Journal of Science, Computing and Engineering Research (JSCER)
Volume-6, Issue- 4, April 2023.
DOI: *https://doi.org/10.46379/jscer.2023.060406*
ISSN: 2708-1079

# B&W Image Colorization Using Computer Vision And Deep Learning

**Dr Palson Kennedy R, Jonisha S, Bersikin Libina T, Vasantha Raja S S**

[1,2,3,4,] Department of CSE, PERI Institute of Technology, Chennai.

**Article Information**

Received : 30 Jan 2023
Revised : 02 Mar 2023
Accepted : 18 Mar 2023
Published : 08 April 2023

**Abstract—** Previous approaches to black and white image colorization relied on manual human annotation, which frequently resulted in desaturated results that were not believable as true colorizations. The project attempts to generate a plausible color version of a greyscale photograph given as input. It's a fully automated process for creating beautiful, lifelike colorization. By framing the challenge as a classification job, it accepts the problem's underlying ambiguity and uses class re-balancing during training to increase the diversity of colors in the end result. The system is trained on over a million color images and is implemented as a feed-forward pass in a CNN during testing. By a large margin, this strategy surpasses earlier methods. It also shows that colorization can be an effective pretext task for self-supervised feature learning when employed as a cross-channel encoder. On a variety of feature learning benchmarks, this technique achieves cutting-edge results

*Corresponding Author:*
*Dr Palson Kennedy*

**Keywords:** ***Colorization, CNN, VGG, Deep learning, Image Coloring***



**Citation: Dr Palson Kennedy R, Jonisha S, Bersikin Libina T, Vasantha Raja S S**. "B&W Image Colorization Using Computer Vision And Deep Learning" Journal of Science, Computing and Engineering Research, 6(4), 18-23, 2023.

## I. INTRODUCTION

Grayscale refers to digital photographs in which each pixel's value represents solely the light's intensity information. Only the darkest black to the brightest white are often displayed in such images. To put it another way, the image only features black, white, and grey hues, with grey having many levels. The value of each pixel in a grayscale image is proportional to the number of bits of data utilized to represent it. Grayscale image is a one-dimension (channel) image and the value of a grayscale image is commonly represented by 8 bits, that is, the pixel value of a pixel is represented by a combination of eight binary values. As a result, pixels have a value range of 0–255,with a total of 256 grayscale levels.

An RGB image is a three-dimension (channel) image, also known as a true colour image, is saved as an m-by-n-by-3 data array that defines the red, green, and blue colour components for each individual pixel. A palette is not used in RGB images. Each pixel's colour is determined by a combination of the red, green, and blue intensities stored in each colour plane at the pixel's location, where the red, green, and blue components are each 8 bits. This results in a possible palette of 16 million colours. True colour image is a term that refers to the precision with which a real-life image can be replicated

We will convert a grayscale or black and white image to a brilliant colour image in this project. However, the purpose of this project is to create a realistic colorization that may potentially trick a human observer, rather than to retrieve the original ground truth colour. As a result, our goal becomes much more attainable: model enough statistical relationships between the semantics and textures of grayscale images and their colour counter parts to yield visually appealing results.

We tackle the problem of automatic image colorization by (a) designingan appropriate objective function that handles the multimodal uncertainty of the colorization problem while capturing a wide variety of colours(b) introducing a novel framework for testing colorization algorithms that could be applied to other image synthesis tasks, and (c) training on a million colour images to set a new high-water water mark on the task. Second, we introduce the colorization taskas a competitive and simple way for self-supervised representation learning,with state-of-the-art results on a variety of benchmarks.

## II. RELATED WORKS

HINT BASED COLORIZATION [1] Levin et al. proposed a simple but effective method that incorporates user colorization hints in a quadratic cost function, requiring that neighboring pixels in space-time with similar intensities have similar colours. The hints are provided in the form of imprecise coloured "scribbles" on the grayscale input image, and the method is capable of producing high quality colorizations with no additional information about the image. Extensions to this approach have improved its performance even more: Huang et al. used adaptive edge

detection to address colour-bleeding issues, Qu et al. used luminance-based weighting of user-supplied hints to improve efficiency for video applications, and Qu et al. extended the cost function to enforce colour continuity over similar textures as well as similar intensities.Welsh et al. proposed an alternative approach that reduces the user's burden even further by requiring only a full-colour example image of similar composition. The algorithm achieves realistic results by matching luminance and texture information between the example image and the target grayscale image, as long as a sufficiently similar image can be found to use as the example image. Regardless of the level of automation, both the "scribble"-based and example-based techniques require significant human assistance, in the form of hand-drawn colour hints or appropriate examples.

