

# Color refinement using deep neural networks for enhancing color recognition in a projector-camera system

Changgu Kang<sup>1</sup>  | Meekyoung Kim<sup>2</sup>  | Sung-Hee Lee<sup>3</sup> 

<sup>1</sup>Gyeongnam National University of Science and Technology, Jinju, South Korea

<sup>2</sup>Daegu University, Gyeongsan, South Korea

<sup>3</sup>Korea Advanced Institute of Science and Technology, Daejeon, South Korea

## Correspondence

Sung-Hee Lee, Korea Advanced Institute of Science and Technology, Daejeon, South Korea.

Email: [sunghlee.lee@kaist.ac.kr](mailto:sunghlee.lee@kaist.ac.kr)

## Funding information

Gyeongnam National University of Science and Technology, Grant/Award Number: 2018 to 2020

## Abstract

In projector-camera systems, object recognition is essential to enable users to interact with physical objects. Among several input features used by the object classifier, color information is widely used as it is easily obtainable. However, the color of an object seen by the camera changes due to the projected light from the projector, which degrades the recognition performance. To solve this problem, we propose a method to restore the original color of an object from the observed color through camera. The color refinement method has been developed based on the deep neural network. The inputs to the neural network are the color of the projector light as well as the observed color of the object in multiple color spaces, including RGB, HSV, HIS, and HSL. The neural network is trained in a supervised manner. Through a number of experiments, we show that our refinement method reduces the difference from the original color and improves the object recognition rate implemented with a number of classification methods.

## KEYWORDS

color refinement, deep neural networks, projector-camera system

## 1 | INTRODUCTION

Projector-camera systems (Figure 1) have been extensively used as a hardware configuration for the table top interface for various applications in media art, education, and entertainment as they effectively visualize information in space and allow users to manipulate visual entities intuitively.<sup>1–8</sup> Applied in augmented reality (AR), the system realizes the projection-based AR and can visualize relevant information on user's hand and objects.<sup>9–11</sup>

In the projector-camera system, user interaction is implemented in many ways. A method using multitouch displays allows users to directly manipulate virtual entities by fingers without using additional devices.<sup>1</sup> Another

popular method uses tangible objects for the user interaction to provide tactile feedback to the users.<sup>9</sup>

A key component for the tangible object-based interaction is the object classification to identify the physical objects. For this, a number of features are used, such as shape, color of an object, or specific markers attached to the object.<sup>11,12</sup> While appropriate features need to be selected according to the characteristics of the system, color is a useful feature especially when multiple objects have the same shape but different colors.

To identify an object by color, the system needs to recognize the original color of the object, but it is a challenging task for the projector-camera system. Because the projector emits lights on top of the object to display image or video, the color of the object seen from the camera deviates from its original color (Figure 2), and the changed color decreases the accuracy of the object



**FIGURE 1** Our projector-camera system configuration

classification. Many applications have avoided this problem by only projecting colors that will not hinder color recognition much or even not projecting any light to the target objects. However, this restriction heavily limits the degree of freedom of the content.

A projector can express various colors by emitting light of a wide spectrum of wavelengths. When the emitted light overlaps with the color of an object, it is difficult to detect the original color accurately by a simple linear operation per channel due to the interference effect between the RGB channels.<sup>13</sup> Moreover, material properties of an object also affect the color change behavior significantly. Therefore, accurate color detection requires expensive equipment operated by professionals, which are not always available for regular projector-camera systems. In the image processing field, a number of studies have been conducted to model and recognize illumination,<sup>14,15</sup> but most of them assumed natural light and thus

are not suitable for the projector light that may have specific colors from the visual content.

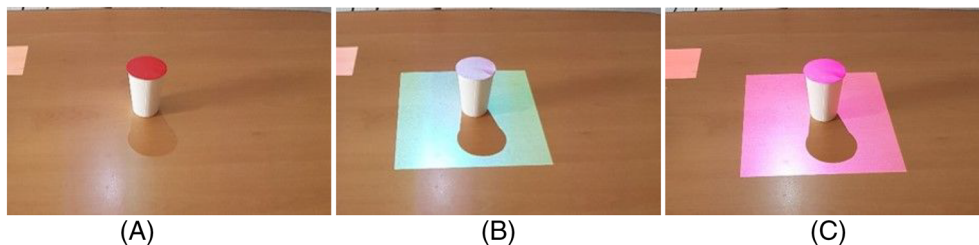
In this paper, we propose a method to estimate the original color from the observed color that has been changed variously by the projection light, and thus increase the object classification accuracy. The proposed method is based on the deep neural network, trained in a supervised manner. For the input to the deep neural network, we use the color of the projection light and the color of the object observed by camera. The observed color of the object is additionally transformed to three color spaces, HSV, HIS, and HSL, and used as input to the neural network as well. The deep neural network outputs the refined color of the object, which then is used as input to the object classifiers.

One advantage of the proposed method is the separation of the object classifier and visual content. In the projector-camera system, training data for a color-based object classifier need be collected while particular visual content is projected to the objects to increase the classification rate. The classifiers often need retraining when the visual content is changed. In contrast, our deep neural network estimates the original color of an object, and thus a classifier is independent from the visual content and does not require re-training due to the content change.

The remaining part of this article proceeds as follows. After reviewing previous research related to our work in Section 2, we first present a system overview in Section 3. In Section 4, we explain data definition and collection for training and testing. Section 5 shows the configuration of neural networks and explain training process. We investigate the effectiveness of the presented method by using a number of classification algorithms in Section 6 and discuss its properties, limitations, and future work in Section 7. We conclude the paper in Section 8.

