Proceedings of the Second International Conference on Applied Artificial Intelligence and Computing (ICAAIC 2023)
IEEE Xplore Part Number: CFP23BC3-ART; ISBN: 978-1-6654-5630-2

Recoloring Grayscale Images using GAN

Aeshita Mathur
Department of Computer Engineering
Delhi Technological University
Delhi, India
aeshitamathur@gmail.com

Ameesha Dabas
Department of Computer Engineering
Delhi Technological University
Delhi, India
ameeshadabas2002@gmail.com

Parikshit Rana
Department of Computer Engineering
Delhi Technological University
Delhi, India
parikshitrana123@gmail.com

Sanjay Kumar
Department of Computer Engineering
Delhi Technological University
Delhi, India
sanjay.kumar@dtu.ac.in

2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC) | 978-1-6654-5630-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICAAIC56838.2023.10140749

***Abstract*—In this paper, we intend to develop an efficient method for Colorizing Grayscale Images such that the output images are close to their real colors. The main objective is to produce a precise image should be produced, which means that the output image should be as close to its natural color as possible. In colorization, there is no aim to recover the ground truth color; rather, the aim is to create a plausible colorization that is useful to the user, even if it differs from the ground truth color.In order to solve this problem, we use Convolutional Neural Networks (CNN) and a variety of deep learning techniques. Instead of using vectors drawn at random from the probability distribution, we employ Conditional Generative Adversarial Networks (GAN). This will enable it to extract different features and understand the correlations between them in order to predict colored images.**

***Index Terms*—Deep Learning, Colorization, Gray-scale, Generative Adversarial Network, Convolutional Neural Networks, Feature Extraction, Transfer Knowledge,image-to-image translation**

I. Introduction

Colorization is the technique of adding colour to a gray scale image or video with the help of a computer. The procedure for adding colour to a grayscale image entails giving an image with just one dimensional variation three dimensional (RGB) pixel values. The mapping between intensity and colour is not unique, and colorization is ambiguous in nature, needing some level of human input or outside information since distinct colours might have the same luminance/ intensity value but differ in hue or saturation [1]. The purpose of colorization, it is vital to remember, is not to recreate the real ground truth colour, but rather to create a believable colorization that the user finds helpful, even if the colorization deviates from the actual ground truth colour. Considering how much information is lost when a colour image is reduced to its underlying grayscale representation (two out of the three colour dimensions), colorization might appear to be a daunting undertaking. Numerous hints for sound colorization are provided by the semantics of a picture scene. Because it uses scene semantics for picture categorization and object recognition, deep learning is a successful method for colorization. Note that the human eye would attempt to discern distinctions between two images if a true colour image and an artificial one were side-by-side. In recent times, machine learning and deep learning techniques have produced improved results for many real-life problems like computer vision, image processing, fake news detection, online social networks, and many others [2]–[7].

This paper intends to create an effective technique for ”colourizing grayscale images using Deep learning methods that will provide output images that are fairly accurate representations of their actual hues.

- The final image needs to be precise. Precise refers to how closely the colours in the produced image should match their actual natural state.
- It is crucial to remember that colorization's objective is to create a believable colorization that the user finds helpful even if the colorization deviates from the ground truth colour rather than to retrieve the actual colour of the ground truth.

A. Motivation

The task of colourizing a grayscale image is challenging. There is frequently no ”right” achievable hue given a certain visual. Due to how much information is lost when a colour image is converted to its underlying grayscale representation (two out of the three colour dimensions), colorization might appear to be a daunting undertaking. By enabling colorization of ancient, black-and-white photos and films and improving interpretation of CCTV camera footage, astronomical photography, and electron microscopy, uses of such a system offer a new kind of entertainment. Deep learning, which already makes use of scene semantics for picture categorization and object recognition, has the potential to be an effective tool for colorization. A manual method of colorization dates back to the early 19th century [8].The theories and techniques created in the area of colourizing grayscale photographs have been applied to improve the effectiveness of object detection systems. Motivations for Deep Architectures:

- Insufficient depth can hurt- The number of nodes in the graph that are necessary for shallow architecture (SVM, NB, KNN, etc.) calculations and parameter learning may

Authorized licensed use limited to: Northeastern University. Downloaded on September 16,2024 at 20:56:16 UTC from IEEE Xplore. Restrictions apply.

increase significantly. Many functions that are efficiently represented in deep architectures cannot be efficiently represented in shallow ones.