DEEP COLORIZATION [2] We intend to use the large amount of image data available on the internet in our proposed method to fully automate the colorization process with no human intervention. Neural networks have shown great promise in learning a hierarchical model required for image understanding, so we turn to them. Cheng et al. proposed using per-pixel patch, DAISY, and semantic 2 features to train neural networks to predict chrominance values for each pixel, with joint bilateral filtering to smooth out accidental image artefacts.This method outperformed example-based methods that used hand-selected examples when trained on a large-scale image database with a simple Euclidean loss function against the ground-truth chrominance values. Ryan Dahl took it a step further, eschewing potentially limited image segmentation features in favour of a convolutional neural network pretrained for image classification as a feature extractor in a novel residual-style architecture that directly outputs full colour channels for the input image. The approach produced mixed results when trained on the ImageNet database with a Euclidean loss function on the chrominance values: the predicted colours were almost always reasonable, but they also tended toward desaturated and even brownish colours in general. In this case, the Euclidean loss function most likely resulted in "averaging" of colours across similar objects.

GENERATIVE ADVERSARIAL NETWORKS [3] The adversarial modelling framework, first suggested by Good fellow et al., is a method for training a neural network model that estimates the generative distribution pg(x) over input data x. We use neural network G (z;g) with parameters g to represent a mapping from an input noise variable with distribution pz(z) to a point x in the data space, and we use neural network D(x; d) to represent a mapping from a point x in the data space to the probability that x came from the data rather than G(z; g).Radford et al. used the adversarial framework to train convolutional neural networks as generative models for images, establishing the viability of DCGANs with tests on class-constrained datasets like the LSUN bedrooms dataset and human faces scraped from the web.

INFERENCE FROM LITERATURE REVIEW- Colorization algorithms differ primarily in how they gather and treat data in order to predict the grayscale-colour correlation. Given an input grayscale image, non-parametric approaches define one or more colour reference images (supplied by the user or automatically retrieved) to be utilised as source data. The colour is then transferred onto the input image using the Image Analogies framework from comparable parts of the reference image(s) [18,19,20,21]. Parametric techniques, on the other hand, learn prediction functions from huge datasets of colour images during training, posing the problem as either regression onto continuous colour space [22,1,2] or quantized colour value classification. Our method learns to identify colours as well, but with a larger model that has been trained on more data.

## III. METHODOLOGY

### A. CNN Architecture

A convolutional neural network (CNN, or ConvNet) is a type of artificial neural network used to interpret visual imagery in deep learning. Based on the shared-weight architecture of the convolution kernels or filters that slide along input features and give translation equivariant responses known as feature maps, they are also known as shift invariant or space invariant artificial neural networks (SIANN). Surprisingly, most convolutional neural networks are only equivariant under translation, rather than invariant. Image and video recognition, recommender systems, image classification, image segmentation, medical image analysis, natural language processing, brain-computer interfaces, and financial time series are just some of the areas where they can be used.

Multilayer perceptron are regularised variants of CNNs. Multilayer perceptron is typically completely connected networks, meaning that each neuron in one layer is linked to all neurons in the following layer. These networks' "complete connectedness" makes them vulnerable to data overfitting. Regularization, or preventing overfitting, can be accomplished in a variety of methods, including punishing parameters during training (suchas weight loss) or reducing connectivity (skipped connections, dropout, etc.) CNNs take a different approach to regularisation: they take advantage of the hierarchical pattern in data and use smaller and simpler patterns imprinted in their filters to assemble patterns of increasing complexity.

As a result, CNNs are at the lower end of the connectivity and complexity spectrum. The connection arrangement between neurons in convolutional networks was motivated by biological processes in that it mirrors the organisation of the animal visual brain. Individual cortical neurons only respond to stimuli in a small area of the visual field called the receptive field. Different neurons' receptive fields partially overlap, allowing them to cover the whole visual field.In comparison to other image classification methods,

CNNs require very little pre-processing. This means that the network learns to optimise the filters (or kernels) by automatic learning, as opposed to hand-engineered filters in traditional techniques. This lack of reliance on prior

information or human intervention in feature extraction is a significant benefit.

