Indonesian Journal of Electrical Engineering and Computer Science
Vol. 22, No. 1, April 2021, pp. 563~570
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v22.i1.pp563-570

e-SimNet: a visual similar product recommender system for E-commerce

Ssvr Kumar Addagarla, Anthoniraj Amalanathan
School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India

Article Info

Article history:

Received Jan 17, 2021
Revised Mar 4, 2021
Accepted Mar 11, 2021

Keywords:

ANNOY
e-SimNet
Image similarity
SqueezeNet
Visual recommendations

ABSTRACT

Visual similarity recommendations have an immense role in E-commerce portals. Fetching the appropriate similar products and suggesting to the buyers based on the product image's visual features is complex. Here in our research, we presented an efficient E-commerce similar product network model (e-SimNet) for visually similar recommendations. To achieve our objective, we have performed image feature extraction and generating embeddings using deep learning techniques and built an Index tree using the approximate nearest neighbor oh yeah (ANNOY) algorithm. Further, we have fetches top-N the near similar items using distance measure. We have benchmarked our model in terms of accuracy, error rate, and results show that better than other state-of-the-art approaches with 96.22% of accuracy.

Corresponding Author:

Anthoniraj Amalanathan
School of Computer Science and Engineering
Vellore Institute of Technology (VIT)
Vellore, Tamil Nadu, India
Email: aanthoniraj@vit.ac.in

1. INTRODUCTION

With the consistent growth in the E-commerce market, especially in Indian and other Asian countries, there is a tremendous need for visual recommendations. In this pandemic COVID-19 situation, most users are transformed their buying needs from traditional shopping to online shopping ranging from essential commodities to branded goods [1-3]. Traditionally, most E-commerce recommendation systems are utilizing machine learning-based techniques, and few popular E-commerce portals such as Amazon, Flipkart, and Myntra are utilizing deep learning-based approaches approximations. Most machine learning-based recommender systems have limitations due to the lack of proper visual descriptions using text-based searches by the buyers. Many researchers have proposed image-based supervised and unsupervised recommendations using dimensionality reduction techniques such as principal component analysis (PCA), Singular value decomposition (SVD) [4-7]. Later, computed the clustering-based dimensionally transformed data points and calculated the distance measure to fetch similar products. These models are analyzed using the performance measures such as accuracy, error rate, silhouette coefficient, and Calinski-Harabasz (CH) score [8]. These methods have some limitations. Significantly if the images' orientations change after the training, we get inappropriate recommendations.

Here we have proposed an efficient deep learning based visual similar recommender system for E-commerce. Our approach has applied data augmentation and extracted the features by using based on convolutional neural networks (CNN) model [9]. CNN based models have proven in many fields with better accuracy results compared with traditional machine learning approaches. The convolution process has a

better understanding of the image level pixels to be considered in terms of edges, colors, patterns, texture. Later, we have built the binary forest of trees using an approximate nearest neighbor algorithm for better approximation results. Experimentation and results are carried out for our model by benchmarking with other state-of-the-art approaches using deep learning techniques. The proposed outcome for our visual recommender system, as shown in Figure 1.

Figure 1. The proposed outcome for our visual recommendations

Image retrieval based on the color histograms using local feature regions was proposed by Wang *et al*. and computed a similar distance measure for the color images to retrieve the images. The authors have analyzed the model using accuracy and compared it with the existing methods [10]. A bag of word approach using color SIFT is proposed and compared with the Hessian affine and SIFT based methods and tested on holiday and UKB benchmark datasets [11]. A new color descriptor of image using the Census transform histogram is proposed for the global shape information and intensity values [12]. A CNN based online image search was proposed by considering small samples from the SUN 397 dataset and achieved the classification accuracy of 51% for pseudo labeled data [13]. A deep hash-based image search using learning-deep hash and supervised deep hash techniques are proposed [14]. This approach was used to extract the image embeddings and further applied the deep hash techniques to retrieve the near similar images.

