A Study on Applying the SRCNN Model and Bicubic Interpolation to Enhance Low−Resolution Weeds Images for Weeds Classification

Vo Hoang Trong, Yu Gwang−hyun, Dang Thanh Vu,
Lee Ju−hwan, Nguyen Huy Toan, Kim Jin−young

Abstract

In the image object classification problem, low−resolution images may have a negative impact on the classification result, especially when the classification method, such as a convolutional neural network (CNN) model, is trained on a high−resolution (HR) image dataset. In this paper, we analyze the behavior of applying a classical super−resolution (SR) method such as bicubic interpolation, and a deep CNN model such as SRCNN to enhance low−resolution (LR) weeds images used for classification. Using an HR dataset, we first train a CNN model for weeds image classification with a default input size of 128×128. Then, given an LR weeds image, we rescale to default input size by applying the bicubic interpolation or the SRCNN model. We analyze these two approaches on the Chonnam National University (CNU) weeds dataset and find that SRCNN is suitable for the image size is smaller than 80×80, while bicubic interpolation is convenient for a larger image.

Keywords : super−resolution | weeds classification | convolutional neural network | deep learning

I. INTRODUCTION

CNN provides significant results on many problems, such as object classification[1], detection[2], and SR[3]. Considering the classification case, many models have dramatically advanced the state−of−the−art and applied to many computer vision problems. However, the success of CNN depends on the architecture and the dataset. If the dataset contains many high−quality images, there is a high possibility that LR images interrupt model performance. Interpolation is a traditional classical approach to increase the quality of the image[4,5]. It is applied to many image processing problems such as satellite image[6], biometrics[7], and MRI image enhancement[8]. With the strong development of deep neural networks, many models have been released to improve images' resolution, such as the generative adversarial network (GAN)[9], deep residual networks[10], and deep convolutional networks[11].

In this paper, we analyze the case of LR weeds classification problems by first using the VGG model that already trained in the HR weeds dataset. Then, an LR image test dataset is used to evaluate CNN performance. We reconstruct HR from LR images by either using a bicubic interpolation or applying the SRCNN model. We use the VGG model because of its simple architecture and uncomplicated to analyze features' behaviors through layers. On the SR problem, many models([9−11]) have a complex architecture and a vast number of parameters to improve the perceptual quality of the LR image for a human observer. However, to enhance an LR image for classification, only specific features, such as texture patterns or edges, should be strengthened [12], leads to a slight CNN model, such as SRCNN is used for the SR purpose. Experiments show that there exists a threshold where one approach is more effective than the other. For LR square images, SRCNN shows better performance on classification than bicubic interpolation when the size of the image is smaller than 80. In contrast, bicubic interpolation appears to be practical in the opposite case.

* This work was carried out with the support of the “Cooperative Research Program for Agriculture Science and Technology Development (Project No. PJ01385501)”, Rural Development Administration, Republic of Korea.
* Member, Department of Electronics and Computer Engineering, Chonnam National University, Gwangju, Republic of Korea.

Manuscript : 2020. 07. 23
Revised : 2020. 09. 24
Confirmation of Publication: 2020. 10. 12
Corresponding Author : Kim Jin−young,
e−mail : beyondi@chonnam.ac.kr

The organization in this paper is as follow: First, we introduce experiments that used interpolation and deep neural network for image enhancement in Section 2. Then, the overall information about bicubic interpolation and SRCNN are shown in Section 3. Section 4 describes our experiment, including descriptions of the CNU weeds dataset, configurations to train VGG model for classification and SRCNN model for the single image SR problems, and evaluations of 2 image enhancement approach. Finally, the conclusion of our work is shown in Section 5.

II. RELATED WORK

Many approaches apply interpolation techniques based on sampling theory to SR problems. Anbarjafari and Demirel[4] used the discrete wavelet transform to obtained high-frequency subband images. Then, they applied interpolation on these images and combined them with the input image to get an HR image via the inverse discrete wavelet transform. Hwang and Lee[5] noticed that pure interpolation schemes led to blurring images, so they used local feature information to calculate the inverse gradient and applied it to conventional bilinear and bicubic interpolation. Interpolation techniques are also adopted in real-world problems. Demirel and Anbarjafari[6] used their proposed method in [4] to enhance the resolution of satellite images. Reddy *et al.* [7] applied cosine transforms and bilateral interpolation to create a higher resolution from multi-frame LR visual images.

