International Journal on Advanced Science Engineering Information Technology

Vol.13 (2023) No. 3
ISSN: 2088-5334

RGB Channel Combinations Method for Feature Extraction in Image Analysis

Seo-El Lee [a], Seong-Un Cho [b], Kyungyong Chung [b], Hyun Yoo [c,*]

[a] Department of Public Safety Big data, Kyonggi University, Suwon, 16227, Republic of Korea
[b] Division of AI Computer Science and Engineering, Kyonggi University, Suwon, 16227, Republic of Korea
[c] Contents Convergence Software Research Institute, Kyonggi University, Suwon, 16227, Republic of Korea
Corresponding author: *rhpa0916@gmail.com

Abstract— **Latest image analysis deep learning algorithms use diverse methods to extract features from images based on the Convolution Neural Network (CNN). CNN has a convolution layer consisting of RGB as three overlapping channels in the feature extraction process, and such architecture enables the backbone network to flow without losing each hue information. Therefore, 3D input data consisting of 3 channels to process the RGB channel consists of a large-scale neural network with many layer blocks. This processing method exhibits high accuracy. However, in terms of practicality, it results in big inefficiencies such as memory overhead and computational overhead. This study proposes the RGB Channel Combinations Method for Feature Extraction in Image Analysis to resolve such inefficiencies. This is a method that converts the RGB value into one tensor structure by combining each weight and bias and makes it possible to pass through the backbone network without damaging hue information. Based on the experiment results, it is confirmed that the accuracy decreased by 1.42% compared to the pre-existing method, but the number of parameters used by the input layer decreased. It is confirmed that the pre-processing used in the proposed method gained an additional computational overhead, but the number of input parameters decreased to 1/3 in the feature extraction stage performed afterward. As the proposed method applies to all image analysis algorithms, its expandability is extremely high and can process a large amount of image data.**

Keywords— **Convolution neural network; rgb channel; feature extraction; video analysis; computer vision.**

Manuscript received 29 Oct. 2022; revised 11 Dec. 2022; accepted 15 Mar. 2023. Date of publication 30 Jun. 2023.

I. INTRODUCTION

Deep learning is combined with diverse fields and is shown excellent performance. Technology obtained through applying deep learning to image analysis makes it possible to recognize and classify objects appearing in images and is used in the fields of crime prevention, traffic, and manufacturing requiring image analysis [1]. In the involved fields, as immediate control over situations requiring more than simple image analysis is frequently required, real-time image analysis is required [2]. However, as image data itself is big data that require large memory, there is a limitation in that the response speed is slow in terms of processing [3]. In addition, an image holds both temporal and spatial features. This means that the present and future points must be considered and that the relationship between the objects appearing in an image has a significant meaning. Therefore, there is a difficulty it is necessary to obtain comprehensive results by taking into consideration all temporal and spatial contexts [4].

The pre-existing CNN-based image analysis technology configures a convolution layer consisting of RGB as three overlapping channels to preserve color images or image features [5]. In general, RGB has a value ranging from 0 to 255 depending on each pixel and can be made to include larger information using floating points as normalization is performed. The pre-existing RGB extraction method creates a channel per hue, normalizes this value, and enters it into the tensor structure for use. This enables learning while maintaining the spatial information in the image dimension conversion process and enables each hue information to pass through the backbone network without causing any loss [6]. Although many overlapping layer blocks exhibit high accuracy, such method's RGB channel processing is insufficient in practice. In particular, when processing large-scale neural networks, memory, and computational overheads serve as limitations [7].

To overcome such limitations, Lingling et al. [8] proposed a new method that combines NSCT (Nonsubsampled Contourlet Transform) and CNN to classify RGB channel images. NSCT enhances classification accuracy by extracting features from an input image on diverse scales. In addition, as it can express statistical features, it decreases the information redundancy of an RGB channel image. The involved method extracts the NSCT-based coefficient of an image that captures the statistical features of each channel. The involved method exhibits high accuracy, and its expandability applies to diverse related studies. However, its running time is too great compared to the basic CNN-based image classification method, and there is a limitation in that particular images require more computational costs.

