



Eunkyeong Kim D, Jinyong Kim D, Hansoo Lee D and Sungshin Kim *D

Department of Electrical and Electronics Engineering, Pusan National University, Busan 46241, Korea; kimeunkyeong@pusan.ac.kr (E.K.); skes1234@pusan.ac.kr (J.K.); hansoo@pusan.ac.kr (H.L.) * Correspondence: sskim@pusan.ac.kr; Tel.: +82-51-510-2374

Abstract: Artificial intelligence technologies and robot vision systems are core technologies in smart factories. Currently, there is scholarly interest in automatic data feature extraction in smart factories using deep learning networks. However, sufficient training data are required to train these networks. In addition, barely perceptible noise can affect classification accuracy. Therefore, to increase the amount of training data and achieve robustness against noise attacks, a data augmentation method implemented using the adaptive inverse peak signal-to-noise ratio was developed in this study to consider the influence of the color characteristics of the training images. This method was used to automatically determine the optimal perturbation range of the color perturbation method for generating images using weights based on the characteristics of the training images. The experimental results showed that the proposed method could generate new training images from original images, classify noisy images with greater accuracy, and generally improve the classification accuracy. This demonstrates that the proposed method is effective and robust to noise, even when the training data are deficient.

Keywords: data augmentation; data deficiency; adversarial attack; deep learning; color perturbation

1. Introduction

Recent advances in the Internet of Things, big data, cloud computing, and Industry 4.0 are rapidly revolutionizing the manufacturing industry. Industry 4.0, which is related to automation and digitization, is closely linked to smart factories. Smart factories describe the intelligent factory of the future and their core technologies are artificial intelligence (AI) and robot vision systems [1–4].

Owing to the development of AI-related technologies, such as deep learning algorithms, the computational power associated with the growth of graphics processing units, and the collection of big data using large-scale sensor networks, AI has been extensively researched. Furthermore, it has undergone rapid development. Robot vision is the prime perception channel used in manufacturing-related technologies [5]. Accordingly, vision systems have been used in a diverse range of applications in the manufacturing field such as inspection monitoring systems, manipulation, picking and placing, object recognition, and mobile robotics [6–9].

Specifically, combining deep learning, vision systems, and smart factories is a recent trend in the manufacturing sector. Deep learning has been utilized in various manufacturing applications, including autosorting systems, inspection systems, maintenance in mechanical manufacturing, fault classification and diagnosis, and classification systems. For example, to pick and place objects using a manipulator, the objects must be recognized and classified using deep learning and a robot vision system. Object recognition and classification can be achieved using deep learning [10–14].

However, there is a crucial problem in applying deep learning. For example, a network trained to classify objects into categories may be considered. In addition, some noise may



Citation: Kim, E.; Kim, J.; Lee, H.; Kim, S. Adaptive Data Augmentation to Achieve Noise Robustness and Overcome Data Deficiency for Deep Learning. *Appl. Sci.* **2021**, *11*, 5586. https://doi.org /10.3390/app11125586

Academic Editors: Luis Norberto López De Lacalle, Jorge Posada and Federico Divina

Received: 29 April 2021 Accepted: 15 June 2021 Published: 17 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).



be added to the image that falls under a specific category in the trained network. In this case, the noisy image may be misclassified, although the original image without noise is classified correctly. In other words, noise adversely affects the accuracy of image classification. Thus, deep learning models are vulnerable to noise attacks. An autonomous driving system applying deep learning can get into an accident if it is attacked with a noisy image. Such attacks are called adversarial attacks [15,16].

Another problem when applying deep learning algorithms is that it is necessary to prepare sufficient training images to train the deep learning network. Numerous diverse datasets that can be used as training images are available on the Internet. However, in some cases, it is difficult to collect the image data of unusual objects, which are not included in the training dataset of the deep learning model, as opposed to common items. Specifically, in the manufacturing field—for instance, in the electronics industry—many objects are uncommon. There is a limit to capturing and collecting the image data of unusual objects. Therefore, a data augmentation method that can generate images automatically is required to overcome data deficiency. Many researchers are studying data augmentation methods toward this end. Data augmentation improves the generalization capabilities of the deep learning network and the performance of the classification model [17,18].

