TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 28/31 pp402–417
DOI: 10.26599/TST.2024.9010001
Volume 30, Number 1, February 2025

Convolutional Neural Network Image Classification Based on Different Color Spaces

Zixiang Xian, Rubing Huang*, Dave Towey, and Chuan Yue

Abstract: Although Convolutional Neural Networks (CNNs) have achieved remarkable success in image classification, most CNNs use image datasets in the Red-Green-Blue (RGB) color space (one of the most commonly used color spaces). The existing literature regarding the influence of color space use on the performance of CNNs is limited. This paper explores the impact of different color spaces on image classification using CNNs. We compare the performance of five CNN models with different convolution operations and numbers of layers on four image datasets, each converted to nine color spaces. We find that color space selection can significantly affect classification accuracy, and that some classes are more sensitive to color space changes than others. Different color spaces may have different expression abilities for different image features, such as brightness, saturation, hue, etc. To leverage the complementary information from different color spaces, we propose a pseudo-Siamese network that fuses two color spaces without modifying the network architecture. Our experiments show that our proposed model can outperform the single-color-space models on most datasets. We also find that our method is simple, flexible, and compatible with any CNN and image dataset.

Key words: color space; Convolutional Neural Network (CNN); image classification; pseudo-Siamese network

1 Introduction

Computer vision is a popular field in artificial intelligence with practical applications in everyday life, including object classification[1–5], segmentation[6–9], and tracking[10–12]. Image classification is a crucial task in computer vision, and involves classifying an image into predefined categories, such as “cat” or “animal”. It also serves as the basis for more complex tasks, such as object detection and image segmentation.

One of the challenges in image classification is identifying informative features in the image. Traditional computer vision techniques rely on hand-crafted features designed to be invariant to rotation, scaling, and other transformations[13–15]. However, these features may be not enough when attempting to capture more complex patterns and variations in images[13, 15, 16].

Deep neural networks have emerged as a powerful tool for image classification as they can automatically learn features from data. Each layer learns increasingly abstract features that optimize discrimination between different image classes. The final layer’s output is used to predict the image class.

The choice of color space is a critical factor that impacts the performance of deep neural networks in image classification. The Red-Green-Blue (RGB) color space, based on the primary colors red, green, and blue,

• Zixiang Xian, Rubing Huang, and Chuan Yue are with School of Computer Science and Engineering, Macau University of Science and Technology, Macao 999078, China. E-mail: 3220001352@student.must.edu.mo; rbhuang@must.edu.mo; 3220001522@student.must.edu.mo.
• Dave Towey is with School of Computer Science, University of Nottingham Ningbo China, Ningbo 315100, China. E-mail: dave.towey@nottingham.edu.cn.
* To whom correspondence should be addressed.
Manuscript received: 2023-11-07; revised: 2023-12-29; accepted: 2024-01-02

is commonly used in computer vision. However, many other color spaces could be used for image representation, each with its advantages and disadvantages.

Recent research has explored alternative color spaces, beyond RGB, for image classification[1, 17–20]. The Lightness-A-B (LAB) color space, for example, separates image intensity from color information, making it more robust to changes in lighting conditions[18, 19]. Other color spaces, such as Liminance, Red-difference, and Blue-difference chroma (YCbCr)[1] and Hue-Saturation-Value (HSV)[20], have also been investigated.

Kim et al.[21] investigated a traffic light recognition system using six color spaces, based on the Fast Region-based Convolutional Neural Network (Fast R-CNN)[22, 23] and Region-based Fully Convolutional Network (R-FCN)[24] models. They suggested that the appropriate color space and network can improve traffic light detection. Inconsistencies between the RGB color space and human visual perception of color can result in deviations in image quality. To address this issue, converting RGB into human psychology based color spaces (such as HSV) can enhance the perceptual quality assessment of images[25]. To overcome the challenges related to uneven illumination and clutter in natural environments during the segmentation of diseased leaf images, a novel approach combining color-balancing and super-resolution for images has been proposed[26]. Colorization of tumor areas in intracranial tumor Magnetic Resonance Imaging (MRI) grayscale images is a challenging research area in medicine. However, it has been demonstrated that the addition of colors can lead to improvements in automatic colorization[27]. Gowda and Yuan[28] converted RGB images into multiple color spaces simultaneously and used them as inputs to individual Dense Convolutional Networks (DenseNets): They achieved better performance on four image classification datasets. However, systematic and comprehensive experiments are still needed to explore the impact of color space on image classification. Bianco et al.[29], using Convolutional Neural Network (CNN) architectures, examined how suitable color-balancing models can significantly improve the recognition of textures.

The main motivation of this study is to investigate the following two research questions:

(1) In terms of various convolution methods, how does the performance of CNN vary when trained with different color spaces?

(2) Is it possible to enhance CNN performance by using a merged color space, without requiring any modifications to the CNN architecture?

Color spaces are different ways of representing the colors of an image, and they can have a significant impact on the performance of image processing tasks. Different color spaces may emphasize different aspects of the image information, such as luminance, chrominance, hue, saturation, and perceptual quality. The first research question aims to explore how these aspects influence the feature extraction and classification ability of different CNN models with different convolution methods, such as standard, depthwise, and dilated convolutions. Therefore, choosing an appropriate color space for a given task can be crucial for achieving optimal results, as indicated in the first research question. The second research question aims to test whether or not combining two color spaces can enhance the image representation and classification accuracy of CNN models, without significantly increasing the computational complexity or modifying the network structure.

In this paper, we investigate how various color spaces affect the performance of different CNN models (with various numbers of layers and diverse convolution methods). We conduct experiments on four datasets that contain images of natural scenes and objects: Canadian Institute for Advanced Research, 10 classes (CIFAR10, https://www.cs.toronto.edu/kriz/cifar.html), Street View House Number (SVHN, http://ufldl.stanford.edu/housenumbers/), Self-Taught Learning 10, (STL10, https://cs.stanford.edu/acoates/stl10/), and Canadian Institute for Advanced Research, 100 classes (CIFAR100, https://www.cs.toronto.edu/kriz/cifar.html). We compare the performance of five CNN models trained with different color spaces: AlexNet, MobileNet, Very Deep Convolutional Network (VGG), Residual Netwotk (ResNet), and DenseNet. The color spaces we consider are: LAB, YCbCr, Luminance-U-V (LUV), XYZ, Luma-U-V (YUV), Hue-Lightness-Saturation (HLS), HSV, HLS Full Range (HLS FULL), and HSV Full Range (HSV FULL): More details about these color spaces are presented in Section 2.2. We also explore whether or not specific classes of objects are affected by different color spaces, and report on the importance of selecting

an appropriate color space for CNN models. Furthermore, we propose a novel pseudo-Siamese network that can fuse two different color spaces to improve classification accuracy. Our proposed model can be easily implemented, without requiring any changes to the network architecture, and can achieve some improvements over training with only RGB images. We conduct an ablation study to analyze the effect of different color space combinations on the pseudo-Siamese network.

The main contributions of this paper are as follows:

- We provide a comprehensive and systematic analysis of the impact of color space on image classification performance across different CNN models and datasets.
- We propose a pseudo-Siamese network that can leverage two different color spaces to enhance the image classification accuracy, without modifying the network architecture.
- We conduct an ablation study to evaluate the effect of different color space combinations on the pseudo-Siamese network, and identify the best color space pairs for each dataset and model.

The rest of this paper is organized as follows. Section 2 introduces some of the background details for our study, including the basic CNN architectures that are used in our experiments. Section 3 reports on the experiments conducted on the CNNs with different color spaces, and explains why we are motivated to propose the pseudo-Siamese network. Section 4 presents our proposed pseudo-Siamese network, explains the ablation experiment conducted with it, and addresses some potential threats to the validity of our study. Finally, Section 5 summarizes our findings and discusses some potential future work.

2 Background and Related Work

2.1 Convolutional neural network

CNNs have become a mainstream neural network structure used in computer vision, especially for image classification. Despite being introduced by LeCun et al. in 1995, LeNet did not become popular in the field of computer vision due to its limited success on early and small datasets[30]. AlexNet[31] uses group convolution by dividing input and kernel channels into different groups: This can significantly reduce the number of weights, making it possible to train CNNs on larger and more realistic datasets, including the ImageNet dataset[32]. Figure 1 shows an illustration of the AlexNet group convolution.

Figure 2 shows an illustration of depthwise separable convolution[33, 34], which accelerates the learning process by utilizing feature extractors to modify the latent spaces. MobileNets[35] for mobile and embedding vision applications primarily uses depthwise separable convolution to reduce computation.

