

Classification of Garments from Fashion MNIST Dataset Using CNN LeNet-5 Architecture

Mohammed Kayed

Faculty of Computers and Artificial Intelligence, Beni-Suef University, Beni-suef, Egypt, 62511

mskayed@gmail.com

Ahmed Anter

Faculty of Computers and Artificial Intelligence, Beni-Suef University, Beni-suef, Egypt, 62511

sw_anter@yahoo.com

Hadeer Mohamed

Faculty of Science, Beni-Suef University, Beni-suef, Egypt, 62511

hadeer.mohamed98@gmail.com

ABSTRACT

Recently, deep learning has been used extensively in a wide range of domains. A class of deep neural networks that give the most rigorous effects in solving real-world problems is a Convolutional Neural Network (CNN). Fashion businesses have used CNN on their e-commerce to solve many problems such as clothes recognition, clothes search and recommendation. A core step for all of these implementations is image classification. However, clothes classification is a challenge task as clothes have many properties, and the depth of clothes categorization is highly complicated. This complicated depth makes different classes to have very similar features, and so the classification problem becomes very hard. In this paper, CNN based LeNet-5 architecture is proposed to train parameters of the CNN on Fashion MNIST dataset. Experimental results show that LeNet-5 model achieved accuracy over 98%. Therefore, it outperforms both the classical CNN model and the other existing state-of-the-art models in literatures.

Keywords

Deep learning architectures; Fashion MNIST; Fashion Classification; Convolutional Neural Network (CNN); LeNet-5.

1. INTRODUCTION

Over past few years, with the assistance of various layers, deep learning [1] has been widely used and achieved very good results in different domains such as computer vision [2], big data [3], automatic speech recognition [4] and natural language processing [5]. A common architecture of deep neural networks is CNN. CNN is a multi-layer perceptron neural network that extracts properties from the input data and is trained with the neural network back-propagation algorithm. CNN can learn complex, high-dimensional, non-linear mappings from a very large number of data (images). Moreover, CNN gives an excellent classification average for images [6]. The main advantages of CNN are that it extracts the salient features that are never changed, and it is invariant to shifting, scaling and distortions of input data (images). CNN based LeNet-5 architecture has shown very good results in many domains such as image classification [7], pattern recognition [8], computer vision [9] and image segmentation.

One of the most challenging multi-classes classification problems is fashion classification in which labels that characterize the clothes type are assigned to the images. The difficulty of this multi-classes fashion classification problem is due to the richness of the clothes properties and the high depth of clothes categorization as well. This complicated depth makes different labels/classes to have similar features. This paper tries to enhance

the performance of the fashion classification problem on the Fashion-MNIST Dataset [10], which contains 70,000 images (each image is labeled from the 10 categories shown in Figure 1: T-shirt/top, Trousers, Pullover, Dress, Coat, Sandals, Shirt, Sneaker, Bag and Ankle boot).

There are some issues to consider in classification of fashion [11]. First, garments can be easily distorted by lengthening pattern. Second, some garments might be considered as various according to the opinion, and various garments might be considered as same. Third, some garment items are robust to be recovered due to their small size. Fourth, photos can be taken in various cases such as the difference in the angle, light and noise backgrounds. Fifth, some garment classes have similar features and can be fuzzy, such as trouser and tights. Sixth, a garment image is different based on whether it is just a photo of a garment or a photo of the model's wearing garment. Therefore, an algorithm that could be used to get high multi-classes fashion classification performance is of great necessity. As well as this paper gives a brief review of the different CNN models for the classification of the Fashion-MNIST, the major contribution of this paper is that the multi-classes fashion classification problem will be solved by the CNN based LeNet-5 architecture. To the best of our knowledge, this model is not used before for this common MNIST dataset.

The rest of the paper is organized as follows: Section 2 gives a review of the related works. Section 3 describes the used dataset and methodology. Section 4 presents the proposed model. Section 5 presents the experiments and the classification results, while Section 6 concludes our work.

2. RELATED WORKS

Deep learning and CNN have been fully surveyed in [12]. Many CNN architectures have been used in image classification: LeNet [13], Alex Net [14], Google Net [15], VGGNet [16] and ResNet [17]. All of these architectures compete to correctly classifying and detecting images. Neural networks have also been applied to metrics learning with applications in image similarity estimation and visual search. Recently, two datasets have been published. MNIST [18] and Fashion-MNIST datasets for image classification [19] with 70,000 annotated real-life images. In this section, we shall briefly review the works done on the Fashion-MNIST dataset as follows.