## 2 | RELATED WORK

This section reviews previous work related to restoring object colors changed by projector light. After reviewing studies on color features for image understanding, we discuss studies on identifying illumination for color restoration.



**FIGURE 2** A, Original color of an object. B,C, Colors changed by projector beam

## 2.1 | Color space

Color has traditionally been used as an important feature for image processing. Vandenbroucke et al<sup>15,16</sup> proposed image segmentation method using multiple color spaces. Vandenbroucke et al<sup>15</sup> used a hybrid color space that consists of 14 color representations in three channels and proposed a method to select the best color feature. van Erp et al<sup>8</sup> developed a method to select the optimal color space for image segmentation. Vandenbroucke et al<sup>15,16</sup> perform segmentation by selecting an appropriate color space among multiple color spaces. In contrast, we propose a method to refine color of an object by using multiple color spaces concurrently.

Pietikainen et al<sup>17</sup> evaluated color classification performance using Swain and Ballard's method<sup>18</sup> with 3D histogram and a simplified method with 1D histogram in an illumination changing environment, and found that the 1D histogram performs as well as the original method. Shih et al<sup>19</sup> studied color space that can express the whole shape of face for the face recognition. These studies on color features focused on the distribution of color rather than the color itself, or use  $(L^*, a^*, b^*)$  features that require special measurement equipment, and thus are not suitable for our system.

## 2.2 | Illumination

Ambient light changes the color of the image and may cause performance degradation of the recognition algorithm. Therefore, the illumination of a scene has been regarded as an important issue in computer vision research. Mäenpää and Pietikäinen<sup>20</sup> studied image classification by considering external illumination conditions. Joze and Drew and Bianco et al<sup>21,22</sup> proposed methods to find color constancy in illuminated environments.<sup>22</sup> developed a method to estimate illuminants by using convolutional neural networks and create a corrected image. These studies locate a local or global light source in an image to restore color. In contrast, our deep learning-based method refines the object color without

estimating the location of the illumination. This is possible because the projector-camera system has controlled light positions.

## 3 | OVERVIEW

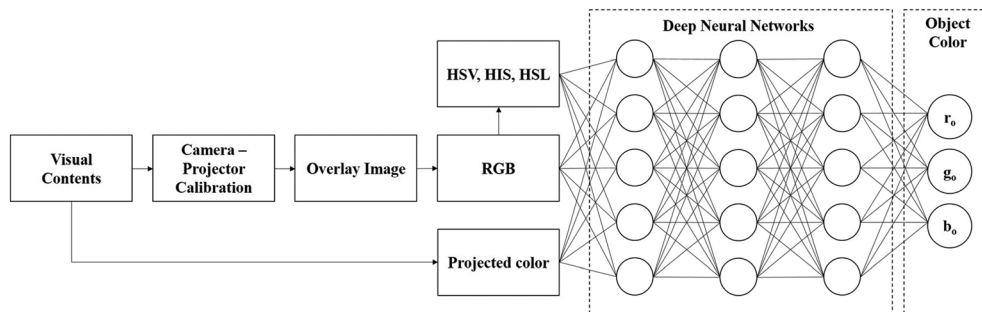
This section describes the system configuration and the overall flow to refine the colors of object. Our implementation on the projector-camera calibration and object detection is introduced here as well.

### 3.1 | System configuration

Figure 3 shows the overview of the system. In the projector-camera system, the calibration process is performed first to match the camera coordinates and the screen coordinates of the projector. An image is projected on the table, and objects are detected when placed on the table. When an object is detected, the color of the object is extracted from the camera. We use the RGB camera of Kinect with auto exposure function enabled. Our color refinement module is made with a deep neural network. The object color from the camera is the overlapped color of its own and the color of the image projected on the object. The neural network receives this RGB color and corresponding HSV, HIS, HSL space colors as inputs. In addition, the color of the projection image is provided as input to the network. The neural network then outputs RGB values of the refined color. Refined colors can be used as input for object recognition or classification, and we present the results of the object classification experiments in Section 6.2.

### 3.2 | Projector-camera calibration

The calibration process finds the coordinate mapping between the camera and the projector image so that we can acquire the position of the object recognized by the camera and the color of the light superimposed on the object. While there is an accurate method<sup>23,24</sup> for the



**FIGURE 3** The flow for our color refinement

projector-camera calibration, we implemented a simple yet effective calibration method.

Equation (1) shows a general formula for calculating a projection matrix for calibration.

$$p' = Hp \quad (1)$$

where  $p$  represents the four corners of the projector image viewed from the camera, and  $p'$  represents the four corner values  $\{(0,0), (0,W), (H,W), (H,0)\}$  for the projector image ( $W$  = image width,  $H$  = image height). We then use the correspondence between  $p$  and  $p'$  corner points to calculate the projection matrix  $H$ . The matrix  $H$  is a  $3 \times 3$  projection matrix in a homogeneous coordinate system and  $p$  and  $p'$  are 3 dimensional vectors.

### 3.3 | Object detection

In order to detect objects on the table, we use the depth image obtained by Kinect camera. The calibration for mapping a depth image to a color image is implemented by using Kinect SDK. We assume that the camera is fixed with respect to the table, and thus the distance from the camera to the table is constant. We convert the depth pixels closer than the table to 1 and remaining pixels to 0. This initial binary image contains much noise, so we denoise the image by applying erosion, Gaussian blur, and dilation operations<sup>25</sup> sequentially. Finally, we use contour detection

**Algorithm 1.** *Object detection in depth images*

**Input:** Depth image  $I$ , Distance from the camera to the table  $D$

**Output:** List of objects  $O$

Binary image  $B \leftarrow \emptyset$

**for**  $y \leftarrow 1$  to  $I.height$  **do**

**for**  $x \leftarrow 1$  to  $I.width$  **do**

$v \leftarrow I(x, y)$

**if**  $v < D$  **then**

$B(x, y) \leftarrow 1$

**else**

$B(x, y) \leftarrow 0$

**end if**

**end for**

**end for**

**Step 1:** Apply erosion, Gaussian blur, and dilation operation on the binary image sequentially

**Step 2:** Convert the resulting image to a binary image for finding contours.

**Step 3:** Apply contour detection operation on the binary image and

find chunks  $C$ .

