Computer Modeling in Engineering & Sciences
DOI:10.32604/cmes.2020.010943
Article

A Hybrid Deep Learning Architecture for the Classification of Superhero Fashion Products: An Application for Medical-Tech Classification

Inzamam Mashood Nasir[1], Muhammad Attique Khan[1,*], Majed Alhaisoni[2], Tanzila Saba[3], Amjad Rehman[3] and Tassawar Iqbal[4]

[1]Department of Computer Science, HITEC University, Taxila, Pakistan
[2]College of Computer Science and Engineering, University of Ha'il, Ha'il, Saudi Arabia
[3]College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia
[4]Department of Computer Science, COMSATS University Islamabad, Wah Campus, Islamabad, Pakistan
*Corresponding Author: Muhammad Attique Khan. Email: attique@ciitwah.edu.pk
Received: 08 April 2020; Accepted: 08 May 2020

Abstract: Comic character detection is becoming an exciting and growing research area in the domain of machine learning. In this regard, recently, many methods are proposed to provide adequate performance. However, most of these methods utilized the custom datasets, containing a few hundred images and fewer classes, to evaluate the performances of their models without comparing it, with some standard datasets. This article takes advantage of utilizing a standard publicly dataset taken from a competition, and proposes a generic data balancing technique for imbalanced dataset to enhance and enable the in-depth training of the CNN. In addition, to classify the superheroes efficiently, a custom 17-layer deep convolutional neural network is also proposed. The computed results achieved overall classification accuracy of 97.9% which is significantly superior to the accuracy of competition's winner.

Keywords: Superheroes; deep convolutional neural network; data augmentation; transfer learning; machine learning

1 Introduction

The graphical comic arts were introduced in the mid of 19th century to explain a story, different characters, events or some particular buildings [1]. These comic arts were initially printed on papers but evolved to the digital characters with the passage of time. At the end of 20th century, these characters were introduced as superheroes in many animated movies and thus the fame of these superheroes increased exponentially [2]. Nowadays, these superheroes are used everywhere, either on comic books, or fashion accessories, school bags or room walls as the sensation of these superheroes is increasing rapidly. With the recent success of machine learning in many fields such as video surveillance [3,4], biometrics [5–7], medical [8,9], agriculture [10–12], social network [13], and few other [14,15], the researchers have drawn their attention towards the understanding and learning the visual features of comic characters to better identify and classify the superheroes. The classification methods simply train on few images before classifying the image into one of the predefined class, while the identification methods extract different features i.e., statistical [16], color [17], geometrical and shape, to identify and locate a character onto

another image [18–20]. This complex learning and empathetic of graphical predictors can help the operative processing, repossession and localization of comic characters. There exist many machine learning techniques to copy the styles of characters [21], avatar creation [22], coloring characters automatically [23] and anime character creation [24].

The DCNN have achieved considerable accomplishments in the field of image processing [25–29] and related domains [30–37]. Many pre-trained CNN models are already proposed, which are trained on large datasets like ImageNet [38], which contains 1000 different classes. AlexNet [39], VGG19 [40] and GoogleNet [41] are few of the famous pre-trained models which are widely used and performed exceptionally well on many domains [42–45]. All of these models are pre-trained on data containing different categories, thus the training data of these model is as complex as their structures [46]. Training these models further with more data and classes will increase the complexity and training time. Therefore, a simple 17-layer deep convolutional neural network is proposed in this article, which trains multiple classifiers to effectively classify the data into most related classes.

Previous techniques had used effective handcrafted features to classify the painting images [47]. Another method was proposed utilizing CNN models to classify the art images. This method initially extracts the deep features and then utilized support vector machine as a classifier to achieve an enhanced classification accuracy [48]. A WikiArt dataset containing 27 art style classes and 1000 artists was used to train the traditional classifiers [49]. CNN classifiers were employed to localize the authorship photographic and illustrative images [50,51]. With the introduction of famous comic dataset Manga109, many researchers employed super resolution classification on this dataset [52–54].

A new custom large-scale dataset was introduced for contemporary artworks having comic images [55]. Comic objects were detected using the concept of general object recognition by detecting four types of comic objects [56]. There are also previous studies for detecting the comic faces [57,58] and comic character detection [59]. Another milestone for comic classification was achieved by utilizing the computational predictors from comic line-segments. The authors also revealed comic classification is fundamentally different from the fine-art classification, so convolutional neural networks are not involved to understand the characteristics of drawing styles in comic lines [60].

Faces in Japanese comics called mangas were identified using Viola-Jones framework [61] and then detected sufficiently [62,63]. The concept of applying the prior techniques for detection and recognition of human faces was proved wrong as it was clearly identified that the size, organ positions and color shades of human are different from the comic characters. Thus an improved and comic face related face detection method was proposed which utilized skin edges and color regions [64]. Another technique utilized the color attributes to detect the comic characters [65]. Graph theory was used to detect the comic characters by representing the color regions as nodes and panels as attributed adjacency graphs [66]. The same idea was implemented using the SIFT features with redundant values to accurately classify the repeated multiple objects [67]. Approximate nearest neighbors (ANN) search and local feature extraction was used for character retrieval [68]. Query-by-example (QBE) model was implemented using a Frequent Subgraph Mining (FSM) techniques for comic detection [69].

1.1 Motivation

The use of superheroes on fashion accessories is increasing worldwide as the demand for items containing a superhero is uprooting. Since, there exist so many superheroes, and one can simply not recognize or memorize them; thus, there is a need for an automated system, which can classify any product having a superhero image and help the consumer to identify and recognize the product that sets primary motivation of the study. Another motivation of this work is to exploit standard datasets for comic classification to set a baseline for the other researchers.

1.2 Applications

Although, the proposed system is tested on product images but not limited to this domain and can solve the many problems of other areas. One of the main applications of this work is to identify all the comic medical images, which involve any medical character, i.e., doctor, nurse, paramedic staff, then classify those images according to these comic characters. This can help to categorize the medical images into a specific folder. Another application may involve on detection of medical comic characters in a movie or video, to identify, locate and describe a medical comic character. This can help to recognize and discriminate a medical comic character from a video.

2 Objectives and Contribution

Classifying the superheroes from multiple types of products is an exciting task due to the placement and size of the images has huge diversity. This task becomes even more difficult when the dataset is extremely imbalanced, and the image sizes are extremely small. The main purpose of this article is to propose an automated system, which not only overcomes these issues but also performs the tasks of training, classification, and prediction with efficiency. The fundamental contributions of this article are:

- A general data balancing algorithm is proposed, which is not limited to the selected dataset only. It calculates the difference between a majority and minority classes, and populates the dataset with augmented images, obtained by performing steps such as image flipping, adding gamma correction and injecting gaussian noise. It increases the training of the model, which ultimately improves the performance of the proposed model.
- A 17-layer deep CNN model is proposed, which contains six (6) convolutional layers with attached ReLU and max-pooling layers and two (2) fully connected layers. The settings of the proposed CNN model are adopted after intensive experiments like increasing and decreasing the total number of convolutional layers and applying max and average pooling.

3 Materials and Proposed Model

This section describes in detail about the selected dataset, preprocessing steps involved, data augmentation, CNN, Network Architecture and Training settings.