Shobhit et al. [20] proposed a model for classification of fashion article images. Convolutional neural network based deep learning architectures are trained to classify images of the Fashion-MNIST dataset. Also, three different CNN architectures used batch normalization and residual skip connections are suggested to accelerate the learning process. The results showed that the

proposed model enhance the accuracy of other literary systems by around 2%.

Michael McKenna [21] suggested a model for addition and comparison of the Sigmoid features, ELUs and ReLUs of missing benchmarks in the Fashion-MNIST data set. First, the missing multi-layer non-convolutional neural feed-forward networks in Fashion-MNIST given a benchmark. Second, testing the efficacy of contemporary activation features (compared with ELU, ReLU and sigmoid). The goals are novel because Fashion-MNIST has many benchmarks, but none with non-deep architectures and ELUs are rarely used than convolutional networks. The results showed that the output was substantially worse than the convolutional benchmarks and that some of the advantages of ELUs and ReLUs were noted when the network was slightly trained.

Shuning Shen [22] used the Short-Term Memory Networks to create a model that uses the Fashion-MNIST dataset for image classification, reducing time consumption and increasing the model's predictive accuracy. Results showed that the LSTM model can fit the dataset with the best precision (88.26 %).

Han et al. [23] create a good benchmark dataset that has all the accessibility of MNIST, namely its little size, and straightforward encoding. Fashion-MNIST images are transformed into the format that matches the MNIST dataset. So, they can work with the original MNIST dataset instantly with all machine learning systems. In addition, Fashion-MNIST is more difficult than simply digit information from MNIST, while MNIST has been trained in greater than 99.7%.

Greeshma et al. [24] proposed a system to classify the fashion products in Fashion-MNIST dataset using HOG features with a multi-class SVM classifier. The results showed that the suggested system provides matching fashion object classification efficiency after implementation of the suggested fashion articles classification system using HOG feature space and multi-class SVM classifier as compared to available literature works.

3. MATERIAL AND METHODS

Convolutional neural network is a class of deep feed-forward artificial neural network which is used mainly for image processing, classification, segmentation and others [25]. It includes three types of layers: convolutional, pooling and fully-connected layers. These layers act as a classifier [26] as shown in Fig 1.

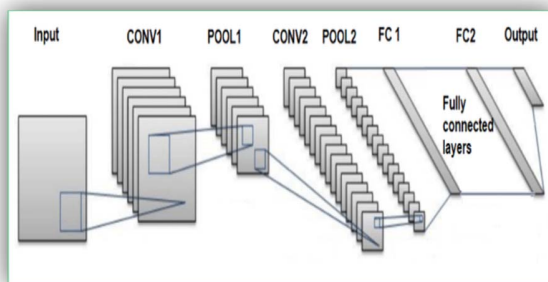


Figure 1. CNN architecture. The gray squares mention the feature maps and the blue squares mention the convolution filter.

CNN architecture has three concepts that make it effective: local receptive areas, weights sharing and down sampling process [27]. There are common CNN architectures such as Alex Net, VGG, ResNet, Dense Net, and LeNet-5 (see [28, 29] for more details).

In this paper, we use the last architecture (LeNet-5) for Fashion-MNIST image classification. To the best of our knowledge, no prior works have used it with this dataset. We use it as it is simple and gives high-performance results in several domains. It is built on local receptive fields, shared weights and special subsampling. In more details, as shown in Fig. 2, LeNet-5 CNN layers are:

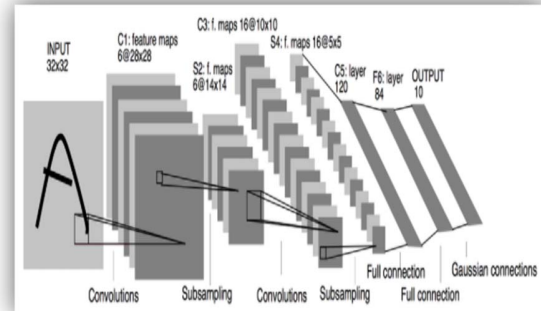


Figure 2. LeNet-5 Architecture

First Layer (C1): The input for this layer is a 32×32 gray scale image. This image passes through the first convolutional layer with six feature maps or filters having size 5×5 and a stride of 1. The image dimensions will be changed from $32 \times 32 \times 1$ to $28 \times 28 \times 6$.