With the development of neural network

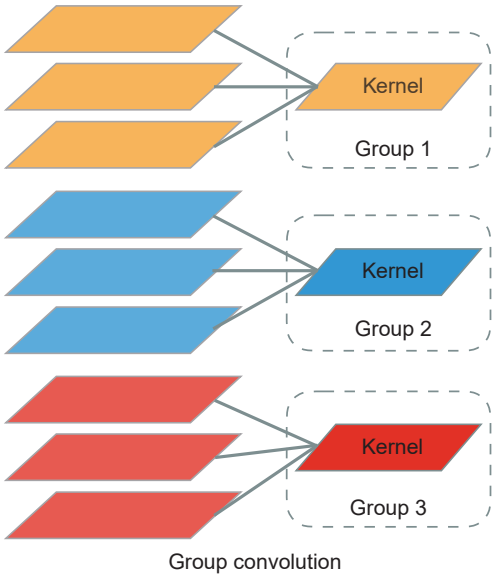

Fig. 1 AlexNet group convolution.

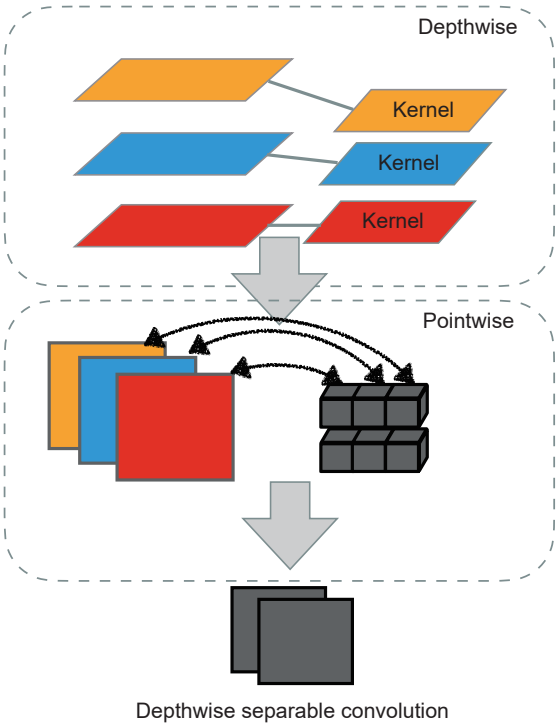

Fig. 2 Depthwise separable convolution.

accelerators, it has become possible to make deeper multichannel, multilayer CNNs with many parameters: Deep CNNs are developing very rapidly in computer vision[36]. Simonyan and Zisserman[37] increased the depth of networks by utilizing repeated Visual Geometry Group (VGG) convolutional blocks.

A deeper network structure typically yields better accuracy[37, 38], but may face the vanishing gradient problem[39]. One solution to this is to add batch normalization layers[40] or normalized initialization parameters[41, 42], which allows the deep neural network to converge for gradient descent with backpropagation[43]. ResNet[44] addresses this problem with the skip-connection method, bypassing signals between layers. The DenseNet[45] was proposed to make CNNs even deeper by connecting each layer to every other layer in a feed-forward fashion[46]. DenseNet can better alleviate the vanishing gradient problem than ResNet[44], while both strengthening feature propagation and reducing the parameters.

All of the methods listed above basically convert the input image to the RGB color space for the classification task. However, conversion to other color spaces is also possible, including, for example, classifying skin color using the YCbCr color space via the Bayesian approach[1]. Vandenbroucke et al.[47] proposed a hybrid color space for classifying pixels in soccer-related color images, aiming to recognize the players' team. Zarit et al.[48] investigated the pixel-classification performance of two skin-detection methods in five different color spaces.

The previous work in this area indicates that different color spaces may deliver different results for machine learning algorithms. This suggests the potential to achieve better performance by combining different color spaces. In Section 3, we evaluate the impact of different color spaces on image classification with four different datasets (CIFAR10[49], SVHN[50], STL10[51], and CIFAR100[49]). We first conduct empirical experiments to explore the effect of different color spaces on different deep learning models in different datasets. We then propose our model, which fuses two different color spaces to improve classification accuracy.

2.2 Color space

This section explores how the RGB color space can be converted to other color spaces. The RGB color space is an additive color space that simulates the emission of light from a source, such as a computer screen or a projector. In this color space, the primary colors of red, green, and blue are combined in various proportions to produce a range of colors, from black (no light) to white (all colors)[52]. In contrast, the CMYK (C stands for cyan, M for magenta, Y_{CMYK} for yellow, and K for key (black)) color space, which is often used in printing[53], uses a subtractive color mechanism, where the more colors are added together, the darker the resulting color becomes. In this color space, the primary colors of cyan, magenta, yellow, and black are mixed in various amounts to produce a range of colors, from white (no ink) to black (all colors). The CMYK color space may have advantages over other spaces, such as RGB, HSV, and YCbCr, for skin color detection: This is because it can exploit the low level of cyan in human skin, which is a distinctive feature that is not present in other color spaces[54]. Conversion between RGB and CMYK is done as follows:

$$K = 1 - \max(R', G', B') \tag{1}$$

$$C = \frac{1 - R' - K}{1 - K} \tag{2}$$

$$M = \frac{1 - G' - K}{1 - K} \tag{3}$$

$$Y_{\mathrm{CMYK}} = \frac{1 - B' - K}{1 - K} \tag{4}$$

where $R' = \frac{R}{255}$, $G' = \frac{G}{255}$, and $B' = \frac{B_{\mathrm{RGB}}}{255}$. R stands for red, G stands for green, and B_{RGB} stands for blue.

The XYZ color space is inspired by human visual perception and color-matching experiments, and the selected colorimetry relies on this process[55]. To convert RGB to the XYZ color space defined by the International Commission on Illumination (CIE)[56], we need to apply a transformation matrix[57]: The X component is a rough indicator of how much red light is present, while the Y_{XYZ} component measures the brightness or luminance with some variation, and the Z component is similar to the blue channel in RGB. The red stimulation can be conceptualized as amounts of the red color in a color model. Luminance is a measure of light in units. It should be noted that the Y_{XYZ} component is often referred to as luminance, although it is not a direct measure of luminance and is subject to some degree of interpretation.

The equation for conversion from RGB space to XYZ space is as follows[58]:

$$\begin{bmatrix} X \\ Y_{\mathrm{XYZ}} \\ Z \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} R \\ G \\ B_{\mathrm{RGB}} \end{bmatrix} =$$

$$\begin{bmatrix} 0.490\,00 & 0.310\,00 & 0.200\,00 \\ 0.176\,97 & 0.812\,40 & 0.010\,63 \\ 0.000\,00 & 0.010\,00 & 0.990\,00 \end{bmatrix} \begin{bmatrix} R \\ G \\ B_{\mathrm{RGB}} \end{bmatrix} \tag{5}$$

The CIE LAB and CIE LUV color spaces are derived from the XYZ color space, which is the basic model defined by the CIE[56]. In the CIE LAB color space, L_{LAB} stands for the lightness of the color ($L_{\mathrm{LAB}} = 0$ means black and $L_{\mathrm{LAB}} = 100$ means white), A stands for chroma (positive values indicate red and negative values indicate green), and B_{LAB} stands for hue (positive values refer to yellow and negative values refer to blue)[55]. In the CIE LUV color space, L_{LUV} stands for luminance, and U and V_{LUV} represent the correlates of chroma and hue color images[55]. CIE LAB adapts CIE XYZ to match human perception better. LUV is a color space that models how colors are perceived by the human eye. It is particularly useful for representing natural scenes, especially for distinguishing different shades of green. Both LAB and LUV can be obtained from XYZ images that have been converted from RGB. Conversion from CIE XYZ to CIE LUV or CIE LAB, and from CIE LUV or CIE LAB to CIE XYZ, is explained by Plataniotis and Venetsanopoulos[55].

The YUV and YCbCr color spaces were created to enable compression of bandwidth for video transmission[59] and color TV broadcasts[60]. YUV is the analogue model for National Television Standard Committee (NTSC) systems, while the YCbCr model is a digital standard[60]. RGB can be converted to YCbCr as follows[61]:

$$\begin{cases} Y_{\mathrm{YCbCr}} = 16 + \dfrac{65.738R}{256.000} + \dfrac{129.057G}{256.000} + \dfrac{25.064B_{\mathrm{RGB}}}{256.000}, \\ \mathrm{Cb} = 128 - \dfrac{37.945R}{256.000} - \dfrac{74.494G}{256.000} + \dfrac{112.439B_{\mathrm{RGB}}}{256.000}, \\ \mathrm{Cr} = 128 + \dfrac{112.439R}{256.000} - \dfrac{94.154G}{256.000} - \dfrac{18.285B_{\mathrm{RGB}}}{256.000} \end{cases} \tag{6}$$

where Y_{YCbCr} represents luminance, Cb represents blue-difference, and Cr represents red-difference.