A framework for image retrieval on a distributed cloud platform using MapReduce has been developed. Here in the process, manual feature extraction has done using SURF. An unsupervised image index was then built using VP Trees for computing distance measure and fetching the images [15]. Gai *et al*. [16] have proposed an associative model using KD-trees and VP-Trees without compromising the memory defects for fetching the images from the database. Rani Saritha *et al*. [17] use deep belief networks to extract the features and classification task. Further applied local sensitive hashing-based techniques to retrieve the images. A limited labeled and unlabeled image were preprocessed using annotation promotion. Then feature extraction was done through CNN and computed the accuracy for a proposed unimodal visual approach for image retrieval. An efficient Feature extraction using Stacked convolutions and residual network for capsual network was proposed for content-based image retrieval (CBIR) [18]. Deep ranking recommendations using Siamese network with VGG19 backbone architecture and angular distance metric used on Fashion MNIST, CIFAR10, exact street2shop datasets are utilized to compute accuracy [19]. Several CBIR [20-22] methods utilize the deep learning approaches using VGGNet, generative adversarial networks (GAN), deep convolutional GAN, infoGAN are used to fetch similar images and also tested on the various big data platforms [23, 24]. Feature extraction using the local binary pattern (LBP) and deep belief networks were proposed for CBIR on multiple datasets [25]. The rest of the paper consists of the 3 sections. In the section 2, proposed research method and approaches are explained, in the section 3, experimentation and results are described and finally presented summarized conclusion about our model in section 4.

2. RESEARCH METHOD

Image-based similar product recommender system is an extensive task to extract the images' visual features and further compute the similarity distance between the input image and a bunch of images. To achieve this process, we have proposed an E-commerce similar image network (e-SimNet) model for visual similar product recommendations, as shown in Figure 2. Our e-SimNet approach initially chosen the convolutional neural network (CNN) approach for the feature learning process and the extraction of image embeddings before the model classification stage. Here in our approach, we have adopted CNN based SqeezeNet [26] Architecture for the feature engineering process to further extract image embeddings from the trained model. SqueezeNet Architecture was designed with two new approaches by considering lower parameters compare with other deep learning architectures. To increase the e-SimNet model's performance, we have adopted the following 3 strategies for the feature extraction process and to speed up the model training.

- Replace 3 × 3 filters with 1×1 filter as 1×1 filter has nine times fewer parameters than a 3×3 filter.
- Consider the lower input channels to 3×3 filters to prune the network.

$$Parameters = (number\ of\ input\ channels) \times (number\ of\ filters) \times (3 \times 3)$$

- For better accuracy, performing the downsampling late in the model training results the higher feature maps.

In Figure 3, Shown the fire module consisting of the 1×1 squeeze filter and a combination of 1×1 and 3×3 expand filters. We have applied the activation function on these convolution operations then concatenates the output from expand layers to given as input to the next fire module. A total of 12,48,424 parameters are reduce to 4,21,098 by utilizing fire module as shown in Table 1.

