IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 17, 2021, accepted December 16, 2021, date of publication December 20, 2021,
date of current version January 10, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3136781

Generation of Novelty Ground Truth Image Using Image Classification and Semantic Segmentation for Copy-Move Forgery Detection

KANG HYEON RHEE
School of Electronics Engineering, Chosun University, Gwangju 61452, South Korea
e-mail: khrhee@chosun.ac.kr

This work was supported by the National Research Foundation of Korea (NRF) under Grant 2021R1A2C1003406.

ABSTRACT Since the ground truth (GT) generated by CNN has pieces of patch information of the learned class, the accurate detection of Copy-Move is ambiguous. With various CNNs for image classification and semantic segmentation, the generated GT images are different yet similar to patch patterns for detecting forgery regions. It is difficult to determine which network model-generated GT image is suitable. Therefore, an optimal GT image is essential in image forensics. The proposed scheme in this paper generates a novelty GT image to solve this problem for the correct detection of Copy-Move forgery. The novelty GT image was configured using image classification and semantic segmentation. The variety of GT images is generated by adopting the state-of-the-art four image classifications and one semantic segmentation in the deep neural network. The proposed scheme implements mainly three tasks: 1) each network model generates the GT images (GT*net*), 2) which are convergence synthesized into one (GT*conv*), and 3) it decomposed again into GT images (GT*decomp*) with a threshold value of the 'Threshold Filter.' Here, the GTnet images involve two pieces of information about the image classification and semantic segmentation of the forgery image. The GTconv has two pieces of information as one GT image. The GTdecomp is decomposed GTconv into various GT images by the threshold value, which is a permeated degree of the information about 'Image classification' and 'Semantic segmentation.' The proposed novelty GT image is accomplished with this operational flow for Copy-Move forgery detection. The results confirmed in the experiment for comparing the performance of the existing GT*net* image, and the GT*decomp* image of the proposed scheme showed that the *Accuracy* and *F1 Score* of the proposed scheme had the maximum improvement rate of 0.4% and 0.2%, respectively. Also, by estimating the proposed CMFD scheme, Area Under the Curve (AUC) is graded as '*Excellent* (*A*)' with a value of 0.9 higher.

INDEX TERMS Copy-move image forensics, forgery detection, ground truth image, image classification, semantic segmentation, deep neural network.

I. INTRODUCTION

In modern social media, digital images contain a vast amount of information as an essential communication medium. While images can easily edit, transmit, and distribute information about our lives, trust in images is difficult to believe. It has emerged as an image forensics problem.

Using image editing tools (Premiere, Final Cut Pro, Vegas, Movavi, and After Effects, they registered trademarks.), a villain can easily manipulate or modify it. Therefore, Cut-Paste, Copy-Move manipulation, and distribution of malicious images cause severe personal infringement.

If there is image manipulation by such aggressive means, a defensive means to detect it is needed. Therefore, Cut-Paste [1], [2] and Copy-Move detection [3], [4] methods are being developed day by day as countermeasures.

Copy-Move operation selects one part of an image and copies it to another region of the same image. This manipulation method is used widely in the field of image forensics.

There are three main methods of detecting Copy-Move as follows:

1) 'Block segmentation': The forgery image to be tested is divided into several blocks, and features are extracted from each block. If blocks divided into $n \times n$ pixel square type have similar characteristics, these blocks

The associate editor coordinating the review of this manuscript and approving it for publication was Zhan-Li Sun.

are regarded as a Copy-Move area. There are some disadvantages: all blocks must be compared and computed, the computational cost is high, and other geometric transformations of moving patches cannot be handled. On the other hand, using the Simple Linear Iterative Clustering (SLIC) algorithm [5], the 2D superpixels of the image are calculated and divided into irregular shapes instead of the regular square division and groups the pixels into regions with similar values.

2) 'Keypoint matching': It is possible to find similar areas in the image by extracting and matching the key points of pixels without dividing the forgery image [6]. Scale-invariant feature transform (SIFT) [7], [8] is the most basic method for detecting Copy-Move. SIFT is an algorithm that extracts features that are invariant to the size and rotation of an image. The basic principle is to extract the SIFT features from two different images and find the corresponding parts by matching the most similar features in the two images. The disadvantage of SIFT is that the region of copy and moving are mismatched due to the bad key points. The expansion of the feature vector dimension for key point generation increases the computing cost and the decision time of the Copy-Move area, so it is not easy to manage the locale smoothly.

3) For pixel-based segmentation classification of images: BusterNet was proposed by Wu *et al.* [9] using the VGG16 net model structure. BusterNet has two branches: namely Simi-Det and Mani-Det. However, there are two drawbacks. One should ensure that both branches correctly locate regions, and the other, the Simi-Det branch, only extracts single-level and low-resolution features due to the four pooling layers in VGG16 [10].

   Chen *et al.* proposed the CMSDNet (copy-move similarity detection network) and the STRDNet (source/target region distinguishment network) [11], which is an atrous convolution instead of the 4th polling layer of VGG16 was used to preserve 'Field-of-views of filters'.

The above detection category is 'Passive forensic detection' methods that can detect the forgery area without needing a ground truth image of the Copy-Move area. On the other hand, the method [12] that uses CNN for Copy-Move detection is being developed. Still, since the class (copy area/move area), the ground truth image classification is trained together in the network model implementation. Hence, the author's view seems to be regarded as 'pre-Active forensic detection.'

The main words of Copy-Move forgery detection from the state-of-the-art [13] showed as a word cloud in Fig. 1.

Essentially, generating a good ground truth to detect the exact temper area of the forgery image is necessary for the image forensics field. To this end, this paper proposes a generation method of novelty ground truth for Copy-Move forgery detection. For this purpose, it adopts the state-of-the-art deep learning CNNs structure of existing 'image classification' [14]–[18] and 'semantic segmentation' [19]. The central concept of this paper is twofold:

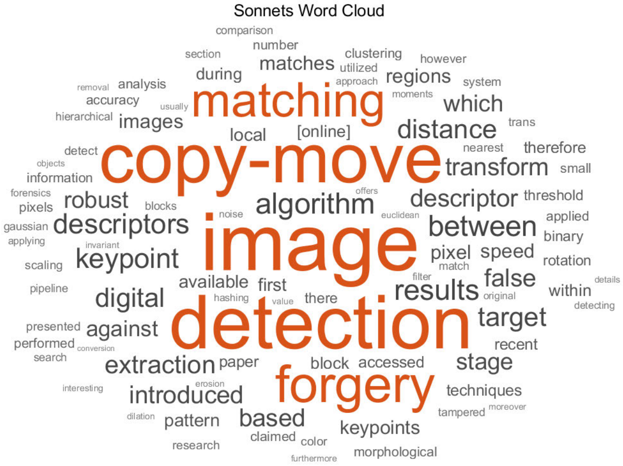


FIGURE 1. Word cloud of copy-move forgery detection in [13].

1) The generated novelty GT image involves two characteristics of 'image classification' and 'semantic segmentation' of a forgery image.
2) Copy-Move detection is considered as 'pre-Active forensic detection,' referring to the ground truth image only when network training is performed, but while testing for forgery detection, the trained forgery detector becomes 'Passive forensic ground truth' in the actual field.

Also, the contributed operations of this paper for forensic detection are as follows:

1) Copy-Move forgery detection using the deep learning CNN structure for 'image classification' and 'semantic segmentation.' The generated various ground truth images converged to be one synthesized pattern.
2) The pattern image above in 1) is decomposed into a novelty ground truth image which can most accurately detect the Copy-Move patch by the 'Crossing Over' operation and the threshold value of the 'Threshold Filter.'
3) The performance evaluation of general forgery detection calculates $\mathcal{Accuracy}$ and $\mathcal{F1\ Score}$ from the detection result of Copy-Move after overlaying the ground truth and the forgery image. However, the many generated ground truth images already include $\mathcal{Accuracy}$ and $\mathcal{F1\ Score}$. Therefore, a novelty ground truth image with high $\mathcal{Accuracy}$ and $\mathcal{F1\ Score}$ is ready to use for forgery detection; thus, CMFD is rapidly performed.

The above contribution in this paper is to reduce the generation cost of a ground truth image and improve the performance of an effective forensic detector.

The rest of this paper is organized as follows: Section 2 briefly introduces the state-of-the-art CNNs for 'image classification' and 'semantic segmentation.' In Section 3, the new scheme of the Copy-Move forgery detection is proposed. Here, a novelty ground truth image is generated with the properties introduced in Section 2. The experimental results are discussed in Section 4, the comparison of performance evaluation with the generated existing and proposed ground truth images. Lastly, Section 5 draws the conclusion and the future research possibilities presented for the area of image forensics.