In recent years, many methods have been introduced to solve the single image SR problem using deep neural networks. Ledig *et al.*[9] proposed SRGAN using GAN to recover low-level features after upscaling the image. Their model is capable of deriving high quality up to 4 times upscaling factors and improving low-features texture from an extremely LR image. Lim *et al.*[10] proposed a compact deep neural network model using the residual connection technique to enhance the image in multi-scale. Dong *et al.*[11] proposed an end-to-end CNN model, used the Y channel of YCbCr color space to map LR and HR images. Their model contains a small number of parameters and achieved an admirable performance than previous methods.

III. METHODOLOGY

1. Bicubic interpolation

Bicubic interpolation is a method to estimate an unknown value from known values. Given a square image I with four sides have length a, suppose we would like to upsample I to the image J with side length is $b > a$, bicubic interpolation use a third-degree polynomial to evaluate missing values in J. Consider the case of 1 dimension, the third-degree polynomial takes the form

$$f(x) = ax^3 + bx^2 + cx + d \tag{1}$$

and derivative function in term of x is

$$f'(x) = 3ax^2 + 2bx + c \tag{2}$$

To find coefficients a, b, c and d requires 4 known values p_0, p_1, p_2, p_3 at 4 different coordinates, such as $x = -1, x = 0, x = 1$, and $x = 2$. Then, we have

$$f(0) = d = p_1 \tag{3}$$

$$f(1) = (a + b + c + d) = p_2 \tag{4}$$

and using $x = -1, x = 2$ to approximate derivative at $f'(0)$ and $f'(1)$

$$f'(0) = \frac{p_2 - p_0}{2} = c \tag{5}$$

$$f'(1) = \frac{p_3 - p_1}{2} = 3a + 2b + c \tag{6}$$

Finally, a, b, c and d are calculated as:

$$\begin{aligned} a &= 2f(0) - 2f(1) + f'(0) + f'(1) \\ &= -\frac{1}{2}p_0 + \frac{3}{2}p_1 \\ &\quad -\frac{3}{2}p_2 + \frac{1}{2}p_3 \end{aligned} \tag{7}$$

$$\begin{aligned} b &= -3f(0) + 3f(1) - 2f'(0) - f'(1) \\ &= p_0 - \frac{5}{2}p_1 + 2p_2 \\ &\quad -\frac{1}{2}p_3 \end{aligned} \tag{8}$$

$$c = f'(0) = -\frac{1}{2}p_0 + \frac{1}{2}p_2 \tag{9}$$

$$d = f(0) = p_1 \tag{10}$$

with known coefficients a, b, c and d, we can use formula (1) to estimate $x \in [0,1]$.

We can apply bicubic interpolation in the case of 2 dimensions by using 5 third-degree polynomials f_1, f_2, f_3, f_4 and f_5. To estimate an unknow value $J(x, y)$, first, we scale pixels from image I to image

J. Then, 4 polynomials are applied on 4 known pixels that lay on 4 lines $y-1, y, y+1, y+2$ (parallel to the $x-$ axis and near to $J(x,y)$). Finally, on each polynomial, a point that vertically direction to $J(x,y)$ is used to estimate coefficients of f_5, and f_5 is being used to evaluate $J(x,y)$. Beside using those 5 polynomials, we can rewrite it as a multivariable polynomial. This polynomial still requires 16 known values to calculate coefficients a_{ij}, which can be taken from I.

$$f(x,y) = \sum_{i=0}^{3}\sum_{j=0}^{3} a_{ij}x^i y^j \qquad (11)$$

2. VGG

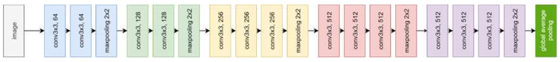


Fig. 1. VGG architecture.

