I.J. Image, Graphics and Signal Processing, **2024, 2, 70-82**
Published Online on April 8, 2024 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijigsp.2024.02.06

Grayscale Image Colorization Method Based on U-Net Network

Zhengbing Hu
School of Computer Science, Hubei University of Technology, Wuhan, 430079, China
E-mail: drzbhu@gmail.com
ORCID iD: https://orcid.org/0000-0002-6140-3351

Oksana Shkurat*
Department of Computer Systems Software, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, 03056, Ukraine
E-mail: shkurat@pzks.fpm.kpi.ua
ORCID iD: https://orcid.org/0000-0001-7633-9121
*Corresponding Author

Maksym Kasner
Igor Sikorsky Kyiv Polytechnic Institute /Department of Computer Systems Software, Kyiv, 03056, Ukraine
E-mail: kasner2001max@gmail.com
ORCID iD: https://orcid.org/0009-0000-9173-092X

Received: 10 October, 2023; Revised: 26 November, 2023; Accepted: 05 January, 2024; Published: 08 April, 2024

Abstract: A colorization method based on a fully convolutional neural network for grayscale images is presented in this paper. The proposed colorization method includes color space conversion, grayscale image preprocessing and implementation of improved U-Net network. The training and operating of the U-Net network take place for images represented in the space of the *Lab* color model. The trained U-Net network integrates realistic colors (generate data of *a* and *b* components) into grayscale images based on *L*-component data of the *Lab* color model. Median cut method of quantization is applied to *L*-component data before the training and operating of the U-Net network. Logistic activation function is applied to normalized results of convolution layers of the U-Net network. The proposed colorization method has been tested on ImageNet database. The evaluation results of the proposed method according to various parameters are presented. Colorization accuracy by the proposed method reachers more than 84.81%. The colorization method proposed in this paper is characterized by optimized architecture of convolution neural network that is able to train on a limited image set with a satisfactory training duration. The proposed colorization method can be used to improve the image quality and restoring data in the development of computer vision systems. The further research can be focused on the study of a technique of defining optimal number of the gray levels and the implementation of the combined quantization methods. Also, further research can be focused on the use of *HSV*, *HLS* and other color models for the training and operating of the neural network.

Index Terms: Image Colorization, Convolution Neural Network (CNN), U-Net Network, Gray Level Quantization, Grayscale Image.

1. Introduction

A lot of problems in image processing, computer vision, and computer graphics can be posed as “improvement” of an input image into a corresponding output image. Traditionally, each of these problems has been solved with separate, special-purpose methods, which are based on image denoising, contrast enhancement, image restoration etc. However, the general task is always the same is to predict pixels from pixels. One of the such image “improvement” problem is an image colorization.

An image can be captured in the presence of the unsatisfactory illumination. Then an image is characterized by:

- low contrast;
- low brightness and lightness;
- low color saturation;

- false color distribution;
- loss of the graphical information;
- appearance of the various distortions;
- false differentiation of the objects and background;
- segmented contours, etc.

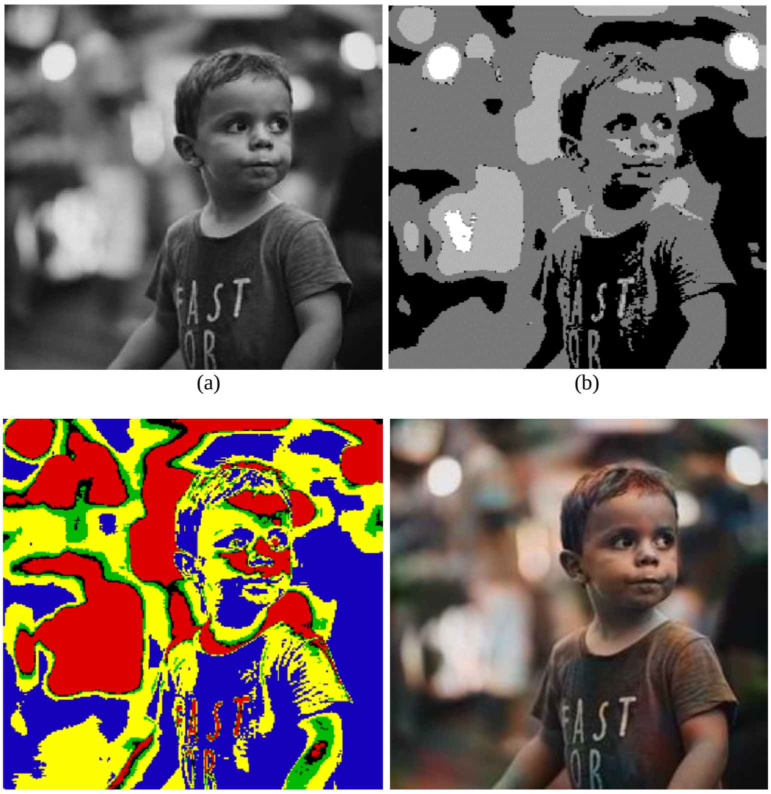
Image colorization is one of the ways to solve these problems. Colorization is a necessary component of visual effects in the film industry, in the archival restoration of valuable historical documents, photographs and videos, as well as in journalism, where color photographs and videos give events greater objectivity and the power to convey information. Image colorization is also an effective tool in the image recognition and computer vision systems [1-6].