3.1 Dataset and Pre-processing

The primary objective of the selected dataset is to classify the 12 superheroes i.e., Antman, Aquaman, Avengers, Batman, Black Panther, Captain America, Catwoman, Ghostrider, Hulk, Ironman, Spiderman and Superman from product images. The dataset is already split into training and testing portions having 5433 and 3375 images respectively. Two-step preprocessing method including the data augmentation and image resizing to a size of $100 \times 100 \times 3$ is adopted in this research work. Fig. 1 illustrates one image from each of the 12 classes.

3.2 Data Augmentation

For training, less images may cause the over-fitting as the training dataset contains five classes with less than 250 images while remaining classes contains 400 or above images, even a class have highest images 1144. The aim of data augmentation is to increase all the minority classes to the size of majority class for a fair and enough training of proposed network. To achieve this, initially all the images of minority classes are flipped horizontally at 90, which balanced the classes like Batman, Captain America, Iron Man and Superman. In the second step, all the remaining classes are augmented using gamma correction by using a fixed gamma-value g at 0.8. This step further balances the classes like Black Panther and Hulk. In the third step, gaussian noise having a variance value of 0.02 is applied on all the minority class images. This step completes the data augmentation method, as now all the classes contain images more

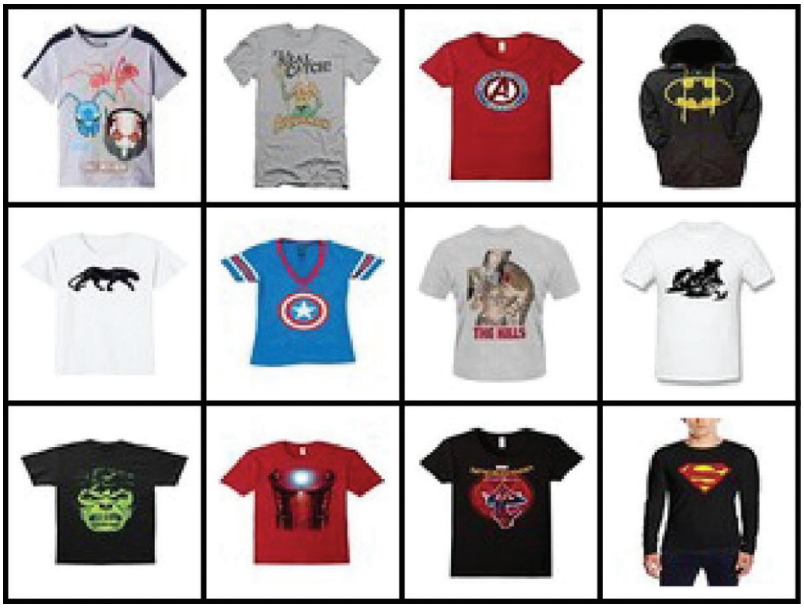

Figure 1: Dataset samples from each class. Left to Right Row 1: (Ant-Man, Aquaman, Avengers, Batman) Row 2: (Black Panther, Captain America, Catwoman, Ghost Rider) Row 3: (Hulk, Iron Man, Spiderman, Superman)

Table 1: Details of dataset before and after augmentation

Class	Original Images	Flip Operation		Gamma Correction		Gaussian Noise		Selected Images
		Created	Total	Created	Total	Created	Total	
Ant-Man	242	242	484	484	968	968	1,936	1,100
Aquaman	202	202	404	404	808	808	1,616	1,100
Avengers	216	216	432	432	864	864	1,728	1,100
Batman	779	779	1,558	–	1,558	–	1,558	1,100
Black Panther	459	459	918	918	1,836	–	1,836	1,100
Captain America	716	716	1,432	–	1,432	–	1,432	1,100
Catwoman	200	200	400	400	800	800	1,600	1,100
Ghost Rider	200	200	400	400	800	800	1,600	1,100
Hulk	413	413	826	826	1,652	–	1,652	1,100
Iron Man	979	979	1,958	–	1,958	–	1,958	1,100
Spiderman	1,144	–	1,144	–	1,144	–	1,144	1,100
Superman	977	977	1,954	–	1,954	–	1,954	1,100
Total	6,527						20,000	13,200

than the majority class. A total of 1144 images are then selected from each class to train the network. The detailed overview of each class is given in Tab. 1 along with the augmentation results. It can be seen that the original dataset contains 6,527 images while the augmented dataset contains 20,000 images. The method of data augmentation is further explained and illustrated in Fig. 2.

