Higher Education and Oriental Studies (HEOS) – Vol 2(6): 40 – 49
www.heos.asia/ojs
e-ISSN 2785-9118
http://doi.org/10.54435/heos.v2i6.85

Application of Improved CNN Technology in Medical Imaging Course

QI DONGKAI*
Illinois Institute of Technology
dqi2@hawk.iit.edu

HUANG HUANG
University of Malaya, Malaysia
erichuangemail@163.com

**Corresponding Author*

Date Received: 23 September 2022 Date Accepted: 21 October 2022 Date Published: 30 November 2022

ABSTRACT

The role of artificial intelligence technology in the medical field has become more and more important. As an important part of artificial intelligence technology, the CNN model in deep learning has become more and more widely used in the field of CT image recognition. In the process of recognizing CT images, the previous CNN model will have the consequences of low recognition accuracy and over fitting, resulting in less obvious recognition effect.This paper proposes an improved CT image recognition method based on 3-D CNN, a convolutional neural network for image classification mainly used to solve the correlation information between images and add new dimension information on human cancer tissue. The method saves valuable education resources on Chinese medicine, reduces the repetitive mechanical workload of medical college lecturers, and improves the efficiency of medical students' knowledge acquisition. Experimental results show that the accuracy of medical CT images can be improved by using 3D-CNN method, thus improving the prediction of the location of cancer areas during CT scans.

Keywords: 3-D CNN model; CT image; depth Learning; medicine Education

INTRODUCTION

In the traditional Chinese medical imaging education model, students would usually undergo numerous years of experience and knowledge accumulation. After the introduction of artificial intelligence (AI) technology however, learning time has effectively become shortened. On the one hand, the application of computer vision technology based on CNN algorithm in medical imaging has, indeed, played a role in helping students in medical schools to identify diseased areas. On the other hand, the application of the technology has also allowed lecturers to forego what was before seen as repetitive and tedious work. More importantly, AI has effectively reduced the probability of diagnostic errors. Traditional methods need aruduous requirements for theoretical and practical capacities of medical personnel and provide a lower efficiency. The establishment of accurate and fast auxiliary system therefore effectively saves medical resources and educational resources.

Technically, the manual annotation of lesion features is no longer applicable due to the limitations of traditional manual annotation in distinguishing CT images, along with the current complex information technology and big data framework. Today’s age of big data has seen a trend in the use of deep learning algorithms to calculate and classify complex data structures. Deep learning ‘s rapid development also represents progress in the field of data analysis: the correct use of deep learning network technology help people obtain patterns from a large number of unordered data features undetected by human beings, allowing more accurate information and opening up another direction of exploration in the medical field.

Using convolutional neural network (CNN) to predict the exact location of disease lesions in CT images and detect abnormalities allows for considerable decision support for both lecturers and students in the medical schools, creating extraordinary breakthroughs in clinical diagnosis. Unlike traditional manual detection, computers use deep learning to automatically learn image features, save learning models, and make model-based judgments based on the models. Its learning speed, learning efficiency, and objectivity are leagues ahead of human intervention: When the system environment is stable enough, the speed of machine learning features is 10 million times faster compared to usual human intervention, and the learning model can even be copied for further efficiency (Bellakhdhar, 2013). Likewise, when the number of learning samples are large and accurate enough, the judgment accuracy of machine learning is also great.

Many theoretical researches and practices show that deep learning can be well applied to medical image processing. For example, the methdo shows ideal results in the diagnosis of gastric and skin cancers. If a certain number of samples can be collected and used for deep learning detection, the diagnostic efficiency can be greatly improved.

Traditional 2D convolution extracts features from a frame of the original image or video. However, many instances often have associated information between multiple images or video consecutive frames, which is the background of the proposed 3D-CNN. 3D-CNN is mainly used to solve the association information between images and add new dimension information. Taking CT image diagnosis based on depth learning as an example, this paper constructs an automatic detection system of tumor lesions in CT images based on 3D-CNN algorithm.

