A Deep Feedforward Neural Network Model for Image Prediction

C.C. Lee[1], M.H.F Rahiman[1,2*], R. A. Rahim [3] and F. S. A. Saad[1,2]

[1]Faculty of Electrical Engineering Technology, Universiti Malaysia Perlis, Pauh Putra Campus, 02600 Arau, Perlis, Malaysia.
[2]Centre of Excellence for Advanced Sensor Technology (CEASTech), Universiti Malaysia Perlis (UniMAP), Perlis, Malaysia.
[3]School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia.

*Email: hafiz@unimap.edu.my

Abstract. Deep learning is a sub-field of machine learning, which inspired by the structure of human brain where it is composed with a lot of neurons that each of them able to perform a simple operation and interact with each other to make a decision. In recent years, deep learning has grown exponentially and have brought revolutionary advances in computer vision-based applications such as image classification, object detection, simultaneous localisation and mapping (SLAM), action and activity recognition, age estimation as well as human pose estimation. Based on the literature review, deep learning techniques applied in the vision-based application such as Convolutional Neural Networks (CNNs), Restricted Boltzmann Machines (RBMs), Autoencoder and Sparse Coding mostly focused on solving the feature extraction and classification problems. However, these techniques do not apply to the image processing applications such as image fusion and image reconstruction that require a quantitative knowledge to solve some problems in such fields. The process of obtaining quantitative knowledge in image fusion and reconstruction is a regression problem. In this case, a deep feedforward neural network is applied to solve the regression problem due to its ability to map complicated nonlinear functions.

1. Introduction

In recent years, deep learning has grown exponentially and have brought revolutionary advances in computer vision-based applications such as image classification, object detection, simultaneous localisation and mapping (SLAM), action and activity recognition, age estimation as well as human pose estimation[1]–[4]. Deep learning is a sub-field of machine learning, which inspired by the structure of human brain where it is composed with a lot of neurons that each of them able to perform a simple operation and interact with each other to make a decision[5]. The deep learning uses multiple layers of non-linear information to learn the features and represent data with abstraction for supervised and unsupervised learning applications[6]. Among the artificial intelligence (AI) techniques, deep learning has gained huge interest among the researchers as it provides higher accuracy in the vision-based tasks such as image classification, semantic segmentation and object detection. Besides, deep learning does not need expert analysis and it also provides flexibility as the neural networks can be retrained using a custom dataset for any use case[1]. Also, deep learning provides automatically feature extraction and classification compared to machine learning as shown in Figure 1[7]. This is because the parameters of the deep learning model are trained jointly and the features are learned by transforming data into abstract representations.

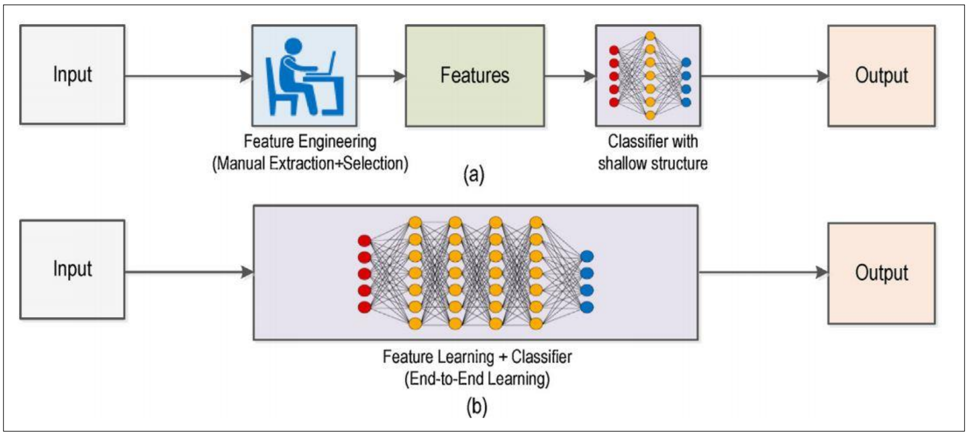


Figure 1: Comparison between two artificial intelligence techniques: (a) machine learning and (b) deep learning [7].