In this study, we apply the VGG net[13] as a CNN classification model. As shown in Fig. 1, this model has 16 layers and the last layer is a 256−dimensional global average pooling (GAP). As explained in [13], a stack of two and three 3×3 convolutional filters have a similar effect with a 5×5 and 7×7 convolutional filter, respectively. Although increasing the network depth leads to better performance, deeper networks often require more training parameters. In VGG architecture, convolutional layers use a small filter size of 3×3, which is efficient in capturing low−level features in earlier layers. Max−pooling layer makes the convolutional layer's output more robust to local features[14], and deeper layers capture high−level features. After each convolutional layer, VGG uses a rectified linear unit (ReLU) as the activation function to increase the non−linear of feature maps. Global average pooling (GAP) is adopted at the end of the model since this layer is robust to the spatial translation of the previous layer. This layer calculates the channel−wise average of the last max−pooling layer to obtain a 512−dimensional feature vector.

3. SRCNN

The single image SR problem aims to enhance the resolution of an LR image. Dong *et al.*[11] developed a technique using CNN, called SRCNN, to solve this problem. Fig. 2 shows the overall scheme to enhance an RGB color image by using the SRCNN model. We first use bicubic interpolation to resize an RGB input image, then convert this image into YCbCr color space. SRCNN uses the luminance channel (Y channel) as the input data since this channel has brightness information of the image that sensitive to the human eyes. SRCNN is a shallow network with three convolutional layers. Unlike VGG, the first layer of SRCN uses 64 9×9 convolutional filters to capture large receptive fields, produce 64 feature maps. These feature maps are passed to the nonlinear activation function (ReLU). Then, 32 $1\times1\times64$ convolutional filters are applied to learn the complex and interaction of cross channel information[15]. Also, this layer (through ReLU) learns a nonlinear dimension mapping of the previous layer to another dimension. Later, a $5\times5\times32$ convolutional filter is applied as a reconstruction layer to produce the HR image.

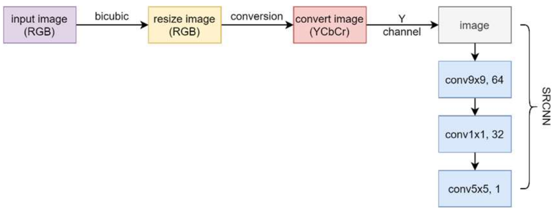


Fig. 2. Overall scheme for single image SR. Input RGB color image is resized by bicubic interpolation, then conversed to YCbCr color space. SRCNN model receive Y channel as the input data and return a single feature map.

IV. EXPERIMENTS

1. Dataset


Fig. 3. Images in the CNU weeds dataset.

We used Chonnam National University (CNU) weeds dataset for experiments. This dataset has 21 species with 208477 square HR images in total[16]. Fig. 4 shows the normalized histogram of image resolutions in this dataset, in which nearly 50% of images that have side length is higher than 711. As Table 1. Contribution of species on the CNU weeds dataset.

Species	Number of images
Amaranthus patulus Bertol.	4138
Amaranthus blitum subsp. oleraceus (L.) Costea	6608
Amaranthus powellii S.Watson	5451
Amaranthus viridis L.	2300
Amaranthus retroflexus L.	6424
Senecio vulgaris L.	23388
Ambrosia trifida L.	23622
Bidens bipinnata L.	804
Ambrosia artemisiifolia L.	1218
Erigeron canadensis L.	24279
Bidens frondosa L.	20679
Sonchus oleraceus (L.) L.	15383
Xanthium orientale L.	13036
Conyza sumatrensis (S.F.Blake) Pruski & G.Sancho	8834
Sonchus asper (L.) Hill	17269
Galinsoga quadriradiata Ruiz & Pav.	24396
Chenopodium ficifolium Smith	1692
Chenopodium album L.	2804
Ipomoea coccinea L.	2300
Veronica polita Fr.	1969
Veronica persica Poir.	1883