BACKGROUND OF THE DEVELOPMENT OF CNN

The Neocognitron model proposed by Japanese scholar Kunihiko Fukushima in 1979 marked the genesis of the research on CNN. After observing the visual cortex of an organism, he designed a similar neural network containing a certain depth structure. The hidden structure of the neural network was composed of alternating simple and complex layers (Bose, 2015). The simple layer was responsible for extracting the image features of the receptive field, and the complex layer was responsible for receiving and feeding back the same features. This conforming neural network has become the prototype of what would later be convolutional neural network .

A relatively well-developed convolutional neural network is the time-delay neural network, proposed by Alexander Waibel in 1987(Chenshuo, 2020). Subsequently, the first two-dimensional neural network was a translation-invariant artificial neural network proposed by Wei Zhang in 1988, which was mainly used to detect medical images. In 1998, the lent-5 model proposed by Yann Lecun resolved the issue of handwritten digit recognition. Later in 2006, its representational learning capability was greatly developed with the introduction of deep learning theory.

CNN's FIELDS OF APPLICATION FIELDS

The application scenarios of convolutional neural networks include machine learning, document analysis, natural language processing, image recognition among others. In deep learning, convolutional neural networks are developed based on back-propagation (BP) neural networks. This allows for the training units of the neural network to be disconnected to the training units of the entire layer, because the training units of the backpropagation are likewise disconnected to the training units of the neural network (Dahl, 2013), ultimately leading to better and faster training results.

In the field of document analysis, because the order of the text hinges on time-dependent and one-dimensional data, we can simplify the trained neural network and let the network extract one-dimensional features. In this case, 1D neural networks are similar to 2D neural networks, hence allowing the transfer of features from 2D convolutional neural networks to 1D convolutional neural networks. This makes it necessary to pool the images in this process to prevent excessive data volume and overfitting (Girshick, 2013).

For text sequences, the convolutional neural network is used to extract local features. Although this allows for extraction of a small segment of features, it is nonetheless primarily limited by the available range. However, the advantage of recurrent neural network is its ability to obtain a long range of relevant information. Combining the advantages of both, a one-dimensional convolutional neural network can extract features this way (Yosinski,2014), allowing both correlation over long ranges and better training speed to reduce the amount of data needed and consider the back-and-forth relationship.

In natural language processing, convolutional neural networks represent the natural language input in a matrix. Becauase it is linear, each row of the matrix can correspond to a word or a character. The width of the filter can be adjusted according to the width of the input matrix, so that the filter can perform feature extraction and convolution operations on the natural language input through a sliding window. In practice, convolutional neural networks are very effective for natural language processing and run very fast compared to other neural network models.

In the field of image recognition, the advantages of convolutional neural networks are evident because they are developed by imitating optical neural networks, allowing them to be directly fed into the original image (Guan, 2010). Convolutional neural networks have numerous advantages because they do not require complex operations such as manual preprocessing of images and extraction of additional features and they also bring image processing as in human intervention of image processing， through the unique fine-grained feature extraction method. The main feature of convolutional neural network however is its high speed. Convolution is a core part of computer images and is implemented in a hardware layer at the GPU level. The representation efficiency of convolutional neural networks is also better, compared to N-grams.

HOW CNN WORKS

A convolutional neural network consists of the following parts: an input layer, a convolutional layer, a relu layer, a pooling layer, a fully connected layer, and an output layer.

There are several feature maps seen in the convolutional layer. Specifically, each feature plane consists of several neurons arranged in a rectangle (Lu, 2019). A neuron is only connected to some neighboring neurons. Neurons in the same feature planeshare weights, thus forming a convolution kernel which is usually initialized in the form of a random decimal matrix. During network training, the convolutional kernel will learn to acquire reasonable weights. This allows for immediate reduction of the connectivity between different layers of the network, thus reducing the risk of overfitting.