Second Layer (S2): It is an average pooling layer with six feature maps of size 14×14 . Each unit in each feature map is linked to a 2×2 block in the identical feature map in C1. Also, S2 has 12 trainable parameters and 5880 connections. The resulting image dimensions will be reduced to $14 \times 14 \times 6$ [30].

Third Layer (C3): Convolutional layers with 16 feature maps having size 5×5 and a stride of 1. Each unit in each feature map is linked to different 5×5 blocks as similar to positions in a subset of S2's feature maps. Finally, the last one loads input from all S2 feature maps.

Fourth Layer (S4): Also, it is an average pooling layer with filter size 2×2 and a stride of 2 with 16 feature maps of size 5×5 . Each unit in each feature map is linked to a 2×2 blocks in the identical feature map in C3, and identical path as C1 and S2. This layer is the same as the second layer (S2) unless it has 32 trainable parameters and 2000 links to the output will be decreased to $5 \times 5 \times 16$.

Fifth Layer (C5): Fully connected convolutional layers with 120 feature maps. Each unit is linked to a 5×5 block on all 16 of S4's feature maps. C5 is classified as a convolutional layer instead of a fully connected layer because if LeNet-5 input were made bigger with everything else protected fixed, the feature map dimension would be larger than 1×1 . Each of the 120 units in C5 is linked to all the 400 nodes ($5 \times 5 \times 16$) in the fourth layer S4.

Sixth Layer (F6): A fully connected layer with 84 units. It is fully linked to C5. It contains 10164 trainable parameters.

Output Layer: A fully connected softmax layer \hat{y} with 10 possible rates identically to the digits from 0 to 9.

3.1 Dataset Description

In this paper, we use Fashion-MNIST dataset which consists of 60,000 training set pictures and 10,000 test set pictures. Each symbol is a gray-scale image of 28×28, linked to 10-category labels as shown in Fig 3. Fashion-MNIST is intended as a direct drop-in replacement of the original MNIST dataset and is used as a benchmark for different machine learning algorithms.

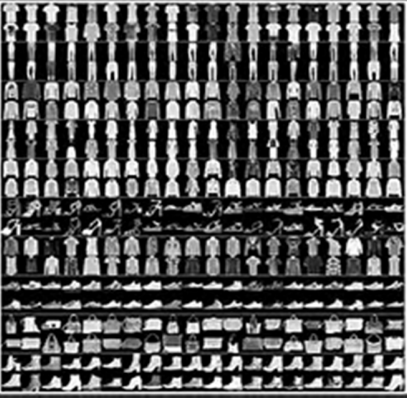
Label	Description	Examples
0	T-Shirt/Top	
1	Trouser	
2	Pullover	
3	Dress	
4	Coat	
5	Sandals	
6	Shirt	
7	Sneaker	
8	Bag	
9	Ankle boots	

Figure 3. Fashion-MNIST Dataset.

3.2 Evaluation Metrics

In this paper, we use different methods to evaluate the proposed model and compare it with the other models. The confusion metrics that are used to display the true labels versus the predicted labels of our test dataset are precision, recall, accuracy and F-measure [31, 32, and 33]. Further metrics could also be used to evaluate the effectiveness of the model such as kappa coefficient, Informedness, Mean Square Error (MAE), Sensitivity, Specificity and confusion metrics. All these metrics are defined as follows.

- (a) **Precision:** The rate of relevant cases through the retrieved cases. It is defined as:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (1)$$

Where TP is the true positive cases and FP is false positive cases.

- (b) **Recall:** The part of relevant cases that have been retrieved through all amounts of relevant cases. It is defined as:

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2)$$

Where FN is the false negative cases.

- (c) **Accuracy:** The ratio of true positive and true negative in all rated cases. It is defined as:

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (3)$$

Where TN is the true negative cases.