Therefore, the RGB Channel Combinations Method for Feature Extraction in Image Analysis is proposed in this study. The proposed method combines channel information corresponding to each hue into one tensor to configure the input color image in RGB format. This enables entering the color image information into the next layer without causing damage. This method differs from the pre-existing method that creates a tensor structure, combining 3 RGB channels into 1 channel. To do so, the two methods used are as follows: a method to connect and normalize the R, G, and B channel values without causing deformation and a method to reduce the pre-existing R, G, and B channel values to grayscale channels. The pre-processing using the proposed method gains an additional computational overhead but is capable of reducing CNN's computation in the complex layer structure created afterward. In addition, it is very high expandability makes the proposed method applicable to diverse deep-learning input layers used for image analysis enabling low memory and low computation.

In this study, a description of the studies and datasets related to RGB channel use, and a description of the proposed method are specified in Chapter 2. The image analysis and performance evaluation results of the proposed method are described in Chapter 3. Finally, the conclusion is specified in Chapter 4.

II. MATERIALS AND METHOD

A. Related Works

Initial image analysis technology uses a fully connected or dense layer to process an input image in a one-dimension [9]. Due to this reason, two-dimensional grayscale images such as MNIST were converted into one-dimensional images for learning [10]. As color images have three channels of R, G, and B, there are limitations in processing such color images with a fully connected layer or a dense layer. Color images have three-dimensional data consisting of width, height, and color. To learn such color images, three-dimensional data needs to be flattened and expressed in one dimension. However, as it is required to adjust the number of dimensions in such a process, there is a risk of losing space-related data information [11]. In addition, there are limitations in learning general images with distortions or different sizes or angles [12]. This is where CNN appears to overcome this matter. CNN minimizes the loss of space data and extracts features [13]. The CNN adjusts the size of input data to a particular size. Then, various filters are used to the adjusted data to extract the feature that most well expresses the input data, and the extracted feature is trained. When a color image is entered, a convolution layer expresses the values corresponding to RGB as a three-dimensional tensor structure ranging from 0 to 255. However, processing complex three-dimensional data leads to a memory overhead problem and a high computational cost problem [14].

In the computer vision field, the RGB channels of color images are diversely used. The CNN-based hue feature extraction method is disadvantageous because it connects features but leaves out the importance or reliability of features from mutually different hues. To resolve this problem, Lu et al. [15] proposed the CCF (color channel fusion) for face recognition. To be able to select more features from each color channel discriminately, this method fuses channels using two-dimension reduction algorithms: principal component analysis and Eigenfeature regularization and extraction. The involved method exhibits a high recognition rate but is limited in that its computation increases. Zhao et al. [16] use a partial channel combination as an input to obtain the hue information and detailed texture information of the objects in an image. The RG, GB, and RB channels are extracted from the raw image and entered into CNN, and the enhanced RG, GB, and RB channels are obtained. Then, abundant detailed information about the image is obtained through the multiscale fusion network using fusion weights. Kim et al. [17] proposed a cervical cancer image processing algorithm through the RGB Channel Superposition Algorithm. The RGB Channel Superposition Algorithm is a method that overlaps channels to create one image. The raw image and the mask images corresponding to the raw image are divided into R, G, and B. They are created as images corresponding to each hue. Then, the involved images are overlapped to create a new image, and the created image is used for learning. The involved methods are advantageous in that the information of each hue feature is used without any loss but are limited in that complex computation occurs in the process of combining channels and reducing dimensions.

Color image processing causes problems such as computational complexity and storage space insufficiency. Due to such problems, attempts have been made from long ago to convert color images into grayscale images [18]. In particular, in the computer vision field, as features extracted from images are used as inputs, they can be processed whether they are in color or grayscale [19]. Many works of literature propose diverse technologies for converting color images into grayscale images. In a study by Kim et al. [20], the color-gray-scale nonlinear global mapping optimized to color images is determined to create grayscale images that preserve the sequence and brightness of the light and shade between adjacent pixels. This determines the brightness, saturation, and nonlinear function of the hue of an image. Once a hue image is provided, the function's parameters are optimized, and the feature discriminability and hue sequence are maintained during conversion from a hue image to a grayscale image. In addition, a nonlinear function is designed to respect the original brightness of the hue during conversion. The proposed method converted the input image into a consistent grayscale image by distinguishing different hue regions but is limited in that the detailed feature information can be lost. In addition, Kumar and Verma [21] propose a method that changes the pixel intensity of a grayscale image by changing the intensity of R, G, and B according to the conversion equation [21]. Kumar et al. [22] propose a conversion method based on the luminosity algorithm. This method modifies the pre-existing method that uses the luminosity algorithm based on human brightness perception. The pre-existing method is limited because it requires a floating-point operator and increases resource utilization. To resolve these problems, a right shift operator is used to move the bit corresponding to each hue to the right, and the obtained final 8 bits are converted into grayscale pixels using the RCA (Ripple Carry Adder). The involved method is advantageous in that it enhances design efficiency but is limited in that its generalization is difficult as it is performed on the FPGA and ASIC.