II. THEORETICAL BACKGROUND

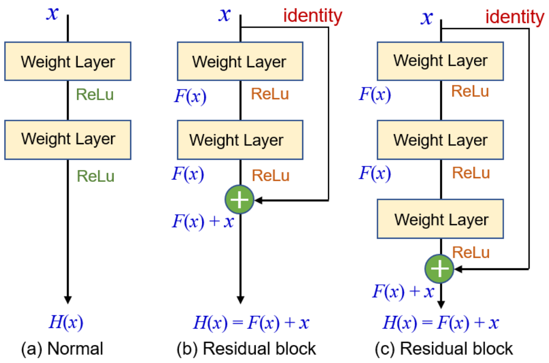
A. ResNet

ResNet [14], [15] is based on the structure of VGG-19 [20] and is a CNN model that is basically used in YoLo model. It won the 2015 ILSVRC (ImageNet Large Scale Visual Recognition Challenge) and was developed by Microsoft. Before ResNet, net models tried to increase performance by stacking many deep layers, but the ResNet author found the Vanishing Gradient Problem (VGP) out of optimization, as shown in Fig. 2(a). This phenomenon in which the influence of gradient on backpropagation rapidly decreases as the model deepens.

Fig. 2(b) shows that a residual block called bottleneck architecture was used to solve this problem.

In this module, the input x is a shortcut structure added directly to the output ReLu $(F(x) + x)$, and the VGP in Fig. 2(a) is solved when the number of layers increases, the input value is forgotten.

And in the version with more than 50 layers, the bottleneck skips connection structure, as shown in Fig. 2(c), is used.



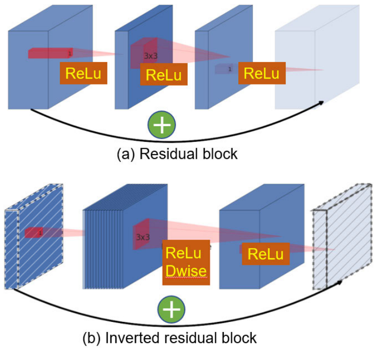
(a) Normal (b) Residual block (c) Residual block

FIGURE 2. Residual block in ResNet [14], [15].

B. MobileNetv2

Google developed MobileNetV2 [16]. Fig. 3(a) is the residual block structure of ResNet, and (b) is the inverted residual black of MobileNetV2.

In Fig 3,(a) is *wide* → *narrow* → *wide type*, and *narrow* makes a bottleneck, whereas (b) is *narrow* → *wide* → *narrow* type. The first feature is a linear bottleneck, and it is added to the last skip connection without going through ReLU. The necessary information is narrow and passed to deeper layers using the skip connection. Also, because skip connection is narrow, memory usage is reduced.



(a) Residual block

(b) Inverted residual block

FIGURE 3. MobileNetv2 [16].

C. XCEPTION

"Extreme Inception," [17] a powerful version of the Inception Architecture, is called 'Xception'. If the size of the network is increased to improve the performance of the deep neural network, 1) overfitting occurs, and 2) computational resources increase. GoogLeNet [21] uses the Inception module as shown in Fig. 4 to solve this problem. Rhee [22] proposed a network model for Cut-Paste forgery detection as an application thereof.

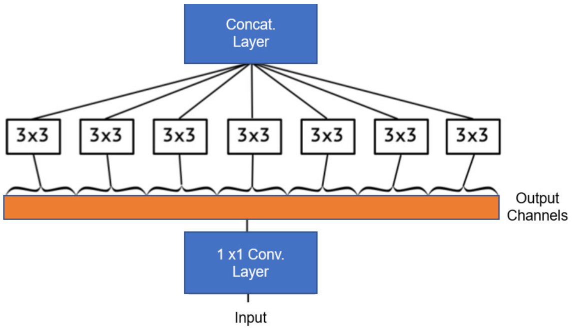
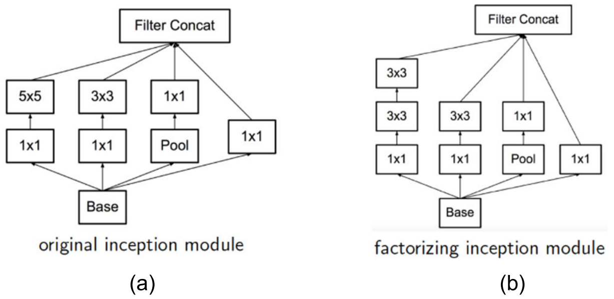


FIGURE 4. Xception [17].

D. INCEPTIONRESNETV2

As a model that surpassed VGG and won 1st place in IRSVRC 2014, a thesis was published under the name of Inception [18], and one of several versions of Inception is GoogLeNet [21]. Fig. 5(a) and (b) present the modules of the original and factorizing module types. Inception v2 has 2.5 times more training parameters than Google Net but is more efficient than VGG.



original inception module (a) factorizing inception module (b)

FIGURE 5. Inception ResNetv2 [18].

E. DEEPLAB V3PLUS STRUCTURE

When generating DeepLab v3plus networks [19] using Xception or Mobilenetv2 primary networks, separable depth convolutions are used in Atrous Spatial Pyramid Pooling (ASPP) and decoder subnetworks [23].

Convolutional layers are used for all other primary networks.

This DeepLab v3plus implementation does not include a global average pooling layer in ASPP but transforms it to a CNN for semantic image segmentation.

The network uses an encoder-decoder architecture, extended convolution. To use a CNN network for semantic segmentation, retrain the segment region by giving it a class.

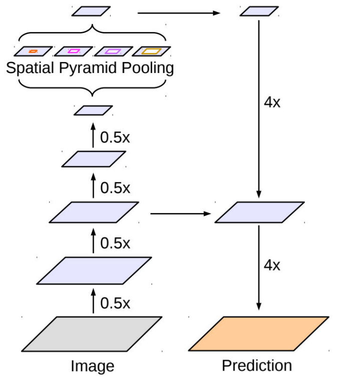


FIGURE 6. **Encoder-decoder with atrous convolution [23].**

F. SEMANTIC SEGMENTATION

Semantic segmentation [24] segments objects in an image into meaningful units. More specifically, it predicts which class each pixel in an image belongs to. This task is sometimes called dense prediction because prediction is performed on all pixels in the image.

If semantic segmentation applies to an image, then know which class each pixel belongs to. That is, the input value of the semantic segmentation algorithm is a color image or a black and white image, and the output value is a segmentation map representing the predicted class of each pixel as in Fig. 7 [25], the semantic labels {1: person, 2: purse, 3: plants/grass, 4: sidewalk, and 5: building/structures}.

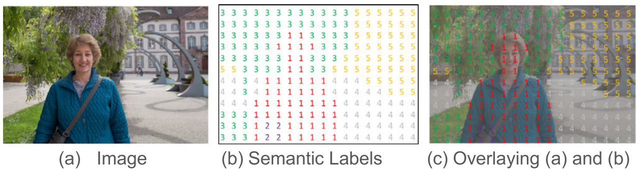

(a) Image (b) Semantic Labels (c) Overlaying (a) and (b)

FIGURE 7. **Semantic segmentation [24].**

G. BUSTERNET

Wu *et al.* proposed a BusterNet [9] with a double-branched DNN structure based on the VGG16 [10] network as shown in Fig. 8, and it is composed of Mani-Det and Simi-Det.

The feature extraction modules of BusterNet use the VGG16 network, and the mask decoder module uses 4 pairs of BN-Inception [9] and performs bilinear upsampling. A correlation module is added to the Simi-Det branch to detect copy patches. This module consists of an autocorrelation layer, a percentile pooling layer, and a batch normalization layer.

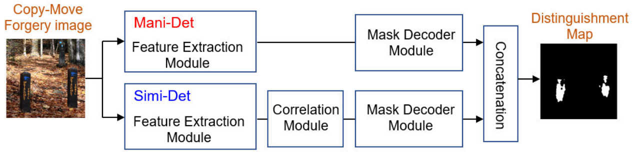


FIGURE 8. **BusterNet [9].**

Mani-Det detects forged area (moved patch), and Simi-Det detects similar patch (copy patch). These two patches are defined into copy and move regions in the distinguishment map. BusterNet has two drawbacks [11]. The one should ensure that both branches correctly locate regions, and the other one, the Simi-Det branch, only extracts single-level and low-resolution features due to the four pooling layers in VGG16. Therefore, the detected forged area may not be correct.

H. SOURCE/TARGET DISTINGUISHMENT

Chen *et al.* proposed the Source/Target Distinguishment [11] as shown in Fig. 9 to compensate for the shortcomings of BusterNet. The parallel scheme BusterNet should ensure that each branch locates regions correctly. In contrast, this structure has the CMSDNet (copy-move similarity detection network) and the STRDNet (source/target region distinguishment network) connected in series. A detection map is inserted between them. That comprises three modules: the feature extraction module, correlation module, and mask decoder module. The peculiarity is that instead of the fourth polling layer of VGG16, an atrous convolution is used to preserve ‘Field-of-views of filters.