**for**  $i \leftarrow 1$  to  $C.count$  **do**

$O.push\_back(C_i)$

**end for**

<sup>26</sup>to detect objects. This relatively simple operations were effective enough to detect object in our application where the objects have simple geometries and similar depths. More advanced methods would be necessary to detect objects in more challenging cases.<sup>14</sup> Algorithm 1 is the pseudo code for the object detection process.

## 4 | DATA DEFINITION AND COLLECTION

In this section, we first show an experiment showing the complexity of color changing phenomena due to the ambient brightness. We then explain features to train a neural network to refine colors and the process to obtain data for training and testing.

### 4.1 | Effect of ambient brightness

In the projector-camera system, an object's color superimposed by the projector light is shown in a different color. It is to be noted that the change of color is affected not only by the projector light but also by the ambient light. Table 1 shows an example that the average error between the original and the observed color varies according to different levels of the ambient brightness, from Level 1 (darkest) to 3 (brightest). The average error of color was 0.073 for the high ambient brightness, and 0.317 for the low brightness. The average error was measured as Equation (2):

$$AverageError = \frac{\sum_{i=1}^N |Color_i - Color_i^{true}|}{N} \quad (2)$$

where  $N$  denotes the number of test data.

As will be discussed later, the color refinement performance and the object classification accuracy are

**TABLE 1** Average error for each RGB channel due to projector color and ambient brightness (level 1: lowest ambient brightness, level 3: highest ambient brightness). Fluorescent light was used for the ambient light

Brightness	R	G	B	Average
Level 1	0.358	0.323	0.272	0.317
Level 2	0.098	0.100	0.123	0.107
Level 3	0.071	0.070	0.077	0.073
Average	0.175	0.164	0.157	0.166



significantly affected by the ambient brightness, and thus we consider this when collecting dataset as follows.

## 4.2 | Data acquisition

We trained a deep neural network to refine object color in a supervised manner. To this end, we prepared ground truth dataset for training and testing. We obtained a dataset in two paths as shown in Figure 4. In the first path, we randomly select a projector color and project the color on the object after the projector-camera calibration. The observed color of the object is converted into HSV, HIS, and HSL (Hue, Saturation, Value, Intensity, Lightness) color spaces to extract a number of physically meaningful features. Because H of the three color spaces is a common input value, only one value is used. Saturation has different values per color space; three separate values are used. In summary, we use 13-dimensional features as the input to the neural network. These are R, G, B, H, S (3), V, I, and L values of observed object color, and the RGB of the projector light.

In the second path, in order to obtain the original color of the object, we do not project any light on the object and record the color observed by the camera. This original RGB color of the object is the ground truth output of the neural network.

## 5 | DEEP NEURAL NETWORK TRAINING FOR COLOR REFINEMENT

In this section, we describe the structure of the neural network for color refinement and how to train the network.

### 5.1 | Neural network structure

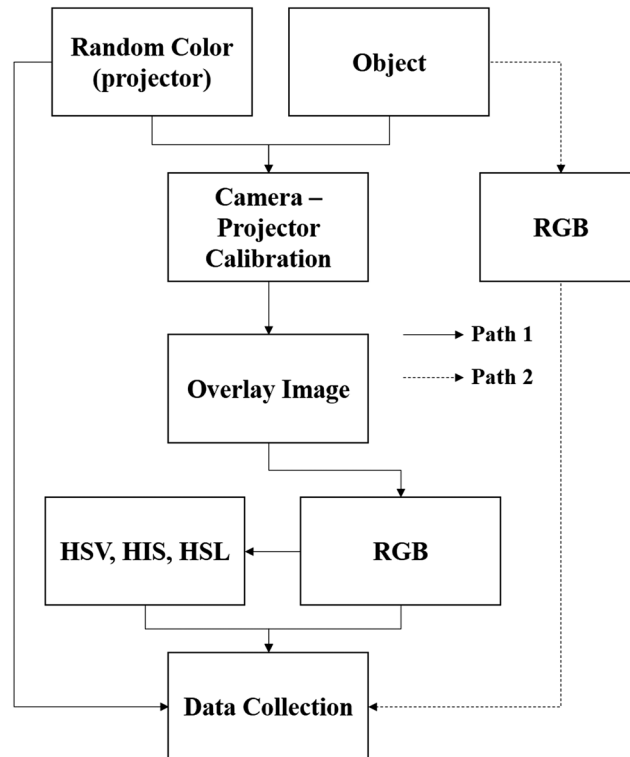
As described in Section 4, the input feature  $X$  for the first layer of the neural network is 13-dimensional data ( $X = \{rp, gp, bp, rc, gc, bc, h, shsv, v, shsi, i, shsl, l\}$ ). The neural network has five layers, and the number of cells in each layer is  $4 \times 13$ . Each layer is fully connected to adjacent layers, and the output  $L_i$  of each layer is as follows.

$$L_i(X_i) = f(W^T X_i + b_i) \quad (3)$$

where  $X_i$ ,  $f$ ,  $W_i$ , and  $b_i$  denote input, activation function, weight, and bias, respectively. We used rectified linear units (ReLUs)<sup>27</sup> as the activation function. The output  $Y$  of the network is the RGB values of the refined color ( $Y = \{ro, go, bo\}$ ).