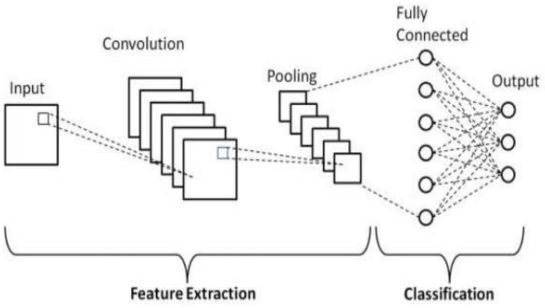



Fig 1. CNN Architecture

### B. VGGArchitecture

VGG stands for Visual Geometry Group, and it is a multilayer deep Convolutional Neural Network (CNN) architecture. The term "deep" refers to the number of layers in VGG-16 or VGG-19, which have 16 or 19 convolutional layers respectively.

The VGG architecture serves as the foundation for cutting-edge object recognition models. The VGGNet, which was created as a deep neural network, outperforms baselines on a variety of tasks and datasets in addition to ImageNet. Furthermore, it is still one of the most widely used image recognition architectures today.

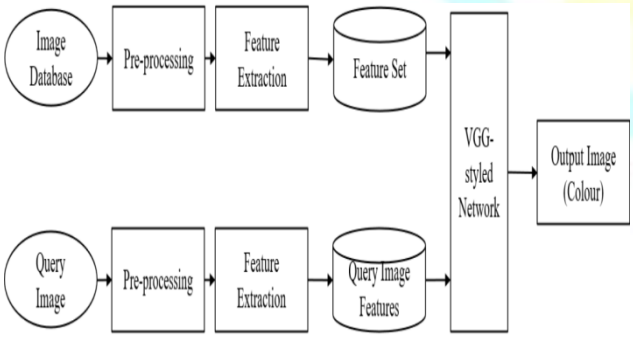



Figure 2. VGG Architecture

VGGNets are built on convolutional neural networks' most important properties (CNN).Small convolutional filters are used to build the VGG network. There are 13 convolutional layers and three fully linked layers in the VGG-16. Let's take a brief look at the architecture of VGG:Input: The VGGNet accepts images with a size of 224224 pixels.

To keep the image input size consistent for the ImageNet competition, the model's authors chopped out the middle 224224 patch in each image.

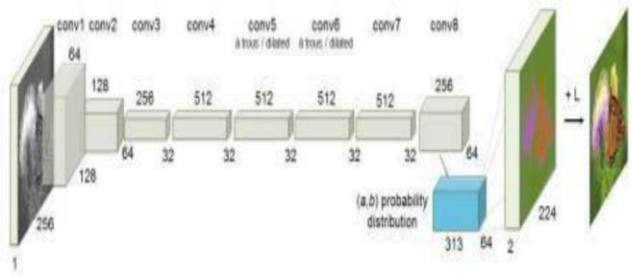
1) Convolutional Layers: VGG's convolutional layers use a small receptive field (33), the smallest size that still captures up/down and left/right movement. In addition, there are 11 convolution filters that operate as a linear transformation of the input. Then there's a ReLU unit, which is a significant AlexNet invention that cuts training time in half. The rectified linear unit activation function (ReLU) is a piecewise linear function that outputs the input if it ispositive and zero otherwise. To maintain spatial resolution after convolution, the convolution stride is set to 1 pixel (stride is the number of pixels shifts over the input matrix).

2) Hidden Layers: The VGG network's hidden layers all use ReLU. Local Response Normalization (LRN) is rarely used in VGG since it increases memory usage and training time. Furthermore, it has no effect on total accuracy.

3) Fully-Connected Layers: The VGGNet is made up of three layers that are all connected. The first two layers each have 4096 channels, whereas the third layer has 1000 channels, one for each class.

The number 16 in the name VGG refers to the fact that it is 16 layers deep neural network (VGGNet). This means that VGG16 is a pretty extensive network and has a total of around 138 million parameters. Even according to modern standards, it is a huge network. However, VGGNet16 architecture's simplicity is what makes the network more appealing. Just by looking at its architecture, it can be said that it is quite uniform.

There are a few convolution layers followed by a pooling layer that reduces the height and the width. If we look at the number of filters that wecan use, around 64 filters are available that we can double to about 128 and then to 256 filters. In the last layers, we can use 512 filters.




