Fashion Image Classification Using Deep Convolution Neural Network

M. S. Saranya(✉) and P. Geetha

Anna University, Chennai, India
saranyasivaraman5@gmail.com, geethap@cs.annauniv.edu

Abstract. We present an analysis of optimal Convolution Neural Network (CNN) for fashion data classification by altering the layers of CNN in this paper. The suggested system employs three deep convolution layers, max pooling layers, and two fully connected layers, as well as dropout layers. While modified layers enhance the test accuracy of Adam optimizer when compared to start-of-art-models. The objective of this work is to address the multi class classification problem and to evaluate the performance of CNN's Adam and RMSProp optimizer. The experiment was carried out using the Fashion-MNIST benchmark dataset. The suggested method has a test accuracy of 92.68%, compared to 91.86% in CNN using the softmax function and 92.22% in CNN utilizing batch normalization.

Keywords: Fashion MNIST · ADAM optimizer · Deep Convolution Neural Network · Image classification

1 Introduction

Object Classification is one of the most prominent applications in computer vision [14]. The fundamental goal of object classification would be feature extraction from photos and categorize them into appropriate classes using any of the available classifiers or classification techniques. Object categorization is a critical issue in a variety of computer vision applications, including image retrieval, autonomous driving, and monitoring. Yan Zhang et al. [1] advocated using stacked sparse auto-encoders (SAE) for obtaining beneficial properties of halftone pictures and furthermore developed an efficient patches extraction method for halftone images in one of their previous research. Anselmo Ferreira et al. [2] have used ad-hoc Convolution Neural Network approaches to classify granite under different resolutions and this was the first approach to compare with texture descriptors.

The fashion-MNIST images were determined by Han Xiao et al. [3] is gray-scale images of 28 * 28 with 70,000 fashion products from 10 class labels. The very first 60,000 photographs are being used for training, while the final 10,000 images were used for testing. The original MNIST dataset is still outstanding for machine learning approaches that could benefit from a direct drop-in change. For the fashion-MNIST pictures data set categorization, several studies were presented.

Emmanuel Dufourq et al. [4] have proposed Evolutionary Deep Networks (EDEN) which is an efficient neuro-evolutionary algorithm to address the increasing complexity.

Published by Springer Nature Switzerland AG 2022
E. J. Neuhold et al. (Eds.): ICCCSP 2022, IFIP AICT 651, pp. 116–127, 2022.
https://doi.org/10.1007/978-3-031-11633-9_10

Alexander Schindler et al. [17] have analyzed five different CNN for fashion image classification to improve the e-commerce applications. Shuning Shen et al. [5] have used Long Short-Term memory network for fashion image classification on Fashion-MNIST benchmark dataset which obtained accuracy of 88.26%. Greeshma K V et al. [6] have explored Hyper-Parameter Optimization (HPO) methods and regularization techniques with deep neural networks for apparel image classification. To characterize the fashion articles in the fashion-MNIST dataset, the authors created three convolutional neural networks. On the benchmark dataset, the method achieves fantastic results.

EnsNet is a new model suggested by Daiki Hirata et al. [7] which is made up of one base CNN and several Fully Connected SubNetworks (FCSNs). Since deep learning necessitates a large amount of data, the insufficiency of photo testing can be exacerbated by various techniques like as rotation, cropping, shifting, and flipping. Kota Hara et al. [8] have proposed a pose-dependent prior model for automatic selection of human body joints and used deep convolution neural network for cloth detection. They have conducted experiments on Fashionista and PaperDoll datasets for image classification.

The rest of the paper is organized as follows: Sect. 2 begins with a comprehensive examination of relevant literature. Section 3 provides an outline of the suggested methodology. Section 4 describes the experimental setting as well as the results of four model performance evaluations using the fashion MNIST dataset. Section 5 describes the conclusion and future work.

2 Related Works

Mustafa Amer Obaid et al. [16] have discussed the fashion image classification using pre-convoluted neural networks. They have conducted experiments on Fashion MNIST. They have achieved 94% accuracy in using pre-convoluted neural networks. The Hierarchical Convolutional Neural Networks model for the fashion and apparel categorization task was presented by Seo Yian et al. [17]. The model achieved better classification results by using VGGNet. They tested their hypothesis on the Fashion MNIST dataset. EnsNET is a new model presented by Daiki Hirata et al. [7] which is made up of one base CNN and many Fully Connected SubNetworks (FCSN). For fashion image categorization, Greeshma K V et al. [18] have classified the fashion products using Histogram of Oriented Gradients (HOG) features with multiclass Support Vector Machine (SVM) classifier. They have conducted experiments on the Fashion MNIST dataset.