### 5.2 | Training

We arbitrarily selected eight colors shown in Figure 5 for training and testing our method. Recognition of object color is influenced by the external brightness as well as



**FIGURE 4** Data collection flow



**FIGURE 5** Colors used for learning and experiments. Each number in the color represents the classes used in the experiment for classification (Section 6.2)

the projector light. Therefore, we collected data with varying external brightness in three levels. We collected a total of 7200 data samples (ie, 300 projector light samples per object color and brightness level). When learning was performed without differentiating the external brightness, the dataset was divided into three groups (A, B, C) of 2400 samples for three-fold cross validation. When considering the external brightness, the dataset was divided by the brightness level, and further divided into three groups (ie, 600 samples for each group) for the three-fold cross validation. We defined the loss function as the mean squared error to train the neural network.

$$\text{Loss}(X, Y; \alpha) = \|Y - \Phi(X; W_i, b_i)\|^2 \quad (4)$$

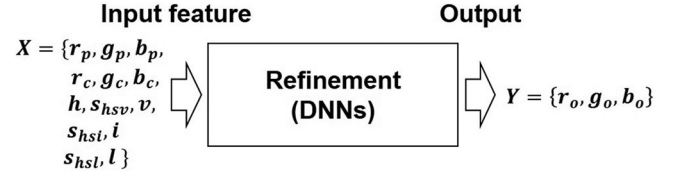
where  $\alpha$ ,  $\Phi$ ,  $X$ , and  $Y$  denote the network parameters  $\alpha = \{W_i, b_i\}$  ( $i = 1 \dots 5$ ), the output of the network, input data, and ground truth output data, respectively. We used Xavier initializer<sup>28</sup> to initialize  $W_i$  and Adam,<sup>29</sup> a stochastic gradient descent algorithm, to optimize  $\alpha$  with the learning rate set to 0.0001. Batch size was set to 50. The dropout probability for each layer was set to 0.8 to prevent overfitting.<sup>30</sup>

## 6 | EXPERIMENTS

This section describes our experiment to evaluate the proposed method. The experiment consists of two parts. The first evaluates the difference between the original color of the object and the refined color, and the second experiment investigates the improvement of the classification rate due to the color refinement.

### 6.1 | Color refinement

We show the experimental results of the difference between the refined color obtained with our method and the original color. Figure 6 shows input features



**FIGURE 6** Input features and output for color refinement

and output for this experiment. Input features have 13 dimensions for various color spaces, and output has three dimensions for the refined object color. For the three-fold cross validation, the dataset was divided into three groups (A, B, and C), two groups were used for training, and the other group was used for testing.

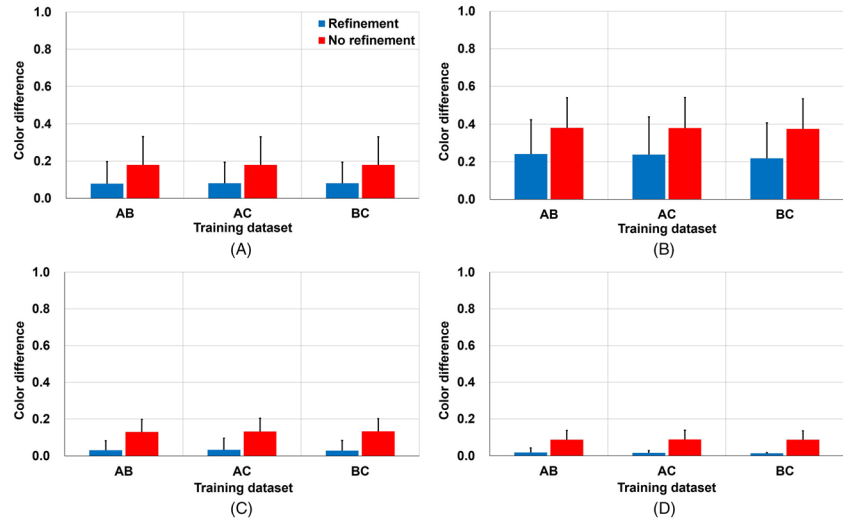
In the first experiment, the differences between the original color and the refined color, and between the original color and the camera-observed color were compared. Experiments were performed with respect to the whole data and each external brightness level, respectively. The difference was computed as the mean squared error (MSE) from the original color.

Figure 7 shows the result of the experiment. Blue and red bars denote the difference of the refined color and the difference of the camera-observed color from the original color, respectively. Figure 7A is the MSE on the entire data. The difference is reduced from 0.179 to 0.079 by color refinement, more than 50% of reduction. Figure 7 B-D are the MSEs for each ambient brightness level. Refinement resulted in difference reduction across all brightness levels. When the external brightness is dark (Figure 7B), the projector light has greater influence, which increases the error. Table 2 shows the mean and standard deviation for all data obtained from the three-fold cross validation. Exp1 and Exp2 are the results with and without refinement, respectively.