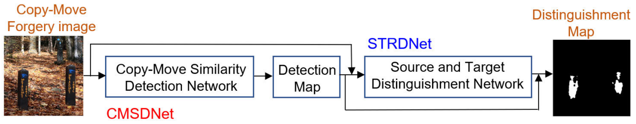

Copy-Move Forgery image
Copy-Move Similarity Detection Network
Detection Map
STRDNet
Source and Target Distinguishment Network
Distinguishment Map
CMSDNet

FIGURE 9. **Source/target distinguishment [11].**

III. PROPOSED COPY-MOVE FORGERY DETECTION SCHEME

A. CONFIGURE TO SEMANTIC SEGMENTATION OF NETWORK STRUCTURE

In this work, a novelty ground truth image to detect a Copy-Move region generated using the deep neural of the state-of-the-art CNN networks {Resnet50 [14], Resnet18 [15], Mobilenetv2 [16], Xception [17], Inceptionresnetv2 [18], and DeepLab v3plus structure [19]} that have excellent performance in ‘image classification’ and ‘semantic segmentation’ were developed, respectively. In Fig. 10, [14]–[18] used for ‘image classification’ are converted into that called '*net*_Model *n*' ($n = 1 \sim 5$) in turn through the [19] used for 'semantic segmentation.' So, the structures of the *net*_Models are newly designed and have the characteristics of 'image classification' and 'semantic segmentation,' both.

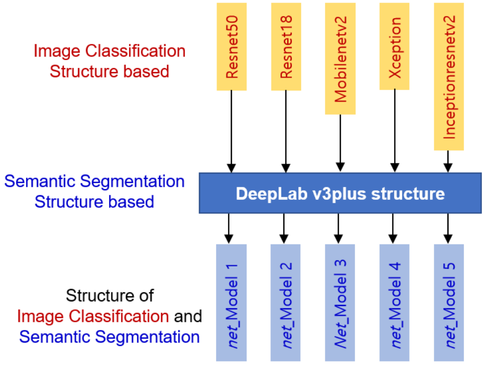


FIGURE 10. Designed *net*_models for image classification and semantic segmentation.

B. CONFIGURE TO NET_MODEL TRAINING

Table 1 shows the composition of the Copy-Move image data set {CoMOFoD [26], CVIP [27], GRIP [28], CASIA [29], COVERAGE [30], MICC-F600 [31], and CPH [32]}, and those augmented Copy-Move images with a shifted eight directions are used for the training of the *net*_Models for the configured the structure of 'image classification' [14]–[18] and 'semantic segmentation' [19].

TABLE 1. Copy-move forgery image data sets.

Image data set	Number of images	Forgery type
CoMoFoD [26]	200	Copy-Move, Scale, Rotate
CVIP [27]	150	Copy-Move, Scale, Rotate
GRIP [28]	80	Copy-Move, Scale, Rotate
CASIA [29]	247	Copy-Move, Scale, Rotate
COVERAGE [30,31]	100	Copy-Move, Scale, Rotate
MICC-F600 [32]	160	Copy-Move, Scale, Rotate, Edge blur,
CPH [33,34]	213	Copy-Move, Scale, Rotate,
Multiple Augmented by eight directions. (←↑→↓↖↗↘↙)	10,200	Generated in Matlab toolbox for deep learning
Total	**11,350 images**	

A total of 11,350 images of the seven image data sets are divided at a (0.7: 0.15: 0.15) ratio for training, validation, and testing of the *net*_Models. Randomly selected 1,275 images of the training data for an augmentation used in the *net*_Models, a copy patch of data image is transformed with {scaled or rotated, or blur}, then moved to another region in an image.

In the manipulation area of the forgery image, copy and move areas are Class 1, and the other innocent area is Class 2. In the ground truth image, each class appears as 'white' and 'black' colors, respectively, shown in Fig. 11.

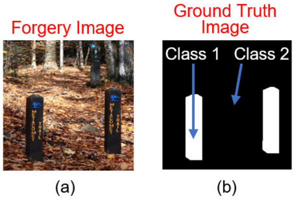


FIGURE 11. Class assignment of GT image.

C. PROPOSED SCHEME

It is training to implement net_Models as a detector for Copy-Move detection. The training option used in the proposed scheme adopts SGDM (Stochastic Gradient Descent with Momentum) [35].

Subsequently, to improve the performance of CMFD, the workflow of the scheme proposed in this paper is shown in Fig. 12. The process from the GT generation of forgery image in *net*_Model (on the upper left) to the decision of the Copy-Move forgery (on the far right), described as follows:

1) The structure of the state-of-the-art five networks [14]–[19] transformed to *net*_Models (No. 1-5) which generate the ground truth image (GT*net*) of the forgery image.
2) All GT*net* images are convergence and synthesized into one GT*conv* image (thick purple line box).
3) Since GT*conv* depth ranges from 1 to 5, the decomposed images (GT*decomp*) are generated by threshold values (*thr* = 1 to 5).
4) GT*net* and original ground truth image (GTorg) become GT^{org}_{net} image by "Crossing Over" operation, which includes true positive (TP), true negative (TN), false positive (FP), and false negative (FN) information to detect the Copy-Move patch, is displayed in 'white', 'black', 'red', and 'green' colors, respectively. The procedure is shown in Fig. 13.
5) With GT^{org}_{net} image, *Accuracy*, Recall, and *F1net Score* are calculated from TP, TN, FP, and FN.
6) GT*decomp* in 3) becomes GT^{org}_{decomp} image through the same procedure of 4) with GT*org*.
7) With GT^{org}_{decomp} image, *Accuracy*, Recall, and *F1decomp Score* are calculated from TP, TN, FP, and FN through the same procedure of 5).
8) By higher value is selected from *F1net* in 5) and *F1decomp Score* in 7), CMFD is determined.

In Fig. 12, procedure 4) is depicted, also procedure 6) has a working same.

D. PROPOSED FORGERY DETECTION METHOD

In Fig. 14, the performance evaluation of Copy-Move forgery detection is the block described in green, and the green line indicates the *Accuracy* and *F1 Score* calculation processing.

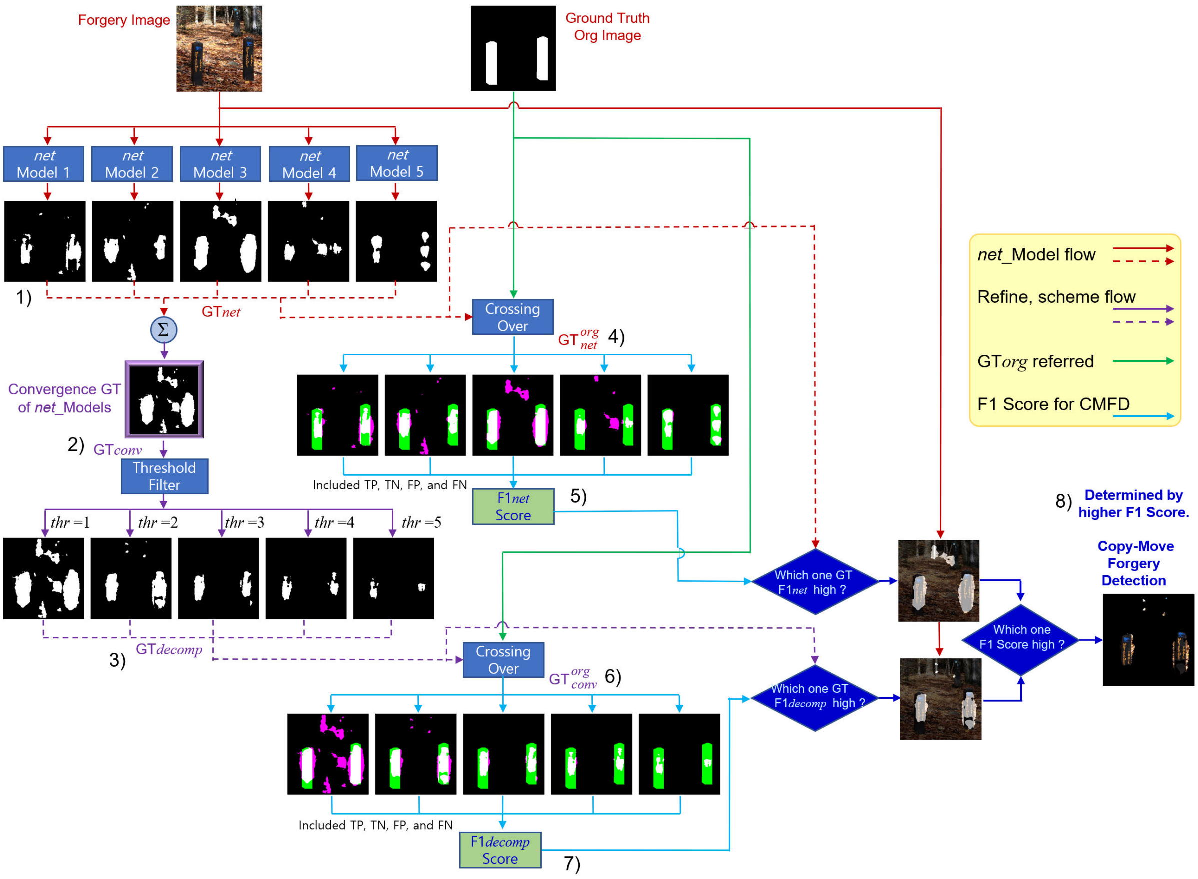


FIGURE 12. The proposed scheme: copy-move forgery detection for semantic segmentation.