Image colorization is a transformation of the image color distribution, color maps of images, generating a new color map etc. Grayscale image colorization is a transformation of a monochrome image into a full-color one with a distribution of natural colors close to the real one. Grayscale image colorization is accompanied by various problems. The main problem is the colorization objectivity. Since image objects can have different colors, there are many possible ways to assign colors to pixels in an image, which means there is no unique solution to this problem. Original grayscale images don't contain enough color information, so the colorization depends on subjective color choices. This can lead to false in the reproduction of real colors. The next problem is the colorization automation. The manual colorization is based on prior knowledge of input image content and manual adjustments of an image. Automatic colorization is based on machine learning techniques, which require a large database of training images. Also, colorization methods based on machine learning can be ineffective in a case of images with non-standard hue distribution or complex textures. The last problem is the colorization of complex images that are characterized by a significant number of color shades (for example, human skin). Such images are difficult to reproduce automatically without losing the naturalness of color distribution (for example, human emotions).

Thus, image colorization defines new opportunities in image processing and reproduction, and there are a sufficient number of problems, which makes the development of the automatic colorization method an actual scientific and practical task.

2. Related Works

There are several approaches to color redistribution of images, in particular gray level quantization, image indexing and image colorization (see Fig. 1).



(a) (b) (c) (d)

Fig. 1. The approaches to image color redistribution: (a) input image; (b) gray level quantization; (c) image indexing; (d) image colorization.

Quantization is based on the redistribution of color component values of image (brightness, lightness) according to defined gray levels. Gray levels are defined by means of image processing methods. There are several approaches to quantization, in particular histogram analysis, threshold method, median method, k-means etc. Image indexing is based on the redistribution of image color according to a transformation result. Such image transformation can be segmentation, contour detecting, object retrieval etc. The color distribution of the output image is changed according to the result of an image transformation method and visualized by means of defined color map. Image colorization is based on to define the image color distribution according to the visualization of real world. The colorization process is based on to define the color coordinate values for each pixel of the image based on its brightness values and the context of the surrounding pixels. The considered approaches to image color redistribution can solve various problems in computer vision systems. The method proposed in this paper is based on the technology of colorization and quantization of gray levels.

The first colorization approach is so-called manual colorization. The manual methods are based on to choose the color in each pixel of the image [7]. In [8], the Bugeau et all developed a manual colorization method based on using pre-defined color inputs and considering an entire colored image as a color example to transfer. The developed method is based on patch descriptors of luminance features and a color prediction model with a general distance selection strategy. In [9], the Welsh et all developed an automated colorization method based on reference color images using the luminance and texture descriptors.

The second colorization approach is so-called exemplar-based colorization. The exemplar-based image colorization methods are based on the automatic color providing from color image to grayscale image [7].

In [10], the Persch et all developed exemplar-based colorization method for image face, which is based on image morphing. This technique is based on the geometric structure of the images and be able to compute a correspondence map between images with similar shapes.

In [11], the Kawulok et all developed manual colorization method based on added scribbles. The developed method is based on determine color propagation paths in the image by minimizing the geodesic distance from the scribbles using Dijkstra algorithm. After that, chrominance blending is performed to colorize the image.

In [12], the Levin et all developed a colorization method based on optimization of a quadratic cost function for a simple premise, that neighboring pixels in space-time that have similar intensities should have similar colors.

In [13], the Irony et all presented a colorization method based on added scribbles. The developed method is based on transferring color from a segmented example image. Firstly, for each pixel is determined which example segment it should learn its color from. In the next, each pixel is assigned a color from the appropriate region using a neighborhood matching metric, combined with spatial filtering for improved spatial coherence.

In [14], the Morimoto et all developed automatic colorization method based on reference color images. The developed method generates various and natural colorized images from an input monochrome image by using the information of the scene structure defined from collected several web images.

In [15], the Charpiat et all developed an automatic colorization method based on reference color images. The developed method based on probability distribution of all possible colors, instead of choosing the most probable color at the local level. In the developed method is predicted the expected variation of color at each pixel and used graph cuts to maximize the probability of the whole-colored image at the global level.

In [16], the Gupta et all developed automated colorization method based on reference color images. As input, the user needs to supply a reference color image which is semantically similar to the target image. The developed method extracts various descriptors from superpixel segmentation and matches them with the ones of the target image with these various descriptors.

In [17], the Deshpande et all developed automatic colorization method based on scene labels. In the developed method is used a LEARCH framework to train a quadratic objective function in the chromaticity maps, comparable to a Gaussian random field.

In [18], the Welsh et all considered path-based colorization methods. For each pixel of the grayscale image, the methods compare the patch centered in this pixel with a set of patches extracted from the luminance channel of the source image [7].

The third colorization approach is based on machine learning techniques. Convolutional Neural Networks (CNN), Generative Adversarial Networks (GAN), Recursive Neural Networks are used in such colorization methods [7].

In [19], the Zhang et all developed CNN-based colorization method, which is based on defining a set of possible colors with associated probabilities for each pixel of a down-sampled image version. Define a color for each pixel in high resolution image version is performed by linear interpolation. Colorization result of proposed method is to compute an annealed-mean in each pixel, independently. CNN-model has been trained on over a million color images of ImageNet. The authors evaluate their method using a “colorization Turing test,” asking human participants to choose between a generated and ground truth color image. Proposed method successfully fools humans on 32% of the trials.