Mohammed Kayed et al. [19] have applied CNN LeNet-5 architecture for the classification of garments. They tested their hypothesis on the Fashion MNIST dataset. The accuracy of the CNN LeNet-5 architecture was 98%. For the fashion-MNIST picture categorization, Khatereh Meshkini et al. [20] have employed different CNN activation functions. They tested their hypothesis on the Fashion MNIST dataset. Zhang et al. [21] investigated LeNet-5, AlexNet, VGG-16, and ResNet, four CNN models. These are utilised in the categorising of fashion images. ResNet has the highest validation accuracy of the four models. They tested their hypothesis on the Fashion MNIST dataset.

3 Proposed System

General system architecture of CNN [9] is shown in Fig. 1. Modified CNN architecture of proposed system is shown in Fig. 2. Input image has been preprocessed using resizing and normalization. Then CNN framework is employed for fashion image classification task.

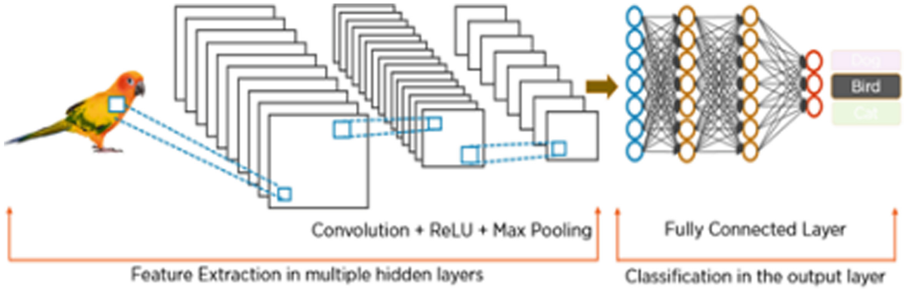


Fig. 1. General system architecture of convolution neural network

3.1 Model Definition

The convolution operations in between two-dimensional picture $\mathrm{I_n}$ and also a two-dimensional kernel k is,

$$S(x, y) = (k * \mathrm{I_n})(x, y) = \sum\nolimits_{\alpha} \sum\nolimits_{\beta} \mathrm{I_n}(\alpha, \beta) k(x - \alpha, y - \beta) \tag{1}$$

The equation becomes with a kernel size of 3×3

$$S(x, y) = (k * \mathrm{I_n})(x, y) = \sum\nolimits_{\alpha=1}^{3} \sum\nolimits_{\beta=1}^{3} \mathrm{I_n}(\alpha, \beta) k(x - \alpha, y - \beta) \tag{2}$$

ReLU is one of the activation functions in CNN. The output is $f(x) = \max(0, x)$. ReLU is computed using

$$f(x) = \begin{cases} x, \textit{ if } x > 0 \\ 0, \textit{ if } x < 0 \end{cases} \tag{3}$$

The pooling layer also known to as the down sampling operation that reduces the dimensionality of the feature map. The parameter used in the max pooling layer is pool size which specifies the 2×2 filter.

The flattening layer is used to convert two-dimensional arrays into one-dimensional array of input so that the flattened input will be given to the final classification layer that is the fully connected layer.

The dropout layer randomly dropping some connections produces several thinned network architecture and one representative network is selected with small weights.

The fully connected layer is used for classification and it is mostly used at the end of the network. Finally, the classification task is performed to generate the output of 10 classes probabilities.

Softmax activation function was used in the neural network for multi-class classification of fashion images.

$$\text{Softmax} f(X_i) = \frac{exp^{X_i}}{\sum_{j=1}^{n} exp^{X_j}} \tag{4}$$

In general, it is the ratio of the exponential of a specific input value to the sum of exponential values of all input values.