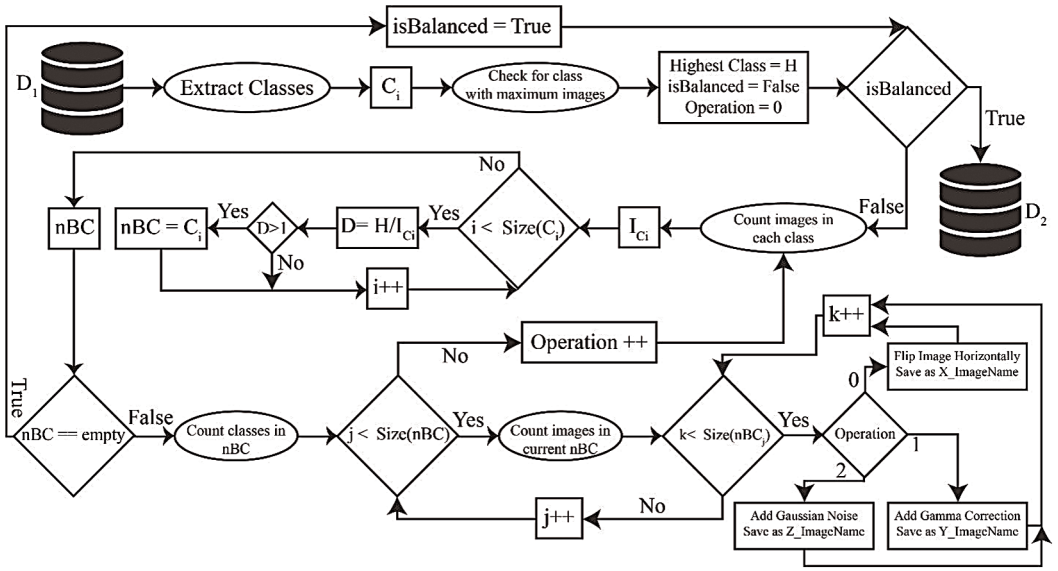

Figure 2: The detailed process of data augmentation

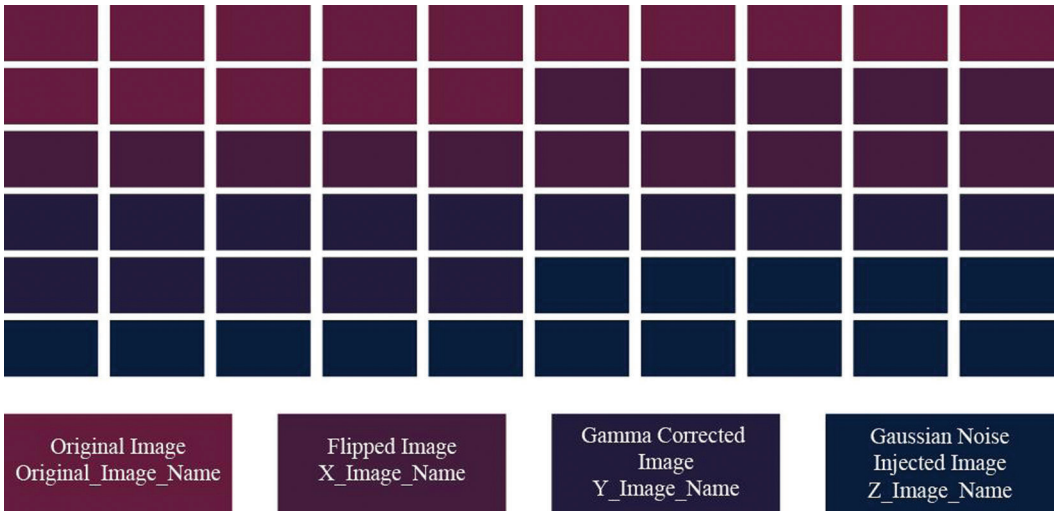

Figure 3: The arrangement of augmented images in related folders

Initially, all the classes are extracted from the dataset and class with maximum images is selected as a threshold value H. Difference between remaining classes and threshold value H is calculated, which is later used to store all the non-Balanced Classes (nBC). These nBC are then forwarded for further processing to obtain three different images from original image under certain conditions. The arrangement of these images after augmentation in relevant folders is presented in Fig. 3.

The purpose of this arrangement is to keep the original images always at the start while the augmented images to follow them, so that the discarded images from the end have overall low impact on training, as the original images are always selected. After the process of augmentation, first 1100 images are sequentially selected to train the proposed network.

3.3 Convolutional Neural Network (CNN)

Textual and non-textual classifications are majorly performed using CNNs as these networks have gained tremendous results due to their deep structures [70]. Parameter sharing, sparse interaction and equivariance have turned these networks advantageous over the traditional shallow networks. A typical CNN is composed of different layers like convolutional, rectified linear units and pooling layers. The number and arrangements of these layers varies from network to network.

3.3.1 Convolutional Layer

Three dimensional inputs and filters are convolved using the convolutional layer. Suppose an input image of size $I_w \times I_h \times I_c$ is convolved along the width and height using a filter of size $F_w \times F_h \times F_c$ where w, h and c denotes the width, height and channels of the both input image and filters. The channel size for both the input and filter must be same to perform the convolution. If the stride for the filter is ω and padding is φ, then width O_w and height O_h of convolved output image can be calculated as:

$$O_w = \frac{I_w - F_w + 2\varphi}{\omega} + 1 \tag{1}$$

$$O_h = \frac{I_h - F_h + 2\varphi}{\omega} + 1 \tag{2}$$

The convolved output is also a three-dimensional with width, height and number of filters in the form of $O_w \times O_h \times O_f$. Every convolutional layer has a ReLU layer as a nonlinear activation function to rectify the output of a convolutional layer. The ReLU function for an output r is defined as:

$$\mathrm{Re}LU(r) = \begin{cases} r, & r \geq 0 \\ 0, & r < 0 \end{cases} \tag{3}$$

3.3.2 Pooling Layer

Pooling operation updates the final output of ReLU activation function by calculating the statistical measures using the nearby output parameters. Performing the pooling operation not only reduces computational burden by reducing the size of parameters but also guarantees that the representation of small translations in input becomes invariant. Suppose an activation set Z has a pooling region P_r in it, then the specific activation set is defined as:

$$Z = \{z_i | i \in P_r\} \tag{4}$$

Max-pooling for this activation set is defined as $Pooling_{max} = \max(Z)$ while the average-pooling is defined as $Pooling_{avg} = \frac{\sum P_r}{|P_r|}$ where $|P_r|$ represents elements in activation set.

3.3.3 Fully Connected Layer

The main propose of fully-connected layers is to combine the learned features of different convolutional kernels in such a way that they form a global representation of the overall image. The neurons of fully-connected layers get fired only when the convolutional features are presented in the features of previous layers. Linear and non-linear transformations are performed on the input data. The linear transformation is represented as:

$$out = w^T . i + b \tag{5}$$

Here, w denotes the weights, i denotes the input from the previous layer and b denotes the biasness. For a non-linear transformation, a sigmoid function is used with the values between 0 and 1. The non-linear

transformations are performed, when the data is binary. As we are dealing with more than two classes, we will be using linear transformations for the fully-connected layers.

3.4 Network Architecture

In the past decade, many pre-trained CNN models are proposed to tackle multiple issues. These networks have performed significantly in many research areas, however, in case of comic classification, none of these CNN models worked effectively because of the complex structure and extensive layers. These extensive layers decrease the efficiency and increase the training time of the algorithm. To cope with this problem, a DCNN with 17 deep layers is proposed in this work. The input layer forwards it to the connected convolutional layers having attached ReLU and max-pooling layers. The size of filters, stride and total number of filters are set by performing multiple experiments. The aim of fully connected layer is to add a bias vector with the multiplication of weight matrix and input. The final, fully connected layer relates to a softmax layer which mainly generalizes the logistic regression. The configuration of proposed CNN is presented in Fig. 4.

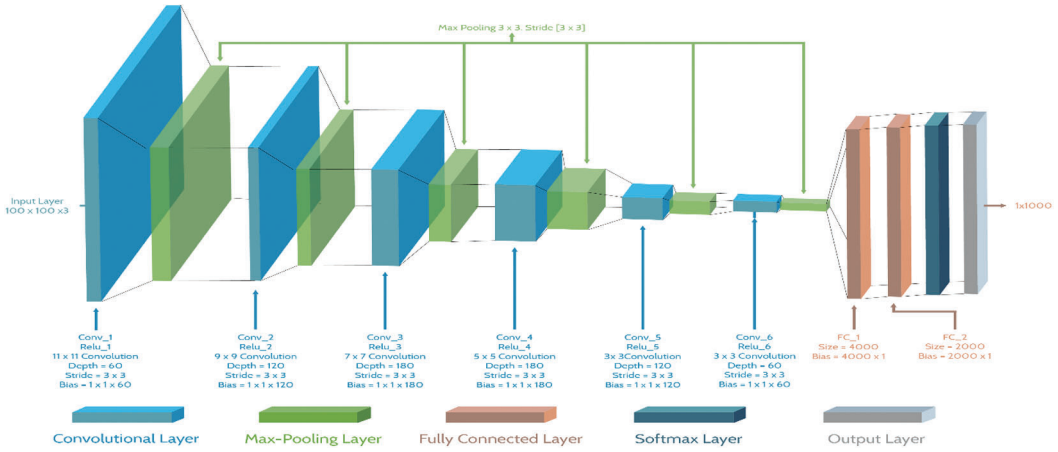

Figure 4: Structure of proposed CNN model

If Prob(m) denotes the probability of prior class and Prob$(n|m)$ denotes the conditional probability for the n^{th} sample within class m, and I denotes the total number of classes in the dataset then probability for a sample can be calculated as:

$$prob(n|m) = \frac{prob(n|m)prob(m)}{\sum_{z=1}^{I} prob(n|z)prob(z)} \tag{6}$$

To simplify the above equation, if we define final probability (F_p) as $F_p = \ln(\text{Prob}(n,m)\text{Prob}(m))$ then:

$$prob(n|m) = \frac{\exp(F_p(m))}{\sum_{z=1}^{I} \exp(F_z(m))} \tag{7}$$

The output provides the predicted class labels for every input based on the training. The class labels are already defined as the class names while training the model. The proposed CNN contained 17 layers, where the input layer accepts an RGB image of $100 \times 100 \times 3$. There are total of 6 convolutional layers, and each layer is followed by a ReLU layer. Different number of filters are applied on these layers. All these 6