### 6.2 | Classification

To investigate the improvement of the classification rate due to our color refinement, we employed four classifiers: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), Gaussian naive Bayes (GNB), and Multilayer perceptron (MLP). All classifiers classify eight objects which have distinctive colors as shown in Figure 5. All classifiers were trained with respect to only the original colors without projector lights. For testing, we compared the error with and without the refinement. That is, without the refinement module, the classifier receives the camera-observed color as input, and with the refinement module, the classifier receives the refined color as input. Figure 8 shows the input features and output of the classifiers with and without the color

**FIGURE 7** Cross validation using three groups of data was performed. The X-axis is the training dataset, and the Y-axis is the difference (0.0 ~ 1.0) between the original color of the object and the object color changed by the projector light. A, Experiments conducted across all external brightness levels. The error is reduced by around 0.18 when the color refinement is used, more than 50% of error reduction than when the refinement was not used. B-D, Experiments for each external brightness level. B, When external brightness was low, the difference from the original color was 0.38 without refinement, and 0.22 with refinement. C and D, The ratio of error difference between with and without refinement increases with the external brightness



**TABLE 2** The difference between the refined color and the original object color (Exp1) and the difference between the camera-observed color and the original object color without refinement (Exp2). The difference of Exp1 was lower than Exp2 in all conditions

Brightness	Type	Mean	STD
Level 1	Exp1	0.232	0.190
	Exp2	0.378	0.160
Level 2	Exp1	0.031	0.056
	Exp2	0.131	0.069
Level 3	Exp1	0.016	0.016
	Exp2	0.088	0.047
All data	Exp1	0.079	0.116
	Exp2	0.179	0.150

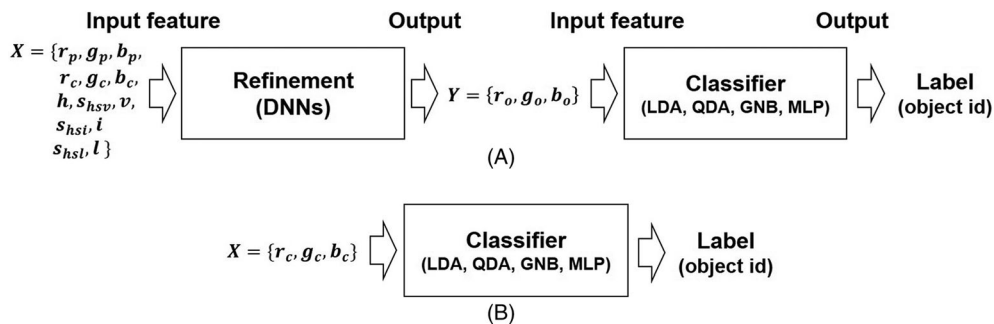
refinement. The three-fold validation method was used as in the previous experiment.

Table 3 shows the result. Although the overall error rates vary depending on the brightness, it is confirmed that the classification error rate decreases when combined with the refinement. When the external brightness was highest, the error rate was 0.13 without the

refinement and 0.01 with the refinement. The error rate was even close to zero with LDA and GNB classifiers.

Tables 4, 5, 6, and 7 compare the confusion matrices obtained with classifiers trained with (b) and without (a) refinement. It is to be noted that the error rates vary significantly with test color when classifiers were trained without refinement. In contrast, the errors are reduced overall and rather regularized when trained with color refinement. For instance, the accuracy for Class 5 was 1.0 when there was no refinement, but it decreased to 0.64 to 0.72 with refinement. In contrast, the accuracy for Classes 1 to 3 has been significantly increased with refinement. This was because the neural network parameters were optimized to increase the overall accuracy across classes.

In addition, we conducted experiments that trained classifiers with expanded input, the same input (13-dimensional features) used to train the neural network. Figure 9 compares the classification error rates of different methods. Exp1 and Exp2 are the experiments that trained with only RGB values of the object color, with and without refinement, respectively. In Exp3, the classifiers were trained with the 13-dimensional data obtained for training the deep neural network, but the color refinement was not



**FIGURE 8** Input features and output of A, the classifier associated with the color refinement and B, the classifier without color refinement

**TABLE 3** Three-fold cross validation results for LDA, QDA, GNB, and MLP methods. The classification results are similar across classification methods. As can be seen from the last column, the classification results are better with the refinement process. Especially, in case of the brightest external environment, classification error rate (green) after refinement process drops down to nearly 0%

Brightness	Type	LDA			QDA			GNB			MLP			AVR
		AB	AC	BC	AB	AC	BC	AB	AC	BC	AB	AC	BC	
Level 1	Exp1	0.73	0.62	0.61	0.74	0.65	0.66	0.74	0.63	0.63	0.74	0.63	0.63	0.67
	Exp2	0.85	0.85	0.85	0.82	0.83	0.82	0.84	0.86	0.84	0.85	0.85	0.85	0.85
Level 2	Exp1	0.05	0.07	0.04	0.05	0.08	0.04	0.05	0.07	0.04	0.05	0.08	0.05	0.06
	Exp2	0.42	0.37	0.42	0.26	0.25	0.24	0.25	0.29	0.28	0.31	0.30	0.31	0.31
Level 3	Exp1	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.00	0.01	0.01	0.00	0.01
	Exp2	0.15	0.22	0.15	0.13	0.14	0.23	0.08	0.07	0.07	0.08	0.09	0.09	0.13
All data	Exp1	0.20	0.20	0.19	0.21	0.28	0.22	0.20	0.28	0.21	0.20	0.21	0.20	0.22
	Exp2	0.38	0.42	0.38	0.36	0.36	0.36	0.39	0.38	0.39	0.35	0.35	0.34	0.38