The HSV and HLS (H stands for hue as the predominant color, S stands for saturation as the color purity, and V_{HSV} or L_{HLS} stands for luminance as the color brilliance) color spaces were designed based on human psychology: They aim to better mimic how humans perceive and describe colors. These color spaces consider attributes such as hue, saturation, and brightness or lightness, respectively. These are more intuitive to humans than the red, green, and blue components used in RGB. HLS and HSV are both color spaces that use hue, saturation, and brightness to represent colors. However, HLS gives more weight to colors that are near white, and thus has a narrower range of saturation levels[52]. This increases the complexity of the model[52]. Conversion from RGB to HSV or HLS can be done as follows[52]:

$$V_{\max} = \max(R, G, B_{\mathrm{RGB}}) \tag{7}$$

$$V_{\min} = \min(R, G, B_{\mathrm{RGB}}) \tag{8}$$

$$H = \begin{cases} 60(G - B_{\mathrm{RGB}})/(V_{\max} - V_{\min}), & \text{if } V_{\max} = R; \\ 120 + 60(B_{\mathrm{RGB}} - R)/(V_{\max} - V_{\min}), & \text{if } V_{\max} = G; \\ 240 + 60(R - G)/(V_{\max} - V_{\min}), & \text{if } V_{\max} = B_{\mathrm{RGB}}; \\ 0, & \text{if } R = G = B_{\mathrm{RGB}} \end{cases} \tag{9}$$

$$H_{\mathrm{HSV}} = H_{\mathrm{HLS}} = H \tag{10}$$

$$L_{\mathrm{HLS}} = \frac{V_{\max} + V_{\min}}{2} \tag{11}$$

$$S_{\mathrm{HLS}} = \begin{cases} \dfrac{V_{\max} - V_{\min}}{V_{\max} + V_{\min}}, & \text{if } L_{\mathrm{HLS}} < 0.5; \\ \dfrac{V_{\max} - V_{\min}}{2 - (V_{\max} + V_{\min})}, & \text{if } L_{\mathrm{HLS}} \geqslant 0.5 \end{cases} \tag{12}$$

$$S_{\mathrm{HSV}} = \begin{cases} \dfrac{V_{\max} - V_{\min}}{V_{\max}}, & \text{if } V_{\max} \neq 0; \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

$$V_{\mathrm{HSV}} = V_{\max} \tag{14}$$

The maximum and minimum values are denoted by $V_{\max}$ and $V_{\min}$ in RGB values, as defined in Eqs. (7) and (8), respectively. The hue in both the HSV and HLS color models corresponds to the variable H in Eq. (9). Lightness (L_{HLS}), specific to the HLS color space, is computed according to Eq. (11). The saturation for HLS and HSV color spaces is determined by Eqs. (12) and (13), respectively. Furthermore, the brightness value (V_{HSV}) within the HSV color model is defined by Eq. (14).

The HSV FULL or HLS FULL color spaces employ a hue range from 0 to 360 degrees to represent the full spectrum of colors. The basic HSV or HLS color space uses a hue range from 0 to 180 degrees. The hue component of a pixel represents its specific color: The

broader range of hues available in HSV FULL or HLS FULL enables more precise and nuanced color representation, but they also require more memory to store the hue values.

Figure 3 presents some sample images in each of the color spaces introduced above.

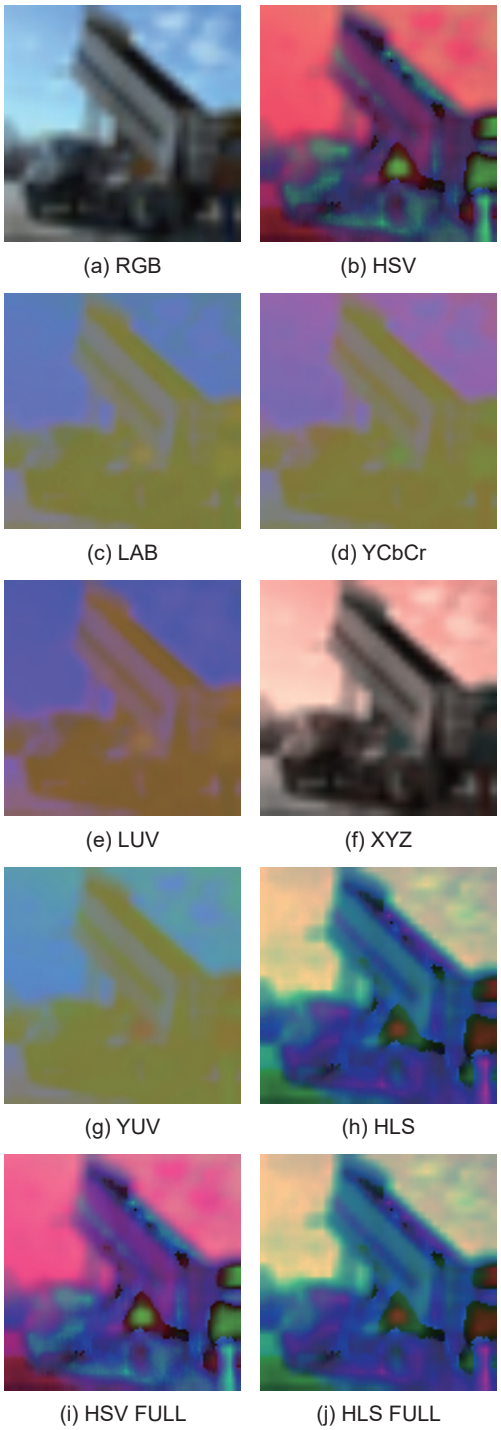

Fig. 3 Sample images from CIFAR10 dataset in different color spaces.

3 Empirical Study

3.1 Experimental setup

Image classification is a fundamental task in computer vision, and its effectiveness can directly determine the effectiveness of other computer vision tasks. To explore the performance of different CNN models and different color spaces in image classification, we used four well-known datasets in our experiments: CIFAR10, CIFAR100, SVHN, and STL10.

Because of the success of deep neural networks in image classification and the marginal differences in performance across simpler datasets[31], our study exclusively used the CIFAR100 dataset to test the efficacy of the VGG and DenseNet models. This dataset is known for its complexity, featuring 100 distinct classes, each with 600 images, necessitating the identification of fine-grained object details and variations. By focusing on this challenging dataset, we were able to more accurately assess the ability of these networks to handle intricate image classification tasks. The remaining datasets were used to validate the neural networks with fewer layers: LeNet5, AlexNet, and MobileNet.

CIFAR10 and CIFAR100 are labeled subsets of the 80 million tiny image datasets collected by Krizhevsky, Nair, and Hinton (https://www.cs.toronto.edu/kriz/cifar.html). CIFAR10 consists of 60 000 32 × 32 color images in 10 classes, with 6000 images per class: We used 50 000 images for training and 10 000 images for testing.

SVHN is a dataset of house numbers extracted from Google Street View images. It has 10 classes, corresponding to the digits 0 to 9. We trained our model on 73 257 digits and tested it on 26 032 digits.

STL10 was collected by Stanford University, and consists of 10 classes of real-world objects in 96 × 96 pixels. The STL10 dataset is still challenging for CNN models, due to its higher resolution. The larger image size includes a broader range of intricate details and variations in object appearances, necessitating more sophisticated feature extraction and increased computational resources for accurate image classification.

3.2 Evaluation metric and training

We used the following metrics to evaluate the image classification performance (TP represents true

positives; TN represents true negatives; FP represents false positives; and FN represents false negatives):

$$\text{Accuracy} = \left(\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}\right) \tag{15}$$

$$\text{Precision} = \left(\frac{\text{TP}}{\text{TP}+\text{FP}}\right) \tag{16}$$

$$\text{Recall} = \left(\frac{\text{TP}}{\text{TP}+\text{FN}}\right) \tag{17}$$

$$\text{F1} = 2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \tag{18}$$

When dealing with imbalanced datasets, accuracy metrics can be misleading due to the uneven representation of certain classes or categories[62]. In such cases, the F1 score (the harmonic mean of precision and recall) becomes especially relevant. By taking into account the algorithm's ability to identify both positive and negative instances of each class, the F1 score provides a more accurate measure of the algorithm's performance on imbalanced datasets[62].

All input images underwent conversion to the target color spaces (including RGB). Furthermore, channel normalization was performed to ensure that pixel-value distributions were consistent across all image channels, reducing potential bias and enhancing the performance of the image-classification algorithms used. The models were trained in batches of size 256, using the Adam optimizer[63] with a linear learning decay scheduler and a momentum value of 0.937. The initial learning rate was set to 1×10^{-3}. The models were trained on an AMD 3500X computer with an Nvidia RTX 3090 graphics card, with 15 epochs in each training.