Based on the literature review, deep learning techniques applied in the vision-based application mostly focused on solving the feature extraction and classification problems in which the output variable are labelled with different classes and it can predict the probability of belonging to a particular class. These techniques are Convolutional Neural Networks (CNNs), Restricted Boltzmann Machines (RBMs), Autoencoder and Sparse Coding[8]. They provide great accuracy in solving closed-end classification problems. However, these techniques do not apply to the image processing applications such as image fusion and image reconstruction that require quantitative knowledge to solve some problems in such fields.

2. Neural networks for the regression problem

The process of obtaining quantitative knowledge in image fusion and reconstruction is a regression problem. Image fusion is a process of combining two or more images of the same object into a comprehensive image. While image reconstruction is a process of converting the signals obtained from multiple projections during the data acquisition phase into an image. Both of the processes are complex and nonlinear. Instead of using other machine learning algorithms such as linear regression, logistic regression, support vector regression (SVR) and regression trees, in this case, a neural network is used to solve the regression problem. This is because neural networks are good in mapping complicated nonlinear functions.

Figure 2 illustrates the framework of deep neural network (DNN) for solving regression problem. The network contains three important layers: an input layer, three hidden layers and an output layer. The output of the network can be expressed as:

$$F(x; \theta) = (f_L \text{ o } \dots \text{ o } f_1)(x) \qquad (1)$$

where each of the f_i is dependent on the parameter $\theta_i \in H_i$, and θ represents the parameter set $\{ \theta_1 , \dots , \theta_L\}$[9]. The optimization of the loss function is one of the important parameters in a neural network. This is because the loss function is designed to predict the performance of a neural network model. In the case of regression, the most common loss function to consider is squared loss, it can be described as:

$$J_R(x, y; \theta) = 0.5\{y - F(x; \theta), y - F(x; \theta)\} \qquad (2)$$

To obtain the new parameters: weights (w) and biases (b), and minimize the errors during network prediction process, a gradient descent step algorithm is applied, given by:

$$V_{\theta L}J(x, y; \theta) = \nabla^*_{\theta L}\ f_L\,(x_L) \cdot D^*_{wL+1}(x_{L+1}) \cdot e_L = \nabla^*_{\theta L}\ f_L\,(x_L) \cdot e_L \qquad (3)$$

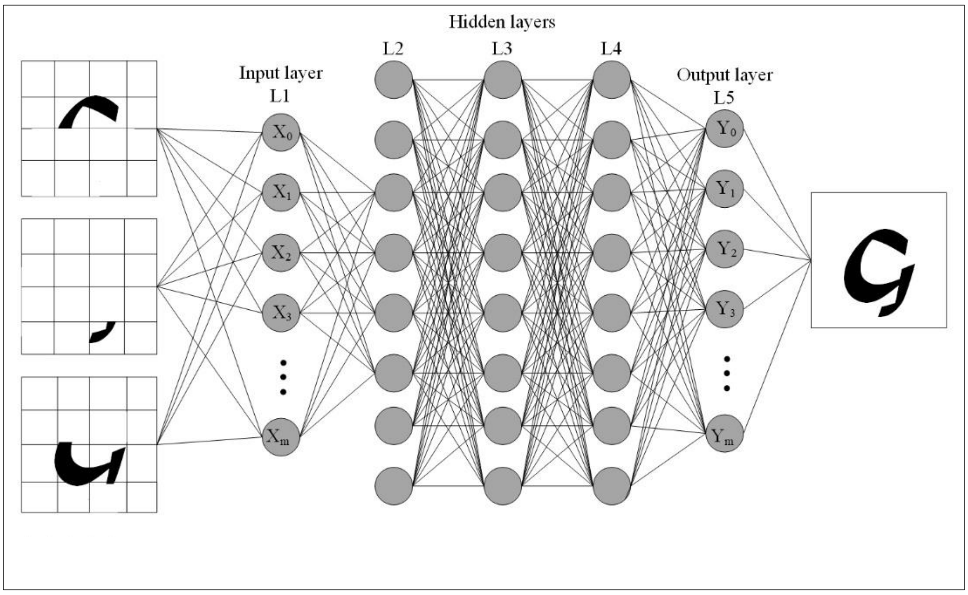
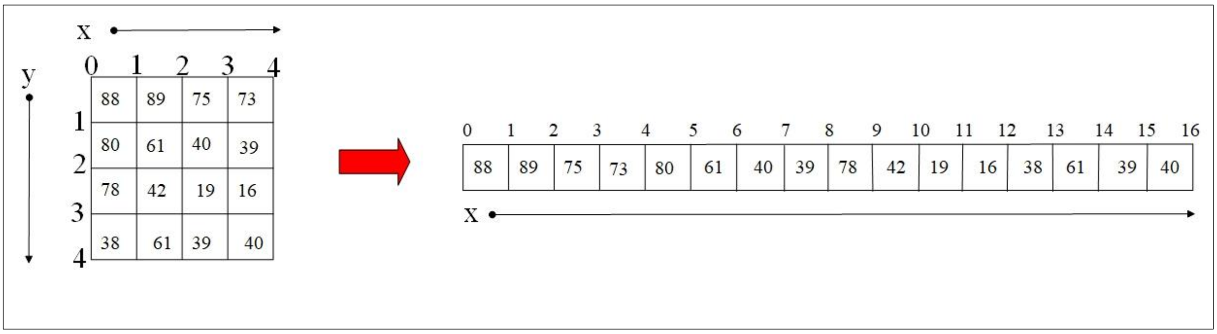