In this paper, a data augmentation method is proposed to achieve noise robustness and overcome data deficiencies. Our study makes three main contributions to the literature as follows:

- Automatic determination of the optimal perturbation range based on image similarity;
- Weight calculation of color perturbation based on the characteristics of the color distribution of training images;
- Data augmentation based on color perturbation and geometric transformations to compensate data deficiency and noisy images.

The proposed method suggests augmenting training data based on the concept of color jittering. Unlike conventional methods, the proposed method applies to perturb the color information by maintaining the similarity with an original image in the special range we suggest and considering color histogram information of the classified objects. Additionally, the reason why we consider the concept of color jittering is that the noise is related to pixel values. The proposed method automatically determines the optimal perturbation range by calculating the image similarity between the original and noisy images. In addition, we analyzed the color distribution based on the histograms of the training images. The weights were calculated using color perturbation based on the color distribution. We then generated new training images using the color perturbation method based on the image similarity to the calculated weights. Subsequently, the pretrained network was retrained using the image generated by the proposed method. To validate the proposed method, we compared its performance with that of the image classification results of the trained models using the test images.

The remainder of this paper is organized as follows: In Section 2, we review related studies on data augmentation, adversarial attacks, and similarity calculations. In Section 3, we describe the proposed data augmentation method that surmounts data deficiencies and guarantees noise robustness. In Section 4, we present the experimental results of the proposed method. Finally, we present our conclusions in Section 5.

2. Related Works

To implement the deep learning network to classify objects for a manipulator for picking and placing or assembling, sufficient training images must be prepared to train the deep learning network. This is because the deep learning network is trained using images for object classification. Accordingly, deficient training images may affect object classification accuracy. However, it is difficult to collect numerous training images, especially because uncommon objects in the manufacturing field are not included in training datasets. Therefore, it is essential to increase the number of training images through data augmentation [19].

Furthermore, noisy images affect classification accuracy. Even if a noisy image is very similar to the original image without noise, the deep learning network may misclassify the noisy images. Therefore, to avoid misclassification, the network must be trained such as to be noise robust. To compensate for these drawbacks, a data augmentation method that exploits image similarity and color perturbation based on the color characteristics of training images is proposed in the paper. This section deals with data augmentation, adversarial attacks, and image similarity. They are related to the proposed method for overcoming data deficiency and achieving noise robustness.

2.1. Data Augmentation

Sometimes data deficiency occurs because the images of objects that should be classified are insufficient or not included in the providing dataset. Data augmentation is being researched to overcome it. To increase the number of training images, data augmentation methods such as image rotation, flipping, scaling, cropping, color jittering, and shearing can be applied [20–22].

Color space transformation methods, such as color perturbation, edge enhancement, and principal component analysis (PCA), are also used. The color perturbation method can be applied by extracting a single-color channel or adding a random value to the color channel. It can also be performed using a color histogram or PCA of the color channel [19]. Therefore, we augmented the training data to counter data deficiency based on color perturbation because this method is simple and has a relatively short computation time.

In our experiment, the objects for image classification were placed on a conveyor. Then, we considered the rotation of the geometric transformation using (1).

$$Rotation = \begin{pmatrix} \cos\Theta & -\sin\Theta\\ \sin\Theta & \cos\Theta \end{pmatrix}$$
(1)

2.2. Adversarial Attack

Recent studies have revealed that deep learning algorithms are vulnerable to adversarial attacks. In 2013, it was found that a certain barely perceptible perturbation could maximize the error in the classification result. Therefore, adversarial attacks indicate that deep learning models may have inherent weaknesses [15,16].

Examples of adversarial attacks are shown in Figure 1. We only added random noise to images with Gaussian-distributed random numbers, and the original images and noisy images were classified using pretrained networks [23]. The experimental results indicate that, after adding noise to images of hot-dogs, the obtained predictions were incorrect: tarantula, cockroach, and mousetrap. The experimental results prove that noisy images may be labeled incorrectly, although the original image may be labeled correctly.

Thus, noise may affect the prediction results adversely and degrade the performance of the classification model. Especially, in cases of object classification in the manufacturing sector, adversarial attacks would reduce the classification accuracy by causing a malfunction that is related to defect and longer working time. To achieve robustness against adversarial attacks to avoid misclassification, we consider Gaussian-distributed random numbers for a weighted color perturbation method.

4 of 17



Prediction: flamingo







Prediction: flamingo

Prediction: hook

Figure 1. Comparison of classification results for original image and noisy image.