3.2 Output Shapes

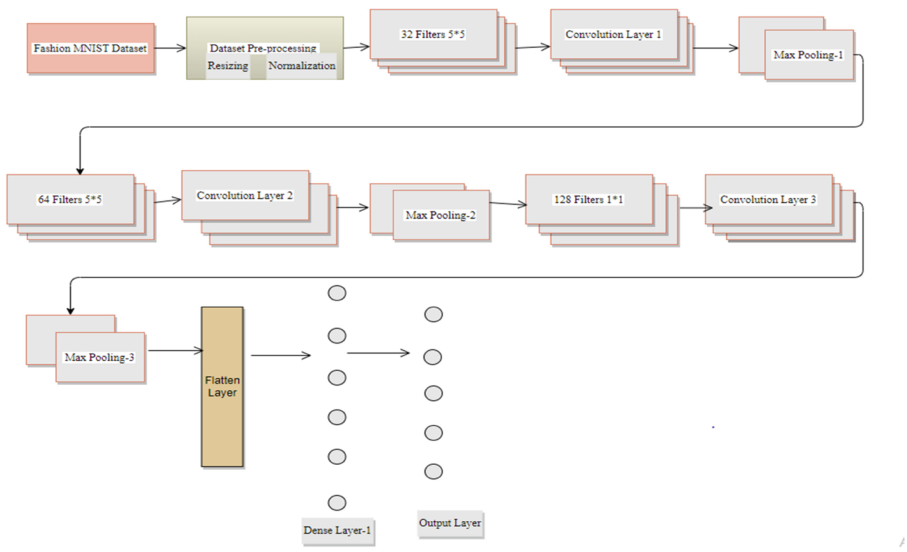


Fig. 2. CNN architecture of proposed system

The suggested CNN model comprises of three convolution layers with 32, 64, and 128 filters, respectively. 5 * 5 kernel sizes are used in Layers 1 and 2, whereas 3 * 3 kernel sizes are used in Layer 3. In ReLU Activation, all three layers are enabled. Following layer 1 and layer 2, the max pooling is handled as 2 * 2 kernel sizes, while layer 3 is applied as 1 * 1 kernel sizes. There is a 0.5% dropout after layer 1 and 2 are applied; this dropout is obtained from layers 1 and 2 to minimize overfitting. Finally, the fully connected neural network is used for classification task, which is then assessed by the softmax activation function to generate the output of 10 classes probabilities (Table 1).

Table 1. Fashion image classification using CNN

Layer	Kernel size	Activation	Output shape	Parameter
Conv2d	5 × 5	ReLU	(None, 24, 24, 32)	812
MP2d	2 × 2	-	(None, 12, 12, 32)	0
Conv2d_1	5 × 5	ReLU	(None, 8, 8, 64)	51264
MP2d_1	2 × 2	-	(None, 4, 4, 64)	0
Conv2d_2	1 × 1	ReLU	(None, 4, 4, 128)	8320
MP2d_2	2 × 2	-	(None, 2, 2, 64)	0
Flatten_1	-	-	(None, 512)	0
Dense_1		ReLU	(None, 10)	5130
Dropout_1	-	-	(None, 10)	0
Dense_2		ReLU	(None, 10)	110
Dropout_2	-	-	(None, 10)	0
Fully connected	-	Softmax	(None, 10)	110

3.3 Performance Evaluation

The proposed model performance is evaluated using the Loss and accuracy metrics.

3.3.1 Loss Function

Loss function calculates the difference between the expected value and ground truth value. Widely used loss function for deep neural network is cross entropy. It is defined as

$$Cross - Entrophy = -\sum_{i=1}^{n}\sum_{j=1}^{m} t_{i,j}\log(p_{i.j}) \tag{5}$$

where $t_{i,j}$ represents the true value that is, If instance i belongs to class j, it is 1; otherwise, it is 0.and The probability score of anticipated class j for relevant instance i is represented by $p_{i.j}$.

3.3.2 Accuracy

Accuracy is a common statistic for evaluating model performance. Accuracy metrics are used to determine how effectively the classifier predicts the output classes and to measure the classification model's efficacy. Accuracy is usually inversely related to error. It is defined as

$$Accuracy = \frac{\mu + \gamma}{\mu + \gamma + \vartheta + \varphi} \tag{6}$$

Here, μ, $\boldsymbol{\varphi}$ denotes the true occurrence for number of true and false predictions respectively and $\boldsymbol{\vartheta}$, $\boldsymbol{\gamma}$ denotes the false occurrence for the number of true and false predictions respectively.

