Research and Comparison of Fire Detection Models Based on Deep Learning and Mixed Dataset

Junkai Lu

School of Computer Science and Engineering, Xi 'an Polytechnic University, Xi 'an, China

pandajunkai@163.com

Abstract. The traditional method of fire detection is based on the data collected by sensors to detect the changes in environmental factors such as temperature and smoke to judge whether a fire has occurred, which has limitations. In recent years, the deep learning model has become more mature, and the convolution neural network (CNN) has the ability to automatically extract image features with greater advantages than traditional methods. Therefore, this paper studies fire detection based on the deep learning model and compares three models. The specific research is as follows: (1) Collecting datasets and architectural pictures through the network and local agents to make the datasets more versatile with the increasing diversity of data; (2) Three models are selected for comparison, including CNN model, VGG16 model and Mobile Net-V2 model. According to the experiment, the deep learning model can effectively identify and detect the fire area. Among the three selected models, the correct rate of Mobile Net-V2 reaches 99.01%. Compared with the other two models, the convolution neural network has poor accuracy. Although the accuracy of the VGG16 model is close to that of Mobile Net-V2, it consumes more computing resources than Mobile Net-V2. In other words, there are too many internal parameters in the model, so Mobile Net-V2 performs best. Based on the research, it is confirmed that three fire detection models involved in this experiment can identify and monitor the fire area efficiently, which has a great guiding significance for the future application of fire prevention..

Keywords: Deep Learning; Fire Detection; CNN; VGG16; Mobile Net-V2.

1. Introduction

As the most common natural disaster, fire endangers the life safety of the public and the development of society. How to improve the accuracy of fire detection and early warning to discover and deal with fire timely has vital research significance. Traditional methods of fire detection usually use sensor detection technology or manual methods for early warning, such as temperature sensor or smoke sensor to monitor the temperature and concentration of smoke particles in the air; Site patrol [1], watchtower detection [2] and holographic image technology [3] are used to prevent forest fires, but these detection technologies can not effectively predict fires because of economic cost, detection performance, operability and other reasons. Meanwhile, these traditional detection methods using sensors have great limitations. For example, sensors can only be used in closed places. When facing open scenes, the sensitivity of sensors decreases, so it is difficult to make timely responses [4].

In recent years, with the rapid development of artificial intelligence technology in the computer field, deep learning has been widely used in computer vision and achieved great success, especially in many fields such as target detection, image classification and face recognition, with traditional methods gradually replaced by deep learning methods [5]. Deep learning, as one of the critical branches of machine learning in artificial intelligence, builds a multi-scale network structure by combining or superimposing a series of simple but nonlinear modules. Each module can transform a scale representation into a more abstract one to process a large amount of data, which is characterized by automatically extracting, learning and reflecting the essence of this group of data [6].

With the explosion of data, the development of big data and the improvement of computer hardware equipment, deep learning has made great achievements in natural language processing [7], computer vision [8] and other fields, which has entered various industries in human society to solve some complicated problems. Hence, using deep learning technology instead of traditional methods of fire detection can conduct efficient and timely early warning and reduce the harm caused by fire to a greater extent. In this study, three deep learning models are used to complete the fire detection on the self-built dataset for comparison.

This paper focuses on: 1. Building a dataset for mixing with network data set after collecting local data; 2. Checking the mixed dataset with the classical model.

2. Dataset

2.1 Network Dataset

The dataset in this paper is formed by combining the network dataset and local dataset. Adopting FIRE Dataset in Kaggle, network dataset is binary classified, which contains images of fire and non-fire. The data in the dataset are stored in two folders. The folder fire_images contains 755 outdoor fire images, some of which contain smoke and flame. The folder non_fire_images contains 244 natural images, such as forests, roads, trees, grass, people, rivers, fog forests, lakes, animals, waterfalls, etc. A total of 999 images are partially shown in Figure 1.