### C. ObjectiveFunction

Classification Our goal is to learn a mapping = ( ) to the two associated colourchannels Y ∈ RH×W×2, where H and W are picture dimensions, given an input lightness channel X ∈ RH×W×1. (Predictions are denoted with a b symbol, but ground truth is not.) This job is completed in CIE Lab colour space. Because the distances in this space model perceptual distance, the Euclidean loss L2(,) between predicted and ground truth colours is a suitable objective function.

$$L_2(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2}\sum_{h,w} \|\mathbf{Y}_{h,w} - \hat{\mathbf{Y}}_{h,w}\|_2^2$$

### D. ClassRebalancing

Because of the appearance of backgrounds such as clouds, pavement, dirt, and walls, the distribution of ab values in natural photographs is significantly biased towards values with low ab values. The empirical distribution of pixels in ab space based on 1.3 million ImageNet training photos. Notice how the number of pixels in natural photos is orders of magnitude more at desaturated values than at saturated levels. Desaturatedab values dominate the loss function if this isn't taken into account. We address the problem of class imbalance by reweighting each pixel's loss at train time based on pixel colour rarity. This is asymptotically equivalent to resampling the training space, which is a common strategy. Based on its closest ab bin, each pixel is weighed by factor $\in \mathbb{R}$ .

$$v(\mathbf{Z}_{h,w}) = \mathbf{w}_{q^*}, \text{ where } q^* = \arg\max_q \mathbf{Z}_{h,w,q}$$

$$\mathbf{w} \propto \left((1-\lambda)\tilde{\mathbf{p}} + \frac{\lambda}{Q}\right)^{-1}, \quad \mathbb{E}[\mathbf{w}] = \sum_q \tilde{\mathbf{p}}_q \mathbf{w}_q = 1$$

### E. Class Probabilities To Point Estimates

Finally, we define H, which transforms the anticipated distribution ˆ into an ab space point estimate ˆ. Taking the mean of the anticipated distribution, on the other hand, yields spatially consistent but desaturated results with an artificial sepia tone. This is understandable, given that choosing the mean after classification has some of the same drawbacks as optimising for a Euclidean loss in a regression framework. We interpolate by re-adjusting the temperature T of the SoftMax distribution and take the mean of the result to try to achieve the best of both worlds. The operation is referred to as taking the annealed-mean of the distribution because it is inspired by the simulated annealing approach.

$$\mathcal{H}(\mathbf{Z}_{h,w}) = \mathbb{E}\left[f_T(\mathbf{Z}_{h,w})\right], \quad f_T(\mathbf{z}) = \frac{\exp(\log(\mathbf{z})/T)}{\sum_q \exp(\log(\mathbf{z}_q)/T)}$$

### F. Implementation

We evaluate the graphics aspect of our algorithm, evaluating the perceptual realism of our colorizations as well as other accuracy measures. We compare our full algorithm to several variants, as well as recent and ongoing work. Colorization is tested as a method for self-supervised representation learning. Finally, we provide qualitative examples based on legacy black and white images. For the most part, the ImageNet database is used in the training process. The database contains millions of images organised into various sets. We trained our model on images totalling 18 gigabytes. The ImageNet database's images have a variety of shapes.

We put this to the test by feeding our fictitious colourized images into a VGG network that had previously been trained to predict ImageNet classes from real colour photos. If the classifier performs well, it means that the colorizations are accurate enough to provide information about the object class. Using a commercial classifier to evaluate the realism of synthesised data.

After removing colours from the input, classifier performance drops from 68.3 percent to 52.7 percent. The performance is improved to 56.0 percent after re-colorizing with our full method (other variants of our method achieve slightly higher results). A VGG classification network fine-tuned on grayscale inputs, for example, achieves a performance of 63.5 percent. This analysis, in addition to serving as a perceptual metric, demonstrates a practical application for our algorithm: by colourizing images with our algorithm and passing them to an off-the-shelf classifier, we can improve performance on grayscale image classification without any additional training or fine-tuning.

As a low-level test, we compute the percentage of predicted pixel colours in abcolour space that are within a thresholded L2 distance of the ground truth. After that, we sweep across thresholds ranging from 0 to 150 to generate a cumulative mass function, integrate the area under the curve (AuC), and normalise. It should be noted that the AuC metric measures raw prediction accuracy, whereas our method seeks plausibility.

Because our model was trained on "fake" grayscale images created by removing ab channels from colour photos, we also tested it on real legacy black and white photographs. Even though the low-level image statistics of the legacy photographs differ significantly from those of the modern-day photos on which it was trained, our model is still capable of producing good colorizations.