B. RGB Channel Combinations Method for Feature Extraction in Video Analysis

The RGB Channel Combinations Method for Feature Extraction in Image Analysis proposed in this study is a method that creates a one-dimensional tensor structure by combining 3 RGB channels into 1 channel. To do so, two methods are used. The first is a method that connects and normalizes the pre-existing R, G, and B channel values without causing deformation. The second is a method that reduces the pre-existing R, G, and B channel values into grayscale channels.

1) The method of normalization with each color value: The first method connects and normalizes the pre-existing R, G, and B channel values without causing deformation. The first method corresponding to such method divides each value into 255 pixels to maintain the hue features of the R, G, and B values. Then, each value is added and normalized to have a value between 0 and 1. Fig. 1 shows the process of the involved method is processed.

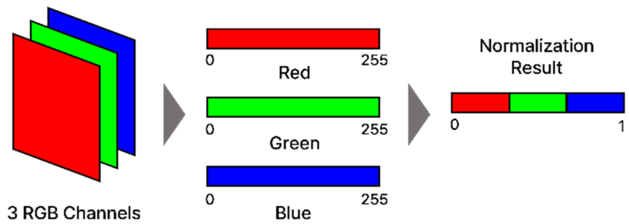

Fig. 1 The process of the first method of normalization with each color values

Formula (1) shows the formula applied to the involved method.

$$Y = \{(R/255) + (G/255) + (B/255\} / 3) \quad (1)$$

The next method puts together the R, G, and B values as one line and divides them by 767, the total size of the array, to normalize them to have a value between 0 and 1. The involved method performs normalization after changing the value corresponding to R into a value between 0 and 255, changing the value corresponding to G into the next value between 256 and 511, and changing the value corresponding to B into the next value between 512 and 767. Fig. 2 shows the process of the involved method.

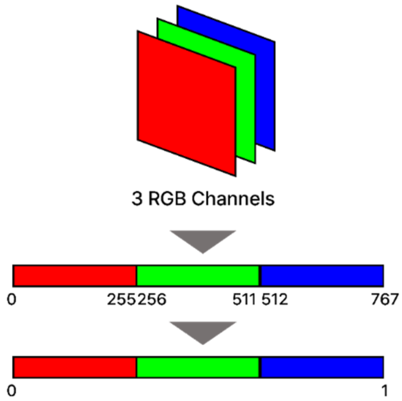

Fig. 2 The process of the second method of normalization with each color values

Formula (2) shows the formula applied to the involved method.

$$Y = (\mathrm{R} + \mathrm{G} + \mathrm{B}) / 767 \quad (2)$$

2) The method of converting each color value to grayscale: The second method converts the RGB hue values into grayscale values. First, the simplest method is the method that adds up the R, G, and B values and converts the obtained mean value into a grayscale value [23]. Fig. 3 shows the process of the involved method.

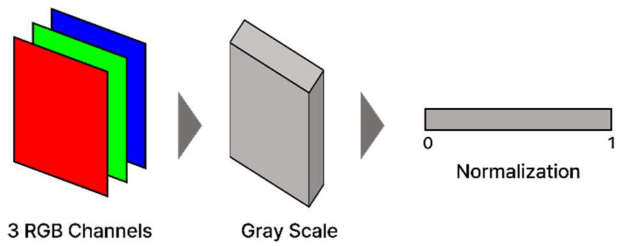

Fig. 3 The process of the first method of converting each color value to grayscale.