- (d) **F-measure:** It merges precision and recall. It is the consistent average of precision and recall. The traditional F-measure is defined as

$$F = 2 \times \frac{(\text{precision} + \text{recall})}{(\text{precision} + \text{recall})} \quad (4)$$

- (e) **Informedness:** It evaluates the discriminative power of the test. It is defined as:

$$\text{Informedness} = \text{Recall} + \text{Inverse Recall} - 1 \quad (5)$$

- (f) **Kappa Coefficients:** Kappa coefficient will be utilized to confirm the existence of the objects that were presented. The Kappa coefficient is a statistical quantity of accuracy that is used to determine qualitative documents and agreement between two raters. It is defined as:

$$\text{Kappa} = \frac{(\text{observed agreement} - \text{expected agreement})}{(1 - \text{expected agreement})} \quad (6)$$

4. EXPERIMENTS AND RESULTS

The full architecture of our proposed model for fashion MNIST image classification is shown in Fig 4. First, the convolution layer takes the input 32×32 gray scale images with 6 feature maps having size 5×5 and a stride one. Then, the second layer takes the input from first convolution layer with a filter size 2×2 and a stride two after passing through a pooling layer. After that, pooling layers are added after the first, the second and the fifth convolution layers. Also, a new fully connected layer (FC3) which takes input from the output of the second fully connected layer (FC2) has been added in this model. Finally, the output of the last layer (FC4) is connected to a softmax layer for the image classification.

This section describes the details of our conducted experiment. It is divided into three main sub-sections. First, it discusses the implementation environment. Second, it shows the experimental results. Finally, a comparison with the other models is presented in the last sub-section.

4.1 Setup Environment

Our experimental environment of this work has been set by following a straightforward process. We apply LeNet-5 model and use Ubuntu 16.4 operating system. In this experiment, all networks are trained and evaluated using Tensor Flow (1.4.0) and Keras (2.2.4) on a machine with NVIDIA GeForce GPUs (GTX 1080) to make the computation faster.

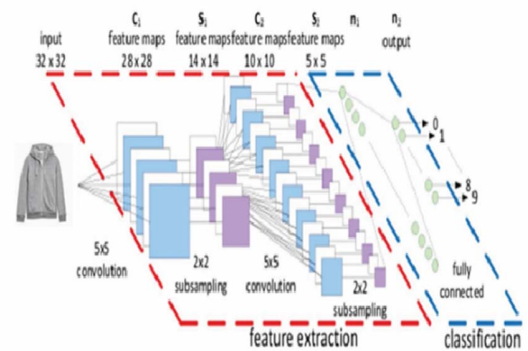


Figure 4. The architecture of our proposed LeNet-5 model.

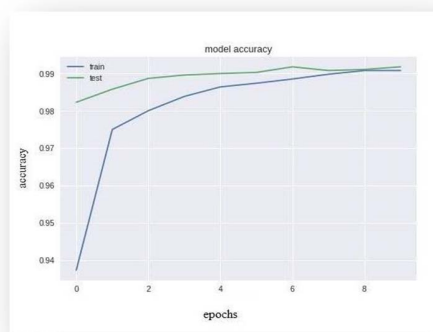


Figure 5. The model accuracy with different number of epochs.



Figure 6. Training accuracy versus validation

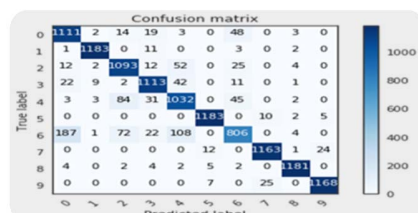


Figure 7. The model confusion matrix.

4.2 Experimental Results and Discussion

After selecting the architecture of the networks, it was necessary to define the learning rate, the batch size and the number of epochs used to train the networks. The batch size is the number of training set samples/instances before the model is updated. The number of epochs describes how many times the algorithm processes a complete dataset. In particular, we use the following hyper-parameters: $\alpha = 0.005$, batch size = 32 and number of epochs = 10.

Figures 5 and 6 show the performance of the LeNet-5 model on the fashion MNIST dataset with different epoch values. As shown in the two figures, our experiments show encouraging results. The accuracy average across all 10 trials was 98.9% for the test set with time-consuming 80 minutes. Figure 5 plots the accuracy of the different training and testing splits for the 10 categories based on the results from the chosen test set. The loss during the validation phase was much higher than the loss during the training phase as displayed in Fig 6. Finally, Fig. 7 shows the confusion matrix for the model. The confusion matrix displays the true labels versus the predicted labels of our test dataset.

