

# BICAME BORNEO INTERNATIONAL CONFERENCE ON APPLIED MATHEMATICS & ENGINEERING

**“Engineering & Applied Science  
Advances for Climate Changes Adaptation & Mitigation”**

BALIKPAPAN, EAST KALIMANTAN, INDONESIA - 10 DECEMBER 2018



2018 2nd Borneo International Conference on Applied Mathematics and Engineering (BICAME) took place 10 December 2018 in Balikpapan, Kalimantan, Indonesia.

IEEE catalog number: CFP18R13-ART  
ISBN: 978-1-5386-7725-4

Copyright and Reprint Permission: Abstracting is permitted with credit to the source. Libraries are permitted to photocopy beyond the limit of U.S. copyright law for private use of patrons those articles in this volume that carry a code at the bottom of the first page, provided the per-copy fee indicated in the code is paid through Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923. For other copying, reprint or republication permission, write to IEEE Copyrights Manager, IEEE Operations Center, 445 Hoes Lane, Piscataway, NJ 08854. All rights reserved. Copyright © 2018 by IEEE.

## Steering Committee

Mohamad Muntaha (Institut Teknologi Sepuluh Nopember & Institut Teknologi Kalimantan, Indonesia)

Subchan Subchan (Kalimantan Institute of Technology, Indonesia)

Sulistjono Sulistjono (Kalimantan Institute of Technology, Indonesia)

## Organizing Committee

# General Chairman

Mochammad Purwanto (Institut Teknologi Kalimantan, Indonesia)

# Co-Chair

Muhammad Ma'arij Harfadli (Institut Teknologi Kalimantan, Indonesia)

# Treasurer

Wahyu Yul (Balikpapan, Indonesia)

# Secretary

Bima Catur Apriadi (Institut Teknologi Kalimantan, Indonesia)

## Technical Committee

Indira Anggriani (Institut Teknologi Kalimantan, Indonesia)

Ariyaningsih Ariyaningsih (Kalimantan Institute of Technology, Indonesia)

Lusi Ernawati Ernawati (Institut Teknologi Kalimantan, Indonesia)

Tegar Palyus Fiqar (Institut Teknologi Kalimantan, Indonesia)

Irma Fitria (Institut Teknologi Kalimantan, Indonesia)

Andhika Giyantara (Kalimantan Institute of Technology, Indonesia)

Adrian Gunawan (Kalimantan Institute of Technology, Indonesia)

Riki Herliansyah (Institut Teknologi Kalimantan, Indonesia)

Andi Idhil Ismail (Institut Teknologi Kalimantan, Indonesia)

Andromeda Laksono (Institut Teknologi Kalimantan, Indonesia)

Menasita Mayantasari (Institut Teknologi Kalimantan, Indonesia)

Rizky Nugroho (Institut Teknologi Kalimantan, Indonesia)

Ridho Prasetyo (Institut Teknologi Kalimantan, Indonesia)

Anggit Rahajeng (Kalimantan Institute of Technology, Indonesia)

Chandra Suryani Rahendaputri (Kalimantan Institute of Technology, Indonesia)

Intan Dwi Wahyu Setyo Rini (Institut Teknologi Kalimantan, Indonesia)

Healty Susantiningdyah (Institut Teknologi Kalimantan, Indonesia)

Author	Session	Start page	Title
<b>A</b>			
Abu Zalata, Mahmoud	2C.6	228	
Achmadi, Bima	3B.4	321	
Ade Indra Saputra, Andika	2D.3	253	
Afandi, AN	2C.4	218	
Ahdan, Muhammad	3C.5	349	
Ahmed, Saif	2A.8	158	
Alameka, Faza	1C.3	70	
	1C.5	82	
Albryansah, Ganang	1D.3	102	
Aldrin Akbar, Mohammad	1D.1	91	
Alfian, Nurkholis	2A.1	123	
Alfian, Nurkholis	2D.6	266	
Alginahi, Yasser	3A.2	286	
Ali, Jahan	3A.2	286	
Alifah, Suryani	2B.8	199	
Amalia, Rahma Nur	3C.3	343	
Andhara, Ade	2B.4	179	
Andika Sujarwo, Nararya	2B.1	164	
Anggarini, Ufafa	2D.1	244	
Anggraeni, Garnis	2B.5	184	
Anggraeni, Merry	3C.1	332	
Anifah, Lilik	3C.2	338	
Anwar, Nadjadji	1B.3	42	
Aphale, Siddharth	1A.2	7	
Arminas, Arminas	3C.5	349	
Astrowulan, Katjuk	3C.3	343	
Atmaja, Lukman	2D.4	258	
Atrinawati, Lovinta Happy	1D.2	96	
Aurelia, Mela	2D.6	266	
Ayunina, Qurrotin	2D.1	244	
<b>B</b>			
B Hi Ibrahim, Muhdi	1D.1	91	
Baffour, Adu	2A.2	128	
Banaget, Cut Keumala	1A.5	23	
Bataona, Daniel	3A.7	295	
Benseghir, Hocine	3A.3	291	
	3A.2	286	
bin Mohd Idris, Mohd Fadzil	3A.3	291	
Budiman, Edy	1C.4	76	
	1C.3	70	
	1D.1	91	
	1C.1	60	
	1C.5	82	
Bura, Romie	3B.1	307	
<b>C</b>			
Caraka, Rezzy Eko	3A.1	280	
<b>D</b>			
Dahlan, Akhmad	2D.8	275	
Damanik, Hillman	3C.1	332	
Defisa, Tomi	3C.1	332	
Dengen, Nataniel	2A.7	153	
Dharmawan, Yudha	1B.1	31	
<b>E</b>			
Elmunsyah, Hakkun	2B.3	174	
<b>F</b>			
Fadli, Sofiansyah	1C.5	82	