2.3. Image Similarity Calculation

To implement a data augmentation method that overcomes data deficiency and is robust against noise, we considered tuning the values of pixels in an image using Gaussiandistributed random numbers. We applied Gaussian noise because it is similar to actual noise. The criterion for the perturbation range was automatically determined by the proposed method. To determine a reasonable range for color perturbation, the peak signalto-noise ratio (PSNR) was used to calculate the weights of the original and noisy images.

The similarity between two images was calculated using the PSNR (2) as follows [24,25]:

$$PSNR = 10 \log_{10} \left(\frac{MAX_{I}^{2}}{MSE} \right)$$

= 20 log₁₀ $\left(\frac{MAX_{I}}{\sqrt{MSE}} \right)$
= 20 log₁₀ $(MAX_{I}) - 10 \log_{10}(MSE)$ (2)

In (2), MAX_I indicates the maximum fluctuation based on the type of image. As the pixel values of our training images ranged from 0 to 255, the value of MAX_I was 255. Equation (3) represents the MSE calculation.

$$MSE = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} [I(i,j) - K(i,j)]^2.$$
(3)

This equation demonstrates that the MSE is associated with the difference between the two images, I and K. Furthermore, this implies a correlation between the PSNR and the difference between both images. Based on this concept, the PSNR was applied to determine the perturbation range and generate new images using a color perturbation matrix.





3. Proposed Method

A data augmentation method using color perturbation based on the color characteristics of the image and Gaussian noise was proposed to make a deep learning network robust against adversarial attacks. The proposed data augmentation method can overcome data deficiencies and improve the classification accuracy. Use of a suitable data augmentation method is key to image classification.

Therefore, the proposed method focuses on color perturbations. To apply color perturbation, we tuned the pixels of the image using the inverse PSNR, Gaussian-distributed random numbers, and weights calculated by histogram. An overview of the proposed method is presented in Figure 2.



Figure 2. Overview of the proposed method.

As shown in this figure, the proposed method can be divided into six parts as follows: capturing image; decision of perturbation range (Figure 3); weight calculation (Figure 6); image preprocessing; color perturbation; geometric transformation. To establish the training image set, the objects to be classified were captured using the vision sensor. Based on the captured images, the perturbation value was calculated to determine the color perturbation range randomly. Further, the weights were calculated by using the histogram of the captured images. These weights were used when calculating the perturbation matrix. Background elimination was performed in image preprocessing. Next, color perturbation and geometric transformation were carried out to augment the training images.



Figure 3. Process of perturbation range calculation.

First, we collected training images captured using a vision sensor. We loaded the training images to calculate the perturbation range automatically. The proposed method suggests a technique for determining an ideal perturbation range. This process is illustrated in detail in Figure 3.

The perturbation range was determined using images extracted arbitrarily from the training dataset. Random noise was generated and incorporated into an image. The PSNR was calculated using the original image and the noisy image. As mentioned previously, the PSNR indicates the similarity between the two images. We used the characteristics to determine the perturbation range. The PSNR was calculated for all randomly extracted images. Following the PSNR calculation, the perturbation range was determined for inverse PSNR data augmentation.

Next, we generated new training images by the adaptive inverse PSNR data augmentation. This process, which is related to the inverse PSNR, was based on the following calculation (4).

$$10 \log_{10} \left(\frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} [I(i,j) - K(i,j)]^2 \right) = 20 \log_{10} \left(MAX_I^2 \right) - PSNR,$$

$$\sum_{i=1}^{m} \sum_{j=1}^{n} [I(i,j) - K(i,j)]^2 = mn * 10^{2 \log_{10} \left(MAX_I^2 \right) - 0.1PSNR}$$
(4)

This equation, which was obtained from (2), yielded the optimal range for color perturbation. As we considered a data augmentation method based on the value of image similarity, we implemented the PSNR equation inversely.

After loading the training images into categories and determining the perturbation value, the alpha channel was extracted from an image. The alpha channel is a criterion for isolating an object from its background. The region of an object is the region of interest (ROI). ROI is extracted to tune only the pixels of an object in an image.

In addition, we analyzed the color characteristics of the training images in each category based on their histograms to improve the classification accuracy. First, we focused on color, a notable characteristic of an image. Figure 4 shows the RGB channel histograms, which are calculated by accumulating the pixel values of the images in each category. These images were included in the training images for object classification using deep learning. However, it was difficult to find distinct characteristics because black was the dominant color of some objects. Therefore, we rewrote the histogram, excluding the black objects. The rewritten histogram is shown in Figure 5.