In [20], the Iizuka et all developed CNN-based colorization method, which is based on merge of the local information dependent on image patches with global priors computed using the entire image. Authors train their model on the Places scene dataset, which consists of 2 448 872 training images and 20 500 validation images. Authors filter the images by removing grayscale images and those that have little color variance with a small automated script. They train model using the batch size is 128 for 200 000 iterations, corresponding to roughly 11 epochs. This takes roughly 3 weeks (on one core of an NVIDIA Tesla K80 GPU). Authors evaluate their model in a user study and find that the output of their model is considered "natural" 92.6% of the time.

In [21], the Larsson et all developed a colorization method which is based on CNN that is trained end-to-end to incorporate semantically meaningful features and the stage of post-processing, a color histogram prediction framework that handles uncertainty and ambiguities inherent in colorization while preventing jarring artifacts. Authors train model for one epoch on the 1.2 million images of the ImageNet training set, each resized to at most 256 pixels in smaller dimension. A single epoch takes approximately 17 hours on a GTX Titan X GPU. At test time, colorizing a single 512x512 pixel image takes 0.5 seconds. The colorization result is 0.165 in terms of PMSE performance.

In [22], the Royer et all developed a probabilistic method of grayscale image colorization, the architecture of which consists of two networks. Deep convolution network for colorizing natural gray images and autoregressive network to produce the final colorization based on modeling complex interactions between image pixels. Authors train model on the CIFAR-10 dataset, which contains 50 000 training images and 10 000 test images of 32x32 pixels, and on ImageNet ILSVRC 2012. This dataset has 1.2 million high-resolution training images spread over 1000 different semantic categories, and a hold-out set of 50000 validation images. In author's experiments are rescaled all images to 128x128 pixels. In the paper is demonstrated the result of colorization, but does not specify the quantitative and qualitative evaluation of colorization.

In [23], the Isola et all proposed implementation of GAN architecture to transform like as image-to-image in particular image colorization, when input and output data is an image. Proposed method based on two CNN, the U-net model to generate data, the PatchGAN to discriminate data. PatchGAN classify structure at the scale of image patches with a $N \times N$ size.

In [24], the Cao et all proposed implementation of GAN architecture to grayscale image colorization. Proposed method consists of novel generator architecture based on CNN with multi-layer noise and multi-layer condition concatenation and the discriminator architecture.

In [25], the Baldassarre et all proposed implementation of two CNN to grayscale image colorization. Instead of training a feature extraction branch from scratch, authors make use of an Inception-ResNetv2 network. The approximately 60 000 images of ImageNet were used for a training model, the optimizer algorithm is Adam. The training of model has been lasted approximately 23 hours. The accuracy of coloration is 54.13% for the qualitative estimation which has been realized as user poll.

In [26], the Nazeri et all proposed implementation of GAN to grayscale image colorization. The approximately 50 000 images of CIFAR-10 were used for a training model, the image size is 32x32 pixels. The approximately 1.8 million images of Places365 were used for a training model, the image size is 256x256 pixels. The accuracy of coloration is 92.5% to employ mean absolute error for Places365 dataset. The accuracy of coloration is 94.9% to employ mean absolute error for CIFAR-10 dataset. Authors also obtained mixed results when colorizing grayscale images using the Places365 dataset. Mis-colorization was a frequent occurrence with images containing high levels of textured details. Network for Places365 dataset was not as well-trained as for CIFAR-10 counterpart due to its significant increase in resolution (256 ×256 versus 32×32) and the size of the dataset (1.8 million versus 50 000).

In [27], the Hong et all proposed implementation of generative model to automatic colorization problem. In experiments of prior learning, authors have been used 150 000 images randomly picked from LSUN, COCO-stuff, ImageNet datasets. Authors reshape each image into 128x128 pixels as preprocessing. Adam is chosen as an optimizer with a learning rate of 0.005 and halved every 5,000 iterations. Subsequently, the proposed model is trained for $1e5$ iterations with the batch size of 8 that takes around 20 hours. The colorization results are 29.46 and 0.9365 in terms of PSNR and SSIM performance.

Thus, the grayscale image colorization is based on priors features which can be extracted from an example image by manual or automatic intervention or can be extracted from a large dataset of color images by means of machine learning techniques.

In this paper, effectively employ CNN model with grayscale image preprocessing method to improve the performance of automatic colorization and the visual perception of colorized images is studied. The proposed colorization method is based on optimized gray-level images and architecture of CNN, that is trained on a limited dataset with a satisfactory training duration.

3. Method Description

The input data of the proposed colorization method is grayscale images. These grayscale images of the size $N \times N$ can be represented in the space of RGB color model with the certain color entropy by three-dimensional matrix $I(i,j,k) = c$, where c is pixel color value according to k-component, $c \in [0;1]$; (i,j) are pixel coordinates, $i = \overline{1,N}$, $j = \overline{1,N}$; k is an order number of the color component of RGB model, $k = 1,2,3$, according to red, green and blue components.

The proposed method of grayscale image colorization includes the following steps:

- Color space conversion
- Grayscale image preprocessing

– Implementation of improved U-Net network
– Inverse color space conversion.

3.1. Color Space Conversion

The color space conversion from *RGB* model into *Lab* model [25] is used in the proposed colorization method.

Step 1. Convert the color space from *RGB* model into *XYZ* model (1).

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

where $R, G, B \in [0; 1]$;
$X \in [0; 0.95], Y \in [0; 1], Z \in [0; 1.089]$.