Table 2: Detail about layers in proposed CNN model

Combinations	Filters	Total Filters	Stride Size	Weight Size	Bias Vector	Activations
Input Layer	–	–	–	–	–	$100 \times 100 \times 3$
Convolutional + ReLU	11×11	60	$[3 \times 3]$	$11 \times 11 \times 3 \times 60$	$1 \times 1 \times 60$	$128 \times 128 \times 60$
Max Pooling	3×3	–	$[3 \times 3]$	–	–	$64 \times 64 \times 60$
Convolutional + ReLU	9×9	120	$[3 \times 3]$	$9 \times 9 \times 60 \times 120$	$1 \times 1 \times 120$	$128 \times 128 \times 120$
Max Pooling	3×3	–	$[3 \times 3]$	–	–	$64 \times 64 \times 120$
Convolutional + ReLU	7×7	180	$[3 \times 3]$	$7 \times 7 \times 120 \times 180$	$1 \times 1 \times 180$	$128 \times 128 \times 180$
Max Pooling	3×3	–	$[3 \times 3]$	–	–	$64 \times 64 \times 180$
Convolutional + ReLU	5×5	180	$[3 \times 3]$	$5 \times 5 \times 180 \times 180$	$1 \times 1 \times 180$	$64 \times 64 \times 180$
Max Pooling	3×3	–	$[3 \times 3]$	–	–	$32 \times 32 \times 180$
Convolutional + ReLU	3×3	120	$[1 \times 1]$	$3 \times 3 \times 180 \times 120$	$1 \times 1 \times 120$	$32 \times 32 \times 120$
Max Pooling	3×3	–	$[1 \times 1]$	–	–	$32 \times 32 \times 120$
Convolutional + ReLU	3×3	60	$[1 \times 1]$	$3 \times 3 \times 120 \times 60$	$1 \times 1 \times 60$	$16 \times 16 \times 60$
Max Pooling	3×3	–	$[1 \times 1]$	–	–	$8 \times 8 \times 60$
Fully Connected	–	–	–	4000×6000	4000×1	$1 \times 1 \times 4000$
Fully Connected	–	–	–	2000×4000	2000×1	$1 \times 1 \times 2000$
Softmax	–	–	–	–	–	$1 \times 1 \times 1000$
Output	–	–	–	–	–	$1 \times 1 \times 1000$

combinations of convolutional and ReLU layers are followed by max pooling layer, which reduced the size of these layers into half. There are two fully connected layers, 'FC1' and 'FC2' which provide total of 4000 and 2000 features respectively. At the end, a softmax and output layers complete the proposed CNN network's architecture. Detailed summary of proposed CNN model is given in Tab. 2.

3.5 Training Settings

The CNN model is trained on NVIDIA GeForce GTX 1080 having an overall 6.1 capability of computation, 1607–1733 MHz clock rate and 7 multiprocessors using MATLAB 2018a. The Stochastic Gradient Descent with momentum (SGDM) algorithm represents the training technique in minibatch size of 64. Learning rate is initially fixed at 0.01 and decreased after every 5 eras by the factor of 5. The momentum is set at 0.7 and maximum epochs are set at 150. A suitable loss function, Cross-Entropy [71] is used as it has performed reasonable for many multiclass issues. To extract the features, FC1 layer is utilized, which extracts 4000 features against a single image.

These parameter settings are selected by performing intensive experiments. Considering the minibatch size of 64, total iterations are set at 150 and 13,200 training images from augmented dataset, which makes

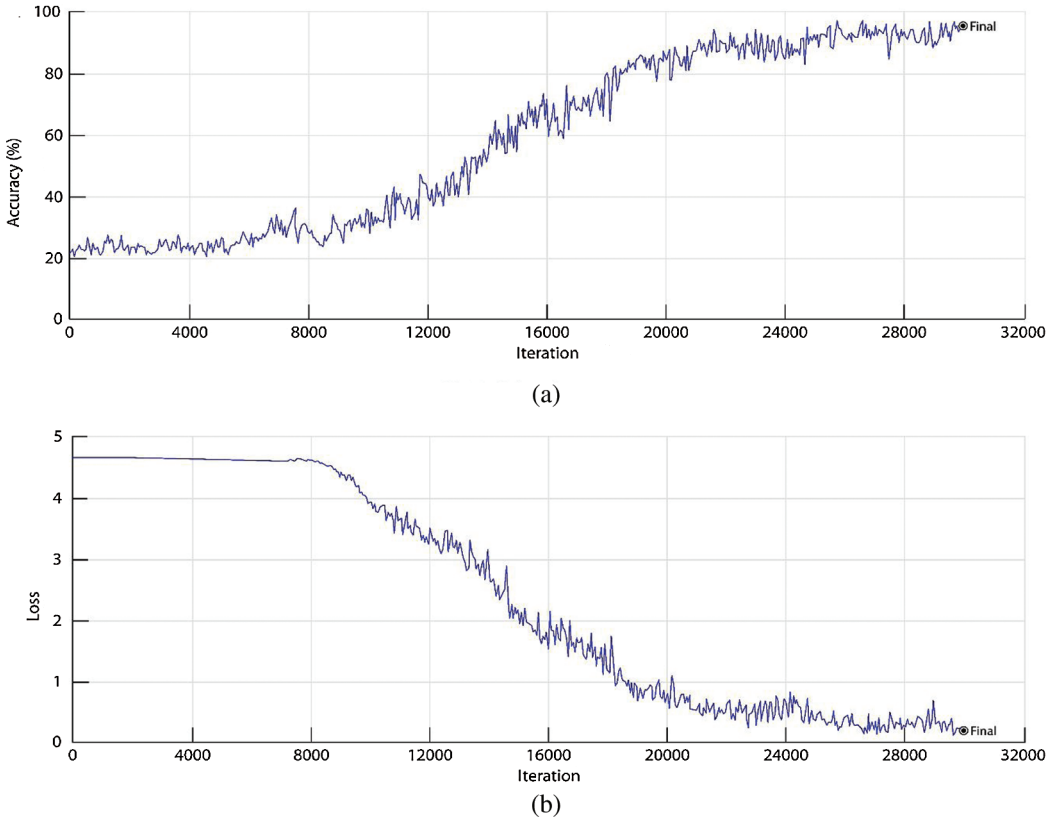

Figure 5: Training accuracy and loss of proposed CNN model. (a) Training Accuracy and (b) Training Loss

$\frac{150 \times 13200}{64} = 30,937$ iterations. The overall training accuracy and loss is illustrated in Figs. 5a and 5b respectively.