4 Experiments and Discussion

4.1 Dataset

In this paper, experiments were carried out using the Fashion MNIST dataset of zalando's article [10–13]. Dataset information is shown in Table 2. The Fashion MNIST dataset contains images from a training dataset of 60,000 samples as well as a test set of 10,000 samples. The size of the grayscale images is 28 $\times$ 28 pixels associated with a 10 class labels.

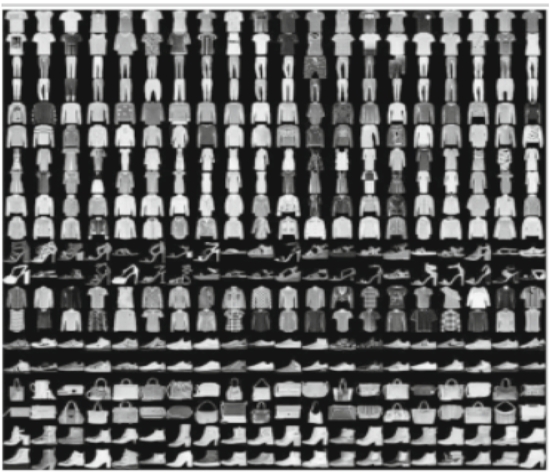
Table 2. Dataset information

Name	Explanation	#Example	Size
train-image-idx3-ubyte.gz	Training sample images	60,000	25 MBytes
train-labels-idx1-ubyte.gz	Training class labels	60,000	140 Bytes
t10k-images-idx3-ubyte.gz	Testing sample images	10,000	4.2 MBytes
t10k-labels-idx1-ubyte.gz	Testing class labels	10,000	92 Bytes

Labels

As indicated in Table 3, each training and testing sample is labelled with one of the class labels.