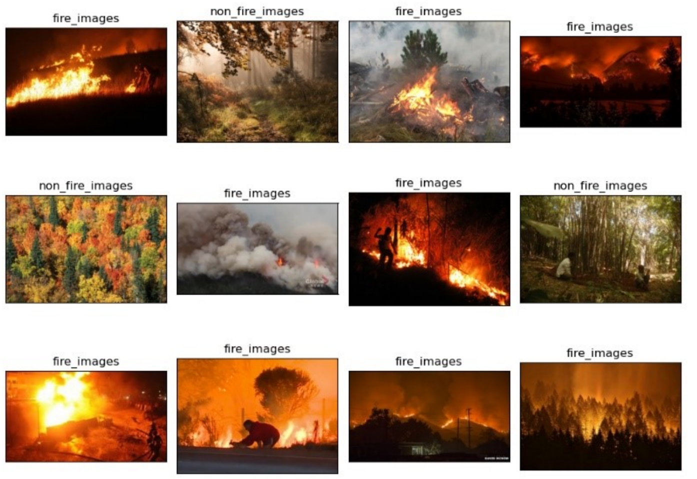


Figure 1 Partial Data

However, the data is inclined, which means that the two types do not have the same number of samples and the number of fire images is far supernatural. Hence, before starting this experiment, the natural map and a small number of fire maps were supplemented, which reduced the data inclination and enhanced the applicability of the dataset.

2.2 Self-Built Dataset

The locally collected dataset refers to the format of the FIRE Dataset on the official website of Kaggle. In other words, the file name is in the form of a picture number and label, with the buildings, shopping malls, parks and landscapes in the author's area collected, and some fire maps obtained through the news. Finally, the data in the dataset is divided into two folders, fire_images and non_fire_images, with some data shown in Figures 2 and 3.

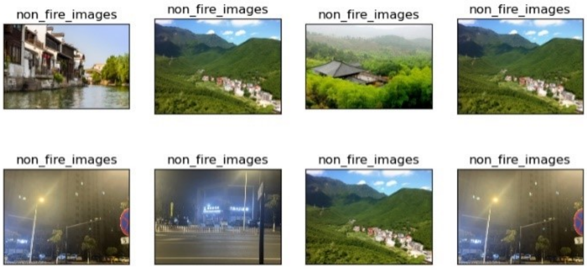


Figure 2 No Fire Occurred

Finally, two datasets are mixed and labeled again. The data format adopted by FIRE Dataset is continued with the picture renamed in the form of “number and label”.

3. Deep Learning Model

3.1 Convolution Neural Network

Convolution neural network (CNN) [9] is a deep learning model, which has been widely used in speech processing [10], computer vision [11], face recognition [12] and other different fields. Compared with the traditional neural network, CNN is similar to a perceptron with a hierarchical structure. Thanks to the convolution kernel in its hidden layer, it has the following characteristics. Parameter sharing and inter-layer connection enable the CNN model to learn multiple features in data with less computation. The input layer, activation layer, pooling layer, convolution layer, connection layer and output layer constitute the basic CNN model [13].