Formula (3) shows the formula applied to the involved method.

$$Y = (\mathrm{R} + \mathrm{G} + \mathrm{B}) / 3 \quad (3)$$

The next method uses the YUV expression, one of the color expression methods. The involved method separately transfers the color brightness and hue information [24]. Fig. 4 shows the process of the YUV expression method.

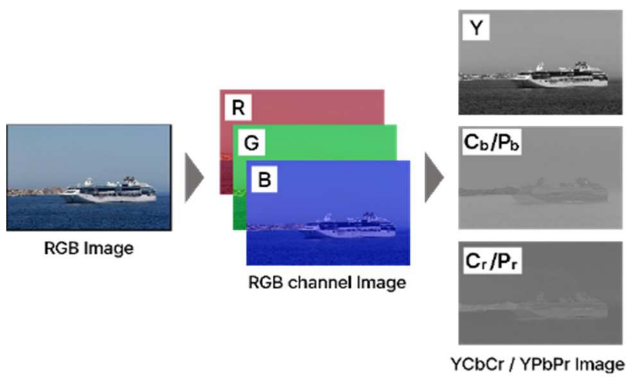

Fig. 4 The process of RGB to YCbCr / YPbPr

First, YCbCr, one of the YUV methods, is used. YCbCr is an expression method made to digitalize CRT, LCDI, PDP, etc. [25]. This method expresses color by using Y (the color brightness information), Cb (the difference between blue color in the color brightness information), and Cr (the difference between red color in the color brightness information). By multiplying the RGB values by the weights, Y, the color brightness information, can be calculated. The weight for each hue value is the quantified value confirming the color to which the human eye is most sensitive. When the R, G, and B values are the same, the human eye perceives G as the brightest and differently perceives R and B as the second brightest and the least bright, respectively. Then, Y is used to convert color images into grayscale images. Formula (4) shows the formula for calculating Y.

$$Y = 0.2126 * \mathrm{R} + 0.7152 * \mathrm{G} + 0.0722 * \mathrm{B} \quad (4)$$

In addition, the YUV methods include YPbPr. If YCbCr is a color expression method for digital systems, YPbPr is a color expression method for analog systems [26]. Similar to the YCbCr method, this method multiplies the R, G, and B hue values by the weights to calculate Y, the color brightness information, and uses the calculated Y value to change a color image to a grayscale image. Formula (5) shows the formula used by YPbPr to calculate Y, the color brightness information.

$$Y = 0.299 * \mathrm{R} + 0.587 * \mathrm{G} + 0.114 * \mathrm{B} \quad (5)$$

In addition, the YUV methods include YPbPr. If YCbCr is a color expression method for digital systems, YPbPr is a color expression method for analog systems [26]. Similar to the YCbCr method, this method multiplies the R, G, and B hue values by the weights to calculate Y, the color brightness information, and uses the calculated Y value to change a color image to a grayscale image. The method is the HSV(B) method, which uses the maximum RGB values to calculate the brightness information, then converts the image into a grayscale image [27]. Formula (6) shows the formula for calculating V(B), the brightness information. The involved method is possible to express more brightness information than the HSL method.

$$V(B) = \mathrm{Max}(\mathrm{R}, \mathrm{G}, \mathrm{B}) \quad (6)$$

The HSL method uses the mean value obtained from the RGB values to calculate the brightness information and then converts the image into a grayscale image [28]. Formula (7) shows the formula for calculating L, the brightness information.

$$L = \{\mathrm{Max}(\mathrm{R}, \mathrm{G}, \mathrm{B}) + \mathrm{Min}(\mathrm{R}, \mathrm{G}, \mathrm{B})\} / 2 \quad (7)$$

III. Result and Discussion

The CIFAR 10 dataset is used for the performance evaluation of this study. CIFAR 10 is a dataset widely used in the machine learning field and includes 60,000 color images with 32x32 pixels [29]. In addition, the images are labeled and classified into ten classes, such as airplanes, vehicles, and birds. Of the involved dataset, 50,000 images are used for training, and the remaining 10,000 are used for testing. As for the configured system, hardware consisting of Intel® Xeon(R) CPU @ 2.20GHz, 16GB Memory, and NVIDIA Tesla T4 is used, and software consisting of Python (Ver 3.8.10) and Tensorflow (Ver 2.11.0) is used to configure a neural network.