Figure 5 shows some objects and their histogram distributions, with respect to the RGB color space. As shown in the figure, the training images were generally red and green, rather than blue. Therefore, we determined that the color perturbation method gave more weight to blue than to red and green. In the proposed method, the abovementioned process was implemented automatically, as shown in Figure 6.



Figure 4. Red, green, and blue (RGB) channel histograms related to images in each category.



Figure 5. Partial training images and histogram of images in its category.

The ideal weights for color perturbation were calculated, as depicted in Figure 6. The weight values indicated that the characteristics of the training images by histogram results were considered during the color perturbation. The summation of the weight values of each channel was three, as shown in Equation (5).

$$W_{red_channel} + W_{green_channel} + W_{blue_channel} = 3$$
⁽⁵⁾

The weights of the $W_{red_channel}$, $W_{green_channel}$, and $W_{blue_channel}$ were initially one. In the case of our training images, the blue channel had a higher weight value than the red and green channels. After histogram analysis, as shown in Figure 6, the weights were changed by the histogram distribution. In our experiment, the $W_{blue_channel}$ had a higher value than the $W_{red_channel}$ and $W_{green_channel}$ because the training images often contain red or green colors.

Next, the perturbation matrix is the perturbation value based on the perturbation range, and the weights were multiplied. A three-dimensional perturbation matrix was generated using Gaussian-distributed random numbers. It was multiplied by the perturbation value and the weights, and combined with a multiplied matrix and an image. The combined image was rotated from 0° to 360°. Rotation was used because the objects were randomly laid on the conveyor. Subsequently, the generated image was saved. This process was repeated until the final range decision value and final image of the final category. When this process was completed, the training dataset was generated through adaptive inverse PSNR data augmentation, and used to retrain the pretrained network for object classification. Transfer learning was deployed to reduce the training time and train the network efficiently [23].

To verify this concept by applying the weights calculated based on the histogram of the objects to be classified, we deployed the color perturbation method by changing the pixels in each channel of the color space. The process is depicted in Figure 7.

The training images were loaded; we isolated the alpha channel from the image to extract the ROI in which an object was contained. The alpha channel was calculated during the preprocessing step through a background elimination process. To eliminate the image background, leaving only the object, we combined the RGB color space and CIE L*a*b* color space. Although the RGB color space is a well-known color space, it is difficult to isolate the object from the background using only the RGB color space. Therefore, we also deployed the CIE L*a*b* color space, which consists of two chromaticity layers and one luminosity layer. One chromaticity layer was related to the red and green axes, and the other was related to the blue and yellow axes. Therefore, we found and eliminated background regions based on background color information using the RGB and CIE L*a*b* color spaces [26].

Next, a one-dimensional perturbation matrix was generated using Gaussian-distributed random numbers. This matrix was multiplied by the range of decision values. The perturbation matrix was combined with the target component, which was extracted by separating the color channel from the image. After combining the perturbation matrix and the target component, the generated image was saved. This process was repeated when the range decision value was the final value. In our experiment, to determine whether the higher weight value of the blue color was effective, all three components—that is, RGB—were individually tested. The experimentally verified results are explained in detail in Section 4.



Figure 6. Partial training images and histogram of images in its category.



Figure 7. Process of color perturbation for components of color space.

4. Experimental Results

The goal of our experiment was to classify the objects into ten categories for grasping the manipulator. We applied a deep learning network for object classification. However, because the objects were not common items, it was difficult to collect sufficient training images to train the deep learning network. Therefore, using the proposed method, training images were generated to improve classification accuracy. There were 32 training images for each category. There were ten categories. The proposed data augmentation method increased the number of training images. In addition, transfer learning was applied for effective learning and to reduce the computational time because it uses a pretrained model that has already learned the features of numerous images. We selected VGGNet with 19 weights layers for transfer learning. VGGNet was already trained by the images of 1000 categories. We cut off and retrained the network using generated images by the methods [23,27].

To verify the proposed method, we compared the classification accuracy of the network trained using the proposed method and the conventional methods. In addition, we trained and compared the deep learning network using the original data and data generated using the proposed method. The test results are presented in the tables. Test images were captured supplementally by placing the objects arbitrarily. They were not included in the training dataset. The classification accuracy is calculated by inputting test images into the trained model.