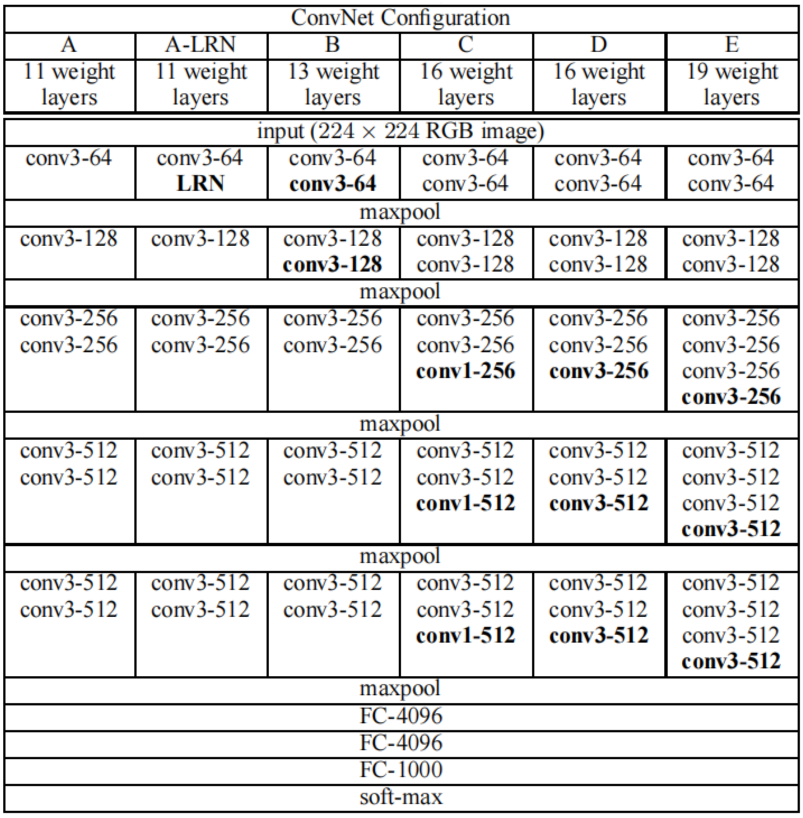
Through the learning process of CNN, the network evolved to a higher scale with more comprehensive and more abstract features, thus being able to recognize images more accurately. Moreover, it is an automatic behavior to extract useful features from data by CNN, so it can optimize the performance of the model. Each layer of CNN has its corresponding functions. By the combination, CNN can automatically learn more complex features from image data, thus realizing more accurate recognition. Table 1 shows the specific structure of the CNN model used in this study.

Table 1 Structure of CNN Layers

Network Name	Output Format	Parametric Quantity
Conv2d	(195, 195, 128)	1664
Conv2d_1	(194, 194, 64)	32832
Max_pooling2d	(97, 97, 64)	0
Conv2d_2	(96, 96, 32)	8224
Max_pooling2d_1	(48, 48, 32)	0
Flatten	(73728)	0
Dense	(128)	9437312
Dense_1	(1)	129
Total Parameters Volume	9480161 (36.16 MB)	

3.2 VGG16

VGG model is a collection of models, which is an important work made by the visual geometry team of Oxford University in the ImageNet Large-scale Visual Recognition Challenge in 2014. In this collection, there are six sub-models of A-E as shown in Figure 3.



ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 3 Six Types of VGG Models

VGG16 is one of the models, whose key innovation is to verify the effectiveness of using extremely small convolution kernels such as 3×3 convolution kernels, so as to build deep architecture and then successfully expand the network depth from 16 layers to 19 layers by stacking multiple convolution layers [14]. One of the most important improvements of this network is that the large-size (large-parameter) convolution kernels are stacked continuously through several small-size (small-parameter) convolution kernels to replace the original large-size convolution kernels. For example, if you want to deeply expand a 5×5 convolution kernel, two 3×3 convolution kernels can be continuously stacked for replacement. To extend a 7×7 convolution kernel deeply, three 3×3 convolution kernels can be continuously stacked for replacement.

Its advantages are as follows. The receptive field is used as the measurement standard. In CNN, the receptive field refers to the area size of the input layer covered by an element in the output result of a certain layer. Based on keeping the same receptive field, by superimposing multiple small convolution layers, the depth of the network can be easily improved, and the ability to extract data features of the model can be increased, and fewer parameters are needed in the network. According to the receptive field formula (1), by stacking two 3×3 convolution kernels with a step size of 1, a 5×5 convolution kernel with a receptive field can be equivalently replaced. Assuming that the number of channels is 1, the number of parameters involved in a 5×5 convolution kernel is 5×5×1. In contrast, the number of parameters involved in two 3×3 convolution kernels is 2×3×3×1. Apparently, using the 3×3 convolution kernel is more advantageous.

$$F(i) = (F(i+1) - 1) \times Stride + K_{size} \tag{1}$$

In this study, the Class D model is used, namely VGG16 [15] as shown in Figure 4.