Fahdian, Ali	3B.4	321
Fansuri, Hamzah	2D.3	253
Fifing, Fifing	2B.5	184
Fikar, Muhammad 'Aizul	3C.7	355
Filiana, Firilia	2B.4	179
Firdaus, Muhammad	1D.1	91
	2B.7	193
	1C.5	82
Firmansyah, Rifqi	3C.2	338
<b>G</b>		
Gharte, Niraj	1A.2	7
Giyantara, Andhika	2C.8	239
	2C.5	223
	3B.4	321
Gultom, Rudy	2B.5	184
	2A.3	134
Gunawan, Adrian	2D.3	253
Gunawan, Gunawan	2B.8	199
Gunawan, Vincentius	2A.4	138
Guo, Jiandong	2A.2	128
<b>H</b>		
Hadi Sirad, Mochammad Apriyadi	2C.1	203
Hadid, Muhammad	1D.3	102
Haeruddin, Haeruddin	1C.4	76
	1C.1	60
Hanafi, Muh	3A.8	301
Hantoro, Ridho	1A.6	27
Haq, Sigit	2B.3	174
Harfadli, Muhammad Ma'arij	1A.5	23
Harsono, Brian	3B.3	317
	3B.2	313
	2C.2	207
Hartono, Hartono	1B.4	47
Hartono, Joko	3B.2	313
	2C.2	207
	3B.3	317
Hasan, Wordh	2A.8	158
Hasanah, Barokatun	2B.4	179
Hidayat, Givan	2C.8	239
Huda, Thorikul	3C.3	343
<b>I</b>		
Ibrahim, Abdullah	3A.3	291
Indriani, Siti	2C.7	233
Irfan, Muhamad	3B.1	307
Irwanto, Dwi	3B.1	307
Islamiyah, Islamiyah	2A.7	153
Ismail, Ika	1B.6	55
Isnanto, R. Rizal	2A.4	138
Izzudin, Muhammad	2C.5	223
<b>J</b>		
Jamil, Muh	1D.1	91
Jintaka, Dhandis R.	3B.3	317
	3B.2	313
	2C.2	207
Juwita, Asmi	1A.3	12
<b>K</b>		
Kabir, Muhammad Nomani	3A.3	291
	3A.2	286
Kanalebe, Herman	2B.6	188