Step 2. Normalize components $X = \frac{X}{0.95}$ and $Z = \frac{Z}{1.089}$, that their values belong to the range $[0; 1]$.

Step 3. Convert the color space from *XYZ* model into *Lab* model (2).

$$\begin{gathered} L = \begin{cases} 116 \times Y^{\frac{1}{3}} - 16, \ Y > 0.008856 \\ 903.3 \times Y, Y \leq 0.008856 \end{cases} \\ a = 500 \times \left(X^{(T)} - Y^{(T)}\right) \\ b = 200 \times \left(Y^{(T)} - Z^{(T)}\right) \end{gathered} \tag{2}$$

where

$$Component^{(T)} = \begin{cases} Component^{\frac{1}{3}}, Component > 0.008856 \\ 7.787 \times Component + \frac{16}{116}, Component \leq 0.008856 \end{cases};$$

$Component = \overline{X; Y; Z}$;
$L \in [0; 100], a, b \in [-127; 127]$.

To represent images by 8 bits depth and integer data type: $L = L \times \frac{255}{100}, a = a + 128, b = b + 128$.

The color space of *Lab* model is formed by lightness component (*L* component) and color components (*a* and *b* components). The color range for *a* component is from green (negative component values) to red (positive component values). The color range for *b* component is from blue (negative component values) to yellow (positive component values).

Lab color model is described by wide range of color values compared to others models (*HSV*, *HLS* etc.) and single component of lightness. Based on *Lab* color model, a trained neural network can integrate more realistic colors (generate data of *a* and *b* components) into grayscale images based on *L*-component data.

The main result of this stage is *L*-component data which can be visualized by grayscale image and represented by two-dimensional matrix $I_L(i,j) = c_L$, where c_L is pixels intensity of *L*-component, $c_L \in [0; 255]$; (i,j) are pixel coordinates, $i = \overline{1, N}$, $j = \overline{1, N}$.

3.2. Grayscale Image Preprocessing

Gray level quantization is used to preprocess images. Gray level quantization is applied to the $I_L(i,j)$ image. The quantization method is Heckbert's median cut method, originally described by Paul Heckbert in [29].

Gray level quantization of proposed colorization method includes the following steps:

Step 1. Define the number of unique gray levels.

Let the number of unique gray levels is 8 for some fragment of an image (see Fig. 2).

49	51	49	28	12
72	62	45	17	27
91	107	128	100	130
91	173	173	200	200
170	190	187	221	255

Fig. 2. The grayscale image fragment.

Step 2. Rank values of image pixel intensity.

The result for Fig. 2 is

$\{12; 17; 27; 28; 45; 49; 49; 51; 62; 72; 91; 91; 100; 107; 128; 130; 170; 173; 173; 187; 190; 200; 200; 221; 255\}$.

Step 3. Define the median value of the array and divide the array on two pats, cutting the median value.

The median value for Fig. 2 is 100. The first part of the array is $\{12; 17; 27; 28; 45; 49; 49; 51; 62; 72; 91; 91\}$. The second part of the array is $\{107; 128; 130; 170; 173; 173; 187; 190; 200; 200; 221; 255\}$.

Step 4. Repeat steps 2 and 3 until the number of arrays equals the number of unique gray levels defined in step 1.

The result for Fig. 2 is shown on Fig. 3.

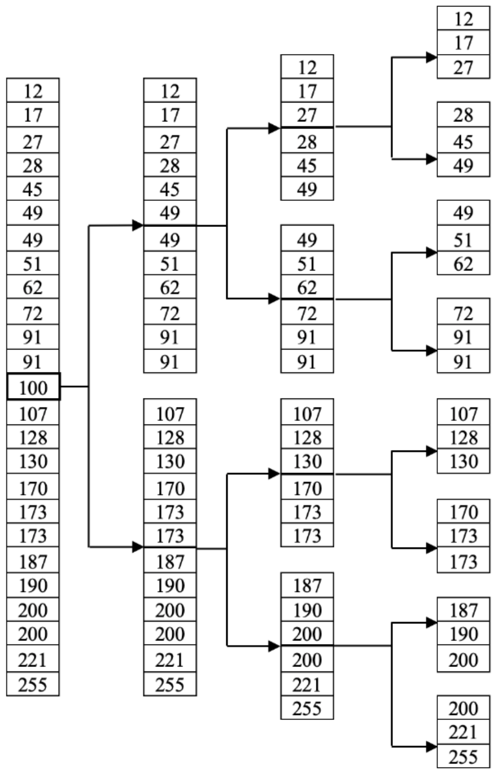


Fig. 3. The unique gray levels study.

Step 5. Define the average values of obtained arrays. These are values of unique gray levels.

The result for Fig. 2 is $\{19; 41; 54; 85; 122; 172; 192; 225\}$.

Step 6. Compute the new values of image pixel intensity according to the minimum distance rule with defined values of unique gray levels.

Fig. 4 is shown the result gray level quantization.

54	54	54	19	19
85	54	41	19	19
85	122	122	85	122
85	172	172	192	192
172	192	192	225	225

Fig. 4. The result of gray level quantization for the image fragment (see Fig. 2).

Applying gray level quantization is optimized training and operating of the convolution neural network, training time is reduced, colorization accuracy is increased. However, there is a problem to define an optimal number of gray levels, an effective quantization algorithm, an acceptable quantization algorithm complexity etc.