- The brain has a deep architecture- Signals go from one part of the visual cortex to the next as it displays a series of regions, each of which includes a representation of the input. Be aware that brain representations fall somewhere between densely dispersed and entirely local. Only 1% of the brain's neurons are simultaneously engaged.
- Cognitive processes seem deep- Human conceptions and ideas are arranged hierarchically. It is in our fundamental nature to first master more basic notions before combining them to express more complex ones.

B. Goals

Generative Adversarial Networks(GAN) are used to colourize photos with the goal of minimising the designer's labor-intensive manual colorization of the grayscale images. For the same objectives, earlier systems employed basic regression and other related techniques. Precision is the second significant issue that we address in our dissertation. Our aim was to design a Web Application that receives a grayscale image as input and creates a colourful image as output. The process of colouring grayscale photographs is a very challenging one. In actuality, there have been several methods used throughout history that have not produced positive outcomes.

In this thesis, we provide a fully automated colorization method that generates accurate hues. We embrace the underlying ambiguity of the problem by expressing it as a classification job and leverage the class rebalancing training time to boost the diversity of colours in the output . The achieved colorization results are excellent and bright. Additionally, because the process is totally mechanical, there is no need for human involvement; all that is required of the user is the input to be coloured.

II. Related Work

A comprehensive study by Žeger et. al. [9] has been done on algorithms with various architectures and levels of user guidance, taking into account objective image quality metrics as well as colorization.time. Due to the combination of human effort and technological advancements regarding neural networks, the user-guided colorization neural network from Zhang et al. [10] provided the most visually convincing results. It was found, however, that the colorization method required extensive human intervention and a great deal of time.

In their method, Nazeri et. al. employed a conditional Deep Convolutional Generative Adversarial Network (DCGAN) in a completely generalized colorization operation [11]. CIFAR10 and Places365 datasets were used for training the network. A comparison was made between the results of the generative model and those of a conventional deep neural network.

In 2014, Goodfellow et. al. introduced Generative Adversarial Networks (GANs) as a novel kind of generative model [12]. During the discussion, it was mentioned that a GAN consists of two smaller networks, the generator and the discriminator. The generator's job is to produce unrecognizable results. The discriminator problem consists of identifying whether the sample belongs to the model distribution generated by the generator, or to the original data distribution. Both subnetworks are trained simultaneously until the generator reliably produces results that the discriminator cannot categorize.

With no manual intervention, Wan et. al. opted for an innovative method of colorizing grayscale photographs [13]. A neural network was used to train the colorization model for the training set by selecting interest points based on SLIC superpixels. In order to colorize input grayscale images, the trained model should first colorize the interest points and propagate the color to the entire image.

Using 3D Printed Low-Index Nanopillars, Wang et. al. created full-color and grayscale paintings [14]. As a result of their study, two-photon polymerization lithography-based 3D printing gave users access to a complete 3D color spectrum with control of hue, saturation, and brightness, as well as producing grayscale shades by adjusting the height, diameter, and spacing of nanopillar arrays [15]

For the colorization process, Zhong et. al. proposed a GECNet-based model based on colorization [16]. As part of the training samples, GECNet converted the input visible images into grayscale images. To retain the features of the original infrared image as well as the rich texture and semantics provided by the visible version, the researchers proposed a feature fusion module. Through cross-modal image generation, their method effectively addressed the modality gap issue in VI-ReID.

A dissertation by Jiancheng An examined the ability to automatically colorize grayscale images using VGG-16 and CNN [17]. In their study, they concluded that color rebalancing is sometimes useful in correcting saturated colors in certain images. Jiancheng proposed to utilize the VGG-16 CNN model based on classification with loss of cross-entropy next year in accordance with the same study [18]. This model produced efficient results.

III. Proposed Methodology

The flowchart shown in Fig. 1 illustrates the method we used to complete the work.

A. Upload Grayscale Image

In order to properly run the model, a grayscale picture is uploaded to the website and further processed by a Python script to change its dimensions.

B. Pre-processing of the uploaded image

The uploaded image is transformed during this stage into a format that may be altered during following stages.