Table 3. Class labels in benchmark fashion MNIST dataset

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

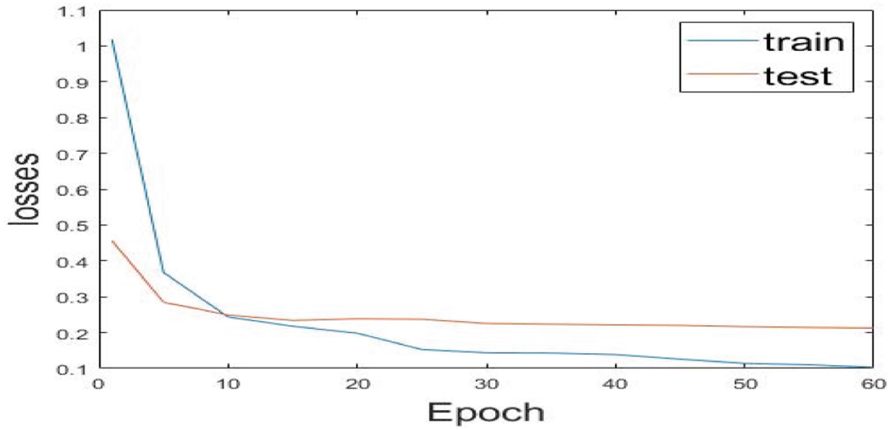


Fig. 3. Loss in ADAM model per epoch

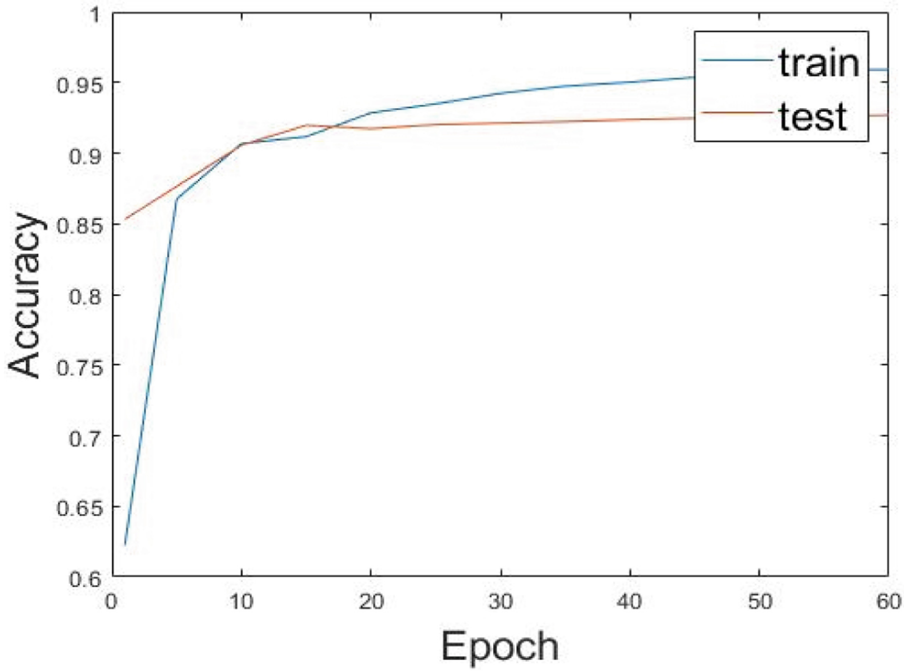


Fig. 4. Accuracy in ADAM model per epoch

4.2 Results

Figure 3 depicts the training and testing losses of the ADAM optimizer model each epoch. Figure 4 depicts the ADAM optimizer model's training and testing accuracy per epoch. Figure 5 depicts the training and testing losses of the RMSProp optimizer model each epoch. Figure 6 depicts the RMSProp optimizer model's training and testing accuracy per epoch. Table 4 compares the classification results of the Fashion MNIST dataset to those found in the literature. Table 5 displays the loss and accuracy on the Adam model per epoch. Table 6 displays the loss and accuracy on the RMSProp model per epoch.