3.3. U-Net Architecture

U-Net is an architecture of Fully Convolutional Network (FCN) formed by a downsampling and an upsampling parts. Fig. 5 is shown the diagram of U-Net network, which is implemented for proposed colorization method.

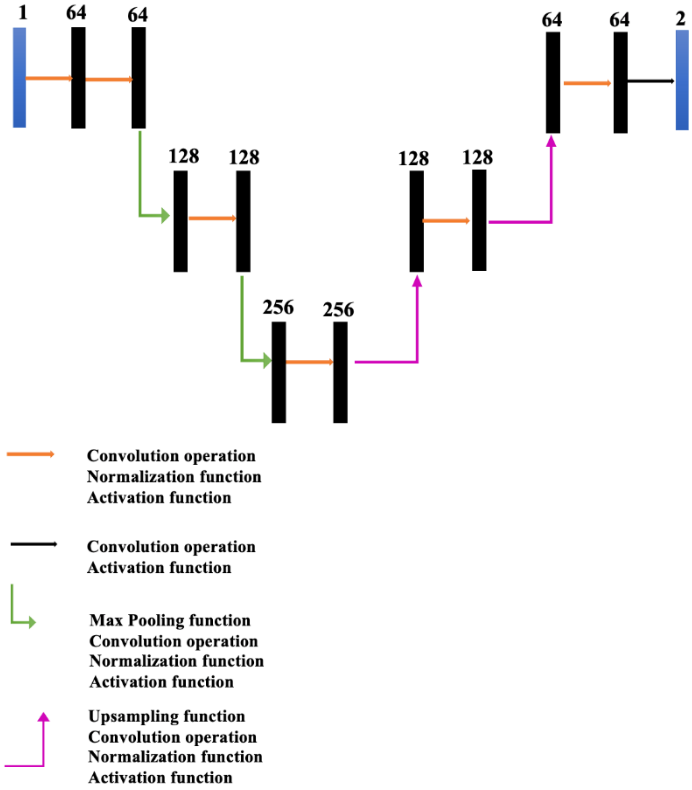


Fig. 5. The U-Net network diagram.

The downsampling part consists of convolution operation, normalization function, activation function and max pooling function. The upsampling part consists of convolution operation, normalization function, activation function and upsampling function.

The convolution operation is the main type of image transformation. The convolution operation is used to extract the image features and build the feature map. In implemented U-Net network, the convolution kernel is 3×3, the filter number of the 1-2 and 9-10 convolution blocks is 64, the 3-4 and 7-8 convolution blocks is 128, the 5-6 convolution blocks is 256 (see Fig. 6).

The application of activation function to the result of convolution layer increases the criterion of nonlinear system. In implemented U-Net network, the logistic activation function $\frac{1}{1+e^{-x}}$ is used. By means of applying logistic activation function the extracted image features are strengthened in the system and thus, they are able to influence the results of the deeper convolutional layers. Before using the activation function, it is mandatory to use the normalization function to obtain an adequate system response (see Fig. 6).

The aggregate function is used to create a downsampled feature map. There are several aggregate functions in particular max pooling, average pooling, ROI (region of interest) pooling etc. The max pooling function, which is based on define the maximum value based on the 2×2 size from the feature map, is used in implemented U-Net network (see Fig. 6).

The upsampling function is used to create an upsampled feature map. There are several interpolation methods for upsampling in particular nearest neighbor, Bed-of-Nails technique, max unpooling, bilinear interpolation, bicubic interpolation, Lanczos interpolation etc. The nearest neighbor interpolation function based on the 2×2 size is used in implemented U-Net network (see Fig. 6).