Kartono, Agus	2A.6	147
	2A.5	143
Kewo, Angreine	2C.3	213
Khaja, Mohammad	2A.8	158
Khan, Mohammad Monirujjaman	2A.8	158
Krisna, Roy	2D.4	258
Kusi, Goodlet	2A.2	128
Kusuma Priyanto, Yun	3B.4	321
<b>L</b>		
Laksono, Andromeda	1B.6	55
Lestari, Dyah	1C.6	87
Liu, Xiufeng	2C.3	213
Lois, Gianda Almyra	1D.2	96
Lulla, Soham	1A.2	7
<b>M</b>		
Mahandi, Yogi	1C.6	87
Mahendra Putra, Gubtha	2B.7	193
Mahfudi, Isa	2D.7	270
Maman Abadi, Agus	1B.4	47
Manembu, Pinrolinvic	2C.3	213
Manulangga, Gloria	3A.7	295
Maria, Eny	2A.7	153
Martoyo, Ihan	2B.6	188
Mudeng, Vicky	2A.1	123
	2B.4	179
	2C.5	223
	3B.4	321
Muhammad, Damris	2D.2	249
Muladi, Muladi	2B.2	169
	2B.3	174
Mulyadi, Mulyadi	1C.2	66
Mulyanto, Karina Berliana	2D.6	266
Munir, Buyung	3B.3	317
	3B.2	313
	2C.2	207
Muntaha, Mohamad	2D.5	263
Murti, SD	1A.3	12
Muttaqin, Adharul	2B.1	164
<b>N</b>		
Natiand, Huda	2C.5	223
Nielsen, Per	2C.3	213
Nilda, Nilda	1C.2	66
Ningrum, Cahyaningtyas	1B.6	55
Nisa, Wahdiyaton	2A.1	123
Nojeng, Syarifuddin	1D.5	113
Novizon, Novizon	2C.8	239
Nugraha, Endah	1A.6	27
Nugroho, Muhammad Agung	2D.8	275
Nur Alam, Sitti	1D.1	91
Nurwahidah, Andi	3C.5	349
	1C.2	66
<b>P</b>		
Pakpahan, Herman	2B.7	193
	1D.4	107
Pardede, Marincan	2B.6	188
Prabasari, Ira	2D.2	249
Prabowo, Rachmad	1B.2	36
	1B.1	31
Prasetyo, Joni	1A.3	12

Pratama, Boby	2B.5	184
	2A.3	134
Pratama, Detak	1A.6	27
Purwanto, Mochammad	2D.5	263
	2D.4	258
Pusfitasari, Memik	2D.4	258
Puspitaningayu, Pradini	3C.2	338
Puspitasari, Novianti	1C.3	70
Putranto, Bambang Purnomosidi Dwi	2D.8	275
<b>R</b>		
Rahmawati, Yuni	2C.4	218
Riawan, Dedet	2C.6	228
Rini, Intan Dwi Wahyu Setyo	2D.3	253
Robandi, Imam	2C.6	228
Rumaherang, W.	1A.4	18
<b>S</b>		
Sahasrabuddhe, Aniket	1A.2	7
Sandy Ade Putra, Leonardus	2A.4	138
Saptaningtyas, Fitriana	1B.4	47
Saputra, Okcy	2A.1	123
Sastrawan, Febrian	1B.5	51
Sendari, Siti	2B.2	169
	2B.3	174
	1C.6	87
Setyadi, Hario	1D.4	107
	2B.7	193
Shah, Jaydeep	1A.2	7
Sholihah, Atti	1A.3	12
Sholikah, Umi	2D.6	266
	2D.3	253
Sinaga, Noval	1C.5	82
Sivaraman, K	1A.1	1
Su'ud, Zaki	3B.1	307
Sudarmadji, Petrisia	3A.7	295
Sudiharto, Dodi Wisaksono	3C.7	355
Sulisetyono, Aries	3C.3	343
Sulistjono, Sulistjono	1B.6	55
	1B.2	36
	1B.1	31
Supriyadi, Asep	2A.3	134
Suryani, Erma	1D.6	117
Suryawan, Sayekti	1D.6	117
Susanto, Tony Dwi	1D.4	107
Susilo, Rizky	2D.7	270
Suyanto, S	1A.6	27
Syafii, Syafii	2C.8	239
Syuhada, Akmal	2D.5	263
<b>T</b>		
Taha, Syamsumarlin	2C.1	203
Tambunan, Handrea	3B.3	317
Taruk, Medi	2A.7	153
	1D.4	107
Taufik, Moch	2B.8	199
Taufiq, Imam	2D.8	275
Taufiqi, Ilham	2C.4	218
Tejawati, Andi	1C.4	76
Thebora, Meyly	1A.3	12
Tofani, Kemas	3B.3	317
	3B.2	313

	2C.2	207
Toharudin, Toni	3A.1	280
Tope, Jayesh	1A.2	7
Triana, Yunita	1B.2	36
	1B.1	31
Triwiyatno, Aris	2A.4	138
Tunas, I Gede	1B.3	42
<b>U</b>		
Ummatin, Kuntum	2D.1	244
Uranus, Henri	2B.6	188
Utama, Bintang	1C.6	87
Utami, Putri	1B.1	31
<b>W</b>		
Wadu, Robinson	3A.7	295
Wardhana, Reza	2B.7	193
	1D.4	107
Wibowo, Ferry Wahyu	3B.5	326
Widagdo, Putut	1D.4	107
	2B.7	193
Widians, Joan	1C.5	82
Widiyaningtyas, Triyanna	2B.2	169
Widodo, Arif	3C.2	338
<b>Y</b>		
Yogaswara, Rachmad	2D.4	258
Yulianto, Fazmah Arif	3C.7	355
Yuwono, Rudy	2B.1	164
<b>Z</b>		
Zaeni, Ilham	1C.6	87
Zainuri, Akhmad	2B.1	164
Zulwisli, Zulwisli	2C.8	239