Table 1 presents the classification accuracy of the network trained using original images. As abovementioned, there were 320 training images, 32 for each category. The experimental results show that the classification accuracy was low when noiseless test images were used because the original images were few. In general, the classification accuracy of noisy images was less than that of noiseless images. Thus, the classification accuracy of noisy images resulted in lower values.

Table 2 presents the classification accuracy of the network trained by the conventional method using original images. Data augmentation was achieved by changing the range of color jittering. In general, color jittering is conducted by adding random values to a color channel. Based on this concept, color jittering #1 #3 add the random values in terms of RGB color space. Color jittering #1 adds positive values to the pixel values of an original image. Color jittering #2 and #3 add positive or negative values with different ranges [19]. When the conventional method was deployed, the number of training images increased by nine times, from 320 to 2880. The experimental results showed that the classification accuracy of the noisy images improved slightly. However, the classification accuracy tended to be low.

To improve the classification accuracy, we eliminated the background and cropped an image by considering the center of the object. Table 3 presents the classification accuracy of the network trained using preprocessed images. The classification accuracy improved slightly over that of the original images and noisy images in Table 1.

Next, we implemented a data augmentation method based on the inverse PSNR, with respect to the RGB color space and rotation. The number of training images increased by 216 times. Table 4 presents the classification accuracy of the network trained using the inverse PSNR, with respect to the RGB color spac, and rotation of the images. As shown in the table, the performance of the trained network improved significantly and the classification accuracy was higher than that of the network trained using the original images and the preprocessed images. Specifically, the classification performance on the noisy images increased significantly. Furthermore, the classification performance on the noiseless images increased.

Table 1. Classification accuracy of the network trained using original images.

Number of	Mathad	Number of	Classification Accuracy			
Experiments	Method	Training Images	Noiseless	Noisy	Total	
1	Training by	320	48.0%	25.0%	36.5%	
2			43.0%	29.0%	36.0%	
3	original images		44.0%	25.0%	34.5%	

Table 2. Classification accuracy of the network trained using the conventional method.

Number of	Mathad	Number of	Classification Accuracy			
Experiments	Method	Training Images	Noiseless	Noisy	Total	
1	Training using images		41.0%	35.0%	38.0%	
2	obtained through	2880	47.0%	37.0%	42.0%	
3	color jittering #1		41.0%	32.0%	36.5%	
1	Training using images		41.0%	37.0%	39.0%	
2	obtained through	2880	43.0%	39.0%	41.0%	
3	color jittering #2		42.0%	39.0%	40.5%	
1	Training using images		29.0%	36.0%	32.5%	
2	obtained through	2880	31.0%	32.0%	31.5%	
3	color jittering #3		29.0%	32.0%	30.5%	

Table 3. Classification accuracy of the network trained using preprocessed images.

Number of	Mathad	Number of	Classification Accuracy		
Experiments	Method	Training Images	Noiseless	Noisy	Total
1	Training using		63.0%	31.0%	47.0%
2	preprocessed	320	52.0%	26.0%	39.0%
3	Images		60.0%	44.0%	52.0%

Table 4. Classification accuracy of the network trained using images obtained through inverse PSNR with RGB color space and rotation.

Number of	Mathad	Number of	Classification Accuracy		
Experiments	Method	Training Images	Noiseless	Noisy	Total
1	Training using		90.0%	60.0%	75.0%
2	inverse PSNR + ROT	69,120	93.0%	62.0%	77.5%
3	images		90.0%	66.0%	78.0%

As mentioned in Section 3, because the objects for image classification generally had more red and green colors than blue, we determined that in the color perturbation method,

a higher weight would most likely result in blue. To provide reasonable evidence, we trained the network using images obtained by applying the inverse PSNR to three channels individually. The classification results are listed in Table 5.

Comparing the above experimental results, the classification results obtained by applying the inverse PSNR to the B channel were better than those obtained from the images with the inverse PSNR applied to the R and G channels. Therefore, we tuned the weights to increase the weight of the B channel and decrease the weights of the R and G channels. We tested two cases in which the weights were tuned. The results are presented in Table 6.

Table 5. Classification accuracy of the network trained using images obtained by applying the inverse PSNR to three channels individually.