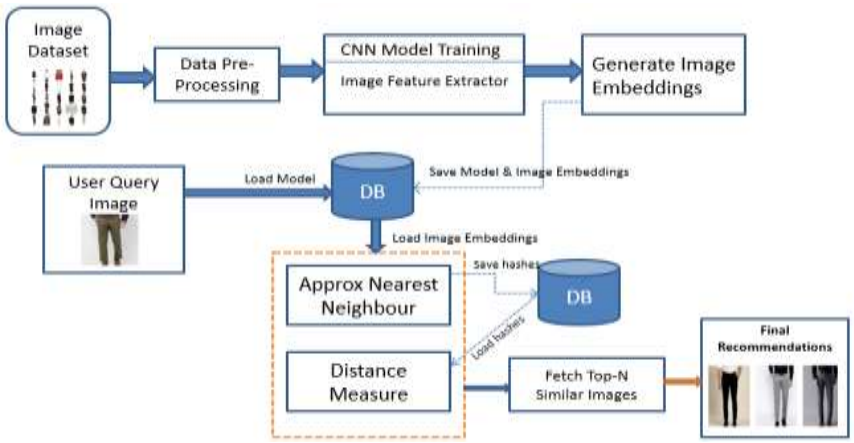

Figure 2. Proposed eSimNet model for visual recommender system

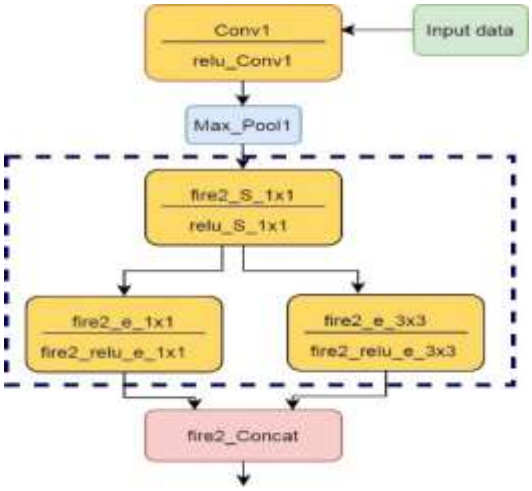

Figure 3. Fire module process in the SqueezeNet architecture

Table 1. Our Custom SqueezeNet Architecture Specifications

Layer Name/Type	Output size	Filter size / stride	Depth	Parameters before Fire module	Parameters after Fire module
Input image	224×224×3				
Conv1	111×111×96	7×7/2(×96)	1	14,208	14,208
Maxpool1	55×55×96	3×3/2	0		
Fire2	55×55×128		2	11,920	5,746
Fire3	55×55×128		2	12,432	6,258
Fire4	55×55×256		2	45,344	20,646
Maxpool4	27×27×256	3×3/2	0		
Fire5	27×27×256		2	49,440	24,742
Fire6	27×27×384		2	104,880	44,700
Fire7	27×27×384		2	111,024	46,236
Fire8	27×27×512		2	188,992	77,581
Maxpool8	13×12×512	3×3/2	0		
Fire9	13×13×512		2	197,184	77,581
Conv10	13×13×1000	1×1/1 (×1000)	1	513,000	103,400
Avgpool10	1×1×1000	13×13/1	0		
	Total Parameters			12,48,424	421,098

2.1. Feature extraction

The feature extraction carries out in the convolution layer, which learns the relationship between the pixels to understand the shape, color, and texture information. It takes the initial input image data $(\mathrm{h} \times w \times d)$ and each feature map is multiplied with a kernel or filter$(\mathrm{f_h} \times \mathrm{f_w} \times \mathrm{d})$. Batch normalization use to normalize the output values from 0 to 1 by computing the mean, variance, normalize, scale and shift as shown:

Algorithm of Batch Normalization using Mini Batches

$Input: Values\ of\ x\ over\ mini\ batch\ B = \{x_1, \ldots . x_k\},\ Parameters\ to\ be\ learned\ Y, \beta$

$Output: \{\mathrm{y_i} = \mathrm{BN}_{\gamma,\beta(\mathrm{x_i})}\}$

1. $compute\ the\ mean\ \mu_B \leftarrow \frac{1}{k}\sum \mathrm{x_i}$
2. $compute\ variance\ \mu_B \leftarrow \frac{1}{k}\sum_{i=1}^{k}(\mathrm{x_i} - \mu_B)^2$
3. $perform\ normalize\ \hat{\mathrm{x}}_i \leftarrow \frac{\mathrm{x_i} - \mu B}{\sqrt{\sigma_B^2 + \epsilon}}$
4. $perform\ scale\ and\ Shift\ \mathrm{y_i} \leftarrow \gamma \hat{\mathrm{x}}_i + \beta = \mathrm{BN}_{\gamma,\beta(\mathrm{x_i})}$