TABLE I
Hyper parameter for performance evaluation

	layer	batch size	epoch
Hyper Parameter	7	128	30

Table 1 shows the hyperparameter optimization results of a convolution neural network model for classification. The optimized results are observed when the number of layers is 7; the batch size is 128, and 30 epochs.

A. *Performance Evaluation*

The performance evaluation compares the basic CNN using three channels to perform image analysis and the CNN using the proposed method. The accuracy and the number of parameters are measured to evaluate the proposed method's performance. The accuracy is based on the confusion matrix and represents the percentage of data for which the predicted and actual values are the same [30].

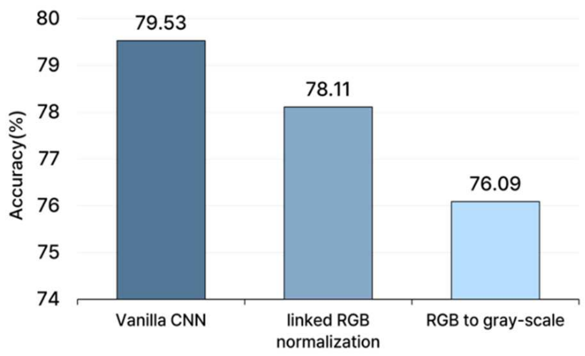

Fig. 5 Comparison of average accuracy of proposed methods with CNN using 3 channels.

Fig. 5 shows the mean accuracy of the vanilla CNN using 3 channels and of the proposed CNN using 1 channel. The vanilla CNN achieved the highest accuracy of 79.53%. The Linked RGB normalization method achieved an accuracy of 78.11% which was 1.42% lower than the Vanilla CNN. The RGB to grayscale method achieved an accuracy of 76.09% which was 3.44% lower than the Vanilla CNN. Such results show that the accuracy of the proposed method is lower than that of the pre-existing CNN using three channels.

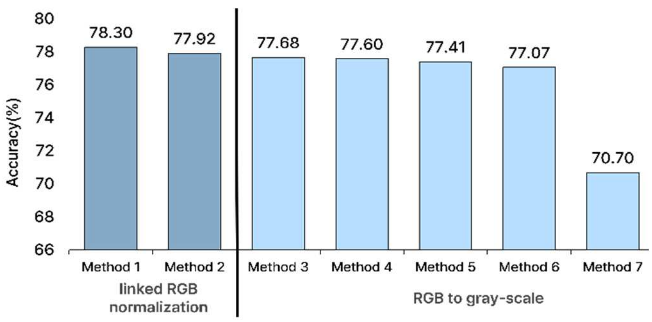

Fig. 6 Comparison of linked RGB normalization and RGB to grayscale

Fig. 6 shows the accuracy of the linked RGB normalization methods and of the RGB to grayscale methods. The Linked RGB normalization method achieved an accuracy that was 2.82% higher on average compared to the RGB to the grayscale method. Based on such results, it is confirmed that, when using the RGB channels as inputs, high accuracy can be achieved by connecting channel values without causing deformation rather than processing them as grayscale values. However, excluding Method 7, which uses the HSV(B) method with the lowest accuracy among the RGB to grayscale methods, it is confirmed that no big difference exists between the two methods. It can be expected that similar accuracy values will be achieved regardless of which method is used.

As the next step, the number of parameters is used to compare the pre-existing CNN using three channels and the CNN using the proposed method. Since the computational cost decreases as the number of parameters decreases, this could decrease the number of computational and memory overheads. The RGB channels are processed on the CNN's input layer. Therefore, the number of parameters used on the involved layer is compared to evaluate the performance of the proposed method.