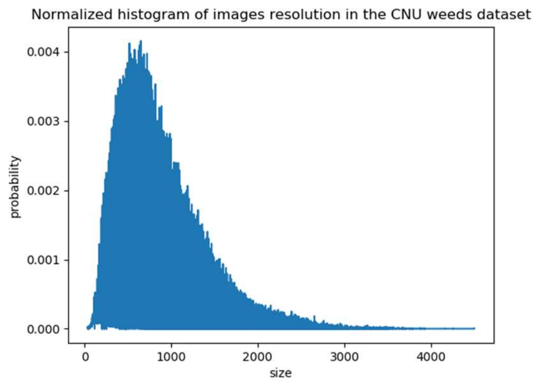


Fig. 4. The probability distribution of images resolution in the CNU weeds dataset.

shown in Table 1, this dataset is imbalanced, where the species has the largest number of images is *Galinsoga quadriradiata* Ruiz & Pav. (around 24300 images). In contrast, species *Bidens bipinnata* L. has the smallest number of images (about 800 images). For classification and SR experiments, we used 60% of images in each species as the training set, 20% was the validate to select the optimal DNN parameters in the classification and SR tasks, and the remain 20% was the test set.

2. Weeds Image Classification

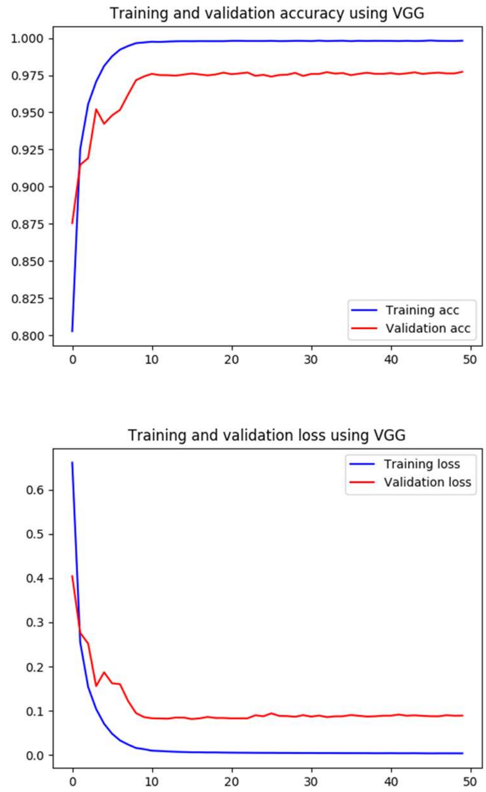


Fig. 5. Training and validation performance on 50 epochs. (up). Accuracy. (down). Loss

We applied transfer learning to train the VGG net used in our previous work in [17] (Fig. 1) for the weeds classification problem. We used parameters trained on the ImageNet dataset as the initial parameters for VGG because this dataset contains weeds images. We furthermore added a GAP layer, followed by a 256-dimensional FC layer, batch normalization layer, ReLU as the activation function, a 21-dimensional FC layer (equal to the number of species), and a softmax layer as the last layer of the VGG net. We used the output of a softmax layer to calculate loss via the categorical cross-entropy loss function. The input of the VGG net was a 128×128 RGB color image, rescaled to [0,1] range. When applying transfer learning on Keras, we froze the first five layers of the VGG net and trained with a batch size of 128 without using any data augmentation techniques. The learning rate was

0.001, and we optimize parameters by using stochastic gradient descent for 50 epochs.

Fig. 5 shows training and validation performances on 50 epochs. VGG started to converge from 10th epoch, and validation accuracy varied around 0.975. After 50 epochs, accuracy, precision, recall, and F1 score on the test set shown in Table 2 ranged between 0.963 and 0.977.

Table 2. Performance on the test set using VGG net

Accuracy	Precision	Recall	F1 score
0.97643	0.96479	0.96396	0.96437

3. Weeds Image Super-Resolution

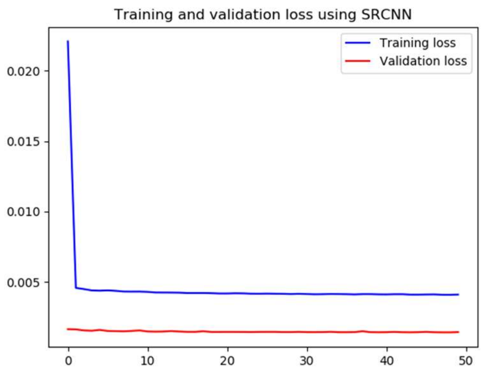


Fig. 6. Loss performance using SRCNN for single image SR.