As we are using a multi-labeled dataset and achieving the non-linearity while training the model, we apply the leaky rectified linear unit (ReLu) activation function. The output is $f(\mathrm{x})$ = max (αx, x) where α = 0.001, which considers the small slope of non-negative values instead of making zero to all the non-negatives as performed by the ReLu function as max (0, x). Due to this dying ReLu problem, we utilized leaky ReLu, which will not miss the necessary pixel values when it performs the backpropagation. Later, to perform the network's downsampling, we used the pooling operation to extract and retain the maximum importance features, as shown in Figure 4.

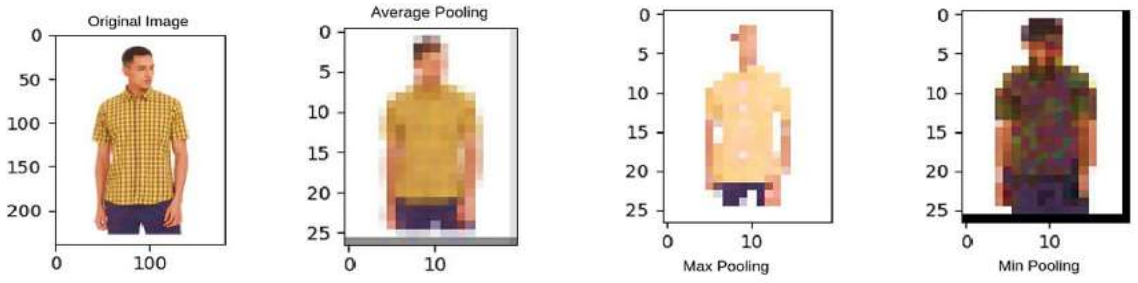

Figure 4. Performance of the various pooling operations on the sample image

2.2. Approximate nearest neighbor

Here we have adopted an approximate nearest neighbor oh yeah (ANNOY) [27] is a method for a faster nearest neighbor search that builds the index based on the random projections. ANNOY uses several forests of trees to compute similar items with better approximation. Mostly the ANNOY loads with mmap, a

memory-mapped method for faster access across the distributed platforms. The following algorithm is for ANNOY computation for finding the Top-N similar images shown below.

ANNOY Algorithm for Similar Product Approximation
Input: Extracted Image Features and k
Output: Top-N Approximate Similar Images
Pre-processing:

1. Split the data points on random hyper plane and construct the forest of binary trees.
2. Perform step 1 for k times and save the index

Querying:

1. Load the ANNOY Index
2. Using priority queue to search all the trees until we have found n items
3. Take union and retain only unique items
4. Calculate Euclidean measure for nearest items
5. Fetch the nearest items based on short distance
6. Recommends the Top-N products

Further, we have computed the Euclidean distance measure of $d(i,s)$ where i and s are points in the euclidean distance space using the following in (1) to compute the near similar items from the search and returns the Top-N similar product recommendation.

$$d(i,s) = \sqrt{\sum_{k=0}^{n}(s_k - i_k)^2} \quad (1)$$

3. EXPERIMENTATION AND RESULTS

In this section, we have carried out experimentation on various E-commerce products dataset. A total of 30K images are collected from the Kaggle fashion dataset and deep fashion image dataset. We have preprocessed the labeled data with 10 categories consist of watches, belts, jeans, trousers, and women-kurta, and each category consists of 3000 images. We have utilized a desktop pc with Core i7 processor with 16GB of RAM and RTX 2070 GPU with 8GB and PyTorch programming environments configured for experimentation. Further, SqueezeNet classification model performance is analyzed through various performance measures as following (2-6).

$$\text{Precision} = \frac{True\ positive}{True\ Positive + False\ Positive} \quad (2)$$

$$\text{Recall (True positive rate)} = \frac{True\ positive}{True\ Positive + False\ Negative} \quad (3)$$

$$\text{Accuracy} = \frac{True\ Negative + True\ positive}{True\ Negative + False\ Positive + True\ Positive + False\ Negative} \quad (4)$$

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

$$\text{Error Rate} = 1 - Accuracy \quad (6)$$

To understand and interpret the features extraction process, we have visualized the initial convolution process, activation feature maps using leaky ReLu, and final convolution based on the image from the E-commerce product dataset. The visualizations, as shown in Figure 5, presented various performance measures of the e-SimNet model in Table 2.