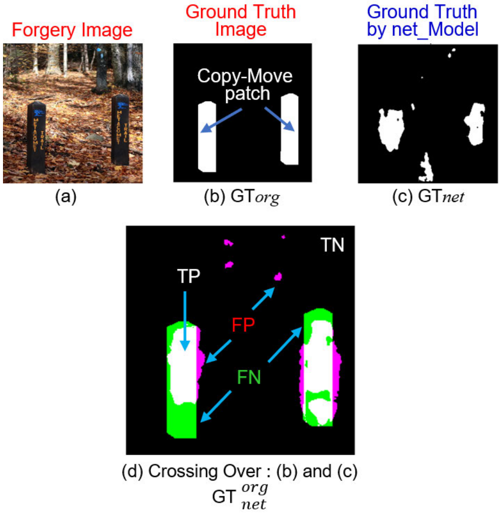


FIGURE 13. 'Crossing over': GT image of original and *net*_model.

The general and proposed methods are configured in (a) and (b), respectively.

In Fig. 14(a), the ground truth image generated by some method is overlaid with the forgery image, and then the evaluation is performed to examine the Copy-Move patch area.

If the result is not satisfactory, another method generates a ground truth image. Overlay with the forgery image again and repeat until the evaluation of the detection result is satisfactory.

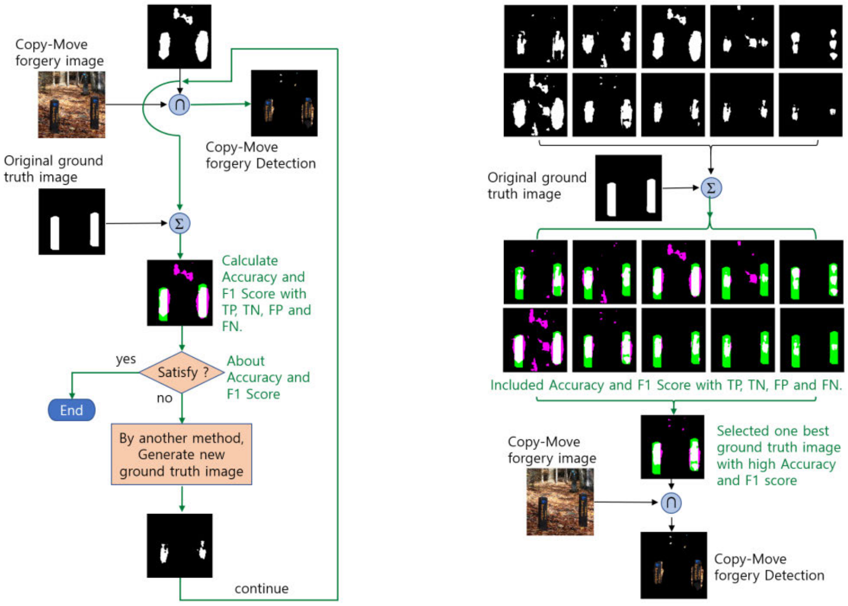
In Fig. 12 (b), on the other hand, the performance evaluation of copy-move forgery detection in the proposed scheme is directly performed by overlaying the generated ground truth images and the forgery image.

IV. EVALUATION OF EXPERIMENTAL RESULTS AND PERFORMANCE

For the experiment of the proposed scheme, an environment is the MATLAB 2021a tool used as simulation software on a PC environment (64bit Win10 Pro, AMD Ryzen9 3950X® 16-Core CPU @3.5GHz, 128GB DDR4 memory, and NVIDIA 2080Ti 11GB Double graphic boards).

A. NET_MODEL TRAINING EVALUATION

The execution of each *net*_Model of the training process configured in Section III-*B* is shown, which includes {Training Loss, Training Accuracy, and Final Validation Accuracy} in Fig. 15 (a) ∼ (e), respectively. These measured variable values are shown {Training time, the number of layers, Final Validation $\mathcal{Accuracy}$} of *net*_Models, respectively in Table 2.



(a) General evaluation method.

(b) Proposed evaluation method.

FIGURE 14. Evaluation processes of copy-move forgery detection between general and proposed method.

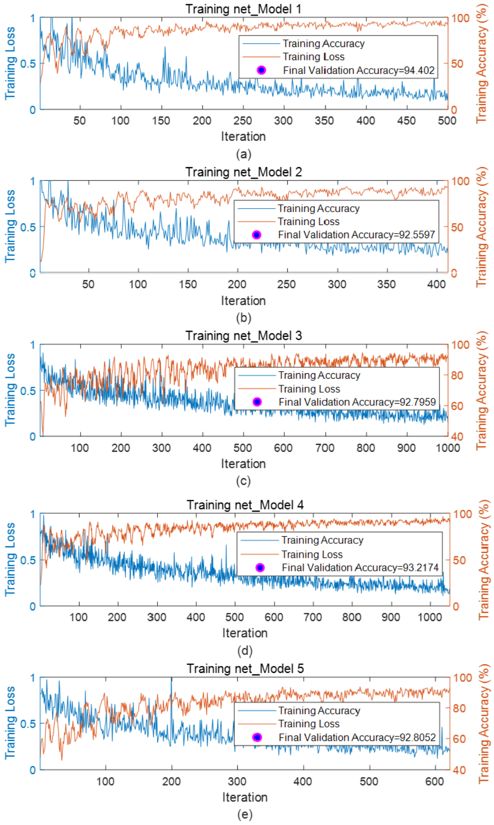


FIGURE 15. Accuracy and loss of training of *net*_models.

B. EVALUATION OF TRAINED NET_MODEL

The Copy-Move forgery images are tested for test preparation in Section III-B on the trained *net*_Model of the proposed scheme.

TABLE 2. Training results.

*net*_Model No.	Training time	Number of Layers / Connections	Final Validation Accuracy
1	16min.57sec.	226 / 227	0.9440
2	11min.22sec.	100 / 113	0.9256
3	43min.07sec.	186 / 201	0.9280
4	26min.13sec.	205 / 222	0.9323
5	145min.16sec.	853 / 956	0.9281

For the CMFD performance evaluation of 1) and 3) generated in the proposed scheme as shown in Fig. 12, the GT images generated in procedure 4) and 6) is a paired pattern of the “Crossing Over” operation (Fig. 13), with TP, TN, FP, and FN are displayed as the following color components.

- White area (TP): The same region of Class 1 of the original GT and the generated GT.
- Black area (TN): The same region of Class 2 of the original GT and the generated GT.
- Red area (FP): Copy-Move area in which the generated GT is erroneously detected.
- Green area (FN): The original GT is the Copy-Move area to be detected, but the generated GT is missing

From procedures 5) and 7), the $\mathcal{F}1\ Score$ is calculated by measurement equations (1) to (7). The meaning of the measurement is described in Table 3.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

TABLE 3. Meaning: measurement items.

Measurement	Meaning where Class 1: ‘white’ and Class 2: ‘black’ in GT image.
Accuracy	In overall predictions, the ratio of correct prediction (regardless, whether the prediction is ‘Class 1’ or ‘Class 2’).
Precision	In the ratio of actual ‘Class 1,’ among the predicted as ‘Class 1’.
Recall	The ratio of predictions of ‘Class 1,’ among actually ‘yes’.
Specificity	The ratio of predictions of ‘Class 2,’ among the actually ‘Class 2’.
Balanced Accuracy	Average of a *Recall* and a *Specificity*.
F1 Score	The harmonize average of ‘*Precision*’ and ‘*Recall*’. (It has a larger value when both values are equally large than when only one of ‘*Precision*’ and ‘*Recall*’ is large)
IoU Score	Jaccard similarity coefficient: A measure of statistical accuracy that penalizes FP. It is the ratio of the number of GT pixels of each class to the number of predicted pixels.

$$Specificity = \frac{TN}{FP + TN} \tag{4}$$

$$Balanced\ Accuracy = \frac{Recall + Specificity}{2} \tag{5}$$

$$F1\ Score = 2 \cdot \frac{Accuracy \cdot Recall}{Accuracy + Recall} \tag{6}$$

$$IoU\ Score = \frac{TP}{TP + FP + FN} \tag{7}$$

Table 4 shows the measured items of *net*_Models according to the meaning of Table 3. *net*_Model No. (1 to 3) are above the average of each measurement item (blue bold: max.). The values in Table 4 are recorded with the final validation after the training.