The training accuracy of 93.5% is achieved on proposed network while the training loss is reduced to less than 1%. The training loss shows that the CNN model is well-trained on training and validation sets. The training accuracy of proposed model is determined, once all the parameters of model are learned and no further learning is in due. This trained network is then used to extract the features of test data, which are later classified to obtain the classification accuracy.

4 Experiments and Results

4.1 Illustration of Data Augmentation

The proposed data augmentation technique initially finds the majority class and calculates the difference of each class with respect to the majority class. Based on this difference, different operations i.e., by image flipping, gamma correction and gaussian noise injection, are performed to generate new images. These operations generate up to 3 new images to enlarge the dataset for training. It also benefits the deep learning algorithms to acquire more consistent features than the original dataset. Fig. 6 demonstrates the results of data augmentation process on 2 different images from 5 minority classes having images under 250. In Fig. 6, (a) represents the original image, (b) represents the flipped image, (c) represents the image after gamma correction and (d) represents the image after gaussian noise injection.

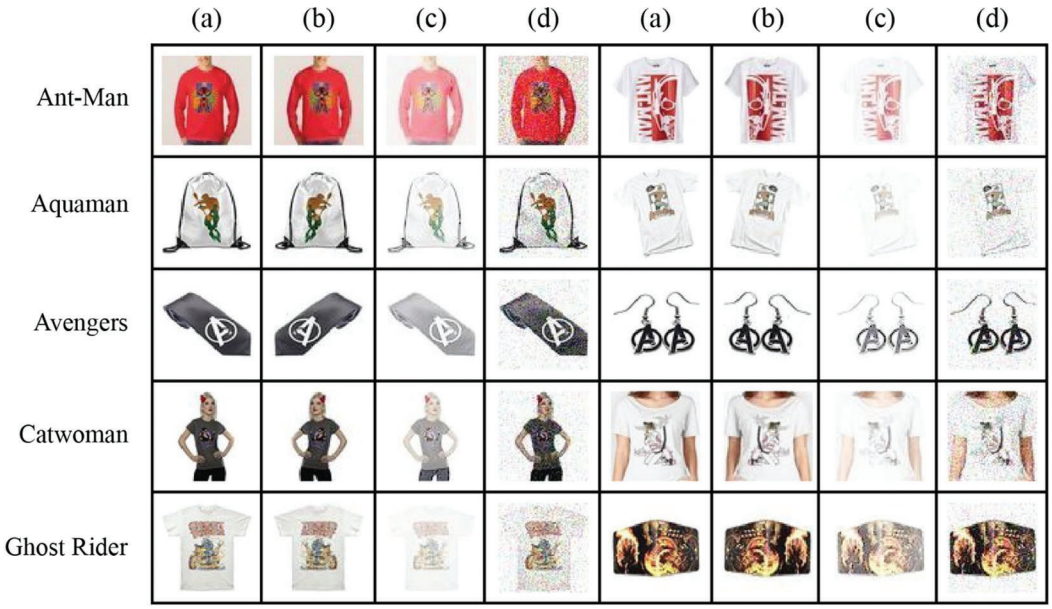

Figure 6: Results of data augmentation operations

4.2 Classification Results

The proposed 17-layer network is used to classify the test data using the trained model. The arrangement of CNN layers like convolutional layer and max-pooling layers plays a vital part in training the model to achieve maximum results. For this purpose, multiple experiments are performed to search for the ideal combination by increasing the depth of network. The network is tested by using 5,6,7,8 and 9 convolutional layers along with ReLU and pooling layers. The highest results are obtained by using the network with 6 convolutional layers. Comparison of these different combinations is shown in Fig. 7 in the form of graph.

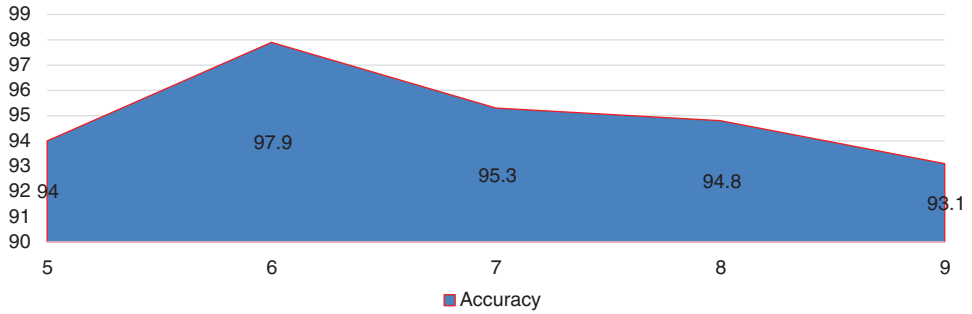

Figure 7: Comparing the impact of decreasing and increasing the convolutional layers

To evaluate the authenticity of proposed network, evaluation parameters like sensitivity, precision, specificity and accuracy are obtained on respective classes for the test data as displayed in Tab. 3. The purpose of extracting the class-wise classification results is to monitor the performance of model on this dataset. There is a lot of inter and intra class similarity in this dataset and decreases the overall efficiency of model. It can clearly be seen that the second class (Aquaman), seventh class (Catwoman) and tenth class (Iron Man) can be perfectly identified having the sensitivity of 100.0%. The class with worst

Table 3: Class-wise classification results

Class	Sensitivity (%)	Precision (%)	Specificity (%)	Accuracy (%)
Ant-Man	92.4	93.4	98.5	98.6
Aquaman	100	97.0	98.3	97.8
Avengers	97.2	95.7	98.9	98.2
Batman	94.8	96.4	97.0	98.3
Black Panther	89.1	92.8	98.6	97.1
Captain America	88.6	96.5	96.2	98.4
Catwoman	100	98.2	97.8	97.0
Ghost Rider	98.4	97.6	98.4	97.8
Hulk	99.0	98.3	99.3	98.9
Iron Man	100	95.9	97.4	97.3
Spiderman	97.5	94.3	97.9	97.4
Superman	99.3	99.8	98.5	97.9
Average	96.3	96.3	98.0	97.9

classification result includes sixth class (Captain America) with sensitivity of 88.6% as this class is mostly misclassified as Spiderman.

The performance of proposed network is compared with 7 classifiers, where ESD performs best by achieving an overall accuracy of 97.9%. These results are obtained on both, augmented and original dataset to verify the impact of data augmentation. The minimum training time is recorded for weighted-KNN with 69.0 seconds while the maximum training time is recorded for LDA with 448.7 seconds. The lowest FNR is 2.1 for ESD classifier and highest FNR is 9.7 for weighted KNN. In terms of sensitivity, 95.3% is highest, recorded for ESD and 90.1% is recorded for LDA. The highest precision is recorded for ESD at 94.3%, while lowest is recorded for cubic SVM at 90.5%. The average prediction time is 0.09 seconds while the minimum prediction time is recorded at 0.04 seconds. Detailed classification results are shown in Tab. 4.

During the testing of proposed method on selected dataset, few images are incorrectly classified, which ultimately degraded the accuracy. All these images have incorrectly predicted labels on the image with yellow background and correct labels under the image in black background. Correctly and wrongly predicted images are shown in Figs. 8 and 9 respectively.