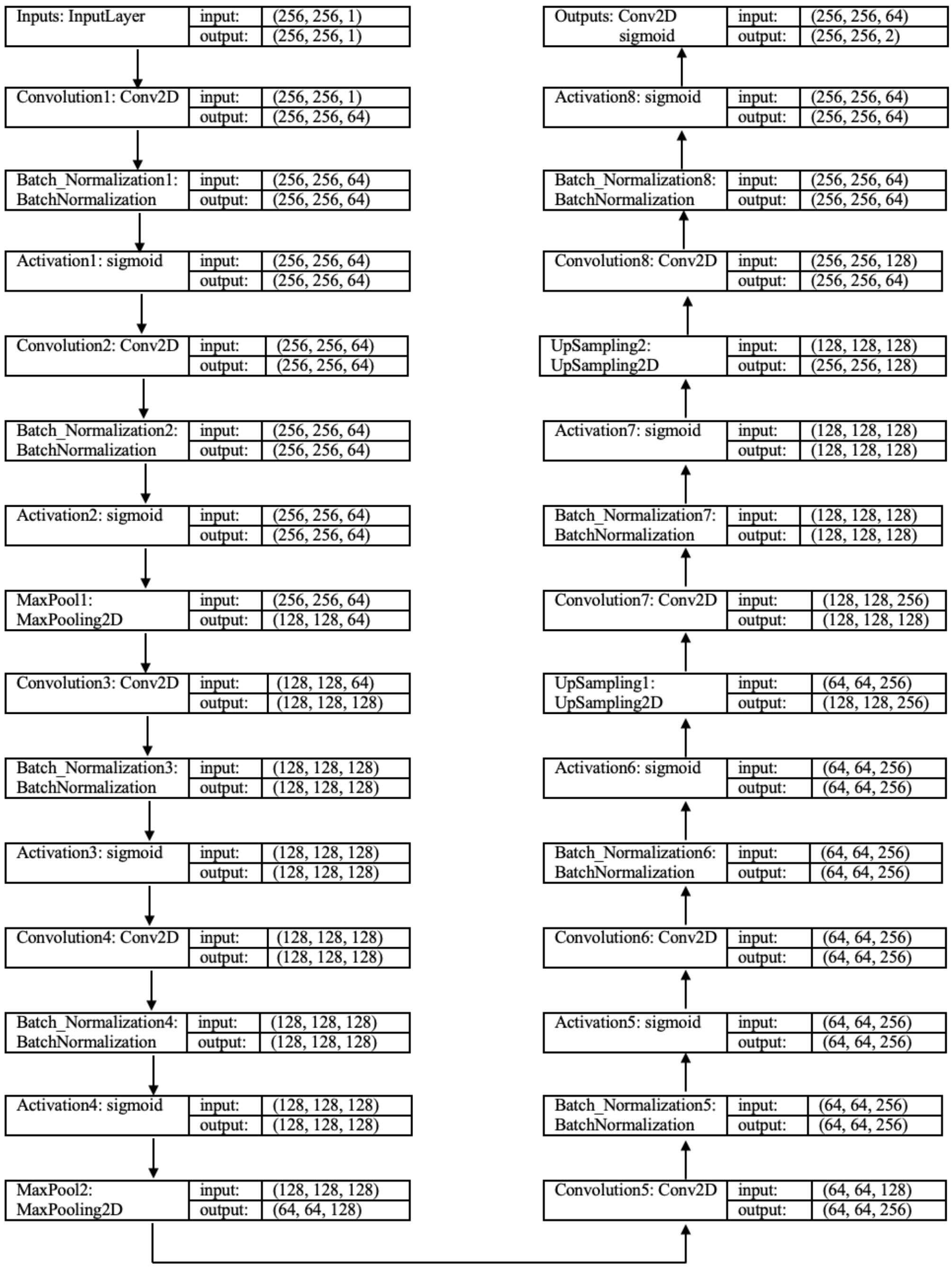


Fig. 6. Building the U-Net model.

3.4. *Inverse Color Space Conversion*

The color space conversion from *Lab* model into *RGB* model [28] is used in the proposed colorization method to visualize the final result.

Step 1. Convert the color space from *Lab* model into *XYZ* model (3).

$$X=\begin{cases} f_x^3\times X_n, f_x>\sqrt[3]{0.008856} \\ \left(f_x-\frac{16}{116}\right)\times\frac{X_n}{7.787}, f_x\leq\sqrt[3]{0.008856} \end{cases}$$
$$Y=\begin{cases} \left(\frac{L+16}{116}\right)^3, f_y>\sqrt[3]{0.008856} \\ \frac{L}{903.3}, f_y\leq\sqrt[3]{0.008856} \end{cases} \quad (3)$$
$$Z=\begin{cases} f_z^3\times Z_n, f_z>\sqrt[3]{0.008856} \\ \left(f_z-\frac{16}{116}\right)\times\frac{Z_n}{7.787}, f_z\leq\sqrt[3]{0.008856} \end{cases}$$

where $X_n = 0.95, Z_n = 1.089$;
$f_x = f_y + \frac{a}{500}, f_y = \frac{L+16}{116}, f_z = f_y - \frac{b}{200}$.

Step2. Convert the color space from *XYZ* model into *RGB* model (4).

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.240479 & -1.53715 & -0.498535 \\ -0.969256 & 1.875991 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{4}$$

where $R, G, B \in [0; 1]$.

4. Practical Implementation

The ImageNet [30] has been used for testing the proposed colorization method. The ImageNet contains 1 281 167 training images, 50 000 validation images and 100 000 test images. The ImageNet is organized according to nature pictures, people, buildings etc. In order to simplify training and reduce running times, only a small subset is used.

The dataset provides 800 training images and 100 validation images of .jpg format. The size of the images is 256x256 pixels. Each image was reduced down to *L*-component data of the *Lab* image to use as a grayscale input, and the original RGB image was used as the target. The training data was further augmented by the gray-level quantization with the:

– 210 unique gray levels;
– 220 unique gray levels;
– 230 unique gray levels.

The results of applying the proposed method are shown in Fig. 7- Fig. 10. The results obtained on identical parameters and their values are the 50 batch size, 2500 iterations, the Adam optimizer.

The model has been implemented with the Keras software library, the TensorFlow deep learning platform and Google Colaboratory, leveraging the Tesla K80 Accelerator GPU to speed up the computations. All training was carried out less than 7 hours from start to finish, with the progressive changing the gray-level for the quantization.

Fig. 7. Examples of applying the proposed colorization method. The method parameters: the batch size is 50, iterations - 2500, Adam optimizer. The dataset provides 800 training images and 100 validation images. The gray level quantization has not been applied.

Fig. 8. Examples of applying the proposed colorization method. The method parameters: the batch size is 50, iterations - 2500, Adam optimizer. The dataset provides 800 training images and 100 validation images. The gray level quantization has been applied. The number of unique gray levels is 210.

Fig. 9. Examples of applying the proposed colorization method. The method parameters: the batch size is 50, iterations - 2500, Adam optimizer. The dataset provides 800 training images and 100 validation images. The gray level quantization has been applied. The number of unique gray levels is 220.