# Real-time Vehicle License Plate Detection by Using Convolutional Neural Network Algorithm with Tensorflow

Imam Taufiq  
Information System  
STMIK AKAKOM  
Yogyakarta, Indonesia  
imamdigmi@gmail.com

Dr. Bambang Purnomosidi  
Informatics Engineering  
STMIK AKAKOM  
Yogyakarta, Indonesia  
bpdp@akakom.ac.id

M. Agung Nugroho  
Informatics Engineering  
STMIK AKAKOM  
Yogyakarta, Indonesia  
nugroho.agung.m@gmail.com

Akhmad Dahlan  
Informatics Management  
Universitas Amikom  
Yogyakarta, Indonesia  
alland@amikom.ac.id

**Abstract**—Detection of vehicle license plates (VLP) become a challenging issue because of the variability in conditions and the types of the license plate. There are several solutions use stationary cameras with a limited angle and specific resolution and also for a specific license plate type. Unfortunately, license plate detection is a challenging issue when vehicles in the open environments and images are taken from a particular range by low cost cameras. Vehicle license plates can be detected with computer vision technology in real-time video conditions. In this paper, we propose vehicle license plates detection system by using convolutional neural network (CNN) with tensorflow. The research was conduct in three step process such as data pre-processing, training/testing process, and interpretation result. Using CNNs algorithm in tensorflow with 25,000 steps and 8 batches on the training process can produce a training model of vehicle license plates detection with high accuracy around 70-99%.

**Keywords**—Convolutional neural network, object detection, license plate, tensorflow

## I. INTRODUCTION

In Indonesia, vehicle license plates (VLP) record by using conventional methods where security guard capture the license plates in devices and manually compare the record with license plates on the spot. Detection of license plates is a challenging issue because of the variability in conditions and the types of the license plate. There are several solutions use stationary cameras with a limited angle and specific resolution and also for a specific license plate type. Additionally, this system only can record the vehicle license plates when they stopped in front of the camera. Unfortunately, license plate detection is a challenging issue when vehicles in the open environments and images are taken from a particular range by low cost cameras. In the open environments, appear difficulties contain disrupted backgrounds, motion blur, appropriate camera setting and location, shadows, and variety in weather and lighting [1]. Vehicle license plates can be detected with computer vision technology in real-time video conditions.

In computer vision, there are some problems such as image classification, object detection, and neural style transfer. Object detection with neural network is still

developing as a technology that duplicates human capability in recognizing thing in form of a picture. One of the subtypes of this neural network which handle a problem on computer vision is a convolutional neural network. This network is trained to find a feature like an edge, corner, and color difference as well as to combine it into more complex shape [2].

Object detection is a subfield of computer vision that uses as a base of machine learning. Detection of objects with convolution is still under development, although this method is more effective.

Based on the availability of datasets and free pre-operation networks Convolutional Neural Network makes it possible [3] to create implementations functional deep neural network without using special hardware.

The plate detection model usually uses several features of artificial images that capture certain morphological, color, or texture attributes [4]. These features can be sensitive to cause interference with the image, and can cause many errors in complex backgrounds or when getting lighting under different conditions. In this research, we tried to overcome this problem by utilizing Convolutional Neural Network (CNN) capabilities.

## II. RELATED WORKS

There are several researches about vehicle plate detection. Anagnostopoulos et al. [1] conducted a survey regarding the development of this technology until 2013. The use of algorithms to identify text in urban areas can also be used in case of license plate detection, even though this algorithm is actually used to solve more general problems [5]. Vehicle license plates have a standard design which has a certain size, shape and color model. In addition, characters have contrasting colors compared to the background color. Special features in images can be explored by identification algorithms that can identify features such as texture, color, shape, and geometry.

Our work is certainly not the first that adopts Convolutional Neural Network on the vehicle license plates. There are some researchers propose a solution to detect vehicle license plates uses convolutional neural networks (CNNs). CNNs are connectionist system type that inspired

from the animal visual cortex mechanism [6]. A CNN-based approach to identify the license plates by using a new output function that allows to calculate the location of the detected plate by adopting the combined results which obtain from the sparse of overlapping regions, and reduce the computation time [7]. The method uses a CNN with a single convolutional layer implemented in a sliding window fashion over small image sub-regions [8]. The CNN was trained to categorize characters as text/non-text, with the outputs unified and then examine based on the form of ratio and size.