The results of max-pooling layer in proposed network are compared with average-pooling. The max pooling provides accuracy of 97.9% while the proposed network provides 96.3% accuracy with average pooling, which clearly degrades the overall classification accuracy by 1.6%. This downfall is because the average-pooling considers all the elements inside the filter to decide while max-pooling only selects the highest feature. In future, other pooling techniques can also be tested to further enhance the results.

4.3 Discussion

In the relevant literature, the researchers focused on extracting hand-crafted features on comic panels or pages. Although, previously proposed techniques have achieved remarkable results, but most of the methods are tested on very few images collected from google or other sources. This research work utilized a standard publicly available dataset which can be used for comparison to validate the methods in this domain. The

Table 4: Comparison of classification results on different classifiers

Classifier	Datasets		Performance Measure				
	Original	Augmented	Accuracy (%)	FNR (%)	Sensitivity (%)	Precision (%)	Training Time (s)
ESD	✓		94.5	5.5	93.6	92.8	53.3
		✓	97.9	2.1	95.3	94.3	96.5
Cubic SVM	✓		90.4	9.6	91.8	90.5	199.3
		✓	92.3	7.7	93.3	92.3	307.1
Fine KNN	✓		91.7	8.3	91.1	91.9	98.4
		✓	93.5	6.5	92.5	92.7	169.2
Weighted KNN	✓		90.3	9.7	91.0	90.6	69.0
		✓	93.3	6.7	93.6	93.9	184.7
Ensemble Subspace KNN	✓		93.1	6.9	92.8	92.2	104.9
		✓	95.9	4.1	95.3	94.0	239.4
Quadratic SVM	✓		91.9	8.1	92.2	90.6	151.8
		✓	92.0	8.0	92.5	91.2	231.5
LDA	✓		91.7	8.3	90.1	92.8	185.2
		✓	93.4	6.6	94.7	93.4	448.7

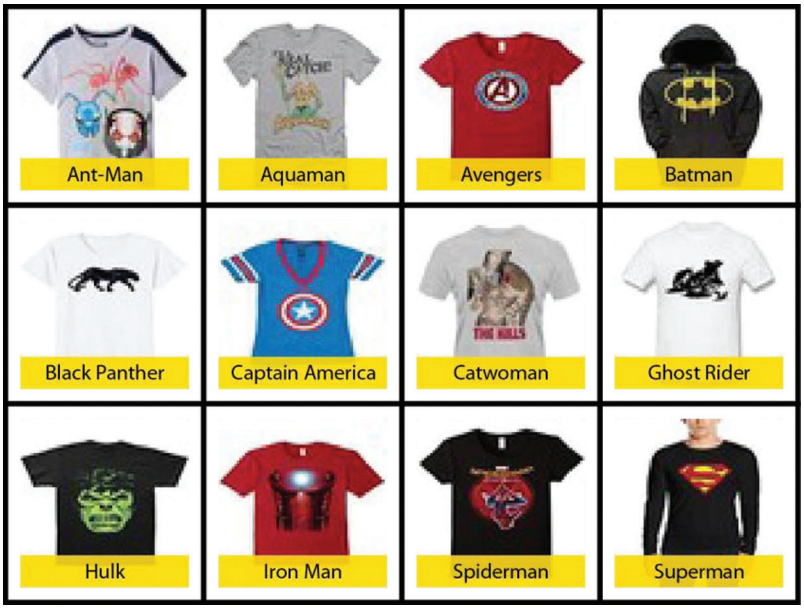

Figure 8: Correctly labeled images using proposed method

selected dataset was presented as a challenge to identify the superheroes on fashion product images. Five winners were selected as a result of solving this challenge, who achieved the highest classification accuracies. The leaderboard on the challenge page contains accuracy scores of around 108 contestants who participated in the trial, with 94.31%, 93.96%, and 93.86% as the first second and third positions.

Figure 9: Incorrect predictions using proposed method (correct labels at the bottom of the image while the predicted label on the image)

The computed results of the proposed model with 97.7% accuracy outperformed the previous results. This improvement in the classification score proves the authenticity of the proposed model.

5 Conclusion

This article proposed a 17-layer deep CNN to classify the superheroes among different fashion product images. A publicly available dataset consists of 12 classes and 8808 images is used to authenticate the performance of the proposed model. The dataset is normalized through augmentation using a proposed augmentation technique, which performs operations like image flipping, adding gamma correction, and Gaussian noise. A 17-layer deep CNN model is proposed containing six convolutional layers with connected ReLU and max-pooling layers and two fully connected layers. Different combinations of convolutional layers and their overall efficiency are also compared along with the effect of maximum and average pooling. The experiments show that six convolutional layers with integrated max-pooling provide better results of 97.9% classification accuracy with an average prediction time of 0.09 seconds. In the future, this network, along with few hand-crafted features, can be utilized to enhance classification results further. This model can also be implemented on other domains to check the validity of integrated layers and depth of the proposed CNN model.

Acknowledgement: I would like to thank our institute, HITEC University, for their ultimate support and resources to make this work possible.

Authors Contribution: Inzamam Mashood and MAK generated this idea and implementation. Majed. A helped in the first draft. T. Saba and A. Rehman perform final proof reading. T. Iqbal is supervising this work.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Farmer, L. S. (2016). Information architecture and the comic arts: knowledge structure and access. *Web Design and Development: Concepts, Methodologies, Tools, and Applications,* pp. 569–588. IGI Global.
2. González-Espada, W. J. (2003). Integrating physical science and the graphic arts with scientifically accurate comic strips: rationale, description, and implementation. *Revista Electrónica de Enseñanza de las Ciencias, 2(1),* 58–66.
3. Khan, M. A., Akram, T., Sharif, M., Javed, M. Y., Muhammad, N. et al. (2019). An implementation of optimized framework for action classification using multilayers neural network on selected fused features. *Pattern Analysis and Applications, 22(4),* 1377–1397. DOI 10.1007/s10044-018-0688-1.
4. Sharif, M., Khan, M. A., Akram, T., Javed, M. Y., Saba, T. et al. (2017). A framework of human detection and action recognition based on uniform segmentation and combination of Euclidean distance and joint entropy-

based features selection. *EURASIP Journal on Image and Video Processing, 2017(1),* 89. DOI 10.1186/s13640-017-0236-8.