**TABLE 4** Confusion matrices for LDA without (a) and with (b) refinement process

(a)								
	1	2	3	4	5	6	7	8
1	0.47	0	0	0	0.03	0	0	0.50
2	0	0.28	0	0	0.16	0	0.09	0.47
3	0	0	0.46	0.02	0.19	0.33	0	0
4	0	0	0	0.59	0.41	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0	0.24	0.76	0	0
7	0	0	0	0	0.34	0	0.49	0.17
8	0	0	0	0	0.30	0	0	0.70
(b)								
	1	2	3	4	5	6	7	8
1	0.78	0.05	0	0	0	0	0.01	0.16
2	0	0.88	0	0.01	0.01	0	0.04	0.06
3	0	0	0.84	0.01	0.05	0.06	0.04	0
4	0	0	0.04	0.76	0.06	0	0.12	0.02
5	0	0	0	0.08	0.72	0.04	0.13	0.03
6	0	0	0.02	0.01	0.09	0.84	0.04	0
7	0	0.01	0	0.06	0.04	0	0.87	0.02
8	0	0.06	0	0.08	0.08	0	0.08	0.69

**TABLE 5** Confusion matrices for QDA without (a) and with (b) refinement process

(a)								
	1	2	3	4	5	6	7	8
1	0.29	0.01	0	0.01	0	0	0.11	0.58
2	0	0.62	0	0.06	0.06	0	0.14	0.12
3	0	0	0.56	0.04	0.14	0.24	0.02	0
4	0	0	0	0.71	0.29	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0.02	0.20	0.77	0.01	0
7	0	0	0	0.03	0.21	0	0.76	0
8	0	0.03	0	0.01	0.24	0	0.35	0.37
(b)								
	1	2	3	4	5	6	7	8
1	0.77	0.08	0	0	0	0	0.01	0.14
2	0	0.93	0	0	0	0	0.04	0.03
3	0	0	0.84	0	0.03	0.07	0.06	0
4	0	0.01	0.03	0.75	0.01	0.02	0.18	0
5	0	0.02	0	0.07	0.64	0.05	0.22	0
6	0	0	0.02	0.01	0.06	0.85	0.06	0
7	0	0.02	0	0.05	0.01	0	0.92	0
8	0	0.15	0	0.07	0.01	0	0.17	0.60

used. As shown in Figure 9, the error rate (0.31) was found to be lower than that of Exp2 (0.38). However, it is still higher than our proposed method (Exp1).

We evaluated the performance according to the number of classes, varying from three to eight, as shown in Figure 10.

Regardless of the class sizes, the error rate is lower with the refinement process. As the number of classes

increases, the error rate increases in general, but the refinement process resulted in lower increasing rate. These experiments suggest the advantages of using the color refinement hold with challenging cases.

Finally, we evaluated the error rates when only one color space was used to compare with our multicolor space method. The error rate with only one color space (HSV, HSI, and HSL) was over 0.6, much higher than that



**TABLE 6** Confusion matrices for GNB without (a) and with (b) refinement process

(a)								
	1	2	3	4	5	6	7	8
1	0.38	0.01	0	0	0.01	0	0.11	0.49
2	0	0.53	0	0	0.15	0	0.10	0.22
3	0	0	0.47	0.04	0.18	0.31	0	0
4	0	0	0	0.66	0.34	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0	0.24	0.76	0	0
7	0	0	0	0.07	0.26	0	0.66	0.01
8	0	0.02	0	0	0.27	0	0.34	0.37

(b)								
	1	2	3	4	5	6	7	8
1	0.78	0.07	0	0	0	0	0.01	0.14
2	0	0.92	0	0.01	0	0	0.04	0.03
3	0	0	0.84	0.01	0.03	0.07	0.05	0
4	0	0.01	0.03	0.78	0.02	0.01	0.15	0
5	0	0.01	0	0.10	0.69	0.04	0.15	0.01
6	0	0	0.02	0.01	0.07	0.85	0.05	0
7	0	0.01	0	0.07	0.01	0	0.90	0.01
8	0	0.14	0	0.10	0.03	0	0.12	0.61

of our method, as shown in Figure 11. This experiment shows that a single color space is insufficient to restore the original color of the object.

The average compute time of the whole refinement process is about 1 ms, with a desktop computer with an Intel Core i7 at 3.7GHz CPU and GTX 1080 Ti 11GB.

## 7 | LIMITATION AND FUTURE WORK

We evaluated the performance of the proposed method through various experiments. This section discusses the limitations of the proposed method and possible future work to improve the method.

As shown in Figure 6 and Table 2, even with the color refinement, the classification rate was limited for low external brightness. In addition, Tables 4–6 indicate that the classification rate for some classes decreases when the refinement was used. Our deep neural network has a rather straightforward structure, and developing more effective network structure that solves these limitation remains as an important future research direction.

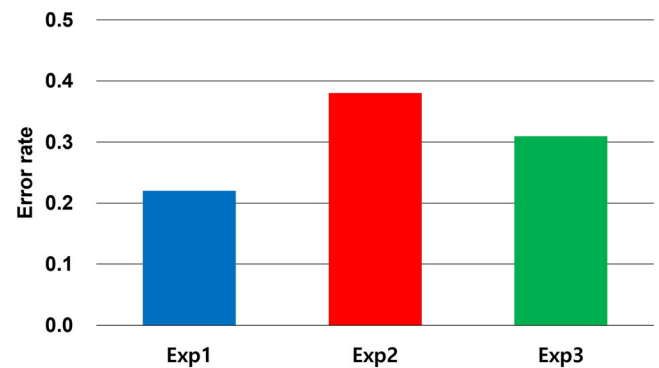
As different projectors have different light intensity, a neural network trained with respect to one type of

**TABLE 7** Confusion matrices for MLP without (a) and with (b) refinement process

(a)								
	1	2	3	4	5	6	7	8
1	0.46	0	0	0	0.02	0	0.01	0.51
2	0	0.36	0	0	0.16	0	0.04	0.44
3	0	0	0.64	0.03	0.19	0.14	0	0
4	0	0	0	0.63	0.37	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0	0.25	0.75	0	0
7	0	0	0	0	0.25	0	0.68	0.08
8	0	0	0	0	0.29	0	0.02	0.70