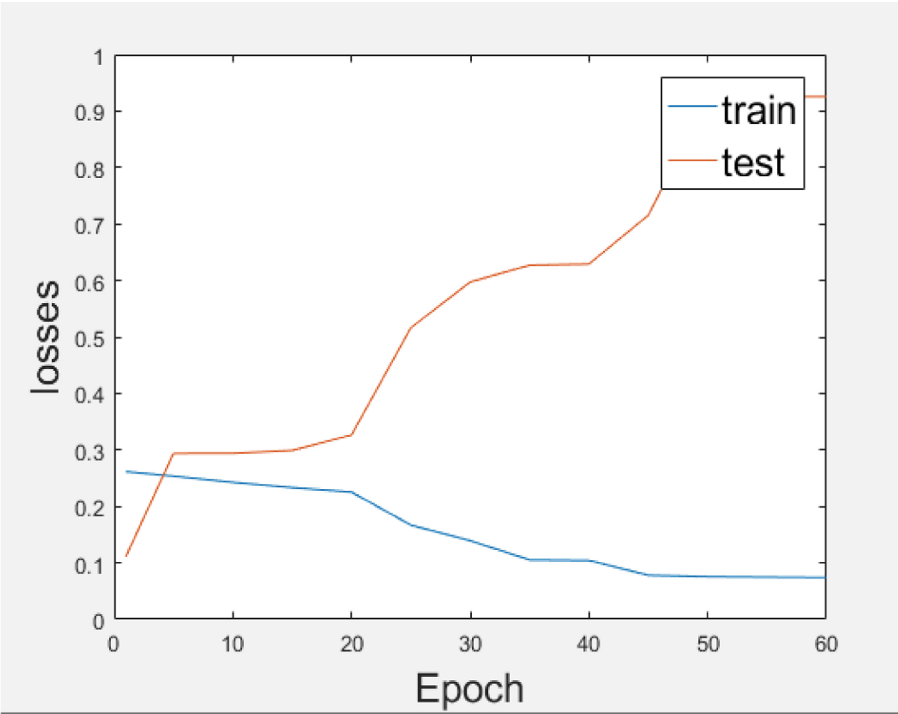


Fig. 5. Loss in RMSProp model per epoch

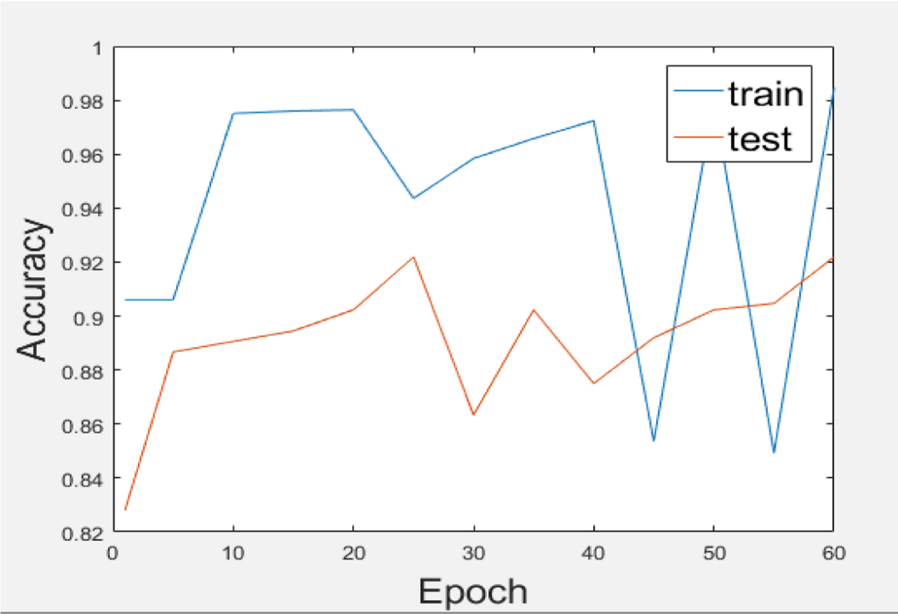


Fig. 6. Loss in RMSProp model per epoch

Table 4. Fashion MNIST dataset classification outcomes comparison with literatures

Model	Validation accuracy in %
3 layer NN [15]	87.23%
SVC with rbf kernel [3]	89.70%
Evolutionary deep learning framework [4]	90.60%
CNN using softmax [14]	91.86%
CNN2 with batch normalization [14]	92.22%
CNN with 3Conv + pooling + 2FC + dropout	**92.68%**

Table 5. Loss and accuracy on Adam model per epoch

Epoch	ADAM optimizer			
	Training		Testing	
	Loss	Acc	Loss	Acc
1	1.0178	0.6224	0.4563	0.8532
5	0.3675	0.8675	0.2846	0.8764
10	0.2436	0.9067	0.2489	0.9056
15	0.2176	0.9117	0.2340	0.9198
20	0.1983	0.9285	0.2386	0.9173
25	0.1528	0.9349	0.2373	0.9202
30	0.1437	0.9423	0.2254	0.9213
35	0.1428	0.9475	0.2232	0.9223
40	0.1386	0.9502	0.2214	0.9237
45	0.1256	0.9537	0.2198	0.9247
50	0.1143	0.9558	0.2164	0.9253
55	0.1105	0.9589	0.2142	0.9263
60	0.1031	0.9592	0.2126	0.9268

Table 6. Loss and accuracy on RMSProp model per epoch

Epoch	RMSProp optimizer			
	Training		Testing	
	Loss	Acc	Loss	Acc
1	0.2614	0.9060	0.1113	0.8281
5	0.2535	0.9060	0.2939	0.8867
10	0.2426	0.9751	0.2942	0.8906
15	0.2332	0.9760	0.2992	0.8945
20	0.2252	0.9764	0.3264	0.9023
25	0.1669	0.9436	0.5165	0.9219
30	0.1392	0.9584	0.5974	0.8633
35	0.1053	0.9658	0.6272	0.9023
40	0.1043	0.9724	0.6292	0.8750
45	0.0782	0.8536	0.7156	0.8920
50	0.0757	0.9794	0.9224	0.9023
55	0.0748	0.8493	0.9248	0.9047
60	0.0740	0.9846	0.9257	0.9219

5 Conclusion

In this paper, proposed system uses three convolution neural network layers. In the suggested approach, the regular CNN's convolution layer is enhanced by expanding the number of layers to three layers with max pooling for fashion image classification. In the future, we will try to use the new benchmark clothing dataset to decide classification algorithms and alternative convolution architectures. Other CNN models can be used to apply the suggested technique in the future, and the work has been expanded to incorporate real-world online clothing imagery.