Figure 2: The framework of deep neural networks (DNN) for solving regression problem.

3. DNN for image prediction

In the case of image prediction, three images which made up of 4x4 pixels are used as the inputs for the feedforward model to predict a new image. The two-dimensional (2D) images are represented as a matrix of pixel values which have an X and Y position as shown in Figure 3. To ease the training process of deep learning model, the 2D images are converted into array pixels which have only one dimension and the colour values are stored in a linear sequence. Next, the array pixels are normalised into the range from 0 to 1 before feeding into the deep learning model.



88	89	75	73
80	61	40	39
78	42	19	16
38	61	39	40

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
88	89	75	73	80	61	40	39	78	42	19	16	38	61	39	40	

Figure 3: Convert a two-dimensional (2D) image into one-dimension array pixels.

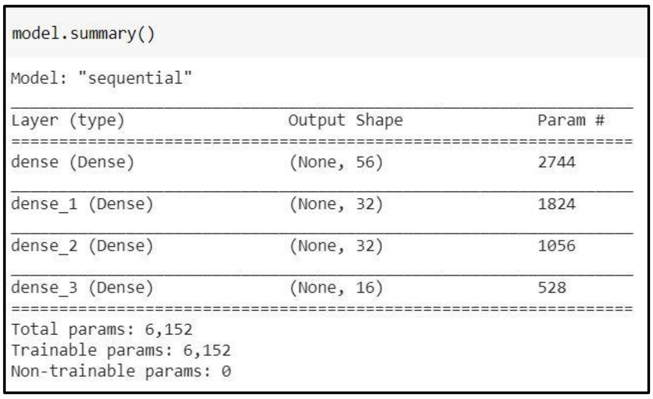
Therefore, the input vectors for the feedforward neural networks model consisted of 48 measurement cases (4). Each element contained a unique value:

$$I = [x_{1,} x_{2,} x_3, x_4, x_5, \dots , x_{48}] \quad (4)$$

The output vector contained 16 elements: pixel values of a new image (5).

$$\Theta = [y_1, y_2, y_3, \dots , y_{16}] \quad (5)$$

Figure 4 illustrates a summary of DNN model for image prediction. The network has 48 inputs, 56 neurons in the first hidden layer, 32 neurons in the second and third hidden layers respectively and 16 outputs. The hidden layer uses a rectified linear unit (RELU) activation. In the output layer, the transfer function is a softmax function.



```
model.summary()

Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 56)                2744
_________________________________________________________________
dense_1 (Dense)              (None, 32)                1824
_________________________________________________________________
dense_2 (Dense)              (None, 32)                1056
_________________________________________________________________
dense_3 (Dense)              (None, 16)                528
=================================================================
Total params: 6,152
Trainable params: 6,152
Non-trainable params: 0
```

Figure 4:A summary of the deep feedforward neural network model.

A dataset of 150 cases was used to train the neural network. The following results apply to the gradient descent training variant. The gradient descent is a popular training algorithm in many cases especially when we have a large neural network. However, the drawback of this algorithm is it requires many iterations for the function.