- Rotate- By rotating the image, the model ensures it does not depend on the image slope for prediction.
- Contrast stretching- The pixel intensity values are stretched to cover the desired range of values using full contrast stretching.
- Histogram Equalization- This technique distributes pixels over a wide range of intensities by utilizing non linear,

Authorized licensed use limited to: Northeastern University. Downloaded on September 16,2024 at 20:56:16 UTC from IEEE Xplore. Restrictions apply.

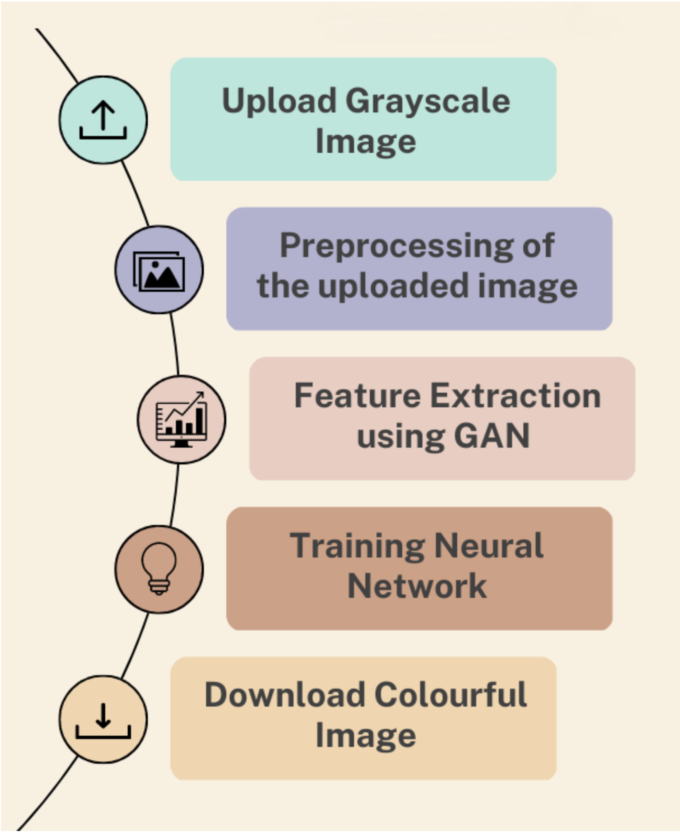

Fig. 1. Flowchart of Methodology applied.

non monotonic transfer functions. This parameter changes the intensity histogram's shape.

C. Feature Extraction using GAN

Randomly produced noise vectors serve as the generator network's input in GANs. Nevertheless, such a GAN will not be effective in the job of picture colorization since our GAN is not intended to create images from random vectors but rather to add colours to already-existing photographs containing only one channel (Black and white). So, adding three RGB channels with appropriate intensities for each colour channel is the fundamental challenge for our GAN. This problem is solved by using conditional GANs, which have one intensity channel and accept black and white pictures as inputs (technically, G(0z—x)). A modified discriminator input is also required for conditional GANs.

Grayscale image and produced image, as well as grayscale image and original picture, are the two inputs that the discriminator accepts. Next it decides if the pairs are bogus or real.

1) Method: The issue that we're working to resolve falls under the umbrella of image-to-image translation with high-dimensional input to high-dimensional output mapping. Regression is really being done at the pixel level with an output condition whose structure is comparable to the input. Hence, the network must have very high spatial similarity between input and output, as well as offer information on each pixel's colour in the original grayscale image.

2) GAN Architecture: This model's network is built using "fully connected networks". Convolutional layers are used in the Generator. Instead of pooling layers, the image is downscaled till it is a vector with a size of 2x2 pixels. The compressed portion is then expanded via upsampling to bring it up to the dimension of the input set of 32 x 32 pixels. A specific kind of deep networks termed encoder-decoder networks, which incorporate encoding and decoding networks for contracting and expanding, and therefore rebuilding the input, are what inspired this approach. This method facilitates network training without using a lot of memory.

The Generator accepts X as an input, which is a dimensioned black and white picture (32 x 32 x 1). The downsampling starts with kernels that have size as 1 and stride as 1. Furthermore, it is then compressed to an image of size (2x2) using a kernel and stride size of 2. After the first layer, this is repeated four more times to create a matrix of dimensions 2 by 2 by 512. With the exception of the last layer, the expansion phases include upsampling the matrix using kernel size 2 and strides 2. To preserve the structural integrity of the image, I and n-i layers were concatenated. For Generator's robust training, 0.5 scale dropouts are performed in the first and second expanding layers.

