat Zaman Sarker and Samjhana Rayamajh “Speed Estimation of Vehicle in Intelligent Traffic Surveillance System Using Video Image Processing” nternational Journal of Scientific & Engineering Research, Volume 5, Issue 12, December-2014, pp 1384 -2390

[3] Mahesh Jangid, Pranjul Paharia and Sumit Srivastava “Video-Based Facial Expression Recognition Using a Deep Learning Approach”

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox “U-Net: Convolutional Networks for Biomedical Image Segmentation” University of Freiburg, Germany

[5] Phan Cao Tho, Duong Minh Chau “The functional area of signalized intersection in urban areas in vietnam” Journal of Science and Technology – University of DANANG, No1(42).2011

[6] https://github.com/MarvinTeichmann/KittiSeg

[7] Sebastian Ruder “An overview of gradient descent optimization Algorithms” Insight Centre for Data Analytics, NUI Galway Aylien Ltd., Dublin