TABLE 4. Measured items of *net*_Models on the training.

*net*_Model No.	Measurement Items				
	Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU	Mean BFScore
1.	**0.940**	0.791	0.630	0.922	**0.537**
2.	0.926	**0.829**	0.614	0.908	0.521
3.	0.928	0.813	**0.639**	**0.923**	0.524
4.	0.932	0.791	0.588	0.899	0.459
5.	0.928	0.770	0.566	0.887	0.473

Fig. 16 shows a confusion matrix to classify ‘Class 1’ (white: Copy-Move patch) and ‘Class 2’ (black: innocent region) for the generated GT*net* images of the entire test forgery images by each *net*_Model (a) ∼ (e), see Fig. 11–1). The detection rate of the Copy-Move region is about 70% of ‘Class 1’, which is more than 50% of the class classification criterion, confirming the validity of semantic segmentation.

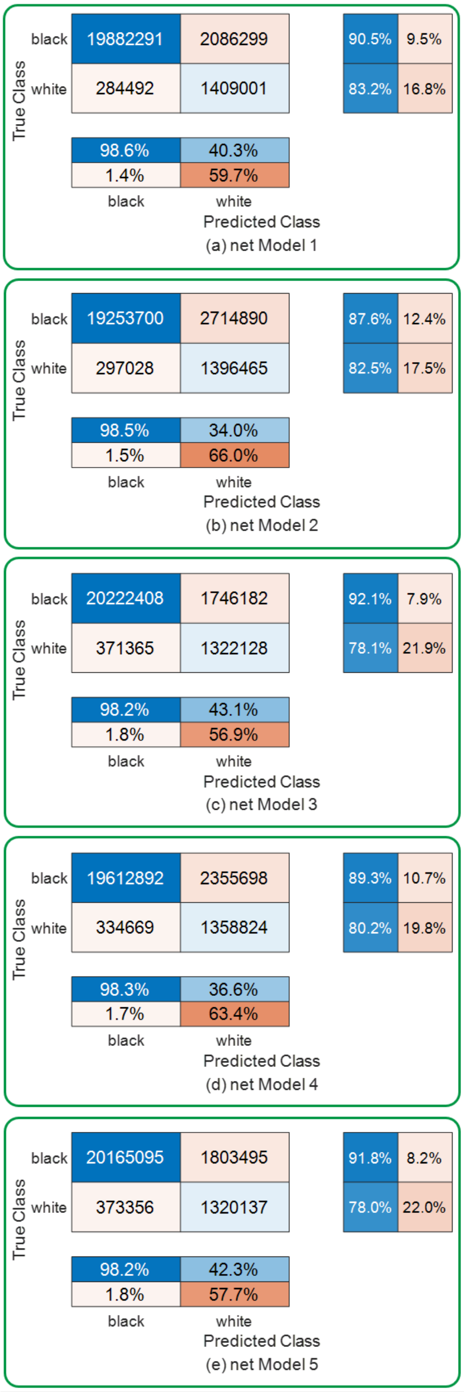


FIGURE 16. Confusion matrix: classification of ‘copy-move’ and background.

C. PERFORMANCE EVALUATION OF COPY-MOVE DETECTION ON TESTING

Fig. 17 ∼ 35 shows the results of detecting Copy-Move by randomly selecting 19 images from the Copy-Move forgery image data set in Table 1. Each Figure is an example of a forgery image to which various manipulation methods and environments of Copy-Move are applied, and the description of the images in (a) ∼ (i) of Fig. 17 ∼ 35 is as follows:

(a) Query, Copy-Move forged image.
(b) GT image of (a).
(c) Copy-Move forgery region of (a) and (b).

(d) Generated GT image by *net*_Model in Fig. 12, 1).
(e) Class overlay image of (a) and (d).
(f) Detection of Copy-Move patch of (a) and (d)

(g) Generated GT image by proposing Scheme in Fig. 12, 3).
(h) Class overlay image of (a) and (g).
(i) Detection Copy-Move patch of (a) and (g)

Here, the brightness of the class overlay image (e) and (h) is differently displayed to distinguish the generated GT image of *net*_Model and the proposed scheme. Also, Table 5 shows how the GT generated from which model became the best GT*decomp* image with how much threshold value.

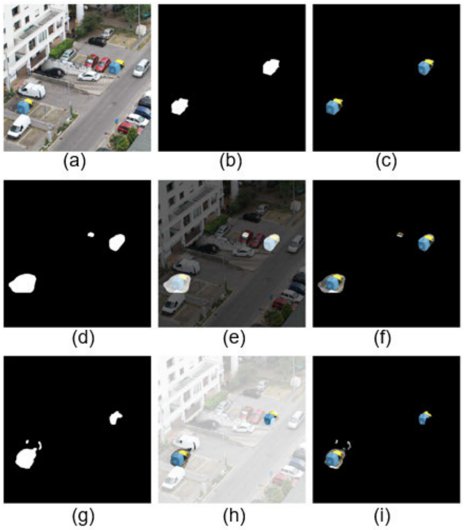


FIGURE 17. Similar object around.

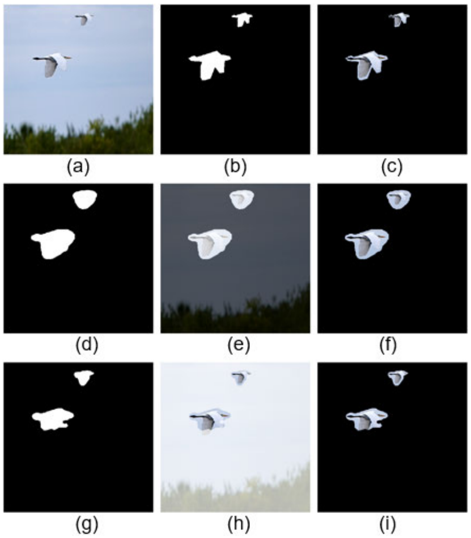


FIGURE 18. Scale.

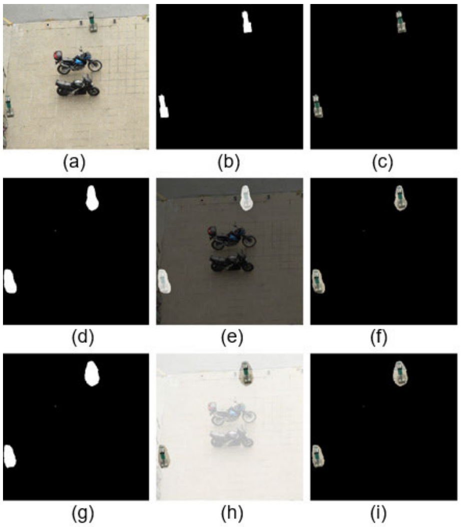


FIGURE 19. Edge region of image.

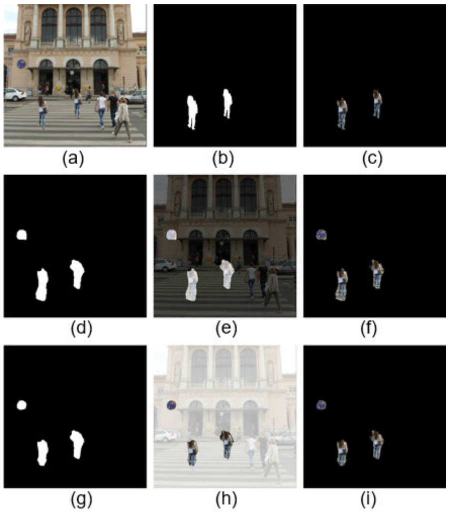


FIGURE 20. Regular background.

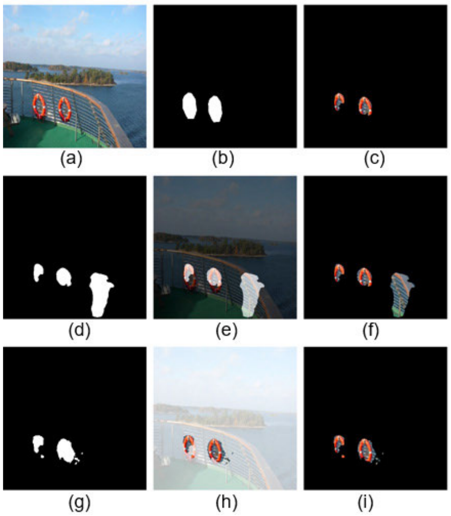


FIGURE 21. Single object.