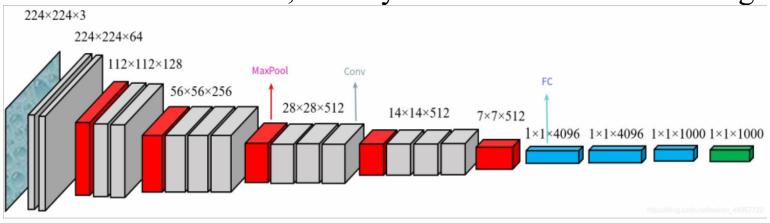


Figure 4 Structure of VGG16

Experiments were conducted. In the end, the Flatten layer, Dropout layer and full connection layer were added to form the final model network as shown in Table 2.

Table 2 Structure of Final Network

Network Name	Output Format	Parametric Quantity
VGG16	(7, 7, 512)	14714688
Flatten	(25088)	0
Dropout	(25088)	0
Dense	(1)	25080
Total Parameters Volume	14739777 (56.23 MB)	

This study adopts the transfer learning method. Firstly, the model weight parameters pre-trained by VGG16 on ImageNet are loaded into this training model. In other words, the knowledge learned from ImageNet is introduced into the current model, and then the model is trained. Feature extraction and knowledge acquisition are conducted on the dataset used in this experiment. Finally, the learned knowledge is processed through the full connection layer to complete the fire detection task.

3.3 Mobile Net-V2

With the emerging neural network, many fields have been completely changed, and the original challenging task of image recognition has made great achievements with amazing accuracy. However, there is a price behind this amazing accuracy. Advanced neural networks often consume a lot of computing resources. Mobile Net-V2 was born under this background, which is a kind of neural network architecture customized for resource-limited environments. The main contribution of the Mobile Net-V2 team is to propose a new neural network module: inverted residuals with linear bottlenecks, namely inverted residuals and linear bottlenecks [16].

As for inverted residuals [17], in the previous practice of residuals block [18], the input image will first pass through a convolution layer to "compress" the number of channels in the feature map, and then the obtained results will pass through a 3×3 convolution calculation layer to extract the features of the input results. The obtained results will be "expanded" at the end by applying a 1×1 convolution calculation layer, so as to increase the number of channels. In other words, it is "compressed" first and then "expanded". Inverted residuals, on the other hand, are the opposite of the residuals block, which is "expanded" first and then "compressed" as shown in Table 3.

Table 3 Structure of Inverted Residuals

Input	Operation	Output
h×w×k	1×1 conv2d, ReLU6	h×w×tk
h×w×tk	3×3 dwise s=s, ReLU6	$\frac{h}{s}\times\frac{w}{s}\times$tk
$\frac{h}{s}\times\frac{w}{s}\times$tk	Linear 1×1 conv2d	$\frac{h}{s}\times\frac{w}{s}\times$k'

Input represents the tensor size of the input image; Operator represents the operation; Output represents the output result. In this way, more information of the original image can be obtained during "expansion", so that more features can be extracted in the second step, thus improving the accuracy.

As for linear bottlenecks [19], to avoid bad factors on image features, Mobile Net-V2 changed the Relu layer that should pass through after convolution to linear layer in the last time of the third convolution operation of residual block to ensure that features are not destroyed.

In this experiment, the network structure of Mobile Net-V2 will be adopted. At the end, two full connection layers with Dropout and a full connection layer with only one neuron will be added to complete the fire detection. The network structure added in this experiment is shown in Table 4.

Table 4 Network Structure with Mobile Net-V2 in the Experiment

Network Name	Output Format	Parametric Quantity
Dropout	(256)	0
Dense	(256)	65792

Dropout_1	(256)	0
Dense_1	(2)	514
Total Data Volume	66360 (259.21 KB)	

4. Comparison of Prediction Ability of Different Models

4.1 Experimental Environment

The experimental environment is Windows 11 and TensorFlow. Meanwhile, the 11th Intel(R)Core(TM)i7-11800H CPU 2.30 GHz processor is used as hardware with 16GB memory and a GeForce GTX 3060 Ti graphics card.