3.3 Result for LeNet5, AlexNet, and MobileNet

The training and testing results presented in Table 1 indicate that, across various color spaces, the training accuracy of each model on the CIFAR10 dataset remained almost the same, with little discernible variation in the recorded values. HSV, HSV FULL, HLS, and HLS FULL perform poorly under the simple neural network LeNet5 in training: about 4.0% less well than RGB. Similarly, they are 0.5% less accurate than RGB with AlexNet and MobileNet for training with CIFAR10. LeNet5 has similar training accuracy scores on the SVHN dataset, regardless of the color space used. MobileNet, like LeNet5, also has similar training scores on the SVHN dataset, for all color spaces. The RGB, LAB, and YCbCr accuracies, with the AlexNet model on the SVHN dataset, are significantly higher than the other color spaces: Only LUV has about 63.0% accuracy.

The analysis of testing accuracy reveals that, across all three models, the RGB color space outperforms all other color spaces on the CIFAR10 dataset. In the second place, the XYZ color space follows closely, while the psychology-based color spaces have the worst performance, consistent with their training-performance accuracy. For the SVHN dataset, YCbCr surpasses RGB in performance when used with the LeNet5 model; YUV achieves the best results with AlexNet; and LUV and XYZ perform best with MobileNet. On the STL10 dataset, results are similar to CIFAR10, with RGB yielding the best performance, followed by LAB or YCbCr in the second place.

There is little difference in the testing accuracy for each color space with LeNet5 across the three datasets. This may be because LeNet5 is a shallow network, with only a few layers, and thus is unable to extract suitable features from the input color spaces.

The recall score is more reflective of the model's classification performance than accuracy or precision, because it may be affected by the number of negative samples in imbalanced datasets. In short, recall can reflect the sensitivity of a specific color space to a specific category of images. On the one hand, different color spaces may have different expression abilities for different image features, such as brightness, saturation, hue, etc. These features may affect the model's ability to recognize some categories of images, thus affecting the recall value. For example, if a color space can better distinguish red from green, then it may have a higher recall for recognizing categories such as watermelons. The RGB color space is generally more reliable, but the recall results in Table 2 reveal certain color spaces to be more effective for specific classes: For example, XYZ has better performance than RGB in MobileNet for Class 8 and Class 10. In addition to RGB, LUV and XYZ also have high recall scores for specific categories, indicating that they have high discrimination rates for certain categories of objects. On the other hand, different datasets and models may have different difficulties and complexities, which will also affect the recall value. For example, if the categories in a dataset have a high similarity, then the model may have more difficulty in distinguishing them, resulting in a lower recall. Similarly, if the

Table 1 Comparison result of LeNet5, AlexNet, and MobileNet on CIFAR10/SVHN/STL10 datasets.

CNN model	Color space	CIFAR10		SVHN		STL10	
		Training accuracy (%)	Test accuracy (%)	Training accuracy (%)	Test accuracy (%)	Training accuracy (%)	Test accuracy (%)
LeNet5	RGB	67.4	63.9	91.8	88.6	51.6	46.8
	LAB	66.4	63.5	91.4	88.6	49.9	46.9
	YCbCr	66.3	62.8	91.5	88.7	50.4	46.2
	LUV	66.9	63.5	91.5	88.6	50.0	45.8
	XYZ	67.8	63.7	91.6	88.4	51.1	46.4
	YUV	66.0	62.6	91.4	88.5	49.9	45.6
	HLS	63.0	59.5	91.2	88.3	51.8	45.8
	HLS FULL	62.0	59.4	91.0	88.4	51.5	45.8
	HSV	62.7	59.5	91.1	88.1	50.1	44.4
	HSV FULL	65.3	62.3	91.4	88.0	52.9	46.1
AlexNet	RGB	99.7	92.1	98.3	96.0	99.9	90.3
	LAB	99.3	89.4	98.1	95.9	99.7	83.9
	YCbCr	99.5	90.1	98.2	96.0	99.7	83.6
	LUV	99.3	89.7	63.6	77.4	99.7	83.8
	XYZ	99.6	91.5	90.4	92.2	99.9	89.3
	YUV	99.4	89.7	98.1	96.1	99.8	83.4
	HLS	99.1	86.3	95.8	94.4	99.7	77.4
	HLS FULL	99.2	86.4	97.6	94.9	99.8	76.9
	HSV	99.1	86.2	94.5	93.7	99.6	75.9
	HSV FULL	99.1	85.0	94.3	93.8	99.5	74.3
MobileNet	RGB	99.8	83.7	99.6	93.4	100.0	90.8
	LAB	99.8	82.0	99.7	93.4	100.0	87.4
	YCbCr	99.8	82.0	99.7	93.5	100.0	87.7
	LUV	99.8	82.2	99.6	93.6	100.0	87.4
	XYZ	99.8	83.4	99.7	93.6	100.0	90.5
	YUV	99.8	82.5	99.6	93.3	100.0	87.3
	HLS	99.8	81.0	99.6	93.1	100.0	84.9
	HLS FULL	99.7	81.2	99.6	93.1	100.0	83.7
	HSV	99.6	78.6	99.5	92.9	100.0	78.9
	HSV FULL	99.4	77.4	99.6	92.7	100.0	77.9

structure or parameters of a model are not complex or deep, then it may not be able to fully utilize the color space information, thus affecting the recall value. Table 2 shows that the more complex and deeper networks work better than the more simple networks, like LeNet5: They have the highest recall scores in multiple categories with the RGB color space. Section 3.4 examines whether or not this same conclusion can be reached for even deeper neural networks, such as VGG and DenseNet.

3.4 Result for VGG and DenseNet

The training configurations for VGG and DenseNet were the same as specified in Section 3.2, except that, to prevent overfitting, these were only trained for 10 epochs.

Table 3 shows that both VGG and DenseNet can achieve 100.0% training accuracy on the CIFAR10 dataset, across all color spaces. Again, RGB has the best performance, in terms of validation accuracy and recall scores, for each category. Because those CNNs have more convolution blocks acting as feature extractors, they can better learn the representation of images, leading to better performance in tasks of image classification.

Because it could be that the CIFAR10 dataset has too few categories, we also evaluated VGG and DenseNet with the larger CIFAR100 dataset. VGG and DenseNet

Table 2 Recall matrix of LeNet5, AlexNet, and MobileNet on the CIFAR10 dataset. Bolds indicate the optimal data.

CNN Model	Color space	Recall									
		Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10
LeNet5	RGB	0.672	0.754	**0.517**	0.433	**0.575**	0.546	0.747	0.668	0.778	0.698
	LAB	**0.686**	0.754	0.481	**0.452**	0.562	0.523	0.737	0.659	0.770	0.724
	YCbCr	0.667	0.723	0.451	0.447	0.519	0.534	**0.780**	0.699	0.756	0.706
	LUV	0.710	0.779	0.494	0.405	0.570	0.487	0.755	0.676	0.743	0.732
	XYZ	0.670	0.744	0.512	0.375	0.545	**0.557**	0.790	0.681	0.774	0.721
	YUV	0.670	0.760	0.458	0.447	0.553	0.531	0.752	0.652	0.759	0.677
	HLS	0.632	0.675	0.427	0.420	0.502	0.499	0.686	0.675	0.735	0.700
	HLS FULL	0.657	0.720	0.456	0.346	0.508	0.482	0.691	0.657	0.745	0.674
	HSV	0.647	0.774	0.469	0.373	0.488	0.518	0.646	0.612	0.767	0.655
	HSV FULL	0.679	**0.767**	0.447	0.403	0.530	0.545	0.649	**0.706**	**0.779**	**0.729**
AlexNet	RGB	**0.931**	0.949	**0.900**	**0.853**	**0.925**	0.858	**0.955**	**0.932**	**0.959**	**0.949**
	LAB	0.889	0.951	0.857	0.778	0.909	0.847	0.932	0.906	0.939	0.931
	YCbCr	0.906	0.944	0.868	0.791	0.903	0.867	0.926	0.920	0.955	0.934
	LUV	0.896	0.959	0.858	0.791	0.906	0.840	0.927	0.917	0.940	0.940
	XYZ	0.934	0.952	0.889	0.809	0.919	**0.880**	0.934	0.947	0.946	0.940
	YUV	0.906	**0.964**	0.866	0.781	0.897	0.845	0.936	0.909	0.947	0.918
	HLS	0.875	0.930	0.796	0.717	0.868	0.781	0.903	0.904	0.935	0.920
	HLS FULL	0.877	0.942	0.783	0.723	0.865	0.813	0.896	0.897	0.928	0.912
	HSV	0.882	0.945	0.813	0.667	0.857	0.817	0.904	0.889	0.927	0.918
	HSV FULL	0.864	0.936	0.787	0.673	0.841	0.803	0.892	0.867	0.924	0.913
MobileNet	RGB	**0.883**	**0.913**	**0.801**	0.679	**0.834**	0.729	0.871	0.858	0.901	0.899
	LAB	0.851	0.905	0.754	**0.683**	0.809	0.722	0.847	0.840	0.898	0.891
	YCbCr	0.826	0.894	0.772	0.674	0.823	0.733	0.860	0.831	0.907	0.885
	LUV	0.846	0.889	0.753	0.678	0.795	**0.738**	0.872	0.847	**0.916**	0.886
	XYZ	0.854	0.913	0.773	0.668	0.833	0.753	0.871	**0.866**	0.898	**0.909**
	YUV	0.848	0.908	0.749	0.669	0.806	0.742	**0.893**	0.841	0.906	0.888
	HLS	0.864	0.909	0.738	0.653	0.772	0.737	0.856	0.825	0.886	0.863
	HLS FULL	0.855	0.904	0.735	0.678	0.775	0.710	0.854	0.844	0.891	0.875
	HSV	0.814	0.896	0.692	0.650	0.758	0.650	0.841	0.826	0.876	0.854
	HSV FULL	0.827	0.874	0.688	0.597	0.774	0.664	0.830	0.804	0.861	0.826

were trained over 15 epochs to enable convergence at the same training accuracy. However, as there were 100 classes in the CIFAR100 dataset, we did not include the recall matrix in our analysis at this stage. Figure 4 shows the results of our experiment.