Number of	Mathad	Number of		Classification Accuracy		
Experiments	Methou	Training Images	Noiseless	Noisy	Total	Average
1	Training using		68.0%	30.0%	49.0%	
2	D DENID ima and	1920	57.0%	34.0%	45.5%	47.5%
3	K PSINK Images		56.0%	40.0%	48.0%	
4	Training using		55.0%	26.0%	40.5%	
5	C DCNIB im a see	1920	60.0%	42.0%	51.0%	46.8%
6	G PSINK Images		61.0%	37.0%	49.0%	
7	Training using		66.0%	46.0%	56.0%	
8		1920	58.0%	38.0%	48.0%	51.2%
9	D F SINK IMages		67.0%	32.0%	49.5%	

Table 6. Classification accuracy of the network trained using images obtained by the adaptive inverse PSNR and RGB color space and rotation.

Number of	Mathad	Number of	Classification Accuracy		
Experiments	wiethod	Training Images	Noiseless	Noisy	Total
1	Training using		92.0%	76.0%	84.0%
2	adaptive inverse	69,120	91.0%	73.0%	82.0%
3	PSNR + ROT images		93.0%	72.0%	82.5%

As shown in the table, the classification accuracy was higher than that of the network trained by other methods. Specifically, the classification performance on the noisy images was higher than 70%. A comparison of tables revealed that the proposed method had the best performance. It is evident that applying weights based on the color tendency of the training images was effective. To validate the proposed method, the network was trained with an equal number of training images of conventional methods. For this, we randomly selected 2880 training images from the images generated by the proposed method. The model is trained three times using selected training images. In case of noiseless images, the average of the classification accuracy of the model trained by the 2880 training images from the images generated by the proposed method was 87.3%. The classification performance on the noisy images was 67.0%. This additional experiment shows that the proposed method could augment the training images effectively.

Figure 8 represents the comparison of the classification accuracy, with respect to average value and best performance. This figure shows that the more the proposed algorithms are combined, the higher the classification accuracy. Furthermore, the proposed method outperformed the conventional method. Therefore, the proposed method could improve the classification accuracy of both noisy and noiseless test images.

We compared the confusion matrices of the experimental results for the training images in Figures 9 and 10. The left and right columns represent the classification results for the original and noisy test images, respectively. The experimental results for the original test images showed that the classification accuracy improved significantly after application of the adaptive inverse PSNR. As depicted in the left column of Figures 9 and 10, the classification accuracy was higher than 90%. The experimental results of the noisy test images showed that the classification accuracy also improved after application of the adaptive inverse PSNR. Specifically, the classification accuracy of the model trained using the original images was 25%. However, the proposed method showed a significant improvement, from 25% to 76%. Therefore, it was demonstrated that the proposed method could effectively compensate for data deficiencies and was robust to noisy images.



Figure 8. Comparison of classification accuracy.



Figure 9. Comparison of confusion matrix #1.



Figure 10. Comparison of confusion matrix #2.

Table 7 presents the precision, recall, specificity, and F1-score of the conventional method and the proposed method. Precision, recall, specificity, and F1-score were calculated using (6).

$$Precision = \frac{True Positive}{True Positive + False Positive}$$

$$Recall = \frac{True Positive}{True Positive + False Negative}$$

$$Specificity = \frac{True Negative}{True Negative + False Positive}$$

$$F1-score = 2\frac{Recall \times Precision}{Recall + Precision}$$
(6)

As shown in the table, in case of the proposed method, precision and recall values have a value closer to one than the conventional method. Therefore, the proposed method could result higher classification performance than the conventional method.

	Noiseless			Noisy				
	Precision	Recall	Specificity	F1-Score	Precision	Recall	Specificity	F1-Score
	0.75	0.30	0.99	0.43	1.00	0.20	1.00	0.33
	0.78	0.70	0.98	0.74	0.24	1.00	0.66	0.39
	1.00	0.10	1.00	0.18	1.00	0.10	1.00	0.18
	0.47	0.80	0.90	0.59	0.50	0.20	0.98	0.29
Conventional Method	0.30	1.00	0.74	0.47	0.47	0.80	0.90	0.59
Conventional Method	1.00	0.50	1.00	0.67	0.31	0.80	0.80	0.44
	-	0.00	1.00	-	-	0.00	1.00	-
	0.35	0.90	0.81	0.50	0.80	0.40	0.99	0.53
	1.00	0.10	1.00	0.18	0.33	0.10	0.98	0.15
	0.75	0.30	0.99	0.43	1.00	0.10	1.00	0.18
	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	1.00	1.00	1.00	1.00	1.00	0.70	1.00	0.82
	1.00	1.00	1.00	1.00	1.00	0.90	1.00	0.95
	0.67	1.00	0.94	0.80	0.36	1.00	0.80	0.53
Proposed Method	1.00	0.70	1.00	0.82	1.00	0.10	1.00	0.18
i toposed Method	0.88	0.70	0.99	0.78	1.00	0.50	1.00	0.67
	0.90	0.90	0.99	0.90	0.67	0.80	0.96	0.73
	1.00	0.90	1.00	0.95	1.00	0.70	1.00	0.82
	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	0.91	1.00	0.99	0.95	0.82	0.90	0.98	0.86