4.2 Loss Comparison

In this experiment, three models are trained on the self-built dataset and the loss changes in the process are recorded. The loss of VGG16 is shown in Figure 5.

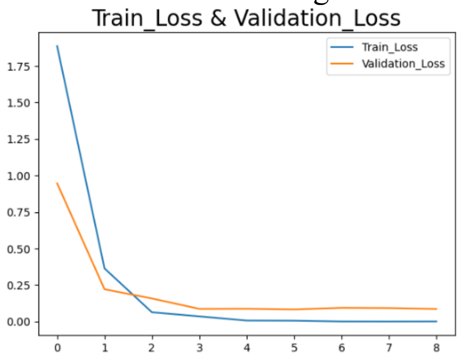


Figure 5 Loss Diagram of VGG16

The vertical axis is Loss and the horizontal axis is the number of epoch iterations. During the training, if the Loss is taken without improvement, the training will be stopped after waiting for three rounds, so that how many epochs the model needs to train to converge can be clearly seen from the figure, which is convenient for observation with the of Loss diagram shown in Figure 5-7.

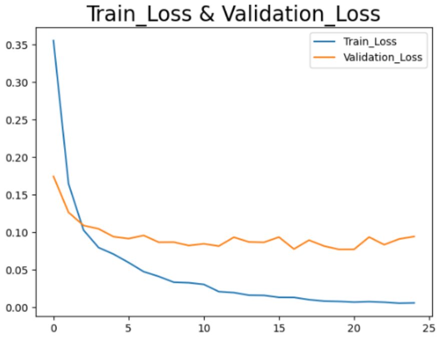


Figure 6 Loss Diagram for Mobile Net-V2

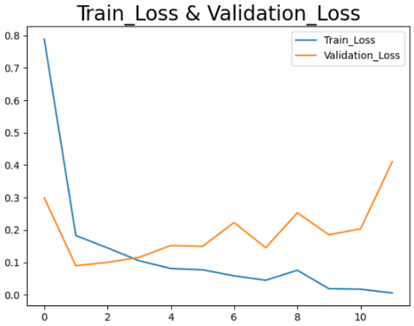


Figure 7 Loss Diagram of CNN

According to Figure 8, the CNN model performs poorly and is unstable, where the loss value even rises linearly in the test set. By comparing Figure 6 with Figure 7, it can be found that although VGG16 converges faster than Mobile Net-V2, that is, VGG16 converges around the 8th epoch, the loss value of Mobile Net is generally smaller than VGG16 with smoother training loss.

4.3 Accuracy Comparison

To further compare the detection ability of the three models, this study recorded their precision, recall and accuracy under fire conditions. The precision, recall, accuracy and average accuracy of the three models under non-fire conditions are shown in Table 5-7.

Table 5 Precision, Recall and Accuracy of Three Models on Fire Images

	precision	recall	accuracy
VGG16	0.99	0.99	0.99
Mobile Net-V2	0.99	1.00	0.99
Ordinary CNN	0.99	0.97	0.98

Table 6 Precision, Recall and Accuracy of Three Models on Non-Fire Images

	precision	recall	accuracy
VGG16	0.98	0.98	0.98
Mobile Net-V2	1.00	0.96	0.98
Ordinary CNN	0.92	0.96	0.94

Table 7 Average Accuracy of Three Models

VGG16	Mobile Net-V2	Ordinary CNN
0.99009	0.99014	0.97520

According to Table 7, 0.97520 as the accuracy of CNN is lower than that of the other two models, while VGG16 and Mobile Net-V2 both reach 0.99. According to Table 5 and Table 6, although the recall of Mobile Net-V2 on the non-fire image is only 0.96, it performs well on the other two indicators. Hence, Mobile Net-V2 performs better, which is more suitable to use.

