Adversarial Attack Based on Perturbation of Contour Region to Evade Steganalysis-Based Detection

HYUNSIK NA, GWONSANG RYU, AND DAESEON CHOI, (Member, IEEE)

1Department of Software Convergence, Graduate School, Soongsil University, Seoul 07027, South Korea
2Department of Software, Soongsil University, Seoul 07027, South Korea
Corresponding author: Daeseon Choi (sunchoi@ssu.ac.kr)

This work was supported in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant through Korean Government [Ministry of Science and ICT (MSIT)] (Robust AI and Distributed Attack Detection for Edge AI Security) under Grant 2021-0-00511, and in part by the National Research Foundation of Korea (NRF) Grant through Korean Government (MSIT) under Grant 2020R1A2C1014813.

ABSTRACT Deep neural networks (DNN) are applied in various fields because they afford good performance. However, they seem vulnerable to adversarial examples. Therefore, studies are actively investigating adversarial defenses to improve the robustness of DNNs as well as adversarial attacks to generate more intensive adversarial perturbations. Most existing attack methods excessively modify images by generating overly large distortions or adding perturbations to regions with little impact on the DNN. As a result, the local smoothness of images is not maintained, and changes can be easily detected using a steganalysis-based detector. In this study, we propose a contour attack method to evade steganalysis-based detection. This method extracts the contour region from an image and adds perturbations only to applicable regions, thus maintaining the local smoothness. It can solve the problems mentioned above and effectively evade steganalysis-based detectors. Our experimental results demonstrate that the local smoothness is better than that in the case of original attack methods and that the detection evasion rate for a steganalysis detector is up to 19.9% higher. Further, structural similarity (SSIM) measurements and image comparisons demonstrate the improved imperceptibility of perturbations. In particular, the SSIM values for images subjected to the contour attack are close to 1.0.

INDEX TERMS Deep neural network, adversarial attack, steganalysis-based detection, local smoothness, imperceptibility.

I. INTRODUCTION Deep neural networks (DNN) have provided remarkable breakthroughs in various fields such as image classification [1], face recognition [2], self-driving cars [3], and radiology [4]. However, DNNs seem vulnerable to adversarial examples; therefore, studies are actively investigating this issue [5]. One adversarial example is an image with an imperceptible perturbation added to it; this may result in misclassification by the DNN. Gradient-based adversarial attacks have been studied using various techniques such as the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm proposed in 2013 [6], fast gradient sign method (FGSM) [7], basic iterative method (BIM) [8], DeepFool [9], C&W attack [10], and decoupled direction and norm (DDN) attack [11]. Studies have also investigated adversarial defenses to cope with adversarial attacks. For example, adversarial attacks have been prevented using concepts such as adversarial training [12], [13], defensive distillation [14], and filtering [15]. First, the adversarial training is performed to combine the loss for the original data with the loss for the adversarial data; consequently, the decision boundary is shifted while considering the adversarial example. This approach is a fast and simple defense method. Next, defensive distillation is performed with a confidence value derived by the DNNs instead of a true label. This approach can reduce the sensitivity of the DNNs to the input perturbation. Finally, filtering can mitigate the effect of adversarial perturbations to refine the input image. However, most of these defenses
have some limitations. First, adversarial training suffers from problems such as an excessive increase in the size of the training dataset, decrease in the classification accuracy of the natural model [12], and vulnerability to other attacks that have not been trained for [16]. Second, defensive distillation provides limited defense against adversarial attacks [10]. Lastly, filtering-based defense methods are vulnerable to adaptive attacks [17]. Therefore, researchers are now focusing on detection-based defense methods such as noise reduction [18], feature squeezing [19], and steganalysis-based detection [20].

Steganalysis-based detection involves extracting steganalysis feature sets from an original image and an adversarial image, training the machine learning detector using them, and finally using binary classification to judge whether the image is perturbed or not. When adversarial images are detected using this method, the key criterion is the local smoothness. Local smoothness refers to the small intensity difference between adjacent pixels, and it is an important factor in natural images. For example, studies have used a subtractive pixel adjacency matrix (SPAM) [21] or a spatial rich model (SRM) [22] as steganalysis feature sets; both these approaches are based on the dependency between two adjacent pixels and whether they are perturbed or not. Most distance-based adversarial attacks [7], [8], [10] do not consider the dependency between adjacent pixels and calculate the perturbation size independently for each pixel. Adversarial perturbations added to particular parts of a smooth region (e.g., sea, solid-colored wall, etc.) in images destroy the local smoothness of the image. Therefore, they can be detected easily by steganalysis-based detection. This detection method outperforms original detectors [18], [19], [23]. In particular, a detector based on SRM feature sets exhibits a high detection rate for the intensive C&W attack. However, this method is vulnerable to adversarial attacks that consider the local smoothness property.

Moreover, the attackers should consider another property, called “imperceptibility,” to generate more subtle adversarial images. If an adversarial image generated by the existing attack method does not consider the imperceptibility, it may be detected by human eyes easily. In other words, such attacks are not effective methods. Therefore, several researchers have studied imperceptible adversarial attacks [24], [25]. However, these methods also have visible distortion against the original image although the imperceptibility is improved.