Usually, there are two forms of sampling: Average set and maximum set. Subsampling can be thought of as a special convolutional process. Convolution and subsampling greatly simplify the complexity of the model and reduce the model parameters.

Another layer, the Relu layer, is a nonlinear mapping of the output of the convolution layer. Because the computation of the convolution layer is linear, it does not adapt well to the nonlinear case (Sharma,2015), which is where the Relu layer gets involved (sharif Razavian,2014).

The fully connected layer is a traditional multi-layer perceptron: the layer re-fits and reduces the loss of feature information. In the output layer, it uses softmax activation or other activation functions. The pooling layer usually follows the convolution of the excitation function to simplify the output of the convolution layer(Buyya, 2009). The pooling function follows the overall statistical characteristics of the neighborhood of an element as the output of the network at a specific location; this downsampling process reduces the number of parameters. Pooling includes both maximum and means pooling, achieving local translation invariance. This means that when the input has a certain translation, the output will not change after pooling. By introducing pooling in Convolution Neural Network, the feature extraction is unaffected by the change of target location.

Gneerally, the Convolution Neural Network takes the original image as the input and convolutes the feature map of the previous layer with a convolution kernel in the convolution layer. The convolution result is then mapped by the activation function to form the feature map of the next layer (Gu,2020). The process can be expressed by the following formula:

$$S(i,j) = \sum_{m} \sum_{n} I(m,n) * K(i-m, j-n) \tag{4.1}$$

Where i is the input, which is generally a two-dimensional image. K is the convolution kernel. m and n are the sizes of the convolution kernel, then the generated feature image is s (I, J) (Chenshuo,2020). To facilitate calculation In practical application, the formula is often transformed into the following forms:

$$S(i,j) = m \sum I(i-m)(j-n) * K(m,n) \tag{4.2}$$

Pooling units compute the values of local blocks in the feature map while neighboring pooling units read data from a small region by moving one row or column (Chenshuo,2020). This reduces the dimensionality of the feature map and allowing the translational invariance of the data to be maintained to some extent while reducing the number of parameters and the computational effort needed in the network(Gu,2020).

Finally, the convolutional neural network classifies the extracted features using a regression model based on a fully connected layer. The current study uses the 3D convolutional network model as a pre-training model in the learning process.

CNN TRAINING PROCESS

The training process of the CNN can be divided into the following two stages: the forward propagation phase and the backward propagation phase(Gu,2020)， More specifically, the above two stages can be realized through the following steps:

Step 1: The convolutional neural network is initialized by weights.
Step 2: Data is extracted and transmitted by convolutional layers, with transmission results better reflecting the characteristics of the original data.
Step 3: Error between the output data and the original data is calaculated, which then determines whether the error tolerance interval is exceeded. When the error value exceeds the interval range, the error value is reversed back to the model and the deviation values of the fully connected and convolutional layers are calculated and adjusted. This is done until the error value is stabilized within the error tolerance interval.
Step 4: The weights according to the error are updated, thus preparing for the second step.

FIGURE 1. Convolutional neural network training flow chart

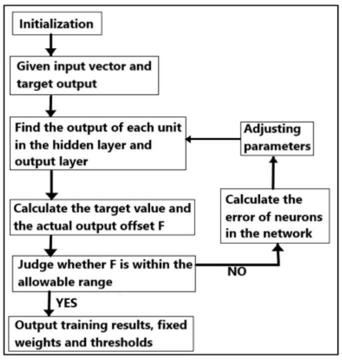
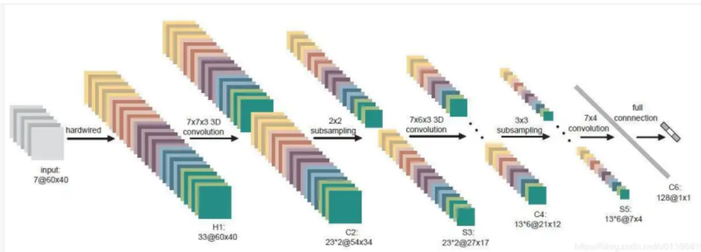