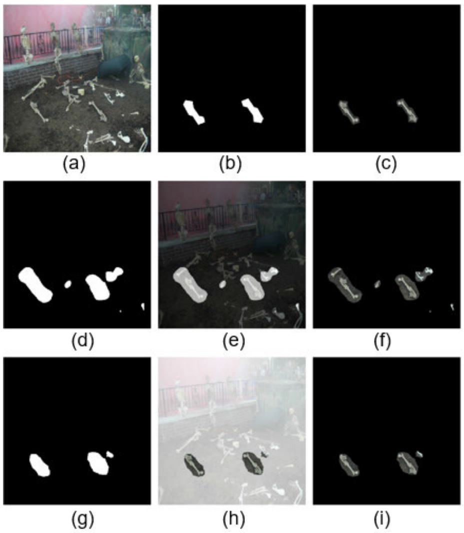


FIGURE 22. Similar multiple objects.

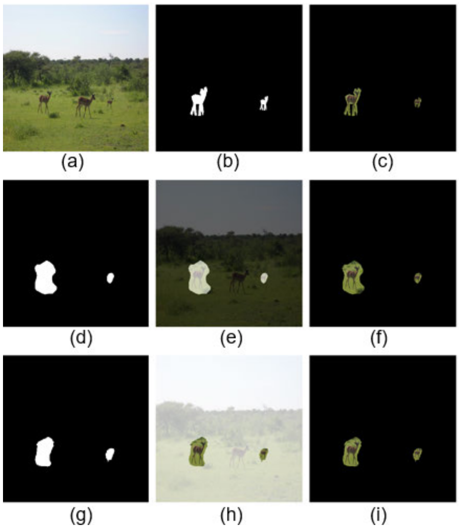


FIGURE 23. Scale.

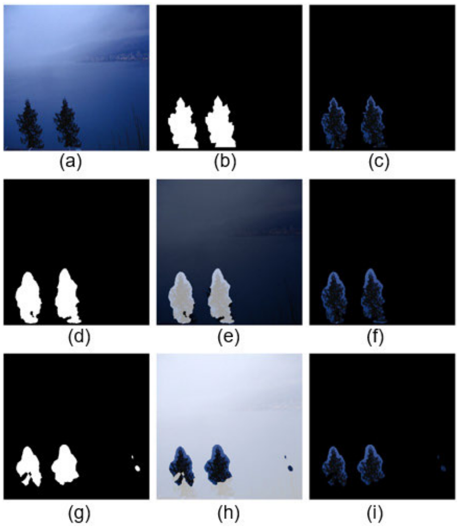


FIGURE 24. Blur.

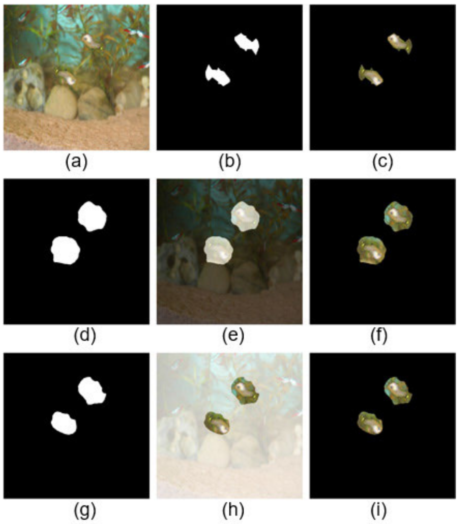


FIGURE 25. Over glass.

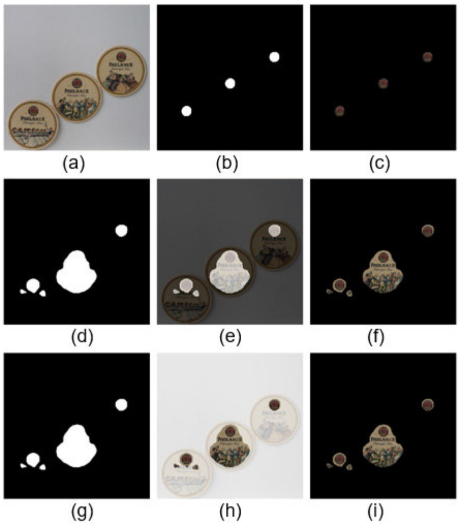


FIGURE 26. Multiple moving patches.

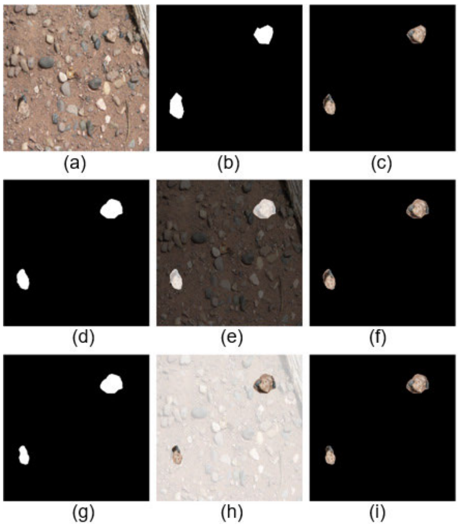


FIGURE 27. Similar texture component.

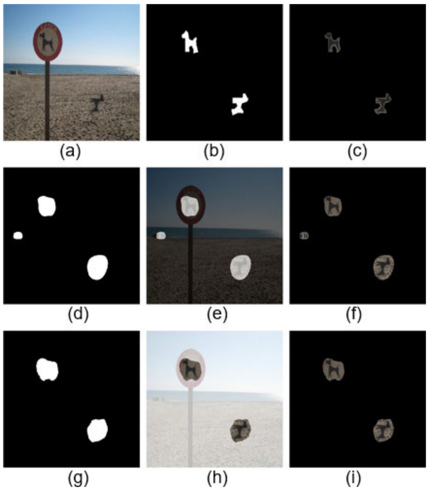


FIGURE 28. Rotate.

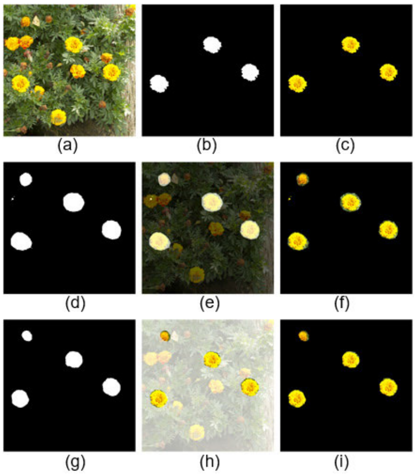


FIGURE 29. Multiple moving patches.

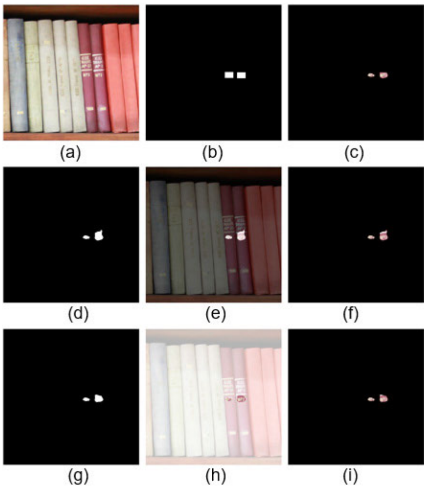


FIGURE 30. Small character.

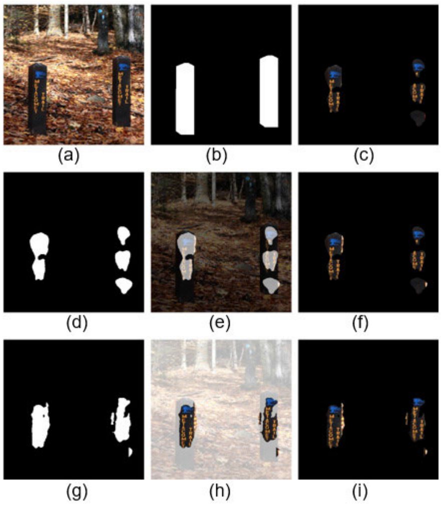


FIGURE 31. Small character.

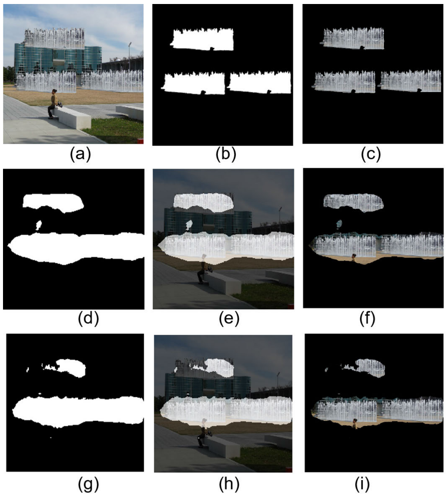


FIGURE 32. Very delicate pattern.