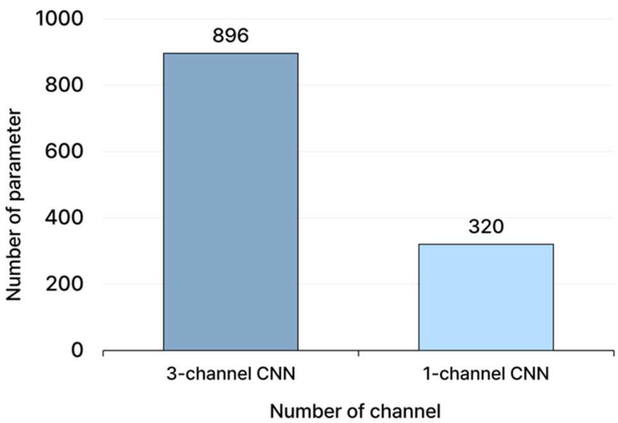

Fig. 7 Comparison of the number of parameters according to the number of channels

Fig. 7 compares the number of parameters on the first convolution layer according to the number of channels. The proposed CNN using 1 channel had 320 parameters, and the pre-existing CNN using 3 channels had 896 parameters, meaning that the proposed CNN using 1 channel had 576 fewer parameters than the pre-existing CNN using 3 channels. Such results show that the improved method decreased the model's complexity.

B. Discussion

This study confirmed that the proposed method's CNN used 576 fewer parameters on the input layer than the pre-existing CNN using three channels. This demonstrates that lower computational costs are required. However, in terms of accuracy, the performance of CNN using the proposed method was lower compared to the CNN using the pre-existing method. Based on such results, it can be expected that decreasing the 3 RGB channels to 1 channel may decrease accuracy but positively impact model complexity and speed reduction.

IV. Conclusion

In this study, the RGB Channel Combinations Method for Feature Extraction in Video Analysis is proposed to resolve the computational and memory overheads resulting from CNN's RGB channel processing. Two methods to combine the 3 RGB channels into 1 channel are proposed. The first method connects and normalizes the pre-existing R, G, and B channel values without causing deformation. Another method decreases the pre-existing R, G, and B channel values to grayscale channels. The one-dimensional tensor-structured channel created through the two methods is used as input to the next layer.

Compared to the pre-existing method using 3 channels, the accuracy was decreased by at least 1.2% and at most by 9%. Based on a comparison between the method for decreasing the channels into grayscale channels and the method for connecting the channels without causing deformation, the method connecting them without causing deformation achieved higher accuracy. Such results show that no significant accuracy difference exists between the pre-existing and proposed methods. Lastly, as a result of comparing the number of parameters used on the input layer between the pre-existing and proposed methods, the proposed method used fewer parameters. Compared to the pre-existing method, the computation complexity was decreased by three times. In addition, as the proposed method applies to deep learning's input layer for diverse image analyses, it exhibits high expandability.

In this study, the input channel was decreased to 1/3 to show that the model's complexity can decrease. However, compared to the pre-existing study, there is a problem that the accuracy and the overall number of parameters showed no significant difference. In the future study, coming up with a configuration capable of decreasing the number of parameters without impacting accuracy will contribute to decreasing the computational and memory costs resulting from diverse image analysis technologies.

Acknowledgment

Basic Science Research Program supported this research through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (No: NRF-2022R1I1A1A01069526)

References

[1] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: analysis, applications, and prospects," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 6999-7019, Dec. 2022.

[2] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3212-3232, Nov. 2019.

[3] S. Peng, S. Sun, and Y. D. Yao, "A survey of modulation classification using deep learning: Signal representation and data preprocessing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 7020-7038, Dec. 2021.

[4] L. Duan, J. Liu, W. Yang, T. Huang, and W. Gao, "Video coding for machines: A paradigm of collaborative compression and intelligent analytics," *Transactions on Image Processing*, vol. 29, pp. 8680-8695, 2020.

[5] Li, Lianlin, and Tie Jun Cui, "Information metamaterials–from effective media to real-time information processing systems." *Nanophotonics*, vol. 8, no. 5, pp. 703-724, Feb. 2019.