For improved training, batch normalisation is performed. LeakyReLU with a slope of 0.2 was employed in our model since it performed better than ReLU activation function. Convolution with a kernel size of 1 and a stride of 1 is performed in the last layer to create the picture dimensions (32 x 32 x 3). We utilize the "tanh" activation function since it has been shown to be more effective than linear activation functions. It produces results in the form of a matrix with values ranging from -1 to 1. During training, the model minimizes the cosine distance between the original image and the predicted image.

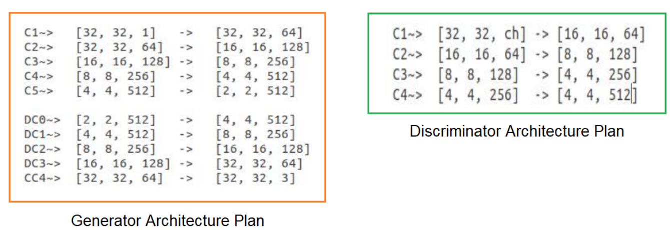

Fig. 2. Generator and Discriminator Architecture Plan.

In order to create the coloured picture for the discriminator, we foremost combine the grayscale image, forecast grayscale image, and ground truth image on the channel axis. Using a convolutional layer with a filter size of 2 x 2 and strides of 2, we downscale the matrix sequentially. Every layer does Batch Normalization, and each layer has a Leaky ReLUactivation function with a slope of 0.2. The final layer is flattened after which a hidden layer consisting of 128 units and an output layer comprising units are linked. The last layer's activation

Authorized licensed use limited to: Northeastern University. Downloaded on September 16,2024 at 20:56:16 UTC from IEEE Xplore. Restrictions apply.

function, known as a "sigmoid," estimates the likelihood that the input picture belongs to the anticipated or actual image.

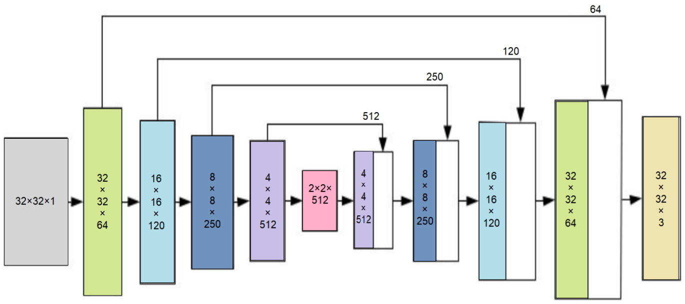

Fig. 3. Generator Network visualization.

D. *Training Neural Network*

We have used Adam's optimizer to our model. The optimizer's learning rate is held to 0.0001. For the implementation of our model, we used the free and open-source Python libraries Tensorflow and Keras. We used the free GoogleColabGPU to train the model. Our batch consists of 50 photos.

1) Batch Normalization: The 'Mode collapse' phenomena is thought to make the GAN challenging to train. The generator was successful in tricking the discriminator into accepting the same produced output as true throughout the model collapse10 phase. As a result, the generator consistently produces similar outputs, which makes the generation lack variation. This behaviour is an undesirable situation during the GANs' training phase. We employ batch normalisation, which has been shown to lower the likelihood of mode collapse, to prevent the aforementioned problem. Nevertheless, as indicated by, batch normalisation is avoided in the discriminator and generator's first layer as well as its final layer.

2) All Convolutional Net: Strided convolutions are utilised rather than spatial pooling. Hence, strided convolutions enable the model to learn its own upsampling and downsampling rather than relying on predetermined downsampling and upsampling. This method has been demonstrated to improve training efficiency and aid in the network's learning of key invariances using only convolutional layers.

3) Leaky ReLu Activation Function: We have employed the LeakyReLU activation function whenever activation function is used since it performs better than conventional ReLU.

4) Mode Collapse avoidance strategy: In our work, we discovered that the generator produced images utilising the same colour scheme and colour grids at various points during the training. This was the result of "mode collapse," as it was previously described. In addition to batch normalisation, we employed a cutting-edge strategy in which we trained the generator to avoid utilising the same colours repeatedly. In a way, this serves as the generator's "reverse training," as we instinctively instruct it to steer clear of consistently employing the same colours.