Table 2. Comparison of SqueezeNet model performance

Model Name	Accuracy	Top-5 Accuracy	Training Loss	Validation Loss	Error Rate
vgg_19_bn	0.7925	0.9610	0.6341	4.4162	0.2075
vgg_16_bn	0.7142	0.9422	0.7519	20.9078	0.2858
densenet121	0.7334	0.9225	0.7324	99.8531	0.2666
efficientNet	0.9347	0.9836	0.9528	0.7495	0.0653
resnet18	0.8020	0.9799	0.8302	12.2099	0.1980
resenet152	0.9475	0.9984	0.3047	13.7449	0.0525
SqueezeNet	0.9622	1.0000	0.7554	0.1126	0.0378

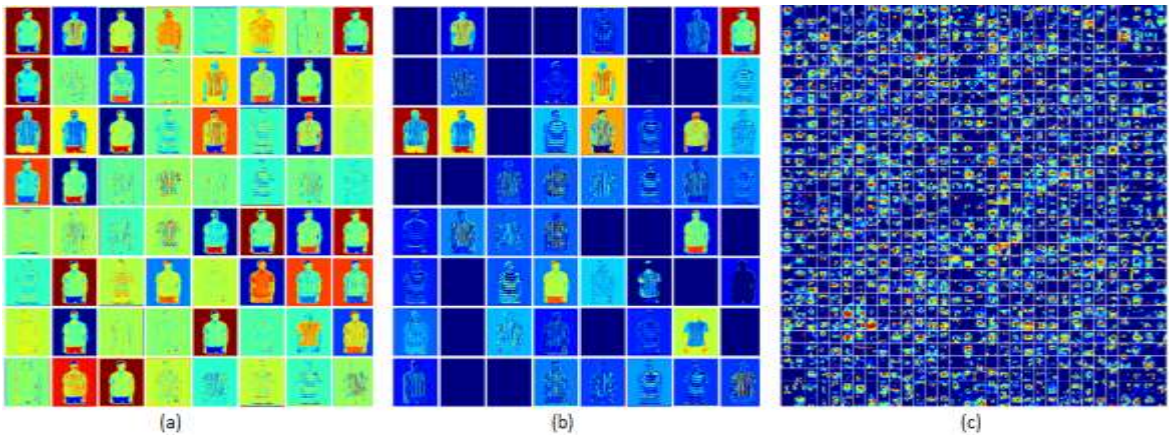

Figure 5. Presents various visualization for (a) initial convolution, (b) leaky ReLu, and (c) final convolution

Later, we have computed the ANNOY for an approximate nearest neighbor search to get the final Top-N recommendations. Table 3, shows the performance comparison concerning index build time from the image embeddings, query time to search the items from the forest of binary trees using a priority queue, and accuracy to fetch the appropriate recommendations. Finally, we have presented the Top-5 similar visual recommendations for our e-SimNet model as shown in Figure 6.