FIGURE 2. 3D-CNN Working Principle



Advantages of convolutional neural networks:

- Input image is well aligned with network topology.
- Shows excellent performance despite using fewer parameters.
- Avoids explicit feature extraction and implicitly learns from the training data.
- Simultaneously performs feature extraction and pattern classification and are generated simultaneously in training.
- Weight sharing reduces the training parameters of the network, reduces the complexity of the network structure, and provides greater applicability.
- Obtains features with good classification without manual selection of features and training weights.
- Can be directly inputted to the network, therein avoiding the complexity of data reconstruction during feature extraction and classification.

EXPERIMENTAL STEPS AND EXPERIMENTAL DATA

The environment of this experiment was as follows:

Locale: Python 3.7
Compiler: Jupiter notebook
Deep learning environment: tensorflow2.4.1
Graphics card (GPU): NVIDIA Geforce GTX 1660

This experiment ran through the Anaconda 3-4.4.1 system and used TensorFlow as the development environment and GPU as the carrier for model training.The advantage of using GPU is its ability to reduce the training time of the model. TensorFlow it is an open source software library that uses data flow graphs for numerical computation, and is Google's second-generation machine learning system that includes extended support for deep learning which supports numerical computation on GPUs and CPUs. TensorFlow is widely used due to its high efficiency and wide range of applications.

Traditional use cases for CNNs are RGB images (3 channels). 3D CNNs take 3D volumes or frame sequences (e.g. slices from CT scans) as input and extract features from them. While a traditional CNN extracts the representation of a single image and places it in a vector state (latent space), a 3D CNN extracts the representation of a set of images, which is required to make predictions based on volumetric data (e.g. CT scans). 3D CNNs consider the temporal dimension (e.g. 3D context). This method of learning the representation from volumetric data helps to find the correct label, which is achieved through 3D convolution. The steps of the experimental setup include the following:

DOWNLOAD NSCLC RADIOMICS GENOMIC DATASET

Since training 3D convolutional neural network is time consuming, a subset of the NSCLC-Radiomics-Genomics dataset was used which consisted of CT scans with gene expression and relevant clinical data.

Simultaneously, to simplify the experimental results, the dichotomy was used to judge the location of cancer in the clinical data. Specifically, the experimental results were expressed as either left or right, and the corresponding probability were given.

LOADING DATA

The files were provided in Nifti format (with the .nii extension). The nibabel package was used to read these scans. For future data processing, the original image was adjusted by 90 degrees and the three indices of width, height, and depth were adjusted. Then, the path of CT scan was read from the class directory, with the CT scan and its shape visualized. Because a CT scan has many slices, in the loaded data, the adjacent slices therefore show a gradual change.