E. *Generate Output*

We have a grayscale layer as input once model training is complete, and we anticipate the model to forecast an RGB image as output.

IV. Result

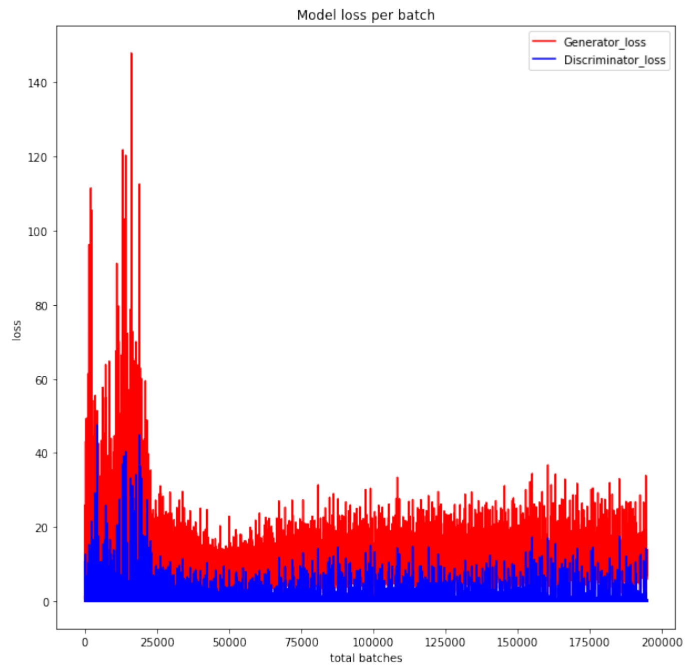

Fig. 4. Graphical representation of model loss per batch.

Fig. 4 shows the losses of both the generator and the discriminator of the implemented model. As can be observed from the graph, initially the generator loss is very high while the discriminator loss is very low thereby signifying the convergence failure problem and thus showing that the discriminator was winning at the start. As the number of batches increases, it can be seen that a state of near-equilibrium is achieved as the losses of both generators and discriminators seem to achieve a low value, and their relative difference also decreases.

We can deduce the following from observations collected from Fig. 5:

- In addition to producing promising results, the software provides some useful insights.
- Even though the output images may not represent the ground truth colors, they are vivid and accurate.
- In terms of performance, the software was able to handle a variety of images, such as human life, infrastructure, nature, etc.
- As the random inputs demonstrate, overfitting is not present.

V. Conclusions

The accuracy of the images mainly depends on the architecture of the model used and the training of the model, and

Authorized licensed use limited to: Northeastern University. Downloaded on September 16,2024 at 20:56:16 UTC from IEEE Xplore. Restrictions apply.

Fig. 5. Colorization of input image with respect to epochs.

as you can see the more promising results were obtained as the number of epochs increased. With the help of generative adversarial networks, we were able to successfully automatically colourize grayscale photos to a visually acceptable level. The GAN-created synthetically coloured images resembled the source pictures rather well. Not only is increasing the number of epochs sufficient, but we also need a large dataset to avoid overfitting problems. Efficient use of resources saves time and improves accuracy. For example, using the system GPU instead of the CPU saves hours. Adding more convolutional layers makes the model more accurate. Even after adding a significant number of convolutional layers, the output size is still large enough for efficient processing. In this case, pooling or downsampling is a good candidate to further compress the output of the convolutional layer without significant loss of information. Although our model can be used effectively for coloring small images, they are not reliable because it takes a long time to work with large images under normal conditions and increases the cost requirements.

VI. Future Scope

- Color Precision - Although the results provided by the color accuracy model were satisfactory, the still needs more model training to improve accuracy.
- With the advent of better deep learning methods, such as DCGAN (Deep Convolutional Generative Adversarial Network), models can be further refined to improve color accuracy.
- The ideas proposed in this paper can be extended to coloring in images, old movies, videos, and CCTV footages.
- We also want to add more convolution and pooling layers to make the model more accurate.