Character recognition uses Convolutional Neural Network trained in a large number of group data can improve success rate better than matching technique to recognize the character. Using CNN's nine-layer network, and the transportation vehicle dataset from multiple perspectives may become an outcome which is indicated that a nine-layer network can achieve the solution on vehicle identification. A deep learning framework such as caffe might improve the accuracy of vehicle identification about 92.2% [9].

### III. RESEARCH METHOD

This research apply the Neural Convolutional Network (CNN) algorithm. Convolutional Neural Network is the development of Multilayer Perceptron (MLP) which is designed to prepare two-dimensional data. CNN is the Deep Neural Network's type where data is distributed to the network that is two-dimensional data, so linear operations and weighting parameters on Convolutional Neural Network (CNN) is different. The operation of Neural Convolutional Network (CNN) linear is using convolution operations, while the weights no longer in one-dimension, but a four-dimensional shape which convolution kernel collection.

This research was conducted in three steps of the process that illustrates in Figure 1 such as pre-processing data, training / testing process, and interpretation results.

#### A. Data pre-processing

In data pre-processing, there are several step such as collecting data, labeling data, converting datasets to various type, and labeling map. The pre-processing process uses a mean-substraction process where the input patches (both are owned by the train and test sets) are zero centered by subtracting the mean computed on the entire training set. Given  $N$  training images, each denoted by  $x \in R^{h \times \omega \times c}$ , we can denote the mean-substraction step as follows[10]:

$$x = x - \hat{x}, \text{ where } \hat{x} = \frac{1}{N} \sum_{i=1}^N X_i. \quad (1)$$

The type of data is image data which is taken using a camera. The data taken is image data from the license plate. First, the data collection that use about 480 images from various types of license plate. After that, the images should be labeled in the process image labeling which is the initial stage where the input dataset is labeled or identifier (sign) for the purpose of storing image information then stored in an XML file with PASCAL VOC format. Then, the datasets will convert from XML to TFRecord which use as datasets training in tensorflow.

#### B. Training Process

The training stage is the main stage where neural networks are trained to learn patterns that are expected to produce identification of the object.. The result is expected high in accuracy. Tensorflow is the system that uses for training process. During the training process, tensorflow records that processing steps done after 25,000 steps. It is provide the information of loss which generated in each step and each step is completed with an average time is 1.5 seconds step.

In the first stage of convolution input is to read the value of the pixel's weight then multiply with the kernel value in a 3x3 matrix size and it shifts 1 column to the last of a 3x3 matrix to decide the maximum value.

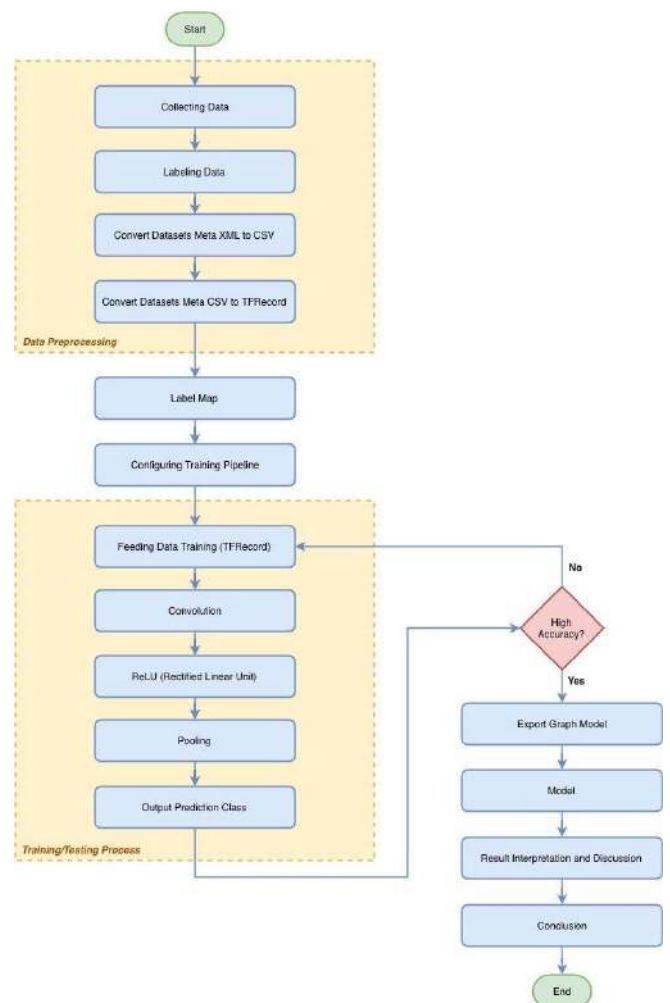


Fig. 1. Research Process

#### 1) Convolution layer

Convolutional layers are layers that carry out the same convolution process as the convolution process in image processing algorithms. Where  $I_i$  is the input image and  $h$  is the kernel convolution, the output image for  $I_o$  convolution process can be written as follows [11]:

$$I_o[m, n] = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} I_i[i, j], h[m, n] \quad (2)$$

with  $[m, n]$  is pixel value at coordinate  $(m, n)$ . The process of training in CNN will learn  $h$ , as kernel, also known as convolutional layer parameters.