We used the SRCNN model for training the single weeds image SR. We trained this model from scratch and initialized training parameters using the Xavier uniform initializer[17]. On each batch, images on the training set were first downsampled randomly between 32 and 120 on the side length, then rescaled to 128×128 by using bicubic interpolation and transformed to YCbCr color space. We normalized the Y channel to $[0,1]$ range, and we used it as the input data to the SRCNN model. We trained this model with the batch size of 32, used mean square error (MSE) as the loss function, and used RMSprop to optimize training parameters. The learning rates varied cyclically between 10^{-6} and 5.5×10^{-5} by following triangle policy[18]. After completed training on each epoch, we evaluated the model by averaging five times the evaluation results on the validation set. Each time, we downsampled all images on this set to 32, 54, 76, 98, and 120. After training for 100 epochs, we kept parameters that produced the lowest MSE (which was 0.00143) on the validation set. When applying these parameters to the test set, the MSE was 0.00078, evaluated with a similar policy on the validation set. Fig. 6 shows training and validation loss on 50 epochs, and Fig. 7 shows image examples using SRCNN to enhance LR images that had a side length of 32, 76, and 120. These images are produced by taking the feature map output of the SRCNN model ("*conv5x5*" layer) and multiplying to 255, then transforming all feature map values into $[0,255]$ by using equation (12). Finally, we replaced images' Y channel to this transformed feature map and converted back to RGB color space to produce the SR image.

$$f(x) = \begin{cases} 0; \text{if } x \leq 0 \\ 255; \text{if } x \geq 255 \\ x; \text{otherwise} \end{cases} \tag{12}$$

4. SR for Classification

In this step, we experimented on applying SR techniques to classification purposes. We downsampled images on the validation and test set to every side length from 32 to 128, enhanced images using bicubic interpolation and SRCNN, then fed to the VGG model for classification. Fig. 8 shows the average classification loss and accuracy on each side length. It indicates that there exists a threshold in which a method performs better than the other. On the validation set, if images had side lengths smaller than 80, using SRCNN to enhance images before classification showed better performances than using bicubic interpolation because SRCNN could recreate important features information. On the other hand, bicubic interpolation showed slightly better performance than SRCNN when the side length was larger or equal to 80. This situation came from the fact that SRCNN, using convolutional operators, enhanced features on the image even those features laid on the background, which were not ideal for classification. Furthermore, SRCNN could not enhance low-level features on the weeds surface, which might interrupt classification results. Fig. 9 demonstrates images for this behavior. In the case of images with an input size of 56×56, SRCNN could recover many essential features on the weeds surface with higher quality than using bicubic interpolation. However, with images that had size of 104×104, SRCNN could not increase the quality of low-level