Table 7. Precision, recall, specificity, and F1-score of the conventional method and the proposed method.

The additional experiment was conducted about the illumination change of the images when captured. One of common applications of the smart factory is AOI inspection. It is important to maintain a stable lighting source for AOI inspection. As per the experimental results, in case of the trained model using original images, the classification accuracy is 45.89%. The classification accuracy of the trained model by the conventional method is 43.67%, and that of the trained model by the proposed method is 82.89%. The experimental results show that the classification performance of the proposed method is higher than other trained models; namely, it represents that the proposed method could be robust against the illumination change.

5. Conclusions

We developed a data augmentation method to overcome training data deficiency, achieve robustness against noise that causes classification errors, and improve the image classification accuracy. To examine the effect of the characteristics of the training images, we employed the adaptive inverse PSNR, which automatically determined the ideal perturbation range and weights for the color perturbation method, which were used to generate new training images. Weights are useful for classifying noisy images because the weights are greater than the color distribution.

Experiments were performed to compare the classification models trained by the original data, the data generated by the conventional method, and the data generated by the proposed method using the test images. The experimental results showed that the images generated based on the adaptive inverse PSNR using the rotation method were more effective than the training data generated by the proposed method. In addition, the experimental results showed that the proposed method could generate new images as training data effectively, was robust against noisy images, and improved the image classification accuracy. In future, we plan to develop a data augmentation method that considers the characteristics of each image to overcome data deficiencies in the manufacturing domain. Further, we will apply the proposed method in other AI models such as object detection or semantic segmentation and so on.

Author Contributions: Conceptualization, E.K. and H.L.; methodology, E.K.; software, E.K.; validation, E.K., H.L. and J.K.; formal analysis, E.K.; investigation, E.K.; resources, E.K. and J.K.; data curation, E.K. and J.K.; writing—original draft preparation, E.K.; writing—review and editing, S.K.; visualization, E.K. and H.L.; supervision, S.K.; project administration, S.K.; funding acquisition, S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Technology Innovation Program (10073147, Development of Robot Manipulation Technology by Using Artificial Intelligence) funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to undergoing project.

Acknowledgments: This research was supported by BK21PLUS, Creative Human Resource Education and Research Programs for ICT Convergence in the 4th Industrial Revolution, and was supported by the Technology Innovation Program (10073147, Development of Robot Manipulation Technology by Using Artificial Intelligence) funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea) and was supported by Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0008473, HRD Program for Industrial Innovation).

Conflicts of Interest: The authors declare no conflict of interest.