### 2) Rectified Linear Unit

The activation function preference in convolutional layer could offer high outcome for the networks. In this research, we use *ReLU* (Rectified Linear Unit) as activation function for all layers.

$$f(x) = \max(x, 0) \quad (3)$$

Where  $x$  is the input neuron,  $0$  in the formula *ReLU* is a linear corrected unit, if the input is less than  $0$ . So, if the input is greater than  $0$ , the output is the same as the input. When it is get a positive input, the derivative is only  $1$ , in other words, activation only thresholds the values.

If the output produces a low level of accuracy, it will be going to re-training process. if the output produces high accuracy, it will continue to the next step that is testing data in the program. This training uses a small amount of data than before. After that, the testing process is beginning as convolution step and ending with the predicted output class with a picture of the detection of the vehicle license plates.

### 3) Pooling layers

The aim of the pooling layers is to reduce feature maps' resolution. Max pooling is intended to teach convolutional neural networks to recognize all the different image poses. To be able to do this, the network requires a property known as spatial variance. This property can make the network have the ability to recognize objects in the image so there is no need to recognize differences in image texture, distance when the image is taken, angle, or otherwise. The objective of max pooling is allowing the convolutional neural network to discover the VLP images when presented with the image in any variety. The max pooling function:

$$\alpha_y = \max_{N \times N} (\alpha_x^{n \times n} u(n, n)) \quad (4)$$

The window function  $u(a, b)$  is for the input patch, and calculate the maximum in the proximity.

After completing the previous process, in this process it should have a pooled feature map with multi-dimensional tensor. Then the feature map is downsized to one-dimensional tensor known as the flattening process. The goal is to make it easier to enter data on the artificial neural network.

### 4) Output prediction class

At this stage, an accuracy level is needed to recognize objects to be classes. This process will create a fully connected layer with weight ( $\alpha$ ) and biases ( $\beta$ ) declarations as random normal distributions. The fully connection process works by using neurons in the fully-connected layer to detect some features on the object, then become a value. This value transforms into VLP classes to identify whether the object VLP or not. All inputs will use the standard operation:

$$z = \alpha x + \beta \quad (5)$$

The level of accuracy in determining the possibility of VLP classes is use the softmax function where converting K-dimensional vector  $x$  has original values to original values in the form of a vector with a range  $(0,1)$ , sum is 1.

$$f(\alpha)_j = \frac{e^{\alpha_j}}{\sum_{n=1}^N e^{\alpha_n}} \quad (6)$$

The propose of softmax function in convolutional neural network to determine the probability degree of each classes.

## IV. RESULT AND DISCUSSION

### A. Architecture and Algorithm

#### 1) Convolutional Layer

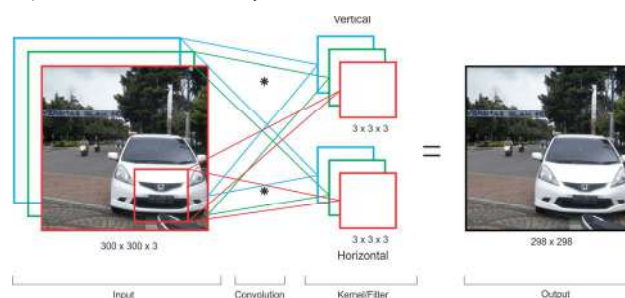


Fig. 2. Convolutional Layer

Convolution is a way to combine two series of numbers that produce the third series of numbers. In this research, two numbers of numbers are in the input and kernel or filter while the series of numbers that all three are output. Inputs and kernels or filters both have a series of numbers in the form of a matrix. In series input the numbers are obtained based on the existing color level of each pixel while the kernel or the series number filter is adjusted by the researchers needs. There are several types of kernels or filters commonly used, such as identity operations, edge detection, sharpen, box blur, Gaussian blur and so forth.