[6] S. Zhang, W. Huang, and C. Zhang, "Three-channel convolutional neural networks for vegetable leaf disease recognition," *Cognitive Systems Research*, vol. 53, pp. 31-41, 2019.
[7] N. Dryden, N. Maruyama, T. Moon, T. Benson, M. Snir, and B. Van Essen, "Channel and filter parallelism for large-scale CNN training," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1-20, Nov. 2019.
[8] F. Lingling, Z. Hanyu, Z. Jiaxin, and W. Xianghai, "Image classification with an RGB-channel nonsubsampled contourlet transform and a convolutional neural network," Neurocomputing, 396, pp. 266-277, 2020.
[9] M. Masana, X. Liu, B. Twardowski, M. Menta, A. D. Bagdanov, and J. van de Weijer, "Class-incremental learning: survey and performance evaluation on image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Oct. 2022.
[10] S. Jiang *et al.*, "An Optimized Deep Neural Network Detecting Small and Narrow Rectangular Objects in Google Earth Images," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 1068-1081, 2020.
[11] Y. Ming, X. Meng, C. Fan, and H. Yu, "Deep learning for monocular depth estimation: A review," *Neurocomputing*, vol. 438, pp. 14-33, May 2021.
[12] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Computational Intelligence and Neuroscience*, 2018.
[13] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9, 2015.
[14] Z. Cheng, Q. Yang, and B. Sheng, "Colorization using neural network ensemble," *IEEE Transactions on Image Processing*, vol. 26, no. 11, pp. 5491-5505, Nov. 2017.
[15] Z. Lu, X. Jiang, and A. C. Kot, "A Color Channel Fusion Approach for Face Recognition," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 1839-1843, 2015.
[16] L. Zhao *et al.*, "Color Channel Fusion Network for Low-Light Image Enhancement," *IEEE International Conference on Image Processing (ICIP)*, pp. 1654-1658, 2021.
[17] Y. J. Kim *et al.*, "RGB Channel Superposition Algorithm with Acetowhite Mask Images in a Cervical Cancer Classification Deep Learning Model," *Sensors*, vol. 22, no. 9, 3564, 2022.
[18] M. Mittal *et al.*, "An efficient edge detection approach to provide better edge connectivity for image analysis." *IEEE Access*, vol. 7, pp. 33240-33255, 2019.
[19] D. Singh, K. Vijay, and K. Manjit, "Densely connected convolutional networks-based COVID-19 screening model." *Applied Intelligence*, vol. 51, pp. 3044-3051, 2021.
[20] Y. Kim, C. Jang, J. Demouth, and S. Lee, "Robust color-to-gray via nonlinear global mapping," in *ACM SIGGRAPH Asia 2009 papers*, pp. 1-4, 2009.
[21] T. Kumar and K. Verma, "A Theory Based on Conversion of RGB image to Gray image," *International Journal of Computer Applications*, vol. 7, no. 2, pp. 7-10, 2010.
[22] K. Kumar, R. K. Mishra, and D. Nandan, "Efficient Hardware of RGB to Gray Conversion Realized on FPGA and ASIC," *Procedia Computer Science*, vol. 171, pp. 2008-2015, 2020.
[23] J. Hu, Q. Jiang, R. Cong, W. Gao, and F. Shao, "Two-branch deep neural network for underwater image enhancement in HSV color space," *IEEE Signal Processing Letters*, vol. 28, pp. 2152-2156, 2021.
[24] J. F. Yang, K. T. Lee, G. C. Chen, W. J. Yang, and L. Yu, "A YCbCr color depth packing method and its extension for 3D video broadcasting services," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 9, pp. 3043-3053, 2019.
[25] P. Fan, G. Lang, B. Yan, X. Lei, P. Guo, Z. Liu, and F. Yang, "A method of segmenting apples based on gray-centered RGB color space," *Remote Sensing*, vol. 13, no. 6, pp. 1211, 2021.
[26] N. Dubey, H. Modi, "A robust discrete wavelet transform based adaptive watermarking scheme in YCbCr color space against camcorder recording in cinema/movie theatres," *Engineered Science*, vol. 15, pp. 116-128, 2021.
[27] N. Danapur, S. A. A. Dizaj, and V. Rostami, "An efficient image retrieval based on an integration of HSV, RLBP, and CENTRIST features using ensemble classifier learning," *Multimedia Tools and Applications*, vol. 79, no. 33-34, pp. 24463-24486, 2020.
[28] A. Garg, X. W. Pan, L. R. Dung, "LiCENt: Low-light image enhancement using the light channel of HSL," *IEEE Access*, vol. 10, pp. 33547-33560, 2022.
[29] Krizhevsky, Alex, and Geoffrey Hinton. "Learning multiple layers of features from tiny images." (2009): 7.
[30] A. Luque, A. Carrasco, A. Martín, and A. Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognition*, vol. 91, pp. 216-231, 2019.