It can be observed from Fig. 4 that RGB has the highest validation accuracy for both CNNs, even though all color spaces had the same accuracy during the training. HLS, HSV, HLS FULL, and HSV FULL are the four worst-performing color spaces, which matches our findings in Section 3.3. The XYZ space consistently ranks the second, which also aligns with our results in Section 3.3. Based on our earlier experiments, we propose fusing a model trained in RGB with one trained in another color space: We can train this pseudo-Siamese CNN using XYZ and RGB with the CIFAR100 dataset. More details about this pseudo-Siamese CNN will be discussed in Section 4.

4 Proposed Method

4.1 Architecture of proposed method

The Siamese neural network is a type of neural network that uses the same weights to process two input vectors in parallel, to compute their similarity. In our research, we trained a Siamese CNN to work with two different color spaces. To accomplish this, we needed to create two different copies of the network's

Table 3 Recall matrix of VGG and DenseNet on CIFAR10 dataset. Bolds indicate the optimal data.

CNN model	Color space	Recall										Training accuracy (%)	Validation accuracy (%)
		Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10		
VGG	RGB	**0.940**	**0.967**	**0.907**	**0.824**	**0.948**	0.896	**0.971**	0.938	**0.961**	**0.950**	1.0	93.3
	LAB	0.926	0.958	0.857	0.813	0.907	0.860	0.953	**0.940**	0.939	0.945	1.0	91.0
	YCbCr	0.933	0.962	0.886	0.824	0.921	0.885	0.951	0.931	0.949	0.945	1.0	91.9
	LUV	0.917	0.960	0.869	0.814	0.927	0.851	0.949	0.917	0.951	0.940	1.0	91.0
	XYZ	0.941	0.958	0.902	0.824	0.946	**0.899**	0.964	0.938	0.957	0.949	1.0	92.8
	YUV	0.934	0.962	0.873	0.814	0.931	0.861	0.947	0.932	0.944	0.938	1.0	91.0
	HLS	0.922	0.961	0.852	0.790	0.901	0.827	0.931	0.932	0.934	0.938	1.0	89.9
	HLS FULL	0.917	0.959	0.859	0.786	0.900	0.850	0.928	0.924	0.941	0.937	1.0	90.0
	HSV	0.919	0.958	0.841	0.761	0.887	0.829	0.935	0.929	0.951	0.938	1.0	85.9
	HSV FULL	0.910	0.947	0.840	0.771	0.902	0.816	0.915	0.918	0.953	0.939	1.0	89.1
DenseNet	RGB	0.899	0.933	**0.838**	**0.756**	**0.867**	**0.781**	**0.914**	0.899	**0.931**	0.908	1.0	87.3
	LAB	0.894	0.923	0.774	0.710	0.864	0.747	0.894	0.877	0.925	0.884	1.0	84.9
	YCbCr	**0.904**	0.929	0.813	0.716	0.859	0.740	0.879	0.870	0.919	0.888	1.0	85.2
	LUV	0.888	0.927	0.792	0.724	0.843	0.781	0.888	0.860	0.929	**0.919**	1.0	85.5
	XYZ	0.890	0.923	0.821	0.736	0.870	0.780	0.908	**0.900**	0.927	0.908	1.0	86.6
	YUV	0.876	**0.934**	0.816	0.706	0.852	0.777	0.895	0.882	0.927	0.896	1.0	85.6
	HLS	0.879	0.905	0.765	0.650	0.836	0.734	0.872	0.849	0.913	0.879	1.0	82.8
	HLS FULL	0.857	0.904	0.776	0.664	0.803	0.741	0.888	0.859	0.895	0.873	1.0	82.6
	HSV	0.827	0.884	0.734	0.659	0.778	0.711	0.841	0.829	0.877	0.864	1.0	80.0
	HSV FULL	0.839	0.876	0.722	0.605	0.774	0.696	0.836	0.802	0.858	0.834	1.0	78.4

weights, one for each color space. This is referred to as "pseudo"-Siamese, since it involves duplicating the weights in a way that is not typical in a Siamese neural network. The architecture of the pseudo-Siamese CNN for prediction is shown in Fig. 5.

The two parts of the Siamese CNN share the same structures (instead of weights) and have softmax layers as the final layers. The networks were fed with the tensors in two color spaces for each image, one of which was RGB. The outputs of the softmax layers from both networks were combined, with the class showing the maximum probability selected as the final prediction. For example, if the RGB model outputs Class 1 with a probability of 0.7, and the other color space model outputs Class 2 with a probability of 0.9, then the overall prediction result from the model is Class 2. This is called the all-max operation in this paper. Our experiments in Sections 3.3 and 3.4 revealed that some color spaces yielded high recall scores for specific categories. Therefore, during prediction, we applied the aforementioned max operation only to the predictions of the target category, while using the RGB color space predictions for other categories. We refer to this approach as the class-max operation, where, for instance, the 8-max operation indicates that only the predictions of Class 8 were obtained from the non-RGB color spaces, while the predictions for other classes came only from the RGB color space.

Since we had two weights, we calculated the loss function $\mathcal{L}$ as follows:

$$\mathcal{L} = -\frac{1}{N}\sum_{i}\sum_{c=1}^{\mathcal{M}}\frac{1}{\lambda_1+\lambda_2}[\lambda_1 \times Y_{i,c}\log(P_{i,c}(\text{RGB})) + \lambda_2 \times Y_{i,c}\log(P_{i,c}(\text{Other}))] \tag{19}$$

where N is the number of entries in the dataset, $\mathcal{M}$ is the number of classes, $Y_{i,c}$ is the target label, $P_{i,c}$ is the prediction probability (for RGB or the other color space), and λ_1 and λ_2 are trainable hyperparameters for setting the ratio loss for each part of the network, respectively. We used an initial setting of $\lambda_1 = 0.5$ and $\lambda_2 = 0.5$.

Using the configuration specified above, we set up an ablation experiment to compare our proposed models (using all-max or class-max prediction) with the original CNN models.