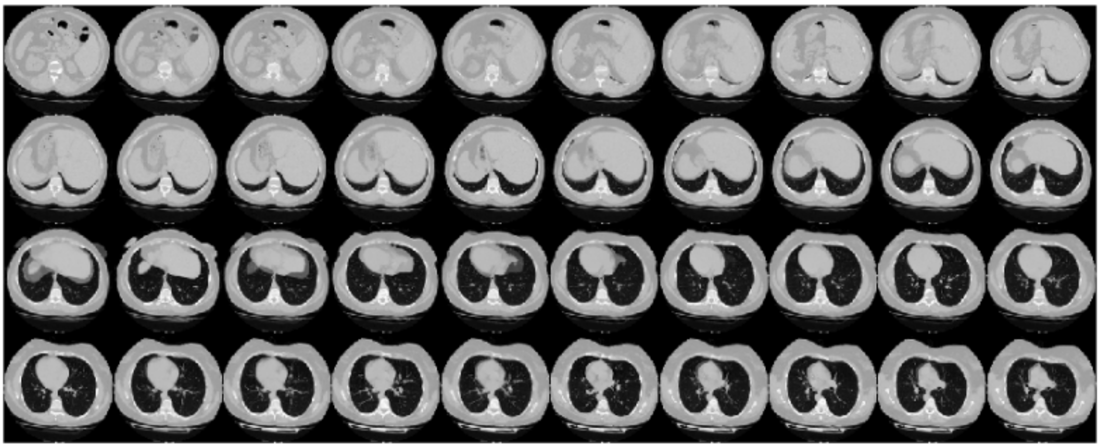
BUILD TRAINING AND TEST DATASETS

Several auxiliary functions were defined to process the data. These auxiliary functions can read, scan, and assign labels from the class directory at this stage. Finally, the data set was divided into either training or testing subsets. In this experiment, the number of samples in training and experiment is 28 and 12, respectively.

PRE-PROCESSING AND DATA EXPANSION

CT scans store raw voxel intensity in Hounsfield units (HU), ranging from -1024 to above 2000 in this dataset. Above 400 were bones with different radiointensity, which is used as a higher bound. A threshold between -1000 and 400 is commonly used to normalize CT scans. The CT scans are also augmented by rotating and blurring. There are different kinds of preprocessing and augmentation techniques widely used; this example only shows a few simple ones for initial discussion. In this experiment, when defining the training and test data loader, the training data was transferred through the enhancement function of random rotation or fuzzy volume, which is then finally re-scaled to a value between 0 and 1.

FIGURE 3. Montage Visualization of CT Scanning Slices



3D-CNN MODEL

Blocks were defined to allow easier comprehension of the model. Because this is a 3D-CNN, 3D convolutions are used. The following is the data model of 3D-CNN used in this experiment:

FIGURE 4. 3D-CNN Model

Model: "3dcnn"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 128, 128, 64, 1)]	0
conv3d (Conv3D)	(None, 126, 126, 62, 64)	1792
max_pooling3d (MaxPooling3D)	(None, 63, 63, 31, 64)	0
batch_normalization (BatchNo	(None, 63, 63, 31, 64)	256
conv3d_1 (Conv3D)	(None, 61, 61, 29, 64)	110656
max_pooling3d_1 (MaxPooling3	(None, 30, 30, 14, 64)	0
batch_normalization_1 (Batch	(None, 30, 30, 14, 64)	256
conv3d_2 (Conv3D)	(None, 28, 28, 12, 128)	221312
max_pooling3d_2 (MaxPooling3	(None, 14, 14, 6, 128)	0
batch_normalization_2 (Batch	(None, 14, 14, 6, 128)	512
conv3d_3 (Conv3D)	(None, 12, 12, 4, 256)	884992
max_pooling3d_3 (MaxPooling3	(None, 6, 6, 2, 256)	0
batch_normalization_3 (Batch	(None, 6, 6, 2, 256)	1024
global_average_pooling3d (Gl	(None, 256)	0
dense (Dense)	(None, 512)	131584
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 1)	513

Total params: 1,352,897
Trainable params: 1,351,873
Non-trainable params: 1,024

TRAINING MODEL

Notably, the number of sample are really small (only 40) and no random seed is specified. Hence, the expected results may be very different.