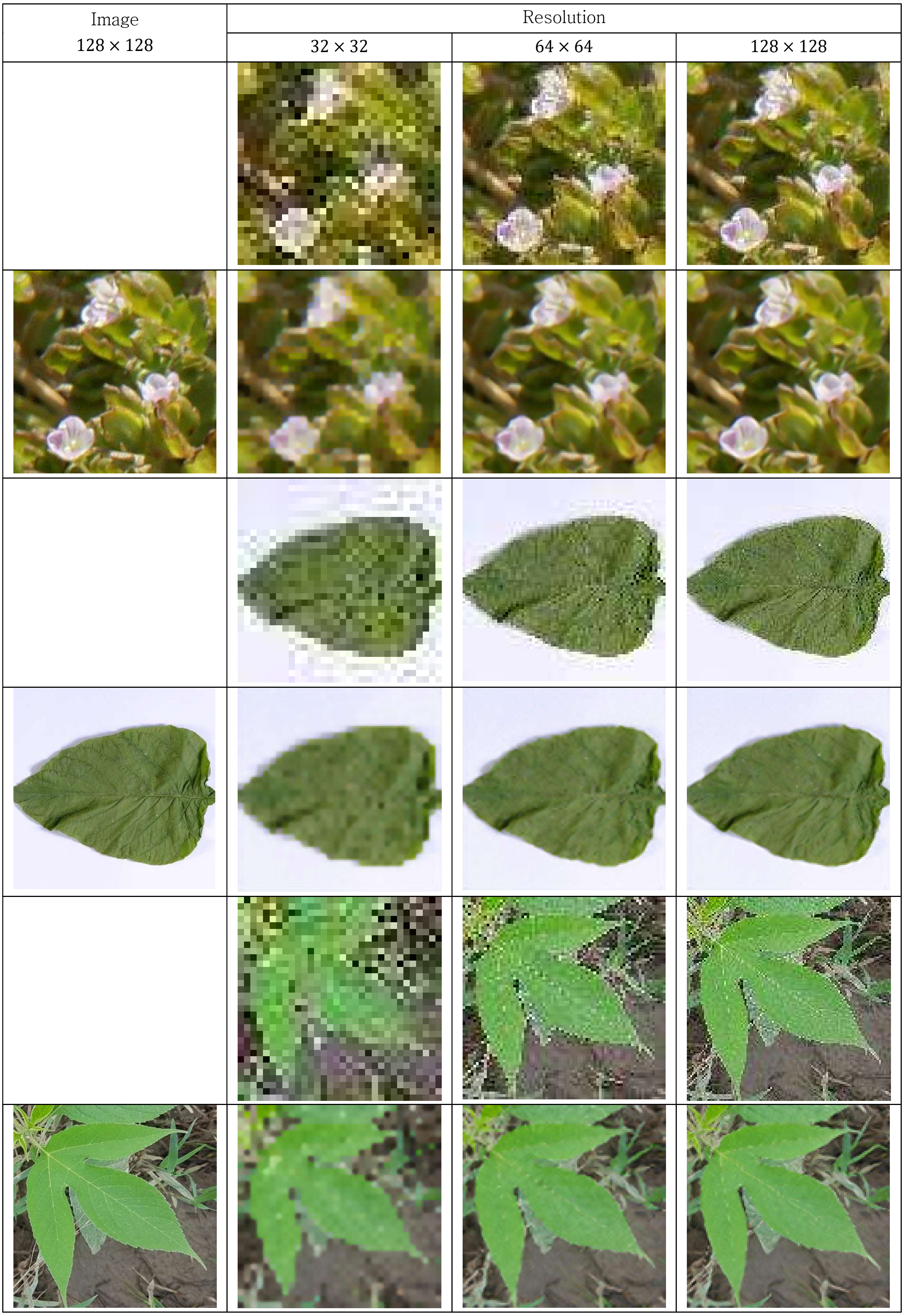


Fig. 7. Experiment results using SRCNN for single image SR. Left column. Original input image (128 × 128). Right columns. SR result after downsampling input image.

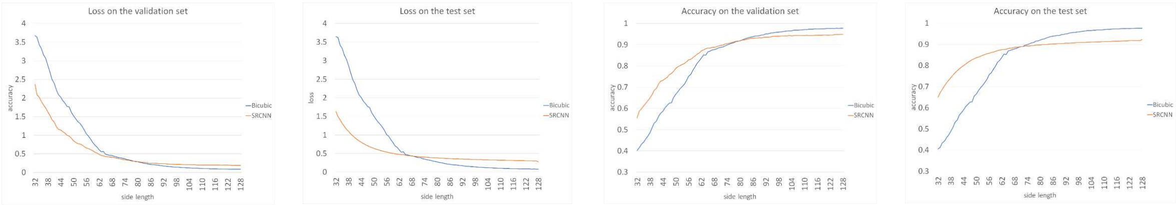


Fig. 8. Performance of using bicubic interpolation and SRCNN to enhance image for classification.