Table 3. Comparison of approximate nearest neighbors

ANN Algorithm	Query Time in Seconds	Average Accuracy	Index Build Time in Seconds
LSHF	0.00735	0.536	0.7450
FLANN	0.00027	0.561	0.1954
ANNOY	0.00287	1.0	16.3237

Figure 6. Final Top-5 recommendations from the e-SimNet model

4. CONCLUSION

This paper presented an e-SimNet visual recommender system for better and accurate similar product recommendations for E-commerce products. We have developed our model by utilizing the deep learning techniques and approximation nearest neighbors for fetching out the Top-N recommendations. As

observed in the model training, Squeezenet is outperformed compared with other popular models such as ResNet, VGG, EfficientNet and DenseNet architectures with 96.2% accuracy and 0.0378% of error rate. Fetching for Top-N recommendations using approximation methods such as ANNOY, LSH, and FLAN is computed. Whereas in terms of index-built time and query time, LSH and FLAN are better than ANNOY, but in terms of accuracy, ANNOY gives the best results for our visual recommender system.

REFERENCES

[1] I. Y. Wulansaria and N. B. Parwantob, "Asian E-Commerce Engages Global Trade Openness: The Role of Information and Communications Technology, Social, and Security Indicators," *Int. J. Innov. Creat. Chang.*, vol. 11, pp. 110-136, 2020.

[2] A. Bhatti, H. Akram, H. M. Basit, A. U. Khan, S. M. Raza, and M. B. Naqvi, "E-commerce trends during COVID-19 Pandemic," *Int. J. Futur. Gener. Commun. Netw.*, vol. 13, no. 2, pp. 1449-1452, 2020.

[3] Statista, "eCommerce-Asia | Statista Market Forecast," 2020. [Online]. Available: https://www.statista.com/outlook/243/101/ecommerce/asia. [Accessed: 02-Dec-2020].

[4] H. Kim, S. R. Sohn, and J. Kim, "Revisiting Gist-PCA Hashing for Near Duplicate Image Detection," *J. Signal Process. Syst.*, vol. 91, no. 6, pp. 575-586, 2019.

[5] I. E. Kaya, A. Ç. Pehlivanl\i, E. G. Sekizkarde\cs, and T. Ibrikci, "PCA based clustering for brain tumor segmentation of T1w MRI images," *Comput. Methods Programs Biomed.*, vol. 140, pp. 19-28, 2017, doi: 10.1016/j.cmpb.2016.11.011.

[6] M. Mateen, J. Wen, S. Song, Z. Huang, and others, "Fundus image classification using VGG-19 architecture with PCA and SVD," *Symmetry (Basel).*, vol. 11, no. 1, pp. 1-12, 2019, doi: 10.3390/sym11010001.

[7] S. K. Addagarla and A. Amalanathan, "Probabilistic Unsupervised Machine Learning Approach for a Similar Image Recommender System for E-Commerce," *Symmetry (Basel).*, vol. 12, no. 11, pp. 1-18, 2020, doi: 10.3390/sym12111783.

[8] V. Tyagi, "Similarity Measures and Performance Evaluation," in *Content-Based Image Retrieval*, Springer, 2017, pp. 63-83.

[9] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, pp. 1-87, 2019, doi: 10.1007/s10462-020-09825-6.

[10] X. Y. Wang, J.-F. Wu, and H.-Y. Yang, "Robust image retrieval based on color histogram of local feature regions," *Multimed. Tools Appl.*, vol. 49, no. 2, pp. 323-345, 2010.

[11] C. Wengert, M. Douze, and H. Jégou, "Bag-of-colors for improved image search," in *Proceedings of the 19th ACM international conference on Multimedia*, 2011, pp. 1437-1440, doi: 10.1145/2072298.2072034.

[12] W.-T. Chu and C.-H. Chen, "Color CENTRIST: a color descriptor for scene categorization," in *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, 2012, pp. 1-8, doi: 10.1145/2324796.2324837.

[13] M. Kolá\vr, M. Hradiš, and P. Zemč\'\ik, "Deep learning on small datasets using online image search," in *Proceedings of the 32nd Spring Conference on Computer Graphics*, 2016, pp. 87-93, doi: 10.1145/2948628.2948633.

[14] J. Lu, V. E. Liong, and J. Zhou, "Deep hashing for scalable image search," *IEEE Trans. image Process.*, vol. 26, no. 5, pp. 2352-2367, 2017.