Figure 4. Histogram of Images

## IV. CONCLUSIONS

Image colorization is a niche computer graphics task, but it is also an example of a difficult pixel prediction problem in computer vision. We demonstrated that colorization using a deep CNN and a well-chosen objective function can produce results that are indistinguishable from real-world colour photos. Not only does our method produce useful graphics, but it can also be used as a pretext task for representation learning. Despite being only trained to recognize colours, our network learns a representation that is surprisingly useful for object classification, detection, and segmentation, outperforming other self-supervised pre-training methods. The goal of this project is to create an output image that is similar to but not identical to the input image. To avoid overfitting, various transformations such as image zooming and flipping were used. The model is used to extract high-level features. Colorization of historical videos will be one of our future projects. Using this technique, the old documentaries will appear more visually appealing.

## REFERENCES

[1] Cheng, Z., Yang, Q., Sheng, B.: Deep colorization. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 415–423

[2] Dahl, R.: Automatic colorization. In: http://tinyclouds.org/colorize/.(2016)

[3] Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: Proceedings of the 28th international conference on machine learning (ICML-11). (2011) 689–696

[4] Agrawal, P., Carreira, J., Malik, J.: Learning to see by moving. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 37–45

[5] Jayaraman, D., Grauman, K.: Learning image representations tied to ego-motion. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1413–1421

[6] Pathak, D., Kr¨ahenb¨uhl, P., Donahue, J., Darrell, T., Efros, A.: Context encoders: Feature learning by inpainting. In: CVPR. (2016)

[7] Lotter, W., Kreiman, G., Cox, D.: Deep predictive coding networks for video prediction and unsupervised learning. arXiv preprint arXiv:1605.08104 (2016).

[8] Atul Kumar Dwivedi, Deepali Virmani, Anusuya Ramasamy,Purnendu Bikash Acharjee, Mohit Tiwari" Modelling And Analysis Of Artificial Intelligence Approaches In Enhancing The Speech Recognition For Effective Multi-Functional Machine Learning Platform – A Multi Regression Modelling Approach " Journal of Engineering Research - ICMET Special Issue, 2022-04-06.

[9] Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: Proceedings of the

[10] IEEE International Conference on Computer Vision. (2015) 1422–1430

[11] Wang, X., Gupta, A.: Unsupervised learning of visual representations using videos. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 2794–2802.

[12] Prathima Chilukuri , J.R. Arun Kumar , R. Anusuya , M. Ramkumar Prabhu. "Auto Encoders and Decoders Techniques of Convolutional Neural Network Approach for Image Denoising In Deep Learning" Journal of Pharmaceutical Negative Results, 13(4), 1036–1040. https://doi.org/10.47750/pnr.2022.13.04.142 ,November 4, 2022.

[13] Donahue, J., Kr¨ahenb¨uhl, P., Darrell, T.: Adversarial feature learning. ArXiv preprint arXiv:1605.09782 (2016)

[14] Gupta, R.K., Chia, A.Y.S., Rajan, D., Ng, E.S., Zhiyong, H.: Image colorization using similar images. In: Proceedings of the 20th ACM international conference on Multimedia, ACM (2012) 369–378

[15] Deshpande, A., Rock, J., Forsyth, D.: Learning large-scale automatic image colorization. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 567–575.

[16] S. Bharathi, A. Balaji, D. Irene. J, C. Kalaivanan and R. Anusuya, "An Efficient Liver Disease Prediction based on Deep Convolutional Neural Network using Biopsy Images," 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2022, pp. 1141-1147, doi: 10.1109/ICOSEC54921.2022.9951870.

[17] Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. European Conference on Computer Vision (2016).

[18] R. Anusuya, M. Ramkumar Prabhu, Ch. Prathima, J. R. Arun Kumar" Detection of TCP, UDP and ICMP DDOS attacks in SDN Using Machine Learning approach" Journal of Survey in Fisheries Sciences, Vol. 10 No. 4S (2023): Special Issue 4.

[19] Iizuka, S., Simo-Serra, E., Ishikawa, H.: 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 (Proc. of SIGGRAPH 2016) 35(4) (2016)

[20] Hariharan, B., Arbel´aez, P., Girshick, R., Malik, J.: Hypercolumns for object segmentation and fine-grained localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 447–456

[21] Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv preprint arXiv:1606.00915 (2016)

[22] Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. International Conference on Learning Representations (2016).

[23] J. R. Arunkumar, S. Velmurugan, B. Chinnaiah, G. Charulatha, M. Ramkumar Prabhu et al., "Logistic regression with elliptical curve cryptography to establish secure iot," Computer Systems Science and Engineering, vol. 45, no.3, pp. 2635–2645, 2023.

[24] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision 115(3) (2015) 211–252

[25] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Advances in neural information processing systems. (2014) 487–495
[26] Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)