5. Sharif, M., Khan, M. A., Faisal, M., Yasmin, M., Fernandes, S. L. (2018): A framework for offline signature verification system: best features selection approach. *Pattern Recognition Letters*.
6. Khan, M. A., Sharif, M., Javed, M. Y., Akram, T., Yasmin, M. et al. (2017). License number plate recognition system using entropy-based features selection approach with SVM. *IET Image Processing, 12(2),* 200–209. DOI 10.1049/iet-ipr.2017.0368.
7. Khan, M. A., Akram, T., Sharif, M., Shahzad, A., Aurangzeb, K. et al. (2018). An implementation of normal distribution based segmentation and entropy controlled features selection for skin lesion detection and classification. *BMC Cancer, 18(1),* 638. DOI 10.1186/s12885-018-4465-8.
8. Wang, S. H., Muhammad, K., Hong, J., Sangaiah, A. K., Zhang, Y. D. (2020). Alcoholism identification via convolutional neural network based on parametric ReLU, dropout, and batch normalization. *Neural Computing and Applications, 32(3),* 665–680. DOI 10.1007/s00521-018-3924-0.
9. Jia, W., Muhammad, K., Wang, S. H., Zhang, Y. D. (2019). Five-category classification of pathological brain images based on deep stacked sparse autoencoder. *Multimedia Tools and Applications, 78(4),* 4045–4064. DOI 10.1007/s11042-017-5174-z.
10. Sharif, M., Khan, M. A., Iqbal, Z., Azam, M. F., Lali, M. I. U. et al. (2018). Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection. *Computers and Electronics in Agriculture, 150,* 220–234. DOI 10.1016/j.compag.2018.04.023.
11. Iqbal, Z., Khan, M. A., Sharif, M., Shah, J. H., ur Rehman, M. H. et al. (2018). An automated detection and classification of citrus plant diseases using image processing techniques: a review. *Computers and Electronics in Agriculture, 153,* 12–32. DOI 10.1016/j.compag.2018.07.032.
12. Zhang, Y. D., Dong, Z., Chen, X., Jia, W., Du, S. et al. (2019). Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation. *Multimedia Tools and Applications, 78(3),* 3613–3632. DOI 10.1007/s11042-017-5243-3.
13. Tuan, T. M., Chuan, P. M., Ali, M., Ngan, T. T., Mittal, M. (2019). Fuzzy and neutrosophic modeling for link prediction in social networks. *Evolving Systems, 10(4),* 629–634. DOI 10.1007/s12530-018-9251-y.
14. Sethi, J. K., Mittal, M. (2019). A new feature selection method based on machine learning technique for air quality dataset. *Journal of Statistics and Management Systems, 22(4),* 697–705. DOI 10.1080/09720510.2019.1609726.
15. Wang, S. H., Zhang, Y., Li, Y. J., Jia, W. J., Liu, F. Y. et al. (2018). Single slice based detection for Alzheimer's disease via wavelet entropy and multilayer perceptron trained by biogeography-based optimization. *Multimedia Tools and Applications, 77(9),* 10393–10417. DOI 10.1007/s11042-016-4222-4.
16. Sharif, M., Tanvir, U., Munir, E. U., Khan, M. A., Yasmin, M. (2018). Brain tumor segmentation and classification by improved binomial thresholding and multi-features selection. *Journal of Ambient Intelligence and Humanized Computing,* 1–20.
17. Akram, T., Khan, M. A., Sharif, M., Yasmin, M. (2018). Skin lesion segmentation and recognition using multichannel saliency estimation and M-SVM on selected serially fused features. *Journal of Ambient Intelligence and Humanized Computing,* 1–20.
18. Bar, Y., Levy, N., Wolf, L. (2014). Classification of artistic styles using binarized features derived from a deep neural network. *European Conference on Computer Vision. Cham: Springer,* pp. 71–84.
19. Gatys, L. A., Ecker, A. S., Bethge, M. (2016). Image style transfer using convolutional neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 2414–2423.
20. Chen, D., Yuan, L., Liao, J., Yu, N., Hua, G. (2017). Stylebank: an explicit representation for neural image style transfer. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 1897–1906.
21. Chen, Y., Lai, Y. K., Liu, Y. J. (2018). Cartoongan: generative adversarial networks for photo cartoonization. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 9465–9474.
22. Wolf, L., Taigman, Y., Polyak, A. (2017). Unsupervised creation of parameterized avatars. *Proceedings of the IEEE International Conference on Computer Vision,* pp. 1530–1538.

23. Hensman, P., Aizawa, K. (2017). cGAN-based manga colorization using a single training image. *14th IAPR International Conference on Document Analysis and Recognition. IEEE, vol. 3,* pp. 72–77.
24. Jin, Y., Zhang, J., Li, M., Tian, Y., Zhu, H. et al. (2017). Towards the automatic anime characters creation with generative adversarial networks. arXiv preprint arXiv: 1708.05509.
25. Khan, M. A., Javed, M. Y., Sharif, M., Saba, T., Rehman, A. (2019). Multi-model deep neural network based features extraction and optimal selection approach for skin lesion classification. *International Conference on Computer and Information Sciences. IEEE,* pp. 1–7.
26. Sharif, M., Attique Khan, M., Rashid, M., Yasmin, M., Afza, F. et al. (2019). Deep CNN and geometric features-based gastrointestinal tract diseases detection and classification from wireless capsule endoscopy images. *Journal of Experimental & Theoretical Artificial Intelligence,* 1–23.
27. Dash, S., Acharya, B. R., Mittal, M., Abraham, A., Kelemen, A. (2020). *Deep learning techniques for biomedical and health informatics*. Cham: Springer.
28. Mittal, M., Arora, M., Pandey, T., Goyal, L. M. (2020). Image segmentation using deep learning techniques in medical images. *Advancement of Machine Intelligence in Interactive Medical Image Analysis*, pp. 41–63. Singapore: Springer.
29. Zhang, Y. D., Zhang, Y., Hou, X. X., Chen, H., Wang, S. H. (2018). Seven-layer deep neural network based on sparse autoencoder for voxelwise detection of cerebral microbleed. *Multimedia Tools and Applications, 77(9),* 10521–10538. DOI 10.1007/s11042-017-4554-8.
30. Khan, M. A., Khan, M. A., Ahmed, F., Mittal, M., Goyal, L. M. et al. (2020). Gastrointestinal diseases segmentation and classification based on duo-deep architectures. *Pattern Recognition Letters, 131,* 193–204. DOI 10.1016/j.patrec.2019.12.024.
31. Khan, M. A., Sharif, M., Akram, T., Bukhari, S. A. C., Nayak, R. S. (2020). Developed Newton-Raphson based deep features selection framework for skin lesion recognition. *Pattern Recognition Letters, 129,* 293–303. DOI 10.1016/j.patrec.2019.11.034.
32. Sharif, M. I., Li, J. P., Khan, M. A., Saleem, M. A. (2020). Active deep neural network features selection for segmentation and recognition of brain tumors using MRI images. *Pattern Recognition Letters, 129,* 181–189. DOI 10.1016/j.patrec.2019.11.019.
33. Khan, M. A., Sharif, M., Akram, T., Yasmin, M., Nayak, R. S. (2019). Stomach deformities recognition using rank-based deep features selection. *Journal of Medical Systems, 43(12),* 329. DOI 10.1007/s10916-019-1466-3.
34. Mittal, M., Goyal, L. M., Kaur, S., Kaur, I., Verma, A. et al. (2019). Deep learning based enhanced tumor segmentation approach for MR brain images. *Applied Soft Computing, 78,* 346–354. DOI 10.1016/j.asoc.2019.02.036.
35. Li, Z., Wang, S. H., Fan, R. R., Cao, G., Zhang, Y. D. et al. (2019). Teeth category classification via seven-layer deep convolutional neural network with max pooling and global average pooling. *International Journal of Imaging Systems and Technology, 29(4),* 577–583. DOI 10.1002/ima.22337.
36. Pereira, P. M., Fonseca-Pinto, R., Paiva, R. P., Assuncao, P. A., Tavora, L. M. et al. (2020). Skin lesion classification enhancement using border-line features—The melanoma *vs.* nevus problem. *Biomedical Signal Processing and Control, 57,* 101765. DOI 10.1016/j.bspc.2019.101765.
37. Lu, S., Xia, K., Wang, S. H. (2020). Diagnosis of cerebral microbleed via VGG and extreme learning machine trained by Gaussian map bat algorithm. *Journal of Ambient Intelligence and Humanized Computing,* 1–12. DOI 10.1007/s12652-020-01789-3.
38. Deng, J., Dong, W., Socher, R., Li, L. J., Li, K. et al. (2009). Imagenet: a large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition. IEEE,* pp. 248–255.
39. Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems,* pp. 1097–1105.
40. Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv: 1409.1556.
41. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. et al. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 1–9.