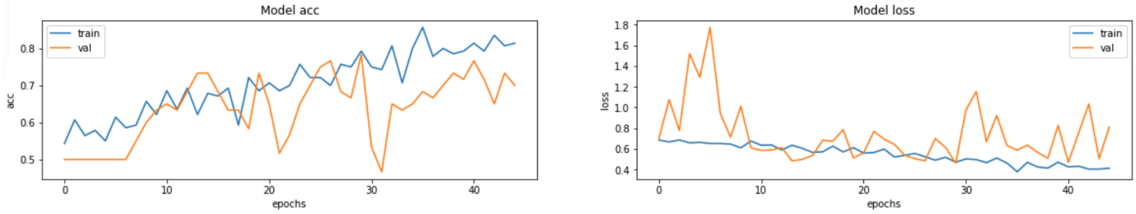
FIGURE 5. 3D-CNN Model Training

```
Epoch 29/100
70/70 - 17s - loss: 0.5197 - acc: 0.7500 - val_loss: 0.6136 - val_acc: 0.6667
Epoch 30/100
70/70 - 17s - loss: 0.4725 - acc: 0.7929 - val_loss: 0.4640 - val_acc: 0.7833
Epoch 31/100
70/70 - 17s - loss: 0.5037 - acc: 0.7500 - val_loss: 0.9758 - val_acc: 0.5333
Epoch 32/100
70/70 - 17s - loss: 0.4973 - acc: 0.7429 - val_loss: 1.1538 - val_acc: 0.4667
Epoch 33/100
70/70 - 17s - loss: 0.4673 - acc: 0.8071 - val_loss: 0.6699 - val_acc: 0.6500
Epoch 34/100
70/70 - 18s - loss: 0.5123 - acc: 0.7071 - val_loss: 0.9229 - val_acc: 0.6333
Epoch 35/100
70/70 - 17s - loss: 0.4639 - acc: 0.8000 - val_loss: 0.6345 - val_acc: 0.6500
Epoch 36/100
70/70 - 17s - loss: 0.3799 - acc: 0.8571 - val_loss: 0.5880 - val_acc: 0.6833
Epoch 37/100
70/70 - 17s - loss: 0.4704 - acc: 0.7786 - val_loss: 0.6361 - val_acc: 0.6667
Epoch 38/100
70/70 - 17s - loss: 0.4262 - acc: 0.8000 - val_loss: 0.5670 - val_acc: 0.7000
Epoch 39/100
70/70 - 17s - loss: 0.4154 - acc: 0.7857 - val_loss: 0.5101 - val_acc: 0.7333
Epoch 40/100
70/70 - 17s - loss: 0.4721 - acc: 0.7929 - val_loss: 0.8272 - val_acc: 0.7167
Epoch 41/100
70/70 - 17s - loss: 0.4265 - acc: 0.8143 - val_loss: 0.4710 - val_acc: 0.7667
Epoch 42/100
70/70 - 17s - loss: 0.4335 - acc: 0.7929 - val_loss: 0.7532 - val_acc: 0.7167
Epoch 43/100
70/70 - 17s - loss: 0.4056 - acc: 0.8357 - val_loss: 1.0338 - val_acc: 0.6500
Epoch 44/100
70/70 - 17s - loss: 0.4061 - acc: 0.8071 - val_loss: 0.5066 - val_acc: 0.7333
Epoch 45/100
70/70 - 17s - loss: 0.4141 - acc: 0.8143 - val_loss: 0.8115 - val_acc: 0.7000
```

VISUALIZING MODEL PERFORMANCE

Here the model accuracy and loss for the training and the validation sets are plotted. Because the test set is class-balanced, accuracy provides an unbiased representation of the errors.

FIGURE 6. Model accuracy and loss of training set and test set



EXPERIMENTAL RESULT

After training completion, the model needed to be validated. In the test set, the model yielded correct results by probabilistic analysis.

FIGURE 7. Running results

```
This model is 17.34 percent confident that CT scan is normal
This model is 82.66 percent confident that CT scan is abnormal
```

Experimental results show that the 3D-CNN model works well and the prediction is accurate.