References

- Tao, F.; Qi, Q. New IT driven service-oriented smart manufacturing: Framework and characteristics. *IEEE Trans. Syst. Man Cybern.* Syst. 2017, 49, 81–91. [CrossRef]
- Ghobakhloo, M. The future of manufacturing industry: A strategic roadmap toward Industry 4.0. J. Manuf. Technol. Manag. 2018, 29, 910–936. [CrossRef]
- 3. Hozdić, E. Smart factory for industry 4.0: A review. Int. J. Mod. Manuf. Technol. 2015, 7, 28–35.
- 4. Shi, Z.; Xie, Y.; Xue, W.; Chen, Y.; Fu, L.; Xu, X. Smart factory in Industry 4.0. Syst. Res. Behav. Sci. 2020, 37, 607–617. [CrossRef]
- 5. Frese, U.; Hirschmüller, H. Special issue on robot vision: What is robot vision? J. Real Time Image Process. 2015, 10, 597–598. [CrossRef]
- 6. Edinbarough, I.; Balderas, R.; Bose, S. A vision and robot based on-line inspection monitoring system for electronic manufacturing. *Comput. Ind.* **2005**, *56*, 986–996. [CrossRef]
- Nair, A.; Chen, D.; Agrawal, P.; Isola, P.; Abbeel, P.; Malik, J.; Levine, S. Combining self-supervised learning and imitation for vision-based rope manipulation. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 2146–2153.[CrossRef]
- Pedersen, M.R.; Nalpantidis, L.; Andersen, R.S.; Schou, C.; Bøgh, S.; Krüger, V.; Madsen, O. Robot skills for manufacturing: From concept to industrial deployment. *Robot. Comput. Integr. Manuf.* 2016, 37, 282–291.
- 9. Zakhama, A.; Charrabi, L.; Jelassi, K. Intelligent Selective Compliance Articulated Robot Arm robot with object recognition in a multi-agent manufacturing system. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 1–15. [CrossRef]
- Wang, T.; Yao, Y.; Chen, Y.; Zhang, M.; Tao, F.; Snoussi, H. Auto-sorting system toward smart factory based on deep learning for image segmentation. *IEEE Sens. J.* 2018, 18, 8493–8501. [CrossRef]
- 11. Li, L.; Ota, K.; Dong, M. Deep learning for smart industry: Efficient manufacture inspection system with fog computing. *IEEE Sens. J.* **2018**, *14*, 4665–4673.
- 12. Pech, M.; Vrchota, J.; Bednář, J. Predictive Maintenance and Intelligent Sensors in Smart Factory. Sensors 2021, 21, 1470. [CrossRef]
- Lee, K.B. Cheon, S.; Kim, C.O. A convolutional neural network for fault classification and diagnosis in semiconductor manufacturing processes. *IEEE Trans. Semicond. Manuf.* 2017, 30, 135–142. [CrossRef]
- 14. Kwon, O.; Kim, H.G.; Ham, M.J.; Kim, W.; Kim, G.; Cho, J.; Kim, N.I.; Kim, K. A deep neural network for classification of melt-pool images in metal additive manufacturing. *J. Intell. Manuf.* **2020**, *31*, 375–386. [CrossRef]
- Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; Vladu, A. Towards deep learning models resistant to adversarial attacks. *arXiv* 2017, arXiv:1706.06083. [CrossRef]
- 16. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.
- 17. Acción, Á.; Argüello, F.; Heras, D.B. Dual-Window Superpixel Data Augmentation for Hyperspectral Image Classification. *Appl. Sci.* **2020**, *10*, 8833.
- 18. Baldominos, A.; Saez, Y.; Isasi, P. A survey of handwritten character recognition with mnist and emnist. *Appl. Sci.* **2019**, *9*, 3169. [CrossRef]
- 19. Shorten, C. Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. J. Big Data 2019, 6, 60. [CrossRef]

- 20. Qi, H.; Liang, Y.; Ding, Q.; Zou, J. Automatic Identification of Peanut-Leaf Diseases Based on Stack Ensemble. *Appl. Sci.* 2021, 11, 1950. [CrossRef]
- Urbonas, A.; Raudonis, V.; Maskeliūnas, R.; Damaševičius, R. Automated identification of wood veneer surface defects using faster region-based convolutional neural network with data augmentation and transfer learning. *Appl. Sci.* 2019, *9*, 4898.
 [CrossRef]
- 22. Hussain, Z.; Gimenez, F.; Yi, D.; Rubin, D. Differential data augmentation techniques for medical imaging classification tasks. In Proceedings of the AMIA Annual Symposium Proceedings, Washington, DC, USA, 6–8 November 2017; pp. 979–984. [CrossRef]
- 23. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv 2014, arXiv:1409.1556.
- 24. Hore, A.; Ziou, D. Image quality metrics: PSNR vs. SSIM. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369.
- 25. Huynh-Thu, Q.; Ghanbari, M. Scope of validity of PSNR in image/video quality assessment. Electron. Lett. 2008, 44, 800-801.
- Kim, E.K.; Lee, H.; Kim, J.Y.; Kim, S. Data Augmentation Method by Applying Color Perturbation of Inverse PSNR and Geometric Transformations for Object Recognition Based on Deep Learning. *Appl. Sci.* 2020, 10, 3755. [CrossRef]
- Hussain, M.; Bird, J.J.; Faria, D.R. A study on cnn transfer learning for image classification. In Proceedings of the UK Workshop on Computational Intelligence, Nottingham, UK, 5–7 September 2018; pp. 191–202. [CrossRef]