42. Saba, T., Khan, M. A., Rehman, A., Marie-Sainte, S. L. (2019). Region extraction and classification of skin cancer: A heterogeneous framework of deep CNN features fusion and reduction. *Journal of Medical Systems, 43(9),* 289. DOI 10.1007/s10916-019-1413-3.
43. Khan, M. A., Sharif, M., Akram, T., Raza, M., Saba, T. et al. (2020). Hand-crafted and deep convolutional neural network features fusion and selection strategy: An application to intelligent human action recognition. *Applied Soft Computing, 87,* 105986. DOI 10.1016/j.asoc.2019.105986.
44. Khan, M. A., Sharif, M. I., Raza, M., Anjum, A., Saba, T. et al. (2019). Skin lesion segmentation and classification: A unified framework of deep neural network features fusion and selection. *Expert Systems,* e12497. DOI 10.1111/exsy.12497.
45. Khan, M. A., Akram, T., Sharif, M., Javed, K., Rashid, M. et al. (2019). An integrated framework of skin lesion detection and recognition through saliency method and optimal deep neural network features selection. *Neural Computing and Applications, 29,* 1–20. DOI 10.1007/s00521-019-04514-0.
46. Khan, M. A., Javed, K., Khan, S. A., Saba, T., Habib, U. et al. (2020). Human action recognition using fusion of multiview and deep features: An application to video surveillance. *Multimedia Tools and Applications, 10,* 1–27. DOI 10.1007/s11042-020-08806-9.
47. Johnson, C. R., Hendriks, E., Berezhnoy, I. J., Brevdo, E., Hughes, S. M. et al. (2008). Image processing for artist identification. *IEEE Signal Processing Magazine, 25(4),* 37–48. DOI 10.1109/MSP.2008.923513.
48. Saleh, B., Elgammal, A. (2015). Large-scale classification of fine-art paintings: learning the right metric on the right feature. arXiv preprint arXiv: 1505.00855.
49. Tan, W. R., Chan, C. S., Aguirre, H. E., Tanaka, K. (2016). Ceci n'est pas une pipe: a deep convolutional network for fine-art paintings classification. *IEEE International Conference on Image Processing. IEEE,* pp. 3703–3707.
50. Thomas, C., Kovashka, A. (2016). Seeing behind the camera: Identifying the authorship of a photograph. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 3494–3502.
51. Hicsonmez, S., Samet, N., Sener, F., Duygulu, P. (2017). DRAW: deep networks for recognizing styles of artists who illustrate children's books. *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval,* pp. 338–346.
52. Lai, W. S., Huang, J. B., Ahuja, N., Yang, M. H. (2017). Deep laplacian pyramid networks for fast and accurate super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 624–632.
53. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y. (2018). Residual dense network for image super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 2472–2481.
54. Haris, M., Shakhnarovich, G., Ukita, N. (2018). Deep back-projection networks for super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 1664–1673.
55. Wilber, M. J., Fang, C., Jin, H., Hertzmann, A., Collomosse, J. et al. (2017). Bam! The behance artistic media dataset for recognition beyond photography. *Proceedings of the IEEE International Conference on Computer Vision,* pp. 1202–1211.
56. Ogawa, T., Otsubo, A., Narita, R., Matsui, Y., Yamasaki, T. et al. (2018). Object detection for comics using manga109 annotations. arXiv preprint arXiv: 1803.08670.
57. Chu, W. T., Li, W. W. (2017). Manga facenet: Face detection in manga based on deep neural network. *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval,* pp. 412–415.
58. Nguyen, N. V., Rigaud, C., Burie, J. C. (2018). Digital comics image indexing based on deep learning. *Journal of Imaging, 4(7),* 89. DOI 10.3390/jimaging4070089.
59. Nguyen, N. V., Rigaud, C., Burie, J. C. (2017). Comic characters detection using deep learning. *14th IAPR International Conference on Document Analysis and Recognition.* IEEE, vol. 3, pp. 41–46.
60. Chu, W. T., Chao, Y. C. (2014). Line-based drawing style description for manga classification. *Proceedings of the 22nd ACM International Conference on Multimedia,* pp. 781–784.
61. Viola, P., Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision, 57(2),* 137–154. DOI 10.1023/B:VISI.0000013087.49260.fb.

62. Sun, W., Kise, K. (2013). Detection of exact and similar partial copies for copyright protection of manga. *International Journal on Document Analysis and Recognition, 16(4),* 331–349. DOI 10.1007/s10032-013-0199-y.
63. Sun, W., Kise, K. (2010). Similar partial copy detection of line drawings using a cascade classifier and feature matching. *International Workshop on Computational Forensics. Berlin, Heidelberg: Springer,* pp. 126–137.
64. Takayama, K., Johan, H., Nishita, T. (2012). Face detection and face recognition of cartoon characters using feature extraction. *Image, Electronics and Visual Computing Workshop,* pp. 48.
65. Khan, F. S., Anwer, R. M., Van de Weijer, J., Bagdanov, A. D., Vanrell, M. et al. (2012). Color attributes for object detection. *IEEE Conference on Computer Vision and Pattern Recognition. IEEE,* pp. 3306–3313.
66. Rigaud, C., Guérin, C., Karatzas, D., Burie, J. C., Ogier, J. M. (2015). Knowledge-driven understanding of images in comic books. *International Journal on Document Analysis and Recognition, 18(3),* 199–221.
67. Sun, W., Burie, J. C., Ogier, J. M., Kise, K. (2013). Specific comic character detection using local feature matching. *12th International Conference on Document Analysis and Recognition. IEEE,* pp. 275–279.
68. Iwata, M., Ito, A., Kise, K. (2014). A study to achieve manga character retrieval method for manga images. *11th IAPR International Workshop on Document Analysis Systems. IEEE,* pp. 309–313.
69. Le, T. N., Luqman, M. M., Burie, J. C., Ogier, J. M. (2015). A comic retrieval system based on multilayer graph representation and graph mining. *International Workshop on Graph-Based Representations in Pattern Recognition,* pp. 355–364. Cham: Springer.
70. Arshad, H., Khan, M. A., Sharif, M. I., Yasmin, M., Tavares, J. M. R. et al. (2020). A multilevel paradigm for deep convolutional neural network features selection with an application to human gait recognition. *Expert Systems,* e12541.
71. Shore, J., Johnson, R. (1980). Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. *IEEE Transactions on Information Theory, 26(1),* 26–37. DOI 10.1109/TIT.1980.1056144.