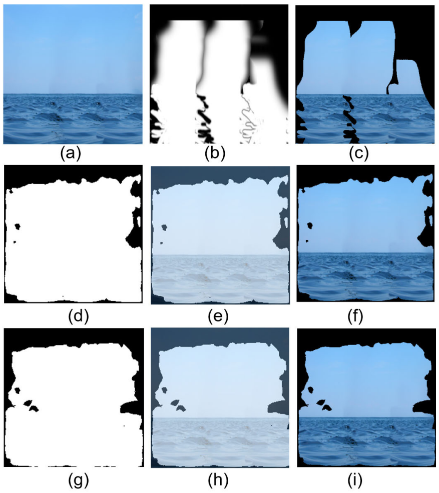


FIGURE 33. Texture is very similar to its surroundings.

D. COMPARISON OF GT IMAGEs BETWEEN RETRANSFORMED NET_MODEL AND PROPOSED SCHEME FOR COPY-MOVE DETECTION

The experimental results were measured with six items {*Accuracy*, *Balanced Accuracy*, *Precision*, *Recall*, *F1 Score*, and *Specificity*} in Table 6 (next page). As a result of comparing the generated GT images of each *net*_Model and the proposed scheme, the performance was somewhat lower than that of *net*_Model 1; meanwhile, it was superior to the results of

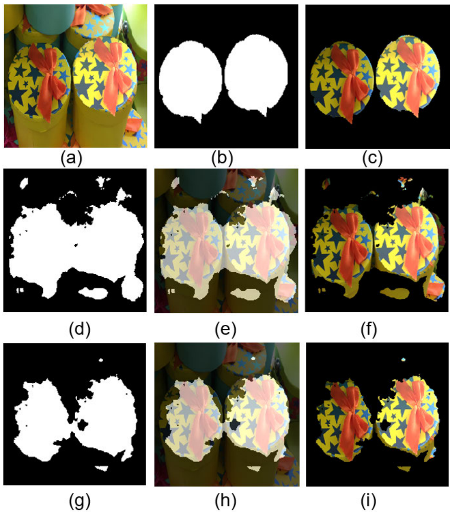


FIGURE 34. Luminance change of copy-move patch.

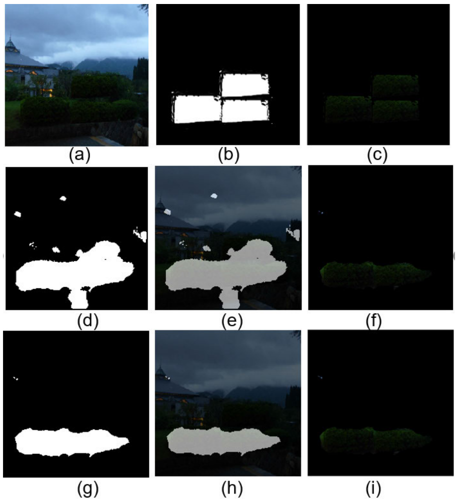


FIGURE 35. Copy-move patch with low intensity.

TABLE 5. Selected net_model no. and threshold value.

FIGURE No.	Generated GT*net* image from which *net*_Model No.	Generated GT*decomp* image, *threshold* value.
FIGURE 17	5	3
FIGURE 18	3	4
FIGURE 19	5	4
FIGURE 20	1	5
FIGURE 21	3	3
FIGURE 22	5	5
FIGURE 23	3	4
FIGURE 24	1	3
FIGURE 25	3	5
FIGURE 26	5	5
FIGURE 27	1	5
FIGURE 28	5	4
FIGURE 29	4	5
FIGURE 30	2	5
FIGURE 31	5	3
FIGURE 32	3	5
FIGURE 33	5	5
FIGURE 34	4	3
FIGURE 35	2	4

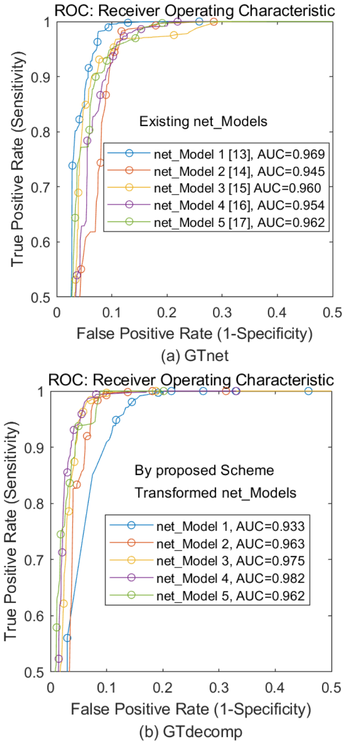


FIGURE 36. ROC curves: (a) GT*net* and (b) proposed GT*comp*.

other *net*_Model 2 ∼ 5 (red), and it was confirmed that the improvement rate was increased (blue bold).

The final validation accuracy in Table 4 is the accuracy of the verification process after *net*_Model training, and the accuracy in Table 6 is the results of test processing with the trained *net*_Model. The maximum $\mathcal{Accuracy}$ is a value of 0.9523 (red bold on highlight).

Fig. 36 presents the estimated proposed scheme with ROC curves: (a) the GT*net* classification of the existing *net*_Models and (b) the GT*comp* classification of the proposed scheme.

The aspect of the whole AUC values confirms that outstanding of the AUCs with a value of 0.9 higher. Thus, the evaluation [36] of the proposed CMFD scheme was graded as '***Excellent*** (*A*)'.

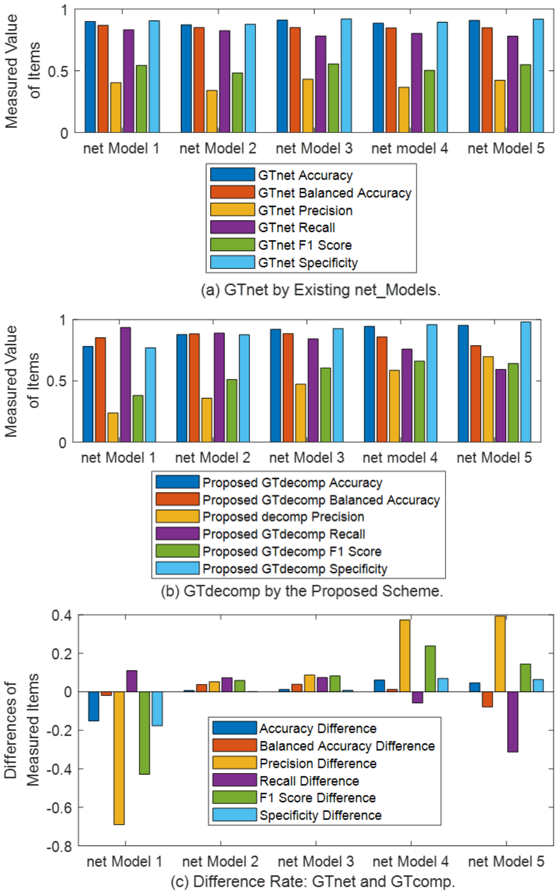
In particular, it was also confirmed that all measurement items of *net*_Model 2 and 3 were '*excellent*' (red) by increasing the improvement rate.

The measurement results in Table 6 present a graph for visual aid. In Fig. 37, GT*net* (a), and GT*comp* (b) are shown, respectively, and the improvement rates of each *net*_Models are shown in (c).

The performance evaluation of the proposed CMFD scheme was compared with the state of the arts [9], [11]

TABLE 6. Comparison between proposed scheme and *net*_model on the test processing.

Generated GT	Measurement Items					
	Accuracy	Balanced Accuracy	Precision	Recall (Sensitivity)	*F1 Score*	Specificity
*net*_Model l. [11] GT	0.8996	0.8684	0.4040	0.8319	0.5439	0.9049
Proposed GT	0.7812	0.8521	0.2390	0.9350	0.3807	0.7693
Improvement Rate	-0.1516	-0.0191	-0.6904	**0.1102**	-0.4287	-0.1763
*net*_Model 2. [12] GT	0.8726	0.8502	0.3406	0.8241	0.4820	0.8763
Proposed GT	0.8780	0.8832	0.3594	0.8892	0.5119	0.8771
Improvement Rate	**0.0062**	**0.0373**	**0.0522**	**0.0732**	**0.0583**	**0.0009**
*net*_Model 3. [13] GT	0.9103	0.8506	0.4318	0.7808	0.5561	0.9204
Proposed GT	0.9211	0.8850	0.4728	0.8428	0.6058	0.9272
Improvement Rate	**0.0117**	**0.0388**	**0.0866**	**0.0735**	**0.0819**	**0.0073**
*net*_Model 4. [14] GT	0.8861	0.8476	0.3668	0.8027	0.5035	0.8926
Proposed GT	0.9440	0.8584	0.5854	0.7585	0.6608	0.9584
Improvement Rate	**0.0613**	**0.0126**	**0.3734**	-0.0583	**0.2380**	**0.0686**
*net*_Model 5. [15] GT	0.9078	0.8488	0.4236	0.7799	0.5490	0.9178
Proposed GT	**0.9523**	0.7870	0.6976	0.5940	0.6416	0.9800
Improvement Rate	**0.0467**	-0.0785	**0.3927**	-0.3129	**0.1443**	**0.0636**



(a) GTnet by Existing net_Models.