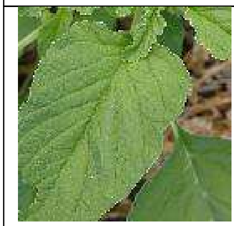
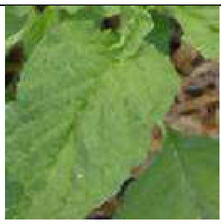
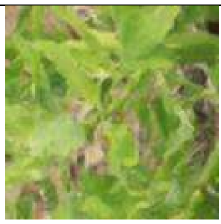
Original image	SRCNN		Bicubic interpolation	
128×128	56×56	104×104	56×56	104×104
				
				
				

Fig. 9. Example of using SRCNN and bicubic interpolation to enhance $56 \times 56, 104 \times 104$ input test images to 128×128. In this example, those images have size 56×56 get correct classification with SRCNN and incorrect classification with bicubic interpolation, while images have size 104×104 get reverse results.

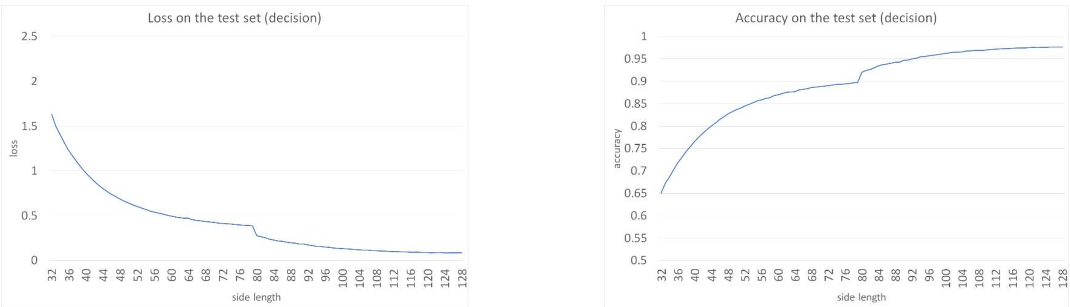


Fig. 10. Performance of classification on the test set by decision: Using SRCNN to enhance images have side length smaller than 81, and bicubic interpolation otherwise.

features laid on the surface, while bicubic interpolation still maintains significant features for classification. After experimenting on 2 SR approaches, we decided to apply SRCNN to enhance images that had the side length smaller than 80 and apply bicubic interpolation on the other case.

Fig. 10 shows performances on the test set by using this decision, in which the CNN model could classify weeds in LR images with slightly high accuracy and low on loss.

V. CONCLUSION

In this paper, we examined applying the SRCNN model and bicubic interpolation to enhance LR weeds images for weeds classification. We used the VGG model already trained in the HR CNU weeds dataset to analyze those techniques. Experimented with SRCNN and bicubic interpolation on the CNU weeds dataset showed that SRCNN achieved higher performance when images side length was smaller than 80. In contrast, bicubic interpolation showed slightly better otherwise. By the properties of SR techniques, SRCNN can replicate relevant features on small weeds images. Still, it may wrongly recover low-level features on larger weeds images, interrupting feature descriptions on the VGG model. This study shows promising real-world applications and can be applied when the input image's resolution is lower than those trained on the VGG model.

REFERENCES

[1] Noh Sun-Kuk, "Classification of Clothing Using Googlenet Deep Learning and IoT based on Artificial Intelligence," *Smart Media Journal,* vol. 9, no. 3, pp. 41-45, 2020

[2] So Yeon Jeon, Jong Hwa Park, Sang Byung Youn, Young Soo Kim, Yong Sung Lee, Ji Hye Jeon, "Real-time Worker Safety Management System Using Deep Learning-based Video Analysis Algorithm," *Smart Media Journal,* vol. 9, no. 3, pp. 25-30, 2020

[3] Minh-Trieu Tran, Guee-Sang Lee, "Super-resolution in Music Score Images by Instance Normalization," *Smart Media Journal*, vol. 8, no. 4, pp. 64-71, 2019

[4] G. Anbarjafari and H. Demirel, "Image super resolution based on interpolation of wavelet domain high frequency subbands and the spatial domain input image," *ETRI journal*, vol. 32, no. 3, pp. 390-394, 2010

[5] J. W. Hwang and H. S. Lee, "Adaptive image interpolation based on local gradient features," *IEEE signal processing letters*, vol. 11, no. 3, pp. 359-362, 2004