In Figure 2 on the input section, the image has a height of 300 and a width of 300. In Figure 2 also has depth, the depth of 3 corresponds to the basic color channel is red, green and blue. The convolution layer is established by running a filter on it. Filter is another block or cube with a smaller height and width but it has a similarity on the depth that is swept out on the base of the image or the original image. Filters are used to determine what patterns will be detected which are subsequently converted or multiplied by the values in the input matrix, the values in each column and the row in the matrix depend heavily on the type of pattern to be detected.

To be able to better understand the workings of the convolution process, the researcher will use the sample of numbers on the input due to the limitations of writing with the size 300x300 then the researcher uses the sample series of numbers on input with size 6x6 and use kernel or filter for vertical edge detection operation with size 3x3.

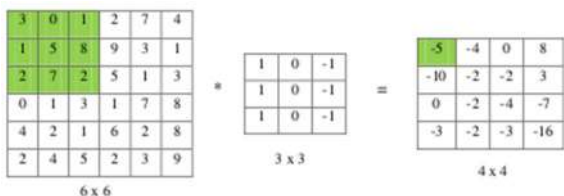


Fig. 3. Convolutional Layer

Calculations on the convolution process with 3x3 filter size is started from the top left corner and then sliding window to the bottom left corner. The filter used by the researcher in Figure 3 is no more than a set of weights, ie  $3 \times 3 = 9 + 1$  bias = 10 weights. In each position, the number of calculated pixels uses the formula and then the new value is obtained. A single filter will generate a 4x4x1 size volume as shown in Figure 2.

The image size generated from the convolution process progressively shrinks sequentially or continuously, it is not so good because its size will be very small. In addition, it will limit the use of large size filters as it will result in faster size reductions. To prevent this problem, researchers generally only use step 1.

2) Activation Layer

This stage is a value calculation with Activation Function used to find non-linear value at convolution value. The Activation Function uses *ReLU* to identification of VLP. *ReLU* formula used is with an input neuron or node. The number 0 in the *ReLU* formula is the linear unit that is corrected if the input is less than 0. That is, if the input is greater than 0, the output is equal to the input. To better understand *ReLU* can be illustrated with the picture below:

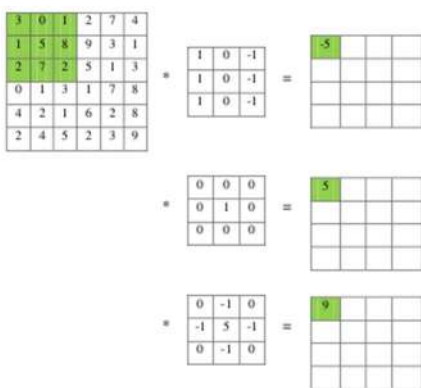


Fig. 4. Activation Layer

The first kernel  $(3 \times 1) + (1 \times 1) + (0 \times 5) + (8 \times (-1)) + (2 \times 1) + (7 \times 0) + (2 \times (-1)) = -5$ , in the second kernel  $(3 \times 0) + (1 \times 0) + (5 \times 1) + (2 \times 0) + (2 \times 0) = 5$  and on the third kernel  $(3 \times 0) + (0 \times (-1)) + (1 \times 0) + (1 \times (-1)) + (5 \times 5) + (8 \times (-1)) + (2 \times 0) + (7 \times (-1)) + (2 \times 0) = 9$ . Since the value the highest obtained using the three kernels, then used is the number 9 on the next output, as well as with other columns and rows.

3) Pooling Layer

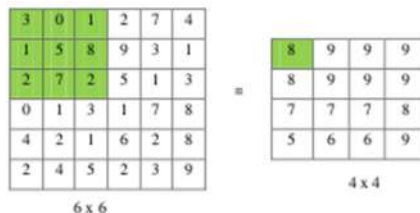


Fig. 5. Pooling Layer

Pooling layer in Figure 4 is a pooling layer using Max Pooling method. Pooling layer itself is useful for reducing image size. As seen in figure, there is a layer with size  $6 \times 6$ , if the researcher uses 3x3 filter with stride 1 then obtained the result of Max pooling with size 4x4.