(b) GTdecomp by the Proposed Scheme.

(c) Difference Rate: GTnet and GTcomp.

FIGURE 37. Comparison of GT*net* and GT*decomp*.

of the pixel-based segmentation classification described in Sections 1 and 2. Table 7 presents the executed results.

The experimental environment is executed under the same conditions as in this paper.

TABLE 7. Comparison of the existing scheme and the proposed.

CMFD Scheme	*Accuracy*	Recall	*F1 Score*	Characteristics
[9]	0.9218	0.9013	0.6315	Parallel structure, VGG16 based.
[11]	0.9425	0.9298	0.6602	Serial structure, 'Field-of-views of filters
Proposed	0.9523	0.9350	0.6608	V3+ structure, Image classification and Semantic segmentation

It was confirmed that the deep learning network structure of the copy and move region patch classification for CMFD has a higher classification rate with the converged method: (image classification + semantic segmentation) against the structures of a parallel and serial configuration, respectively.

V. CONCLUSION

This paper proposed a new scheme for generating a novelty ground truth image detecting Copy-Moved patches in a forged image. The generated GT involves 'Image classification' and 'Semantic segmentation,' both in Copy-Move forgery images.

According to the information of 'Image classification' and 'Semantic segmentation,' the best-fit GT image of the test forgery image is generated.

Variety kinds of forgery patches in Copy-Move {such as Rotate, Scale, and Blur, *etc.*} were included in the data set of the *net*_model training and were detected well in the test stage.

The proposed GT image of the dedicated CNN structures of excellent image classification and semantic segmentation improved the detection {*Accuracy* and *F1 Score*} for the Copy-Move matching. Consequently, the proposed scheme in this paper could be applied more in-depth to the image forensics field (such as Cut-Paste, Inpainting, and Forgery feature extraction, *etc*.).

The detecting method of the Copy-Move patch region of the proposed scheme can be used quickly in an environment of online JIT (Just In Time). It is necessary to advance research on detecting the multiple classes of moving patches in the future.

ACKNOWLEDGMENT

The author would also like to thank the leading researchers in this study field, the hidden reviewers whose meaningful comments, and the official working of Associate Editor, IEEE Access. With the treatments, this paper could be written thoughtfully.

REFERENCES

[1] K. H. Rhee, "Detection of spliced image forensics using texture analysis of median filter residual," *IEEE Access*, vol. 8, pp. 103374–103384, 2020, doi: 10.1109/ACCESS.2020.2999308.

[2] K. H. Rhee, "Forensic detection using bit-planes slicing of median filtering image," *IEEE Access*, vol. 7, pp. 92586–92587, 2019, doi: 10.1109/ACCESS.2019.2927540.

[3] C. Wang, Z. Zhang, Q. Li, and X. Zhou, "An image copy-move forgery detection method based on SURF and PCET," *IEEE Access*, vol. 7, pp. 170032–170047, 2019, doi: 10.1109/ACCESS.2019.2955308.

[4] H. Chen, X. Yang, and Y. Lyu, "Copy-move forgery detection based on keypoint clustering and similar neighborhood search algorithm," *IEEE Access*, vol. 8, pp. 36863–36875, 2020, doi: 10.1109/ACCESS.2020.2974804.

[5] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012, doi: 10.1109/TPAMI.2012.120.

[6] X. Pan and S. Lyu, "Region duplication detection using image feature matching," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 857–867, Dec. 2010, doi: 10.1109/TIFS.2010.2078506.

[7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[8] A. Costanzo, I. Amerini, R. Caldelli, and M. Barni, "Forensic analysis of SIFT keypoint removal and injection," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 9, pp. 1450–1464, Sep. 2014, doi: 10.1109/TIFS.2014.2337654.

[9] Y. Wu, W. Abd-Almageed, and P. Natarajan, "BusterNet: Detecting copy-move image forgery with source/target localization," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 168–184. [Online]. Available: https://openaccess.thecvf.com/ECCV2018_search

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

[11] B. Chen, W. Tan, G. Coatrieux, Y. Zheng, and Y.-Q. Shi, "A serial image copy-move forgery localization scheme with source/target distinguishment," *IEEE Trans. Multimedia*, vol. 23, pp. 3506–3517, 2021, doi: 10.1109/TMM.2020.3026868.

[12] M. Barni, Q.-T. Phan, and B. Tondi, "Copy move source-target disambiguation through multi-branch CNNs," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 1825–1840, 2021, doi: 10.1109/TIFS.2020.3045903.

[13] S. Teerakanok and T. Uehara, "Copy-move forgery detection: A state-of-the-art technical review and analysis," *IEEE Access*, vol. 7, pp. 40550–40568, 2019, doi: 10.1109/ACCESS.2019.2907316.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision—ECCV*, vol. 9908. Cham, Switzerland: Springer, 2016, pp. 630–645. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-46493-0_38

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[17] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258, doi: 10.1109/CVPR.2017.195.

[18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.

[19] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Computer Vision—ECCV 2018* (Lecture Notes in Computer Science), vol. 11211. Munich, Germany: Springer, 2018, pp. 833–851, doi: 10.1007/978-3-030-01234-2_49.

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

[21] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 7–12, doi: 10.1109/CVPR.2015.7298594.

[22] K. H. Rhee, "Composition of visual feature vector pattern for deep learning in image forensics," *IEEE Access*, vol. 8, pp. 188970–188980, 2020, doi: 10.1109/ACCESS.2020.3029087.

[23] L. C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," Dec. 2017, *arXiv:1706.05587*.

[24] Q. Zhao, G. Cao, A. Zhou, X. Huang, and L. Yang, "Image tampering detection via semantic segmentation network," in *Proc. 15th IEEE Int. Conf. Signal Process. (ICSP)*, Dec. 2020, pp. 165–169, doi: 10.1109/ICSP48669.2020.9321086.

[25] J. Jordan. (Aug. 2021). *An Overview of Semantic Image Segmentation*. [Online]. Available: https://www.jeremyjordan.me/semantic-segmentation/

[26] *CoMoFoD—Image Database for Copy-Move Forgery Detection*. Accessed: Aug. 2021. [Online]. Available: https://www.vcl.fer.hr/comofod/comofod.html

[27] *CVIP*. Accessed: Aug. 2021. [Online]. Available: http://www.diid.unipa.it/cvip/?page_id=48

[28] *GRIP*. Accessed: Aug. 2021. [Online]. Available: http://www.grip.unina.it/research/83-multimedia_forensics/90-copy-move-forgery.html

[29] *CASIA V2*. Accessed: Nov. 2021. [Online]. Available: https://paperswithcode.com/dataset/casia-v2

[30] *COVERAGE*. Accessed: Nov. 2021. [Online]. Available: https://paperswithcode.com/dataset/coverage

[31] B. Wen, Y. Zhu, R. Subramanian, T.-T. Ng, X. Shen, and S. Winkler, "COVERAGE—A novel database for copy-move forgery detection," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Phoenix, AZ, USA, Sep. 2016, pp. 161–165, doi: 10.1109/ICIP.2016.7532339.

[32] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, "A SIFT-based forensic method for copy-move attack detection and transformation recovery," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 1099–1110, Sep. 2011, doi: 10.1109/TIFS.2011.2129512.

[33] *CPH*. Accessed: Nov. 2021. [Online]. Available: https://paperswithcode.com/dataset/cph

[34] M. Xu, T. Li, Z. Wang, X. Deng, R. Yang, and Z. Guan, "Reducing complexity of HEVC: A deep learning approach," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5044–5059, Oct. 2018, doi: 10.1109/TIP.2018.2847035.

[35] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," *Proc. 30th Int. Conf. Mach. Learn.*, 2013, vol. 28, no. 3, pp. 1310–1318. [Online]. Available: http://proceedings.mlr.press/v28/pascanu13.pdf

[36] T. G. Tape. (2021). *The Area Under an ROC Curve*. [Online]. Available: http://gim.unmc.edu/dxtests/roc3.htm

KANG HYEON RHEE received the B.S. and M.S. degrees in electronics engineering from Chosun University, in 1977 and 1981, respectively, and the Ph.D. degree in electronics engineering from Ajou University, Suwon, South Korea, in 1991. He is currently a Professor Emeritus with the School of Electronics Engineering, Chosun University, Gwangju, South Korea. His current research interest includes multimedia fingerprinting/forensics. He was a recipient of awards, such as the Haedong Prize from the Haedong Science and Culture Juridical Foundation, South Korea, which he received in 2002 and 2009, respectively.