Fig. 10. Examples of applying the proposed colorization method. The method parameters: the batch size is 50, iterations - 2500, Adam optimizer. The dataset provides 800 training images and 100 validation images. The gray level quantization has been applied. The number of unique gray levels is 230.

RMSE (Root Mean Squared Error), PSNR (Peak Signal-to-Noise Ratio), PA (Pixel Accuracy) criterions have been computed to evaluate the efficiency of proposed colorization method. The evaluation results of the proposed method according to various parameters are shown in Table 1.

Colorization accuracy of the proposed method reachers more than 84.81% in case of applying gray level quantization with the number of gray levels is 220. Although the quantitative evaluation obtains a measure of the performance of the proposed colorization method, there are also interested in how compelling the colors look to a human observer.

Table 1. Proposed Colorization Method Evaluation.

Parameters of the Proposed Colorization Method	Evaluation Criterion		
	RMSE	PSNR	PA
U-Net network Activation function: $\max(0; x)$ Gray level quantization: None	0.1311	26.4190	0.0672
U-Net network Activation function: $\frac{1}{1+e^{-x}}$ Gray level quantization: None	0.1294	26.1365	0.0589
U-Net network Activation function: $\frac{1}{1+e^{-x}}$ Gray level quantization: 210 levels	**0.1244**	27.1280	0.0561
U-Net network Activation function: $\frac{1}{1+e^{-x}}$ Gray level quantization: 220 levels	0.1266	**27.4745**	**0.0546**
U-Net network Activation function: $\frac{1}{1+e^{-x}}$ Gray level quantization: 230 levels	0.1287	27.3249	0.0589

The qualitative measure is performed to evaluate the appearance of some artificially recolored images by means of a user opinion. To this end, there has been chosen several images, which are shown in Fig. 9 and asked, for each of them, the question “Recolored or real image?”. The poll was taken by 16 different users. The results of the qualitative evaluation are shown in Fig 11.

43.75% 31.25% 87.5%

Fig. 11. For each recolored image, is given the percentage of users that answered "real" to the question "Recolored or real image?"

Overall, it is computed that 54.17% of the users classified recolored images as originals. However, the recolored images for the user study were carefully selected from best results.

It is important to highlight the proposed colorization method limitations or where it can't generalize well. The proposed colorization method has been used on the structured dataset by the object type of image, in particular people. Even if more training images are used, it will be difficult for the proposed colorization method to predict the color of the eyes, hair and skin tone. In such cases, the proposed colorization method encourages using grayish colors when it is not sure about what the object is or what color it should be. Using Generative Adversarial Network (GAN) can be a solution to this problem, which can complicate the proposed colorization method.

5. Conclusion and Future Work

The determination of color distribution in grayscale images is performed by implementing the U-Net network, which is trained and operated in space of *Lab* color model with defined range of lightness. The range of lightness is defined in the method of gray level quantization. So, in the proposed colorization method employ CNN model with grayscale image preprocessing method to improve the performance of automatic colorization and the visual perception of colorized images.

The proposed colorization method is based on optimized gray-level images and architecture of CNN, that is trained on a limited dataset with a satisfactory training duration. Colorization accuracy of the proposed method reachers more than 84.81% in case of applying gray level quantization with the number of gray levels is 220. For qualitative evaluation, it is computed that 54.17% of the users classified recolored images as originals.

The further research can be focused on use of other color models for training and operating of the neural network. When using the *Lab* color model, the *L*-component data is used by the model to predict the other two component data (*a* and *b*). The result of the model prediction is a colorful image. When using the *RGB* color model, to convert the input image into a grayscale image is needed firstly. A grayscale image is used by the model to predict data of three components (*R*, *G* and *B*), which is a way more difficult and unstable task due to the many more possible combinations of three numbers compared to two numbers. In the case of 256 unique component values (in an 8-bit unsigned integer image this is the number of brightness choices), predicting the three numbers for each of the pixels is choosing between 256^3 combinations or 16 777 216 choices, while predicting the two numbers is choosing between 65 536 choices. When using the *HSV* or *HLS* color models, the neural network has to predict data of a single component (*Hue* component), however, the training of the neural network is based on much more data, on the data of two components (*Saturation* and *Brightness* components or *Saturation* and *Lightness* components).

The further research can be focused also on study of combined quantization methods, which can increase the colorization accuracy for the case of using non-structured dataset by the image type in the training.

References

[1] Z. Wang, H. Li, and X. Zhang X, "Construction waste recycling robot for nails and screws: Computer vision technology and neural network approach", *Automation in Construction*, Vol. 978, pp. 220-228, 2019.

[2] T.A. Alani, M.Q. Abbood, K.A. Abbar, "Computer Vision-Based System for Classification and Sorting Color Objects," *IOP Conference Series Materials Science and Engineering*, Vol. 745, 2020.

[3] O. Shkurat, Y. Sulema, V. Suschuk-Sliusarenko, A. Dychka, "Image Segmentation Method Based on Statistical Parameters of Homogeneous Data Set," *International Conference of Artificial Intelligence, Medical Engineering, Education*, p.271-281, 2018.

[4] O.S. Shkurat, Ye.S. Sulema, A.I. Dychka, "Complicated Shapes Estimation Method for Objects Analysis in Video Surveillance System," *KPI Science News*, pp. 53–62, 2018.

[5] V.E. Neagoe, "An Optimum 2D Color Space for Pattern Recognition," *International Conference on Image Processing, Computer Vision, & Pattern Recognition*, Las Vegas, USA, 26–29 June 2006, Vol. 2, 2006.