5. Conclusions

In view of the cost of traditional methods of fire detection, the deep learning model is chosen to carry out fire monitoring in this paper, so as to explore and compare the three models. Firstly, the dataset is improved, so that the original dataset in the network adds the picture data of the local area, making the data more versatile with more guiding significance for the local fire detection. After the experiment, it was found that the detection effect of Mobile Net-V2 is the best. Although the convergent epoch is relatively backward, the training speed of each epoch is fast. Experimental data show that the best model, namely Mobile Net-V2, can achieve 99% accuracy, 98% recall and 99.014% accuracy in fire detection on the self-built fire dataset. In future research, the following work will be implemented. First of all, the fire can only detected be after the fire occurs at present, which cannot be predicted in advance. Thus, some situations before the fire will be collected and added to the dataset for training. Secondly, how to combine this model with the current basic settings should be considered. For example, implant the model into the surveillance system of the city and pay attention to the fire in the daily detection of traffic safety.

References

[1] Li, H. S. (2016). Analysis of the current situation of the development of China’s forest fire monitoring system. Journal of Green Science and Technology, (12): 188-190.

[2] Liu, Z. L., Li, Y. J. & Yu, S. J. (2016). Lightning protection design of forest fire prevention watchtower. Theoretical Research in Urban Construction, (29): 63-64.

[3] Tang, R. & Zhang, L. L. (2021). Application prospect analysis of holographic image technology in forest fire prevention and control. China Forest Products Industry, 58(12): 97-99.

[4] Wang, L., Zhao, Q. H. & Zhang, X. R. (2024). A multi-scale object detection algorithm for fires. Computer Simulation, 41(01): 271-276+310.

[5] Wang, L. X., Xia, X., Gao, F. et al. (2023). Research progress of deep learning in forest fire detection. China Forest Products Industry, 60(11): 88-92. DOI:10.19531/j.i ssn1001-5299.202311014.

[6] Yann, L. C., Yoshua, B. & Geoffrey, H. (2015). Deep learning. Nature, 521(7553): 436-444.

[7] Luo, X. (2020). A survey of natural language processing based on deep learning. Intelligent Computer and Applications, 10(4):133-137.

[8] He, K., Zhang, X., Ren, S. et al. (2016). Deep residual learning forimage recognition. Las Vegas: Conference on Computer Vision and Pattern Recognition, 18(3):121-127.

[9] Dhillon, A. & Verma, G. K. (2020). Convolutional neural network: a review of models, methodologies and applications to object detection. Progress in Artificial Intelligence, 9(2): 85-112.

[10] Palaz, D., Magimai-Doss, M. & Collobert, R. (2019). End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition. Speech Communication, 108: 15-32.

[11] Fang, W., Love, P. E. D., Luo, H, et al. (2020). Computer vision for behaviour-based safety in construction: A review and future directions. Advanced Engineering Informatics, 43: 100980.

[12] Li, H. C., Deng, Z. Y. & Chiang, H. H. (2020). Lightweight and resource-constrained learning network for face recognition with performance optimization. Sensors, 20(21): 6114.

[13] Wu, L. (2023). Research on slope disease identification based on convolutional neural network. Nanjing: Master's Dissertation of Nanjing University of Posts and Telecommunications. DOI:10.27251/d.cnki.gnjdc.2023.000155.

[14] Qassim, H., Verma, A., & Feinzimer, D. (2018). Compressed residual-VGG16 CNN model for big data places image recognition. In 2018 IEEE 8th annual computing and communication workshop and conference (CCWC): 169-175. IEEE.

[15] Simonyan, K. & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. ICLR.

[16] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks (conference paper). Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition: 4510-4520.

[17] Zhao, W., Wang, Z., Cai, W. et al. (2022). Multiscale inverted residual convolutional neural network for intelligent diagnosis of bearings under variable load condition. Measurement, 188, 110511.

[18] Shafiq, M. & Gu, Z. Q. (2022). Deep residual learning for image recognition: A survey. Applied Sciences, 12(8972): 8972

[19] Pundir, P. S., Porwal, S. K. & Singh, B. P. (2015). A new algorithm for solving linear bottleneck assignment problem. Journal of Institute of Science and Technology, 20(2): 101-102.