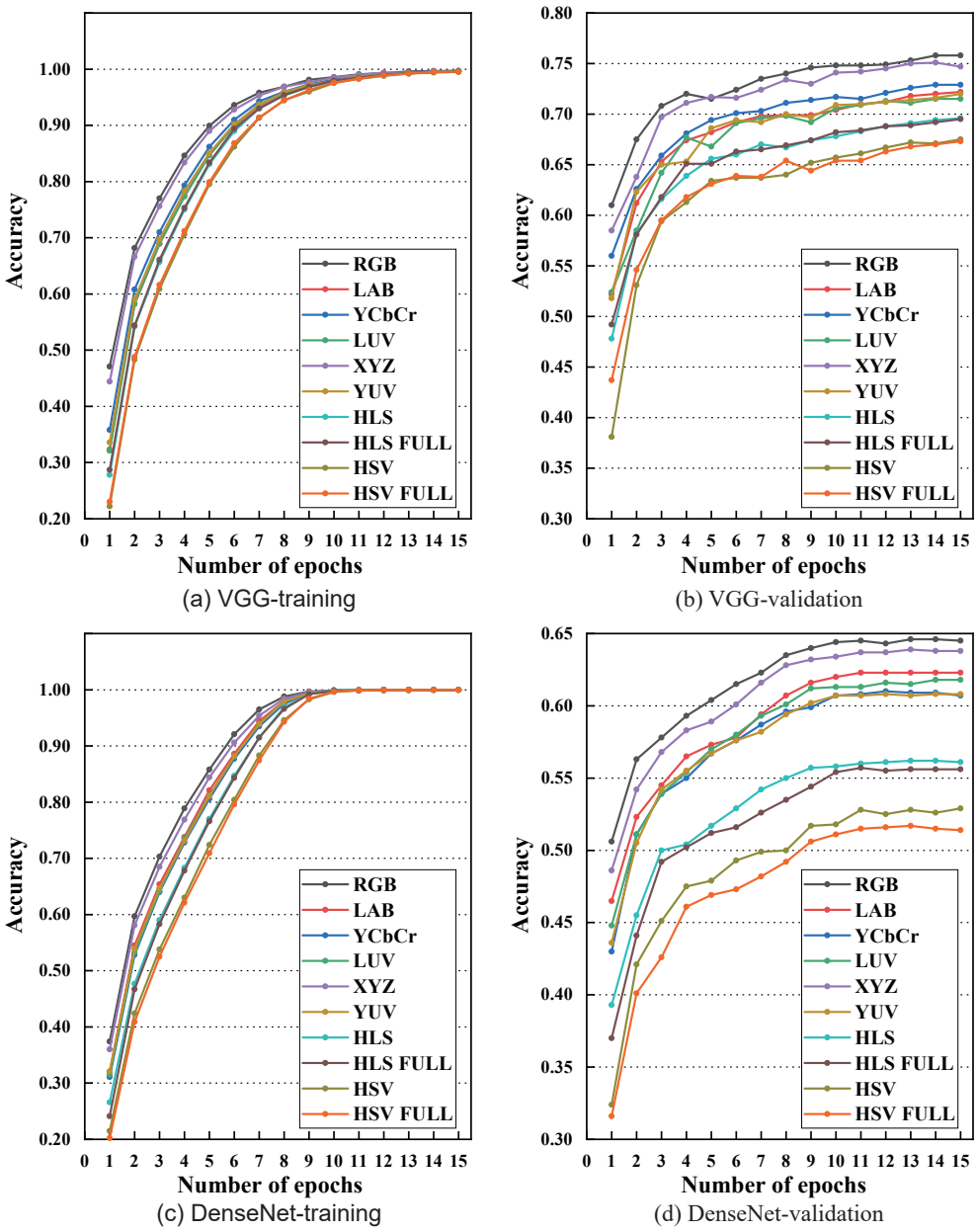

Fig. 4 VGG and DenseNet on CIFAR100 dataset.

4.2 Ablation experiment setup

We conducted ablation experiments on the CIFAR10 dataset for LeNet5, AlexNet, and MobileNet. The pseudo-Siamese CNN was built by combining RGB with the color space that performed the next best, in terms of recall (according to Table 2). The same training configuration as specified in Section 3.2 was used, and all the models were trained on an AMD 3500X computer with an Nvidia RTX 3090 graphic card. The training involved the same number of epochs.

4.3 Results of ablation experiment

Table 4 presents the results of the ablation experiments: In the first column of Table 4, "all" in brackets indicates the use of all-max, and the numbers in brackets indicate the use of class-max. Table 4 shows that our proposed network with all-max prediction outperforms all other networks. In this experiment, the all-max prediction strategy meant that we always output the category prediction with the highest probability in both color spaces for any given category. Although the same pseudo-Siamese CNN is

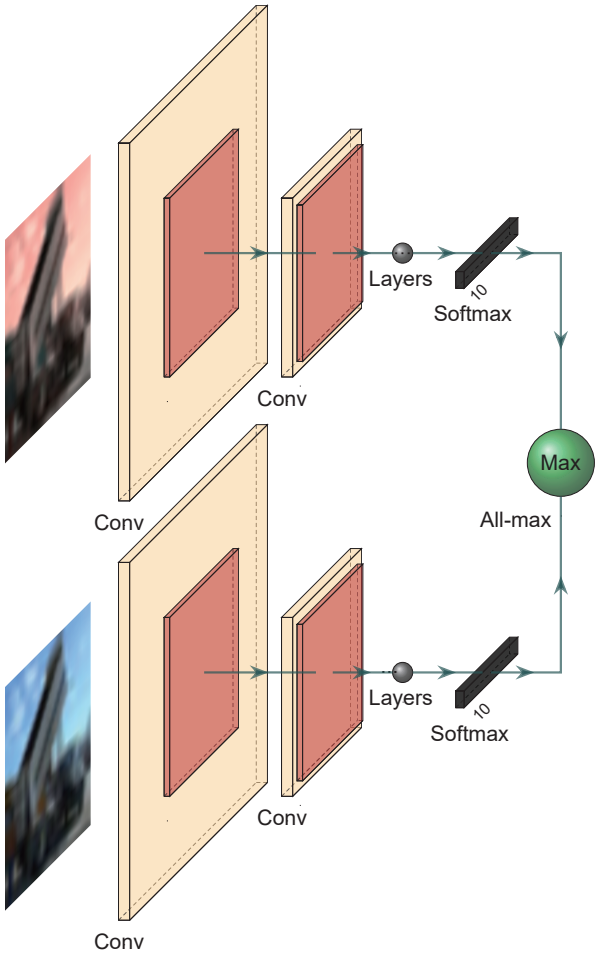

Fig. 5 Pseudo-Siamese CNN architecture. Convolution network is abbreviated as conv.

used, the class-max prediction does not show improved performance. The pseudo-Siamese LeNet5 model, which integrates RGB and HSV FULL color spaces and prioritizes Class 8 and Class 10 for maximum prediction, demonstrates performance parity with the LeNet5 model trained solely on RGB data. However, when the prediction method was changed to all-max, the performance improved. The all-max prediction method uses the complementary information of different color spaces to enhance the classification performance. For instance, the RGB color space could capture the edge and texture features of an object, while HSV captured the hue and saturation features. These features varied in their relevance depending on the scene and lighting conditions. Therefore, the CNN learns to fuse the information from both color spaces during the training process. This is one of the reasons why the all-max prediction method surpasses other methods. The experiments show that combining RGB and XYZ was the best way to train the pseudo-Siamese network. This could be because XYZ compensates for some image features that cannot be correctly recognized in RGB. The XYZ color space captures the luminance and chromaticity features of an object. Luminance is the brightness or intensity of light, and chromaticity is the colorfulness or hue and saturation of light. This also matches our findings in the empirical experiments in Section 3.

Here are answers to the two research questions mentioned in Section 1 that motivated our study:

(1) Regardless of whether group or depthwise separable convolution is used, any neural network that is relatively shallow and only trained in RGB, may still need to better learn some image features. Combining color spaces to train the proposed model can enhance its learning ability and yield better performance.

Table 4 Ablation study on CIFAR10 dataset. Bolds indicate the optimal data.

CNN model	Color space	Accuracy (%)	Precision	Recall	F1
LeNet5	RGB	63.8	0.635	0.638	0.637
	RGB + HSV FULL (all)	**65.8**	**0.654**	**0.658**	**0.655**
	RGB + XYZ (all)	65.8	0.653	0.658	0.655
	RGB + HSV FULL (2, 8, 9, 10)	63.7	0.634	0.637	0.631
	RGB + HSV FULL (8, 10)	63.6	0.634	0.636	0.632
AlexNet	RGB	92.1	0.921	0.921	0.921
	RGB + XYZ (all)	**92.4**	**0.924**	**0.924**	**0.924**
	RGB + YUV (all)	91.9	0.919	0.919	0.919
	RGB + XYZ (6)	91.9	0.920	0.919	0.919
	RGB + YUV (2)	91.8	0.919	0.919	0.918
MobileNet	RGB	83.7	0.836	0.837	0.836
	RGB + XYZ (all)	**85.2**	**0.851**	**0.852**	**0.851**
	RGB + YUV (all)	84.9	0.848	0.849	0.848

(2) Our proposed method combined two color spaces, and did not require modification of the underlying CNN structure. It also delivered better performance.

4.4 Limitation and threat to validity

Because our proposed pseudo-Siamese CNN involves two neural networks, the training overheads can be quite high. This is especially so for deep neural networks, as discussed in Section 3.4. Deep CNNs have been shown to capture sufficient features in RGB alone, making it unnecessary to introduce other color spaces to improve performance. Nevertheless, our proposed method may prove beneficial when using shallow CNNs for image classification.

Our understanding of the neural network's feature learning process in different color spaces is limited: It is not possible to identify the exact features learned, and it lacks interpretability. Exploring this in more depth will form part of our future work.

In this study, we have only validated our approach on four commonly used datasets. We anticipate that there will be other professionally relevant datasets (such as medical image datasets): We look forward to exploring the application of our proposed method to such datasets, which will also form part of our future work.