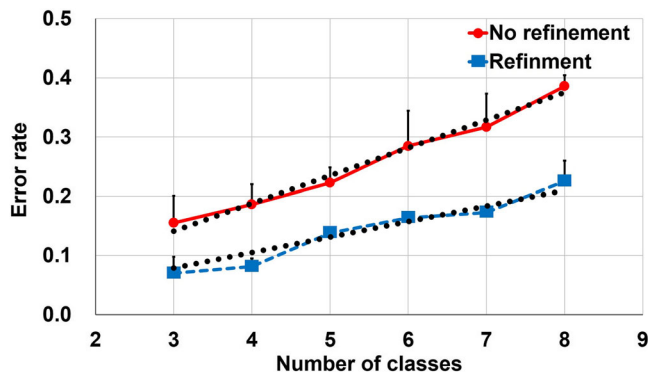
  

(b)								
	1	2	3	4	5	6	7	8
1	0.78	0.02	0	0	0	0	0	0.20
2	0.01	0.81	0	0	0.01	0.01	0.03	0.14
3	0	0	0.85	0.03	0.05	0.05	0.02	0
4	0	0	0.02	0.75	0.06	0.02	0.13	0.02
5	0	0	0	0.06	0.78	0.01	0.14	0.02
6	0	0	0.05	0.01	0.09	0.82	0.02	0.01
7	0	0.01	0.01	0.05	0.06	0.02	0.82	0.04
8	0	0.02	0.01	0.04	0.07	0.01	0.12	0.75

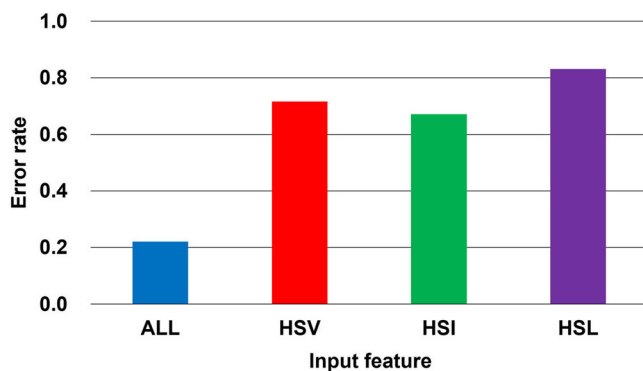
**FIGURE 9** Classification errors by different input features. Exp1 (with refinement) and Exp2 (without refinement) denote the error rates of classifiers that use RGB colors as input. In Exp3, a classifier was trained to receive the 13-dimensional feature vector, which was used to train the refinement module. No refinement was applied in Exp3. The error rate of the proposed method (Exp1) is the lowest

projector may have different performance with other type of projectors, and thus re-training may be necessary.

This limitation could be alleviated by using device-independent features such as  $(L^*, u^*, v^*)$ <sup>31</sup>  $(L^*, a^*, b^*)$ <sup>32</sup>. Moreover, it is worth exploring to find more efficient color space for the color refinement.



**FIGURE 10** Error rates according to the number of classes (3–8). In general, the error rate increases with the number of classes, but the increasing rate is lower when the refinement process was used. Black-dotted lines are linear regression of the error rates



**FIGURE 11** Error rates according to the type of input feature (all features, HSV, HSI, and HSL)

## 8 | CONCLUSION

In this paper, we proposed a method to improve object recognition in the projector-camera system. To overcome the phenomena that the recognition rate decreases as the color of an object changes due to the projected light, we proposed a method to refine the color of the object by using a deep neural network. A number of experiments showed that our method improves the object classification rate significantly regardless of the level of ambient brightness and the size of classes. Our method can contribute to giving more freedom to visual content design for the interactive applications based on projection AR.

## ACKNOWLEDGMENT

This work was supported by Gyeongnam National University of Science and Technology Grant in 2018 to 2020.