[6] H. Demirel and G. Anbarjafari, "Discrete wavelet transform-based satellite image resolution enhancement," *IEEE transactions on geoscience and remote sensing*, vol. 49, no. 6, pp. 1997-2004, 2011

[7] N. Reddy, D. Fahim Noor, Z. Li and R. Derakhshani, "Multi-frame super resolution for ocular biometrics," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018

[8] P. D. Sawant and H. H. Kulkarni, "Resolution enhancement and quality assessment of MRI images using different interpolation techniques," *in 2017 International Conference on Intelligent Computing and Control (I2C2)*, IEEE, 2017, June

[9] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, ... and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017

[10] B. Lim, S. Son, H. Kim, S. Nah and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," *in Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017

[11] C. Dong, C. C. Loy, K. He and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295-307, 2015

[12] V. Sharma, A. N. Diba, B. D., M. S., L. Van Gool and R. Stiefelhagen, "Classification-driven dynamic image enhancement," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4033-4041, 2018

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *in arXiv preprint arXiv:1409.1556.*, 2014

[14] Y. Taigman, M. Yang, M. A. Ranzato and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," *in Proceedings of the IEEE conference on computer vision and pattern recognition,* 2014

[15] M. Lin, Q. Chen and S. Yan, "Network in network," *in arXiv preprint arXiv:1312.4400*, 2013

[16] H. T. Vo, G. H. Yu, H. T. Nguyen, J. H. Lee, T. V. Dang and J. Y. Kim, "Analyze weeds classification with visual explanation based on Convolutional Neural Networks," *Smart Media Journal*, vol. 8, no. 3, pp. 31−40, 2019

[17] V. H. Trong, Y. Gwang−hyun, D. T. Vu and K. Jin−young, "Late fusion of multimodal deep neural networks for weeds classification," *Computers and Electronics in Agriculture*, 2020

[18] L. N. Smith, "Cyclical learning rates for training neural networks," *in 2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, March

Authors

Vo Hoang Trong

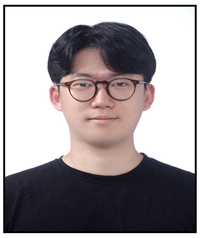
2017 : BS degree in Department of Mathematics, Computer Science, Ho Chi Minh University of Science, Vietnam
2020 : MS degree in Department of Electronic Engineering, Chonnam National University
2020.03 ~: Ph.D student in Department of Electrical Engineering, Chonnam National University
Research interests: Digital signal processing, image processing, machine learning

Yu Gwang−Hyun

2018 : MS degree in Department of Electrical Engineering, Chonnam National University
2018.03 ~ : PhD student in Department of Electronic Engineering, Chonnam National University
2018.12 ~ : Insectpedia C.E.O
Research interests : Digital signal processing, video processing, audio signal processing, machine learning, deep learning.

Nguyen Huy Toan

2012 : BS degree in Department of Electronic Engineering, Thai Nguyen University of Technology, Vietnam
2020 : PhD degrees in Department of Electronic Engineering, Chonnam National University
Research interests : Computer vision, wearable devices, system−based micro−processors, machine learning, deep learning.



Lee Ju−Hwan

2019 : BS degrees in Department of Earth and Environmental Sciences, Chonnam National University.
2019.09 ~ : MS student in Department of Electronic Engineering, Chonnam National University
Research interests : Digital signal processing, video processing, audio signal processing, machine learning, deep learning.

Dang Thanh Vu

2017 : BS degree in Department of Mathematics, Computer Science, Ho Chi Minh University of Science Department, Vietnam
2019.03 ~ : MS student in Department of Electronic Engineering, Chonnam National University
Research interests : Digital signal processing, video processing, audio signal processing, machine learning, deep learning.

Kim Jin−Young

1986 : BS degree in Department of Electronic Engineering, Seoul National University
1988 : MS degree in Department of Electronic Engineering, Seoul National University
1994 : PhD degree in Department of Electronic Engineering, Seoul National University
1995.03 ~ : Professor in Department of Electrical Engineering, Chonnam National University. Research interests : Digital signal processing, video processing, audio signal processing, machine learning, deep learning