All datasets were randomly divided into 3 sets: training, validating and testing in 70:15:15. The training set was used to properly train each of the subsystems. While the validation set was used to determine the moment of stopping the iterative training process. The condition for stopping the learning process was a non-decreasing MSE for the validation set for the next 6 iterations[10]. The test set can be used for independent assessment of network quality after the learning process. The mean squared error (MSE) is evaluated according to (6).

$$\mathrm{MSE} = \frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} (y'_i - y^*_i)^2 \quad (6)$$

Figure 5 presents the performance of the deep feedforward neural network model. The performance of the proposed DNN model achieved 95% and 95.5% of accuracy in the training and validating process. While the losses in the training and validating process decrease gradually. However, the losses in the training process are higher than the validation losses.

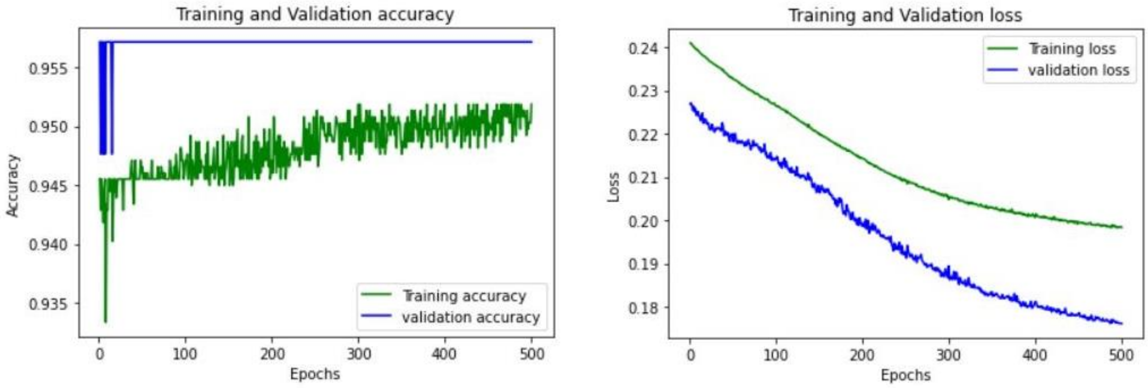


Figure 5: Training and validation result of DNN-Regression Model.

4. Conclusion and future work

In this paper, a deep feedforward neural network is applied to solve the regression problem in the image processing applications such as image fusion and reconstruction which require quantitative knowledge in solving some problems in such fields. The process of obtaining quantitative knowledge in image fusion and reconstruction is a regression problem. In this case, a neural network is applied to solve the regression problem due to its ability to map complicated nonlinear functions. The neural network gives promising results where both of the training and validating accuracy exceed 90%. However, image predictions highly depend on the quality of the training set. Further work will focus on the parameter selection of the learning rate and the minimization of losses in the training process.

Acknowledgements
This research was financed by the Collaboration Research Grant UTM-UniMAP 2018 (Grant No.9023-00002). The authors gratefully thank Universiti Malaysia Perlis (UniMAP) for the facilities and technical support.

References

[1] N. O'Mahony *et al.*, "Deep Learning vs. Traditional Computer Vision," *Adv. Intell. Syst. Comput.*, vol. 943, no. Cv, pp. 128–144, 2020.

[2] M. R. Minar and J. Naher, "Recent Advances in Deep Learning: An Overview," *SpringerBriefs Comput. Sci.*, no. February, pp. 24–34, 2018.

[3] S. Lathuiliere, P. Mesejo, X. Alameda-Pineda, and R. Horaud, "A Comprehensive Analysis of Deep Regression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 9, pp. 2065–2081, 2020.

[4] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep Learning for Computer Vision: A Brief Review," *Comput. Intell. Neurosci.*, vol. 2018, 2018.

[5] N. O'Mahony, T. Murphy, K. Panduru, D. Riordan, and J. Walsh, "Adaptive process control and sensor fusion for process analytical technology," *2016 27th Irish Signals Syst. Conf. ISSC 2016*, no. June 2018, 2016.

[6] L. Deng and D. Yu, "Deep learning: Methods and applications," *Found. Trends Signal Process.*, vol. 7, no. 3–4, pp. 197–387, 2013.

[7] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, vol. 48, pp. 144–156, 2018.

[8] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[9] A. L. Caterini and D. E. Chang, "Deep Neural Networks in a Mathematical Framework," *SpringerBriefs Comput. Sci.*, pp. 24–34, 2018.

[10] T. Rymarczyk, G. Kłosowski, and E. Kozłowski, "Comparison of Selected Machine Learning Algorithms for Industrial Electrical Tomography," 2019.