[15] T. D. T. Nguyen and E.-N. Huh, "An efficient similar image search framework for large-scale data on cloud," in *Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication*, 2017, pp. 1-8, pp. 10.1145/3022227.3022291.

[16] V. E. Gai, V. A. Utrobin, N. V Gai, and I. V Polyakov, "Computer simulations of association-based image search mechanisms basing on theory of active perception," *Opt. Mem. Neural Networks*, vol. 26, no. 1, pp. 77–86, 2017.

[17] R. R. Saritha, V. Paul, and P. G. Kumar, "Content based image retrieval using deep learning process," *Cluster Comput.*, vol. 22, no. 2, pp. 4187-4200, 2019, doi: 10.1007/s10586-018-1731-0.

[18] F. Kinli, B. Ozcan, and F. Kiraç, "Fashion Image Retrieval with Capsule Networks," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, doi: 10.1109/ICCVW.2019.00376.

[19] R. Sharma and A. Vishvakarma, "Retrieving Similar E-Commerce Images Using Deep Learning," *arXiv Prepr. arXiv1901.03546*, 2019.

[20] P. Yin and L. Zhang, "Image Recommendation Algorithm Based on Deep Learning," *IEEE Access*, vol. 8, pp. 132799-132807, 2020, doi: 10.1109/ACCESS.2020.3007353.

[21] S. Camalan, *et al.*, "OtoMatch: Content-based eardrum image retrieval using deep learning," *PLoS One*, vol. 15, no. 5, 2020, doi: 10.1371/journal.pone.0232776.

[22] M. K. Alsmadi, "Content-Based Image Retrieval Using Color, Shape and Texture Descriptors and Features," *Arab. J. Sci. Eng.*, pp. 1-14, 2020.

[23] T. D. T. Nguyen and E.-N. Huh, "Joint index and cache technique for improving the effectiveness of a similar image search in big data framework," *J. Intell. Fuzzy Syst.*, vol. 36, no. 6, pp. 1-12, 2019, doi: 10.3233/JIFS-181760.

[24] B. Ay, G. Ayd\in, Z. Koyun, and M. Demir, "A Visual Similarity Recommendation System using Generative Adversarial Networks," in *2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML)*, 2019, pp. 44-48, doi: 10.1109/Deep-ML.2019.00017.

[25] X. Zhang, "Content-Based E-Commerce Image Classification Research," *IEEE Access*, vol. 8, pp. 160213–160220, 2020, doi: 10.1109/ACCESS.2020.3018877.

[26] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size," *arXiv Prepr. arXiv1602.07360*, 2016.
[27] W. Li, *et al.*, "Approximate nearest neighbor search on high dimensional data-experiments, analyses, and improvement," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 8, pp. 1475-1488, 2019, doi: 10.1109/TKDE.2019.2909204.

BIOGRAPHIES OF AUTHORS

Ssvr Kumar Addagarla is completed his Master degree in 2013 and currently PhD Research Scholar at School of Computer Science and Engineering, Vellore Institute of Technology, India. His current research includes Machine learning, Deep learning, Recommender Systems for developing E-commerce and other industrial applications including IoT. He is an active member in professional societies like CSI and ACM.

Anthoniraj Amalanathan is an Associate Professor at Vellore Institute of Technology (VIT) in School of Computer Science and Engineering. He is also a Director of Software Development Centre (SDC) at VIT, Vellore, India. He received his Ph.D. in Computer Science and Engineering from the Vellore Institute of Technology (VIT). His areas of expertise include Semantic Web, Feature Engineering, Text Mining, Machine Learning, and open-source programming. His research interests are in the use of technology in education and developing open-source software that takes into consideration the unique needs of learners. He is currently doing research on Language Modelling and Chatbot to help students in Academic Activities. He has published many International Journal of repute.