[6] Y. Tayal, R. Lamba, S. Padhee, "Automatic face detection using color based segmentation," *International Journal of Scientific Research*, Vol. 2, pp. 1-7, 2012.

[7] Fabien Pierre, Jean-François Aujol, "Recent Approaches for Image Colorization," in *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging*. Springer International Publishing, 2020.
[8] A. Bugeau, V.-T. Ta, N. Papadakis, "Variational Exemplar-Based Image Colorization," IEEE *Transactions on Image Processing*, Vol.23(1), pp. 298-307, 2014.
[9] T. Welsh, M. Ashikhmin, K. Mueller, "Transferring color to greyscale images," *ACM Transactions on Graphics (TOG)*, Vol. 21(3), pp. 277-280, 2002.
[10] J. Persch, F. Pierre, G. Steidl, "Exemplar-based face colorization using image morphing," *Journal of Imaging*, Vol. 3(4), 2017.
[11] M. Kawulok and B. Smolka, "Competitive image colorization," *IEEE International Conference on Image Processing (ICIP)*, p. 405–408, 2010.
[12] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *ACM Transactions on Graphics*, Vol. 23(3), pp. 689-694, 2004.
[13] R. Irony, D. Cohen-Or, D. Lischinski, "Colorization by example," *Eurographics Symp. On Rendering*, Vol. 2, 2005.
[14] Y. Morimoto, Y. Taguchi, and T. Naemura, "Automatic colorization of grayscale images using multiple images on the web," *ACM Transactions on Graphics*, 2009.
[15] G. Charpiat, M. Hofmann, and B. Schölkopf, "Automatic image colorization via multimodal predictions," *European Conference on Computer Vision (ECCV)*, 2008.
[16] R.K. Gupta, A.Y.S. Chia, D. Rajan, E.S. Ng, H. Zhiyong, "Image colorization using similar images," *ACM International Conference on Multimedia*, pp. 369–378, 2012.
[17] A. Deshpande, J. Rock, and D.A. Forsyth, "Learning large-scale automatic image colorization," *International Conference on Computer Vision (ICCV)*, 2015.
[18] T. Welsh, M. Ashikhmin, K. Mueller, "Transferring color to greyscale images," *ACM Transactions on Graphics*, Vol. 21(3), pp. 277–280, 2002.
[19] R. Zhang, P. Isola, A.A. Efros, "Colorful image colorization," *European Conference on Computer Vision*, pp. 1–16, 2016.
[20] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification," *ACM Transactions on Graphics*, 2016.
[21] G. Larsson, M. Maire, and G. Shakhnarovich, "Learning representations for automatic colorization," *European Conference on Computer Vision (ECCV)*, 2016.
[22] A. Royer, A. Kolesnikov, C.H. Lampert, "Probabilistic image colorization," *British Machine Vision Conference*, 2017.
[23] P. Isola, J.-Y. Zhu, T. Zhou, and A.A. Efros. "Image-to-image translation with conditional adversarial networks," *Computer Vision and Pattern Recognition (CVPR)*, 2017.
[24] Y. Cao, Z. Zhou, W. Zhang, and Y. Yu, "Unsupervised diverse colorization via generative adversarial networks,", 2017.
[25] F. Baldassarre, D.G. Morin, and L. Rodes-Guirao, "Deep Koalarization: Image Colorization using CNNs and Inception-Resnet-v2,", 2017.
[26] K. Nazeri, E. Ng, and M. Ebrahimi, "Image Colorization using Generative Adversarial Networks,", 2018.
[27] K.Hong, W. Li, C. Yang, M. Zhang, Y. Wang, and Q. Liu, "Joint Intensity-Gradient Guided Generative Modeling for Colorization," *IEEE Transactions on Image Processing*, 2020.
[28] OpenCV Documentation. URL: https://docs.opencv.org/3.4/.
[29] P. Heckbert, "Color image quantization for frame buffer display," *Computer Graphics*, vol. 16(3), pp. 297-307, 1982.
[30] ImageNet database. URL: https://www.image-net.org/.

Authors' Profiles

Zhengbing Hu, Prof., Deputy Director, International Center of Informatics and Computer Science, Faculty of Applied Mathematics, National Technical University of Ukraine "Kyiv Polytechnic Institute", Ukraine. Adjunct Professor, School of Computer Science, Hubei University of Technology, China. Visiting Prof., DSc Candidate in National Aviation University (Ukraine) from 2019. Major research interests: Computer Science and Technology Applications, Artificial Intelligence, Network Security, Communications, Data Processing, Cloud Computing, Education Technology.

Oksana Shkurat, Assistant professor at the Department of Computer Systems Software of the Faculty of Applied Mathematics at the Igor Sikorsky Kyiv Polytechnic Institute. Research interests: Image Processing Technologies, Computer Vision Technologies, Artificial Intelligence.

Maksym Kasner, Master at the Department of Computer Systems Software of the Faculty of Applied Mathematics at the Igor Sikorsky Kyiv Polytechnic Institute. Research interests: Software Engineering, Computer Vision Technologies.

How to cite this paper: Zhengbing Hu, Oksana Shkurat, Maksym Kasner, "Grayscale Image Colorization Method Based on U-Net Network", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.16, No.2, pp. 70-82, 2024. DOI:10.5815/ijigsp.2024.02.06