References

[1] Vivek George Jacob, Sumana Gupta. ”Colorization of grayscale images and videos using a semiautomatic approach” , 2009 16th IEEE International Conference on Image Processing (ICIP), 2009.

[2] S. Kumar, M. Kumar,” A study on the image detection using convolution neural networks and TenserFlow”. International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 1080-1083), 2018, IEEE.

[3] A. Bhowmik, S. Kumar, & N. Bhat,”Evolution of automatic visual description techniques-a methodological survey”. Multimedia Tools and Applications, 80(18), 28015-28059 (2021).

[4] S. Kumar, & R. Hanot (2021) ”Community detection algorithms in complex networks: A survey”, In Advances in Signal Processing and Intelligent Recognition Systems, 6th International Symposium, (2020), Springer Singapore.

[5] A. Mallik, A. Khetarpal, & S. Kumar,”ConRec: malware classification using convolutional recurrence”. Journal of Computer Virology and Hacking Techniques, 18(4), 297-313, (2022).

[6] S. Anand, A. Mallik, & S. Kumar, ”Integrating node centralities, similarity measures, and machine learning classifiers for link prediction”. Multimedia Tools and Applications, 81(27), 38593-38621, (2022).

[7] S. Kumar, A. Mallik, & B. S. Panda,”Influence maximization in social networks using transfer learning via graph-based LSTM”. Expert Systems with Applications, 212, 118770, 2023

[8] A. Weltz. The Basics of Hand-Coloring Black-and-White Prints. Accessed: 2020. [Online]. Available: https://www.bhphotovideo.com/explora/photography/tips-and-solutions/the-basics-of-hand-coloringblack-and-white-prints.

[9] I. Žeger, S. Grgic, J. Vuković and G. Šišul, ”Grayscale Image Colorization Methods: Overview and Evaluation,” in IEEE Access, vol. 9, pp. 113326-113346, 2021, doi: 10.1109/ACCESS.2021.3104515.

[10] R. Zhang, J.-Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, and A. A. Efros, “Real-time user-guided image colorization with learned deep priors,” ACM Trans. Graph., vol. 36, no. 4, pp. 111, Jul. 2017.

[11] Kamyar Nazeri, Eric Ng, and Mehran Ebrahimi. 2018. Image Colorization using Generative Adversarial Networks. In book: Articulated Motion and Deformable Objects, pp.8594.

[12] I. Goodfellow, J. PougetAbadie, M. Mirza, B. Xu, D. WardeFarley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in Neural Information Processing Systems 27, Montreal, Quebec, Canada, 2014, pp. 26722680.

[13] S. Wan, Y. Xia, L. Qi, Y. H. Yang and M. Atiquzzaman, ”Automated Colorization of a Grayscale Image With Seed Points Propagation,” in IEEE Transactions on Multimedia, vol. 22, no. 7, pp. 17561768, July 2020, doi: 10.1109/TMM.2020.2976573.

[14] Hao Wang, Qifeng Ruan, Hongtao Wang, Soroosh Daqiqeh Rezaei, Kevin T. P. Lim, Hailong Liu, Wang Zhang, Jonathan Trisno, John You En Chan, and Joel K. W. Yang, “Full Color and Grayscale Painting with 3D Printed Low-Index Nanopillars”, Nano Letters 2021 21 (11), 4721-4729.

[15] N. Bhat, N. Saggu, & S. Kumar S,” Generating visible spectrum images from thermal infrared using conditional generative adversarial networks”. In 2020 5th International Conference on Communication and Electronics Systems (ICCES) (pp. 1390-1394),. 2020, IEEE

[16] X. Zhong, T. Lu, W. Huang, M. Ye, X. Jia and C. -W. Lin, ”Grayscale Enhancement Colorization Network for Visible-Infrared Person Re-Identification,” in IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 3, pp. 1418-1430, March 2022.

[17] J. An, K. K. Gagnon, Q. Shi, H. Xie and R. Cao, ”Image Colorization with Convolutional Neural Networks,” 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Suzhou, China, 2019, pp. 1-4.

[18] An, J., Kpeyiton, K.G. & Shi, Q. Grayscale images colorization with convolutional neural networks. Soft Comput 24, 4751–4758 (2020).

Authorized licensed use limited to: Northeastern University. Downloaded on September 16,2024 at 20:56:16 UTC from IEEE Xplore. Restrictions apply.