## ORCID

Changgu Kang <https://orcid.org/0000-0003-4060-6835>

Meekyoung Kim <https://orcid.org/0000-0002-3750-0346>

Sung-Hee Lee <https://orcid.org/0000-0001-6604-4709>

## REFERENCES

- Creed C, Sivell J, Sear J. Multi-touch tables for exploring heritage content in public spaces. In: *Visual Heritage in the Digital Age*. New York USA: Springer, 2013; pp. 67–90.
- Guo Y, Chu S-C, Liu Z, Qiu C, Luo H, Tan J. A real-time interactive system of surface reconstruction and dynamic projection mapping with rgb-depth sensor and projector. *Int J Distrib Sens Netw*. 2018;14(7):1550147718790853.
- Kaltenbrunner M, Bencina R. Reactivation: a computer-vision framework for table-based tangible interaction. In: *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*. ACM, 2007; p. 69–74.
- Rehg JM, Flagg M, Cham T-J, Sukthankar R, Sukthankar G. Projected light displays using visual feedback. In: *Control, Automation, Robotics and Vision, 2002. ICARCV 2002. 7th International Conference on*, IEEE, 2002 2;926–932.
- Ridel B, Reuter P, Laviolle J, Mellado N, Couture N, Granier X. The revealing flashlight: interactive spatial augmented reality for detail exploration of cultural heritage artifacts. *J Comput Cultural Herit (JOCCH)*. 2014;7(2):6.
- Sukthankar R, Stockton RG, Mullin MD. Self-calibrating camera-assisted presentation interface. In: *Proceedings of International Conference on Automation, Control, Robotics and Computer Vision*, 2000; 2000.
- Sukthankar R, Stockton RG, Mullin MD. Smarter presentations: exploiting homography in camera-projector systems. In: *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, IEEE, 2001 1; 247–253.
- van Erp JBF, Toet A, Meijer K, Janssen J, de Jong A. Subjective user experience and performance with active tangibles on a tabletop interface. In: *International Conference on Distributed, Ambient, and Pervasive Interactions*. Springer, 2015; 212–223.
- Dal Corso A, Olsen, M, Steenstrup, KH, et al. Virtualtable: a projection augmented reality game. In: *SIGGRAPH Asia 2015 Posters*. ACM, 2015; 40.
- Doshi A, Smith RT, Thomas BH, Bouras C. Use of projector based augmented reality to improve manual spot-welding precision and accuracy for automotive manufacturing. *Int J Adv Manuf Technol*. 2017;89(5–8):1279–1293.
- Mewes A, Heinrich F, Kägebein U, Hensen B, Wacker F, Hansen C. Projector-based augmented reality system for interventional visualization inside mri scanners. *Int J Med Rob Comput Assisted Surg*. 2019;15(1):e1950.
- Kaltenbrunner M, Echtler F. The tuio 2.0 protocol: An abstraction framework for tangible interactive surfaces. *Proc ACM Human-Computer Inter*. 2018;2(EICS):8.
- Chen X, Yang X, Xiao S, Li M. Color mixing property of a projector-camera system. In: *Proceedings of the 5th ACM/IEEE International Workshop on Projector Camera Systems*. ACM, 2008; 14.

14. Jalal A, Kamal S. Improved behavior monitoring and classification using cues parameters extraction from camera array images. *Int J Inter Multi Artificial Int.* 2018;5(2).
15. Vandenbroucke N, Macaire L, Postaire J-G. Color pixels classification in an hybrid color space. In *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on.* IEEE, 1998; 1: 176–180.
16. Vandenbroucke N, Busin L, Macaire L. Unsupervised color-image segmentation by multicolor space iterative pixel classification. *J Electron Imaging.* 2015;24(2):023032.
17. Pietikainen M, Nieminen S, Marszalec E, Ojala T. Accurate color discrimination with classification based on feature distributions. In *Pattern Recognition, 1996, Proceedings of the 13th International Conference on.* IEEE, 1996; 3: 833–1838.
18. Swain MJ, Ballard DH. Color indexing. *Int J Comput Vis.* 1991;7(1):11–32.
19. Shih P, Liu C. Evolving effective color features for improving frgc baseline performance. In *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on,* IEEE, 2005; p. 156.
20. Mäenpää T, Pietikäinen M. Classification with color and texture: jointly or separately? *Pattern Recogn.* 2004;37(8):1629–1640.
21. Joze HRV, Drew MS. Exemplar-based color constancy and multiple illumination. *IEEE Trans Pattern Anal Mach Intell.* 2014;36(5):860–873.
22. Bianco S, Cusano C, Schettini R. Color constancy using cnns. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops,* June 2015.
23. Moreno D, Taubin G. Simple, accurate, and robust projector-camera calibration. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on,* IEEE, 2012; 464–471.
24. Yang L, Normand J-M, Moreau G. Practical and precise projector-camera calibration. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR),* IEEE, 2016; 63–70.
25. Jain AK. *Fundamentals of Digital Image Processing* Englewood Cliffs, NJ: Prentice Hall; 1989.
26. Suzuki S. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing.* 1985;30(1):32–46.
27. Nair V, Hinton GE. Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10),* 2010; 807–814.
28. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics,* 2010; 49–256.
29. Kingma DP, Ba, J. Adam: A Method for Stochastic Optimization. *arXiv Preprint arXiv:1412.6980,* 2014.
30. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res.* 2014;15(1):1929–1958.
31. Schwiegerling J. *Field Guide to Visual and Ophthalmic Optics* Washington USA: Spie; 2004.
32. Tominaga S. Color classification of natural color images. *Color Res Appl.* 1992;17(4):230–239.

## AUTHOR BIOGRAPHIES



**Prof. Changgu Kang** received the MS and PhD degree in Computer Science from Gwangju Institute of Science and Technology (GIST) in 2010 and 2017, respectively. He is currently an assistant professor of the Department of Computer Engineering at Gyeongnam National University of Science and Technology. His research interests include character animation, machine learning, motion planning, VR/AR, and interactive media.



**Prof. Meekyoung Kim** received the MS in Mathematics and the PhD degree in Computer Science from Korea Advanced Institute of Science and Technology (KAIST), Korea in 2010 and 2017, respectively, and the BS degree in Mathematics from Kyoungbook National University, Korea, in 2005. She is currently an Assistant Professor of the Division of Computer and Information Engineering, College of Information and Communication Engineering at Daegu University. Her research focuses on creating realistic virtual human combining optimization algorithms, physics, and machine learning.



**Prof. Sung-Hee Lee** received the PhD degree in Computer Science from University of California, Los Angeles, USA, in 2008, and the BS and the MS degree in Mechanical Engineering from Seoul National University, Korea, in 1996 and 2000, respectively. He is currently an Associate Professor with the Graduate School of Culture Technology at KAIST. His research interests include motion generation of avatars, virtual characters, and humanoid robots and modeling/simulation of human characters, specifically for VR/AR, telepresence, computer game, and computer animation.

**How to cite this article:** Kang C, Kim M, Lee S-H. Color refinement using deep neural networks for enhancing color recognition in a projector-camera system. *J Soc Inf Display.* 2019;1–11. <https://doi.org/10.1002/jsid.826>