Table 1 shows the recall, precision and F-measure for the LeNet-5 model in each category. As shown in the table, the results are very high (an average of 99% for each metric). The values of kappa

coefficient, Informedness and Mean square error (MSE) are given in Table 2.

Table 1. LeNet-5 performance on the Fashion-MNIST dataset.

Class	Class label	Precision	Recall	F1 Score
T-Shirt/Top	0	1.00	1.00	1.00
Trouser	1	1.00	1.00	1.00
Pullover	2	0.99	0.99	0.99
Dress	3	0.99	1.00	0.99
Coat	4	0.99	0.99	0.99
Sandals	5	1.00	0.99	0.99
Shirt	6	0.99	0.99	0.99
Sneaker	7	0.99	0.99	0.99
Bag	8	1.00	0.99	0.99
Ankle Boots	9	0.99	0.99	0.99
Overall		0.99	0.99	0.99

Table 2. Evaluation metrics (kappa coefficient, Informedness and MSE) of LeNet-5 on the Fashion-MNIST dataset.

Data set	Kappa	Informedness	MSE
Fashion-MNIST	0.75	0.77	0.66

4.3 Performance Comparison with Other Models

In this subsection, we compare our LeNet-5 model with the other models that have been tested on the Fashion MNIST dataset (such as Support Vector Classifier (SVC) and Evolutionary Deep Learning (EDEN) [34]). Table 3 shows the performance of our model and the other models (from the literatures). As shown in the table, LeNet-5 achieves higher accuracy than the previously used classifiers (SVC and EDEN) and even better than the state-of-the-art classification result using convolutional configuration (CNN2 + Batch Norm + Skip).

Table 3. A comparison between LeNet-5 and the other classification models on Fashion MNIST dataset.

Model (Method)	Test Accuracy
Three-layer Neural Network	87.23%
Support Vector Classifier with rbf kernel	89.70%
Evolutionary Deep Learning Framework	90.60%
CNN using SVM activation function	90.72%
CNN using Softmax activation function	91.86%
CNN with Batchnorm-alization	92.22%
CNN with Batch Normalization. and Residual skip	92.54%
Decision Tree Classifier	79.80%
ExtraTreeClassifier	77.50%
GaussianNB	88.00%

KNeighborsClassifier	85.40%
LinearSVC	83.60%
LogisticRegression	84.20%
RandomForestClassifier	87.30%
SGDClassifier	81.90%
SVC	89.70%
CNN	91.61%
CNN-LeNet-5	98.80%
Multilayer perceptron	78.33%

5. CONCLUSION AND FUTURE WORK

With the growth in deep learning methodologies, image recognition using CNN is excessively applied in fashion domains such as clothes classification, clothes retrieval and automatic clothes labeling. In this paper, we apply LeNet-5 architecture on the Fashion MNIST dataset. LeNet-5 gives a higher performance (an accuracy over 98% was obtained) as compared to other existing models. We plan to conduct a comprehensive comparison between different CNN architectures (such as VGG16) on other clothes datasets (such as Image Net). We also plan to apply LeNet-5 and other CNN architectures on a dataset of real clothes images collected by our self for the evaluation purposes.