5 Conclusion and Future Work

In this paper, we have observed, through extensive experiments, that training CNNs with images from different color spaces results in varying performances. We have demonstrated that different color spaces can affect the accuracy and stability of CNNs, especially for deeper architectures like VGG and DenseNet. If the structure or parameters of a CNN are not complex or deep, then it may not be able to fully utilize the color space information, thus affecting its performance. We have also shown that RGB delivers the most stable result across various datasets and network depths. Moreover, we have analyzed the recall results of different color spaces and identified that some classes are more sensitive to certain color spaces than others. Different color spaces may have different expression abilities for different image features, which may affect CNN's ability to recognize some categories of images. This finding motivated us to design a pseudo-Siamese network that leverages the complementary strengths of two color spaces to enhance classification performance. Our proposed method is simple yet effective: It does not require any modification of the original networks, and it only trains them with two color spaces and a ratio loss function. We have empirically verified that combining RGB and XYZ color spaces produces the best results for our method. We have conducted extensive experiments to explore the effects of the ratio loss function and found that this ratio loss function can maximize the performance of our model.

Our work opens up several promising directions for future research. One of them is to investigate the relationship between color spaces and features for different classes. If such a relationship can be established, it may be possible to augment the features of the images according to their color spaces before training the networks. Another direction is to apply our combined color space technique to other computer vision tasks, such as image segmentation, object tracking, or object detection. We believe that our technique can provide a novel and effective way to improve the performance of CNNs for various computer vision applications.

Acknowledgment

We would like to thank the anonymous reviewers for their valuable comments and suggestions, which have greatly improved the quality of this paper.

This work was supported by the Science and Technology Development Fund of Macau, Macao SAR (Nos. 0021/2023/RIA1 and 0046/2021/A) and the Faculty Research Grant of Macau University of Science and Technology (No. FRG-22-103-FIE). This work was also in part supported by the National Natural Science Foundation of China (Nos. 61872167 and 61502205).

References

[1] D. Chai and A. Bouzerdoum, A Bayesian approach to skin color classification in YCbCr color space, in *Proc. of Intelligent Systems and Technologies for the New Millennium*, Kuala Lumpur, Malaysia, 2000, pp. 421–424.

[2] B. Chen, S. Shi, J. Sun, B. Chen, K. Guo, L. Du, J. Yang, Q. Xu, S. Song, and W. Gong, Using HSI color space to improve the multispectral lidar classification error caused by measurement geometry, *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3567–3579, 2021.

[3] D. S. Y. Kartika and D. Herumurti, Koi fish classification based on HSV color space, in *Proc. Int. Conf. Information & Communication Technology and Systems* (*ICTS*), Surabaya, Indonesia, 2016, pp. 96–100.

[4] E. A. Khan and E. Reinhard, Evaluation of color spaces for edge classification in outdoor scenes, in *Proc. IEEE*

Int. Conf. Image Processing 2005, Genova, Italy, 2005, pp. 952–955.

[5] O. R. Indriani, E. J. Kusuma, C. A. Sari, E. H. Rachmawanto, and D. R. I. M. Setiadi, Tomatoes classification using K-NN based on GLCM and HSV color space, in *Proc. Int. Conf. Innovative and Creative Information Technology* (*ICITech*), Salatiga, Indonesia, 2017, pp. 1–6.

[6] S. Sural, G. Qian, and S. Pramanik, Segmentation and histogram generation using the HSV color space for image retrieval, in *Proc. Proceedings. Int. Conf. Image Processing*, Rochester, NY, USA, 2002, pp. 589–592.

[7] P. Ganesan, V. Rajini, and R. I. Rajkumar, Segmentation and edge detection of color images using CIELAB color space and edge detectors, in *Proc. INTERACT-2010*, Chennai, India, 2010, pp. 393–397.

[8] N. M. Kwok, Q. P. Ha, and G. Fang, Effect of color space on color image segmentation, in *Proc. 2nd Int. Congress on Image and Signal Processing*, Tianjin, China, 2009, pp. 1–5.

[9] A. B. A. Hassanat, M. Alkasassbeh, M. Al-Awadi, and E. A. A. Alhasanat, Color-based object segmentation method using artificial neural network, *Simul. Model. Pract. Theory*, vol. 64, pp. 3–17, 2016.

[10] J. Liu and X. Zhong, An object tracking method based on mean shift algorithm with HSV color space and texture features, *Clust. Comput.*, vol. 22, no. 3, pp. 6079–6090, 2019.

[11] P. Hidayatullah and M. Zuhdi, Color-texture based object tracking using HSV color space and local binary pattern, *Int. J. Electr. Eng. Inform.*, vol. 7, no. 2, pp. 161–174, 2015.

[12] S. Saravanakumar, A. Vadivel, and C. G. Saneem Ahmed, Multiple object tracking using HSV color space, in *Proc. 2011 Int. Conf. Communication, Computing & Security*, Rourkela, India, 2011, pp. 247–252.

[13] R. Azhar, D. Tuwohingide, D. Kamudi, Sarimuddin, and N. Suciati, Batik image classification using SIFT feature extraction, bag of features and support vector machine, *Procedia Comput. Sci.*, vol. 72, pp. 24–30, 2015.

[14] Q. Li and X. Wang, Image classification based on SIFT and SVM, in *Proc. IEEE/ACIS 17th Int. Conf. Computer and Information Science* (*ICIS*), Singapore, Singapore, 2018, pp. 762–765.

[15] Z. Xian, M. Azam, and N. Bouguila, Statistical modeling using bounded asymmetric Gaussian mixtures: Application to human action and gender recognition, in *Proc. IEEE 22nd Int. Conf. Information Reuse and Integration for Data Science* (*IRI*), Las Vegas, NV, USA, 2021, pp. 41–48.

[16] Z. Xian, M. Azam, M. Amayri, W. Fan, and N. Bouguila, Bounded asymmetric Gaussian mixture-based hidden Markov models, in *Hidden Markov Models and Applications*, N. Bouguila, W. Fan, and M. Amayri, eds. Cham, Switzerland: Springer International Publishing, 2022, pp. 33–58.

[17] S. L. Phung, A. Bouzerdoum, and D. Chai, A novel skin color model in YCbCr color space and its application to human face detection, in *Proc. Int. Conf. Image Processing*, Rochester, NY, USA, 2002, pp. 289–292.

[18] I. Philipp and T. Rath, Improving plant discrimination in image processing by use of different colour space transformations, *Comput. Electron. Agric.*, vol. 35, no. 1, pp. 1–15, 2002.

[19] P. A. Herrault, D. Sheeren, M. Fauvel, and M. Paegelow, Automatic extraction of forests from historical maps based on unsupervised classification in the CIELab color space, in *Geographic Information Science at the Heart of Europe*. Cham, Switzerland: Springer International Publishing, 2013, pp. 95–112.

[20] J. Chen and J. Lei, Research on color image classification based on HSV color space, in *Proc. Second Int. Conf. Instrumentation, Measurement, Computer, Communication and Control*, Harbin, China, 2012, pp. 944–947.

[21] H. K. Kim, J. H. Park, and H. Y. Jung, An efficient color space for deep-learning based traffic light recognition, *J. Adv. Transp.*, vol. 2018, p. 2365414, 2018.

[22] R. Girshick, J. Donahue, T. Darrell, and J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014, pp. 580–587.

[23] S. Ren, K. He, R. Girshick, and J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, in *Proc. 28th Int. Conf. Neural Information Processing Systems*, Montreal, Canada, 2015, pp. 91–99.

[24] J. Dai, Y. Li, K. He, and J. Sun, RFCN:Object detection via region-based fully convolutional networks, in *Proc. 30th International Conference on Neural Information Processing Systems*, Barcelona, Spain, 2016, pp. 379–387.

[25] Y. Yuan, G. Zeng, Z. Chen, and Y. Gao, Color image quality assessment with multi deep convolutional networks, in *Proc. IEEE 4th Int. Conf. Signal and Image Processing* (*ICSIP*), Wuxi, China, 2019, pp. 934–941.

[26] S. Khan and M. Narvekar, Novel fusion of color balancing and superpixel based approach for detection of tomato plant diseases in natural complex environment, *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 6, pp. 3506–3516, 2022.

[27] M. Mehmood, N. Alshammari, S. A. Alanazi, A. Basharat, F. Ahmad, M. Sajjad, and K. Junaid, Improved colorization and classification of intracranial tumor expanse in MRI images via hybrid scheme of Pix2Pix-cGANs and NASNet-large, *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 7, pp. 4358–4374, 2022.

[28] S. N. Gowda and C. Yuan, ColorNet: investigating the importance of color spaces for image classification, in *Proc. Computer Vision—ACCV 2018*, Cham, Switzerland, 2019, pp. 581–596.

[29] S. Bianco, C. Cusano, P. Napoletano, and R. Schettini, Improving CNN-based texture classification by color balancing, *J. Imaging*, vol. 3, no. 3, p. 33, 2017.