References

1. Zhang, Y., Zhang, E., Chen, W.: Deep neural network for halftone image classification based on sparse auto-encoder. Eng. Appl. Artif. Intell. **50**, 245–255 (2016). https://doi.org/10.1016/j.engappai.2016.01.032
2. Ferreira, A., Giraldi, G.: Convolutional neural network approaches to granite tiles classification. Expert Syst. Appl. **84**, 1–11 (2017). https://doi.org/10.1016/j.eswa.2017.04.053
3. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
4. Dufourq, E., Bassett, B.A.: Eden: Evolutionary deep networks for efficient machine learning. In: Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech), pp. 110–115. IEEE (2017). https://doi.org/10.1109/RoboMech.2017.8261132
5. Shen, S.: Image classification of Fashion-MNIST dataset using long short-term memory networks. Research School of Computer Science (2018)
6. Greeshma, K.V., Sreekumar, K.: Hyperparameter optimization and regularization on Fashion-MNIST classification. Int. J. Recent Technol. Eng. (IJRTE). **8**(2), 3713–3719 (2019). https://doi.org/10.35940/ijrte.B3092.078219
7. Hirata, D., Takahashi, N.: Ensemble learning in CNN augmented with fully connected subnetworks. arXiv preprint arXiv:2003.08562 (2020)
8. Hara, K., Jagadeesh, V., Piramuthu, R.: Fashion apparel detection: the role of deep convolutional neural network and pose-dependent priors. In: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1–9. IEEE (2016). https://doi.org/10.1109/WACV.2016.7477611
9. Sewak, M., Sahay, S.K., Rathore, H.: An overview of deep learning architecture of deep neural networks and autoencoders. J. Comput. Theor. Nanosci. **17**(1), 182–188 (2020). https://doi.org/10.1166/jctn.2020.8648
10. Leithardt, V.: Classifying garments from fashion-MNIST dataset through CNNs. Adv. Sci. Technol. Eng. Syst. J. **6**(1), 989–994 (2021). https://doi.org/10.25046/aj0601109
11. Teow, M.Y.: Experimenting deep convolutional visual feature learning using compositional subspace representation and fashion-MNIST. In: 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), pp. 1–6. IEEE (2020). https://doi.org/10.1109/IICAIET49801.2020.9257819
12. Pathak, A.R., Pandey, M., Rautaray, S.: Application of deep learning for object detection. Proc. Comput. Sci. **132**, 1706–1717 (2018). https://doi.org/10.1016/j.procs.2018.05.144
13. Gnatushenko, V., Dorosh, N., Fenenko, T.: Fashion MNIST image recognition by deep learning methods. Appl. Ques. Math. Model. **4**(1), 78–85 (2021)
14. Bhatnagar, S., Ghosal, D., Kolekar, M.H.: Classification of fashion article images using convolutional neural networks. In: Fourth International Conference on Image Information Processing (ICIIP), pp. 1–6. IEEE (2017). https://doi.org/10.1109/ICIIP.2017.8313740

15. Zhang, K.: LSTM: An Image Classification Model Based on Fashion-MNIST Dataset (2017)
16. Obaid, M.A., Jasim, W.M.: Pre-convoluted neural networks for fashion classification. Bull. Elect. Eng. Inform. **10**(2), 750–758 (2021). https://doi.org/10.11591/eei.v10i2.2750
17. Seo, Y., Shin, K.S.: Hierarchical convolutional neural networks for fashion image classification. Expert Syst. Appl. **116**, 328–339 (2019). https://doi.org/10.1016/j.eswa.2018.09.022
18. Greeshma, K.V., Sreekumar, K.: Fashion-MNIST classification based on HOG feature descriptor using SVM. Int. J. Innov. Technol. Explor. Eng. **8**(5), 960–962 (2019). https://doi.org/10.35940/ijrte.B3092.078219
19. Kayed, M., Anter, A., Mohamed, H.: Classification of garments from fashion mnist dataset using cnn lenet-5 architecture. In: 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), pp. 238–243 (2020). https://doi.org/10.1109/ITCE48509.2020.9047776
20. Meshkini, K., Platos, J., Ghassemain, H.: An analysis of convolutional neural network for fashion images classification (Fashion-MNIST). In: Kovalev, S., Tarassov, V., Snasel, V., Sukhanov, A. (eds.) Proceedings of the Fourth International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'19). AISC, vol. 1156, pp. 85–95. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-50097-9_10
21. A MNIST-like fashion product database. https://github.com/zalandoresearch/fashion-mnist