6. REFERENCES

- [1] Hu, W., Huang, Y., Wei, L., Zhang, F., & Li, H. (2015). Deep convolutional neural networks for hyper spectral image classification. *Journal of Sensors*, 2015.
- [2] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Image Net classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).
- [3] Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R., & Muharemagic, E. (2015). Deep learning applications and challenges in big data analytics. *Journal of Big Data*, 2(1), 1..
- [4] Chen, X. W., & Lin, X. (2014). Big data deep learning: challenges and perspectives. *IEEE access*, 2, 514-525.
- [5] Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H. G., & Ogata, T. (2015). Audio-visual speech recognition using deep learning. *Applied Intelligence*, 42(4), 722-737.
- [6] Jianming, Z. H. A. N. G., Zhicai, Z. H. A. N., Keyang, C. H. E. N. G., & Yongzhao, Z. (2015). Review on development of deep learning. *Journal of Jiangsu university: natural science editions*, 36(2), 191-200.
- [7] El-Sawy, A., Hazem, E. B., & Loey, M. (2016, October). CNN for handwritten arabic digits recognition based on LeNet-5. In *International Conference on Advanced Intelligent Systems and Informatics* (pp. 566-575). Springer, Cham.
- [8] Yuan, A., Bai, G., Jiao, L., & Liu, Y. (2012, March). Offline handwritten English character recognition based on convolutional neural network. In *2012 10th IAPR International Workshop on Document Analysis Systems* (pp. 125-129). IEEE.
- [9] Xie, L., Wang, J., Wei, Z., Wang, M., & Tian, Q. (2016). Disturblabel: Regularizing cnn on the loss layer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4753-4762).
- [10] Kaggle, (2019), <https://www.kaggle.com/zalando-research/fashionmnist>, April, 2019.
- [11] Hu, W., Huang, Y., Wei, L., Zhang, F., & Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015.
- [12] Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., & Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. *Electronics*, 8(3), 292.
- [13] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
- [14] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).
- [15] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).
- [16] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [17] Li, X., & Cui, Z. (2016, September). Deep residual networks for plankton classification. In *OCEANS 2016 MTS/IEEE Monterey* (pp. 1-4). IEEE.
- [18] Deng, L. (2012). The MNIST database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6), 141-142.
- [19] github, (2019), <https://github.com/zalando-research/fashion-mnist>, Accessed: April, 2019.
- [20] Bhatnagar, S., Ghosal, D., & Kolekar, M. H. (2017, December). Classification of fashion article images using convolutional neural networks. In *2017 Fourth International Conference on Image Information Processing (ICIIP)* (pp. 1-6). IEEE.
- [21] McKenna, M. (2017). A comparison of activation functions for deep learning on Fashion-MNIST. *arXiv preprint arXiv:1708.07747*.
- [22] Shen, S. Image Classification of Fashion-MNIST Dataset Using Long Short-Term Memory Networks.
- [23] Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.
- [24] K V, Greeshma & Sreekumar, K.. (2019). Fashion-MNIST classification based on HOG feature descriptor using SVM. *International Journal of Innovative Technology and Exploring Engineering*. 8. 960-962.
- [25] Ciresan, D. C., Meier, U., Gambardella, L. M., & Schmidhuber, J. (2011, September). Convolutional neural network committees for handwritten character classification.

- In 2011 International Conference on Document Analysis and Recognition (pp. 1135-1139). IEEE.
- [26] Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. In 2017 International Conference on Engineering and Technology (ICET) (pp. 1-6). IEEE.
- [27] Namatēvs, I. (2017). Deep convolutional neural networks: Structure, feature extraction and training. *Information Technology and Management Science*, 20(1), 40-47.
- [28] Khan, A., Sohail, A., Zahoor, U., & Qureshi, A. S. (2019). A survey of the recent architectures of deep convolutional neural networks. *arXiv preprint arXiv:1901.06032*.
- [29] Han, J., Zhang, D., Cheng, G., Liu, N., & Xu, D. (2018). Advanced deep-learning techniques for salient and category-specific object detection: a survey. *IEEE Signal Processing Magazine*, 35(1), 84-100.
- [30] El-Sawy, A., Hazem, E. B., & Loey, M. (2016, October). CNN for handwritten arabic digits recognition based on LeNet-5. In *International Conference on Advanced Intelligent Systems and Informatics* (pp. 566-575). Springer, Cham.
- [31] Zhai, C., Cohen, W. W., & Lafferty, J. (2015, June). Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *ACM SIGIR Forum* (Vol. 49, No. 1, pp. 2-9). ACM.
- [32] Hand, D., & Christen, P. (2018). A note on using the F-measure for evaluating record linkage algorithms. *Statistics and Computing*, 28(3), 539-547.
- [33] Visa, S., Ramsay, B., Ralescu, A. L., & Van Der Knaap, E. (2011, April). Confusion Matrixbased Feature Selection. In *MAICS* (pp. 120-127).
- [34] Dufourq, E., & Bassett, B. A. (2017). Eden: Evolutionary deep networks for efficient machine learning. In *2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)* (pp. 110-115). IEEE.