4) Fully Connected Layer

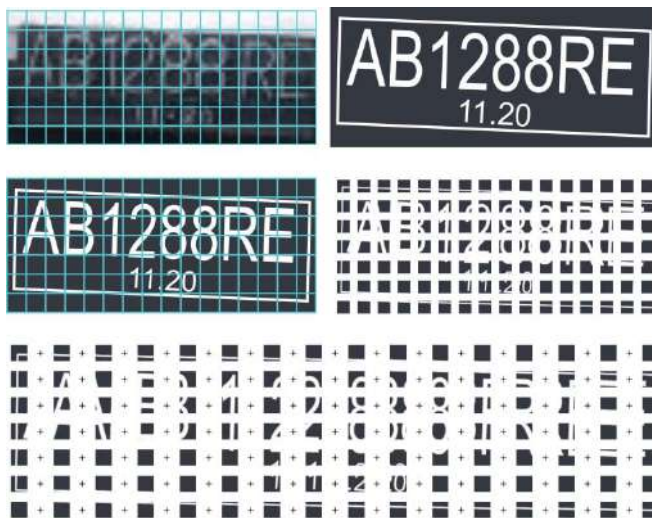


Fig. 6. Fully Connected Layer

In the last process of convolution layer and pooling layer, the network uses a fully connected layer where each pixel is considered as an individual neuron and identical with a regular neural network. The last fully connected layer will have contents as many neurons as the number of classes to predict. In this fully connected layer process the unification of each pixel is considered a Vehicle Number plates which will then become an output consisting of one class label that is a VLP so that the last connected layer has only 1 neuron.

5) Classification

This stage is a classification step on any part of any pixel that has a VLP Mark pattern. In this study researchers not only use images on streaming media consisting of 1 piece of VLP Marks or in other words the researchers used the image on the vehicle that lined up so much that there is a classification process to determine whether the image is a VLP Mark or not.

6) Detection Output

Detection Output is the end result of the VLP detection. In accordance with the output design, researchers get detection output with a description on the green box line and description of the image label with a level of accuracy between 1-99%

## B. Results

The test results of a model that has been in training will be ready to be used to detect the presence of Vehicle License Plate (VLP) on an image frame with the green box marked with the following percentage of accuracy. The following is the result of VLP detection using webcam:



Fig. 7. Result of webcam detection

Based on Figure 7, the results of 99% on the first Vehicle License Plate and 96% on the second Vehicle License Plates. The following is the result of Vehicle License Plates Identification using video:



Fig. 8. Result of real-time video detection

Based on Figure 8 a 98% yield is obtained on the first Sign of Vehicle Number and 97% on the second Vehicle Number Sign. The result of detection of vehicle number marks on webcam or video produces high accuracy value.

## V. CONCLUSION


In this paper, we propose the vehicle license plates detection by using the convolutional neural network with tensorflow. Our model working properly captures VLP with webcam and video in very high accuracy, around 96%-98%. Network architecture used to detect VLP in the real-time video is divided into several layers of the input layer, convolution layer, layer activation, layer pooling, and fully connected layer. In tensorflow, by using 25,000 steps and 8 batches on the training process, will produce a model of VLP detection training in high accuracy. A large number of data sets and the various viewpoints in the image in the training process have an effect on the speed and accuracy of the model results.

## REFERENCES

- [1] Y. Zheng, L. Capra, O. Wolfson, and H. Yang, "Urban computing: concepts, methodologies, and applications," *ACM Trans. Intell. Syst. Technol. TIST*, vol. 5, no. 3, p. 38, 2014.
- [2] C. C. Hau, *Handbook of pattern recognition and computer vision*. World Scientific, 2015.

- [3] H. Li and C. Shen, "Reading car license plates using deep convolutional neural networks and LSTMs," *ArXiv Prepr. ArXiv160105610*, 2016.
- [4] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, "Automatic license plate recognition (ALPR): A state-of-the-art review," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 2, pp. 311–325, 2013.
- [5] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 7, pp. 1480–1500, 2015.
- [6] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
- [7] B. Nassu, F. D. Kurpiel, R. Minetto, and B. T. Nassu, "Convolutional Neural Networks for License Plate Detection in Images," 2017.
- [8] Y.-N. Chen, C.-C. Han, C.-T. Wang, B.-S. Jeng, and K.-C. Fan, "The application of a convolution neural network on face and license plate detection," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, 2006, vol. 3, pp. 552–555.
- [9] X. Luo, R. Shen, J. Hu, J. Deng, L. Hu, and Q. Guan, "A deep convolution neural network model for vehicle recognition and face recognition," *Procedia Comput. Sci.*, vol. 107, pp. 715–720, 2017.
- [10] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, "A guide to convolutional neural networks for computer vision," *Synth. Lect. Comput. Vis.*, vol. 8, no. 1, pp. 1–207, 2018.
- [11] R. F. Rachmadi and I. Purnama, "Vehicle color recognition using convolutional neural network," *ArXiv Prepr. ArXiv151007391*, 2015.



**BIC**  **ME** **BORNEO**  
**INTERNATIONAL**  
**CONFERENCE**  
**ON APPLIED MATHEMATICS & ENGINEERING**

[bicame.itk.ac.id](http://bicame.itk.ac.id) [irsi.or.id](http://irsi.or.id)