CONCLUSION

Predicting the diagnosis of cancer tissue using deep learning technology greatly reduces the repetitive mechanical operation of lecturers in medical schools, improves the learning efficiency of medical students, reduces the burden of medical staff, improves the efficiency and accuracy of cancer tissue diagnosis, and raises the overall survival rate of patients. Currently, exisitng forms of computer technology, equipment enhancement, medical imaging technology, and even deep learning technology are being vigoroulsy developed, are merging with each other, and have broad prospects. Following improved 3D-CNN technology, this study shows good experimental results and completes the prediction of tumor lesions in CT images. The improvement aspects of this project include the following: first, setting dynamic learning, which is then followed by adding early stop strategy. Subsequently, said model becomes more "intelligent" in reducing time which then considerably optimizes data loading. Although this paper mainly studies the prediction of cancer lesions in CT images based on 3D convolutional neural network (notably providing a good push towards progress), there remains various problems to be improved: first, the problem of medical image dataset. The current dataset used in this paper only scratches the surface of the database. Due to the experimental time and tedious data processing, training and learning the entire database reains to be fully carried out, hence the accuracy of data classification is yet to reach the perfect state. With further improvement of the environment, these indicators are expected to be likewise further improved. Second, the final cancer prediction results in this paper can be obtained with high accuracy, but does not accurately show the characteristic lesions. This suggests that future studies should shift towards accurate predictive imaging.

REFERENCES

Bellakhdhar, F. , Loukil, K. , & Abid, M. (2013). Face recognition approach using gabor wavelets, pca and svm. *International Journal of Computer Science Issues (IJCSI)*, *10*(2),201-207.

Bose, S. , Pal, A. , Saharay, R. , & Nayak, J. (2015). Generalized quadratic discriminant analysis. *Pattern Recognition, 48*(8), 2676-2684.

Buyya, R. , Yeo, C. S. , Venugopal, S. , Broberg, J. , & Brandic, I. (2009). Cloud computing and emerging it platforms: vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems, 25*(6), 599-616.

Chenshuo, Yu, X. , Wang, Y. , & Li. (2020). Convolutional neural network visualization in adversarial example attack. *Data Science,12*(1), 247-258.

Dahl, G. E. , Sainath, T. N. , & Hinton, G. E.(2013, October 21). Improving deep neural networks for LVCSR using rectified linear units and dropout. IEEE International Conference on Acoustics. IEEE. Vancouver, British Columbia,Canada.

Girshick, R. , Donahue, J. , Darrell, T. , & Malik, J. (2013). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *IEEE Computer Society,* 580-587.

Guan, L., Liu, W., Yin, X., & Zhang, L. (2010). Traffic Incident Duration Prediction Based on Artificial Neural Network. International Conference on Intelligent Computation Technology & Automation. IEEE Computer Society. Changsha, China.

Gu, Li, Zhang, & Chen. (2020). Path tracking control of field information-collecting robot based on improved convolutional neural network algorithm. Sensors, 20(3), 797.

Lu, J.,Ye, Y., Xu, X., & Li, Q. (2019). Application research of convolution neural network in image classification of icing monitoring in power grid. *EURASIP Journal on Image and Video Processing, 2019*(1).

Sharma, A., & Paliwal, K. K. (2015). Linear discriminant analysis for the small sample size problem: an overview. *International Journal of Machine Learning & Cybernetics, 6*(3), 443-454.

Sharif Razavian, A., Azizpour, B., & Sullivan, J. (2014). CNN features off-he-hell: An astounding baseline for recognition. I/Proceedings of the IEEE Conference on Computer Vi- sion and Pattern Recognition Workshops., 806-813.

Yosinski, J. , Clune, J. , Bengio, Y. , & Lipson, H. . (2014). How transferable are features in deep neural networks? MIT Press, 3320-3328.

ABOUT THE AUTHORS

Qi Dongkai received a Master of Science degree in Artificial Intelligence from the Illinois Institute of Technology. He is a PhD student at the University Malaya (UM) since 2019. His research areas of interest are computer vision, natural language processing and artificial intelligence project management. He has 13 years of experience in financial data project management in commercial banks and the People's Bank of China, and 3 years of experience in artificial intelligence software project development.

Huang Huang received BSc. in Software engineering from WuHan Textile University and MSc. In Data Science from University Malaya (UM) in 2021. He is a PhD candidate in University Malaya(UM) and his major research direction includes natural language processing and fuzzy inference system. He has 2-year experience in software testing.