[30] Y. LeCun, L. D. Jackel, L. Bottou, A. Brunot, C. Cortes, J. S. Denker, H. Drucker, I. R. Subramanian, U. Muller, E. Sackinger, P. Y. Simard, and V. N. Vapnik, Comparison of learning algorithms for hand written digit recognition, in *Proc. 5th Int. Conf. on Artificial Neural Networks*, Perth, Australia, 1995, pp. 53–60.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet

classification with deep convolutional neural networks, *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.
[32] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and F. F. Li, ImageNet: A large-scale hierarchical image database, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*. Miami, FL, USA, 2009, pp. 248–255.
[33] F. Mamalet and C. Garcia, Simplifying ConvNets for fast learning, in *Artificial Neural Networks and Machine Learning*. Berlin, Germany: Springer, 2012, pp. 58–65.
[34] L. Sifre and S. Mallat, Rigid-motion scattering for texture classification, arXiv preprint arXiv: 1403.1687, 2014.
[35] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv: 1704.04861, 2017.
[36] S. N. Gowda, Human activity recognition using combinatorial deep belief networks, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, Honolulu, HI, USA, 2017, pp. 1–6.
[37] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, in *Proc. of the 3rd International Conference on Learning Representations*, San Diego, CA, USA, 2015, pp. 1–14.
[38] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, Going deeper with convolutions, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 1–9.
[39] S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 6, no. 2, pp. 107–116, 1998.
[40] S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in *Proc. 32nd Int. Conf. Int. Conf. Machine Learning*, Lille, France, 2015, pp. 448–456.
[41] K. He, X. Zhang, S. Ren, and J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in *Proc. IEEE Int. Conf. Computer Vision* (*ICCV*), Santiago, Chile, 2015, pp. 1–9.
[42] A. M. Saxe, J. L. McClelland, and S. Ganguli, Exact solutions to the nonlinear dynamics of learning in deep linear neural networks, in *Proc. of the 2nd Int. Conf. Learning Representations*, 2014, pp. 1–22.
[43] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, Backpropagation applied to handwritten zip code recognition, *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.
[44] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 770–778.
[45] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, Densely connected convolutional networks, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 4700–4708.
[46] G. Bebis and M. Georgiopoulos, Feed-forward neural networks, *IEEE Potentials*, vol. 13, no. 4, pp. 27–31, 1994.
[47] N. Vandenbroucke, L. Macaire, and J. G. Postaire, Color pixels classification in an hybrid color space, in *Proc. Int. Conf. Image Processing*, Chicago, IL, USA, 1998, pp. 176–180.
[48] B. D. Zarit, B. J. Super, and F. K. H. Quek, Comparison of five color models in skin pixel classification, in *Proc. Int. Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, Corfu, Greece, 1999.
[49] A. Krizhevsky and G. Hinton, Learning multiple layers of features from tiny images, Tech. Rep. 001, University of Toronto, Toronto, Canada, 2009.
[50] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, Reading digits in natural images with unsupervised feature learning, in *Proc. of the NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, Sierra Nevada, Spain, 2011, pp. 1–9.
[51] A. Coates, A. Ng, and H. Lee, An analysis of single-layer networks in unsupervised feature learning, in *Proc. 14th Int. Conf. Artificial Intelligence and Statistics*, Fort Lauderdale, FL, USA, 2011, pp. 215–223.
[52] N. A. Ibraheem, M. M. Hasan, R. Z. Khan, and P. K. Mishra, Understanding color models: A review, *ARPN J. Sci. Technol.*, vol. 2, no. 3, pp. 265–275, 2012.
[53] S. I. Nin, J. M. Kasson, and W. Plouffe, Printing CIELAB images on a CMYK printer using tri-linear interpolation, in *Proc. SPIE/IS&T 1992 Symp. Electronic Imaging*: *Science and Technology*, 1992, San Jose, CA, USA, pp. 316–324.
[54] D. J. Sawicki and W. Miziolek, Human colour skin detection in CMYK colour space, *IET Image Process.*, vol. 9, no. 9, pp. 751–757, 2015.
[55] K. Plataniotis and A. N. Venetsanopoulos, *Color Image Processing and Applications*. Berlin, Germany: Springer Science & Business Media, 2000.
[56] T. Smith and J. Guild, The C.I.E. colorimetric standards and their use, *Trans. Opt. Soc.*, vol. 33, no. 3, pp. 73–134, 1932.
[57] C. Wyman, P. P. Sloan, and P. Shirley, Simple analytic approximations to the CIE xyz colormatching functions, *J. Comput. Graph. Tech.*, vol. 2, no. 2, p. 11, 2013.
[58] H. S. Fairman, M. H. Brill, and H. Hemmendinger, How the CIE 1931 color-matching functions were derived from Wright-Guild data, *Color Res. Appl.*, vol. 22, no. 1, p. 11–23, 1997.
[59] C. Poynton, YUV and luminance considered harmful, in *The Morgan Kaufmann Series in Computer Graphics*, M. Gross and H. Pfister, eds. San Francisco, CA, USA: Morgan Kaufmann, 2003, pp. 595–600.
[60] A. Ford and A. Roberts, *Colour Space Conversions*, London, UK: Westminster University, 1998.
[61] K. Jack, Color spaces, in *Video Demystified, Fifth Edition*, K. Jack, ed. Burlington, VT, USA: Newnes, 2007, pp. 15–36.
[62] L. A. Jeni, J. F. Cohn, and F. De La Torre, Facing imbalanced data: Recommendations for the use of performance metrics, in *Proc. Humaine Association Conf. Affective Computing and Intelligent Interaction*, Geneva, Switzerland, 2013, pp. 245–251.
[63] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv: 1412.6980, 2014.

Zixiang Xian received the MS degree from Concordia University, Canada, in 2021. Before earning his MS degree, he worked as a senior software engineer at Kingsoft Office Software Corporation (WPS), China. He is currently pursuing the PhD degree at School of Computer Science and Engineering, Macau University of Science and Technology (MUST), China. His research interests include applying deep learning and artificial intelligence techniques to software engineering problems.

Dave Towey received the BA and MA degrees in computer science, linguistics, and languages from Trinity College Dublin, The University of Dublin, Ireland, in 1998 and 2000, respectively, the postgraduate certificates in teaching English to speakers of other languages (PgCertTESOL) from The Open University of Hong Kong, China in 2011, and in higher education (PgCertHE) from University of Nottingham, UK, in 2019, the MEd degree in education leadership from University of Bristol, UK, in 2014, and the PhD degree in computer science from The University of Hong Kong, China in 2006. He has been with University of Nottingham Ningbo China (UNNC) since 2013, where he serves as the associate dean for education and student experience, the deputy head of School of Computer Science, and as deputy director of International Doctoral Innovation Centre. Prior to joining UNNC, he spent eight years working in a variety of roles with a different Sino-foreign university in Zhuhai, China (Beijing Normal University-Hong Kong Baptist University: United International College, UIC). His research interests include technology-enhanced education and software testing (especially adaptive random testing, for which he was amongst the earliest researchers who established the field, and metamorphic testing). He co-founded the International Conference on Software Engineering (ICSE) Workshop on Metamorphic Testing in 2016. He is a fellow of Higer Education Academy (HEA), and a senior member of the ACM and IEEE. At UNNC, he is a core member of Artificial Intelligence and Optimisation (AIOP) research group, and contributes to the Science and Engineering Education (SEE) research group.

Rubing Huang received the PhD degree in computer science and technology from Huazhong University of Science and Technology, Wuhan, China, in 2013. From 2016 to 2018, he was a visiting scholar at Swinburne University of Technology and Monash University, Australia. He is currently an associate professor at School of Computer Science and Engineering, Macau University of Science and Technology (MUST), China. Before joining MUST, he worked as an associate professor at Jiangsu University, China. His research interests include artificial intelligence for software engineering, software engineering for AI, software testing, debugging, and maintenance. He has more than 70 publications in journals and proceedings, including *IEEE Transactions on Software Engineering*, *IEEE Transactions on Reliability*, *Journal of Systems and Software*, *Information and Software Technology*, *Software*: *Practice and Experience*, *Science of Computer Programming*, *IET Software*, *Expert Systems with Applications*, *International Journal of Software Engineering and Knowledge Engineering*, *IEEE Internet of Things Journal*, *Information Sciences*, *The Computer Journal*, *Security and Communication Networks*, ICSE, International Symposium on Software Reliability Engineering (ISSRE), International Conference on Software Testing, Verification, and Validation (ICST), and International Conference on Computers, Software, and Applications (COMPSAC). He is a senior member of the IEEE and the China Computer Federation. More information about him and his work is available online at https://huangrubing.github.io/.

Chuan Yue received the MEng degree in software engineering from Sun Yat-Sen University, Guangzhou, China, in 2013. He is currently pursuing the PhD degree at School of Computer Science and Engineering, Macau University of Science and Technology, Macao, China. His research interests include software engineering and software quality evaluation.