In this study, we propose a contour attack to evade steganalysis-based detection. This approach can improve the imperceptibility of perturbations and reduce the amount of distortion. Our main idea is to avoid adding perturbations to smooth regions where the RGB values of adjacent pixels are similar; instead, perturbations are added to contour regions where the RGB values of adjacent pixels are different. In this manner, smooth regions can be preserved. Toward this end, we propose a simple algorithm for extracting contour regions. Then, we preserve the parts excluded from the contour region and perturb the contour region using the original attack methods. This study makes the following contributions:

- We propose a methodology that can evade steganalysis-based detection by exploiting its vulnerability to attacks that consider the local smoothness property. We discuss the limitations of steganalysis-based detection through the visualization of steganalysis feature sets. Our method can maintain the local smoothness better than existing attack methods.

- We validate the performance of our proposed attack against steganalysis-based detection in terms of the attack success rate, amount of distortion, and perturbation size based on the $L_p$ distance and color distance. Our experimental results indicate that our attack has 19.9% higher detection evasion rate against steganalysis-based detection while maintaining its attack success rate. Further, the amount of distortion was reduced up to 126,584 for FGSM. We analyzed the basis of variation with perturbation size for each attack. Finally, we demonstrated the efficiency of our attack method through comparisons with existing attack methods.

- Through image visualization and statistical metrics like structural similarity (SSIM), we demonstrated that our attack method has higher imperceptibility compared with existing methods. Notably, the evaluation using SSIM is not a standard assessment in the field of adversarial attacks. However, it may provide an idea about real-world applications. Additionally, the contour region can serve as a new perturbation space for generating an adversarial attack in images.

The remainder of this paper is organized as follows. Section 2 discusses existing adversarial attack methods that we use as a baseline before introducing our proposed contour attack. It also explains the steganalysis-based detection method proposed in [20]. Section 3 describes the proposed contour attack method and the process of how the original attack and the contour region are combined. Sections 4 and 5 respectively present the process and results of our experiments to measure the attack ability, robustness, and imperceptibility of our proposed attack. Finally, Sections 6 and 7 respectively summarize the performance and limitations of the proposed method.

II. BACKGROUND & RELATED WORK

A. ADVERSARIAL ATTACKS

Adversarial attacks are of two main types. Untargeted attacks tend to be classified as classes other than the real class $y_{true}$ of the image. In this case, the purpose of an attack is set as $C(x') = y_{pred} \neq y_{true}$, where $C(*)$ is the classification model based on a convolution neural network and $x' = x + \eta$ is the adversarial image obtained through the adversarial perturbation of the original image $x$. By contrast, targeted attacks tend to be misclassified $x'$ as particular classes $y_{target}$ if $y_{target}$ is different from $y_{true}$, that is, $C(x') = y_{pred} = y_{target}$. The present study focuses on untargeted attacks, and a future study will be focused on targeted attacks.
1) FGSM
A previous study analyzed the vulnerability of the linear behavior of a DNN [7] and proved that an adversarial attack that adds a small perturbation in the original image. The FGSM proposed in this study updates each pixel value by the perturbation size $\epsilon$ in the direction of the loss gradient sign for the original image $x$ of the classification model $C(\cdot)$. In other words, $x' = x - \epsilon \times \text{sign}(\nabla_x J(x, y_t))$. FGSM is used as a baseline attack in some studies because it is a one-step attack that requires less time.

2) BIM
BIM [8] is an iterative attack that builds upon a previously proposed attack [7]. In each iteration, BIM adds a small perturbation of size $\alpha$ and crops the maximum perturbation size $\epsilon$ such that the size of perturbations does not exceed $\epsilon$. In other words,

$$
\begin{align}
x_0' &= x \\
x_N + 1 &= x_N - \alpha \times \text{sign}(\nabla_x J(x_N', y_t)) \\
x_N + 1 &= \text{clip}_{x, x}(x_N + 1)
\end{align}
$$

(1)

3) C&W ATTACK
The C&W attack [10] can effectively attack defensive neural networks against adversarial examples. It applies three attacks such as FGSM [7], iterative FGSM (I-FGSM). The DDN can succeed in the attack with fewer iterations compared with the existing C&W attack, and it can vary $\epsilon$ for each iteration. Further, it is calculated by separating the joint optimization, and it generates a powerful adversarial effect based on the gradient from the classification loss or decreases the $L_2$ loss for each iteration. PerC-AL minimizes the perceptual color distance instead of the $L_2$ loss of the existing DDN.

Besides, many researchers are actively studying the adversarial attack techniques from various perspectives. We introduce several attack methods in terms of imperceptibility. The authors of [28] proposed a superpixel-guided attentional adversarial attack method capable of preserving the local smoothness property along with the original attack ability. It was applied in $L_\infty$ attacks such as FGSM [7], iterative FGSM [8], or momentum iterative FGSM [29] with superpixel algorithm and class activation mapping. They evaluated various adversarial defense methods such as the steganalysis-based detection method and image processing-based defense methods. The authors of [30] proposed FineFool, which can generate the perturbations around the contour region through the channel-spatial and pixel-spatial attention maps. The former focuses on the feature distribution and the latter focuses on the pixel region. They achieved high attack accuracy and verified the small distortion by measuring the $L_2$ distances. The authors of [31] proposed GreedyFool for generating an imperceptible and efficient perturbation map. They used $L_0$ normalization to implement a sparse attack, which is perturbed by only a few pixels, and adopted Generative Adversarial Example (GAN) to generate a small distortion map.

First, the PerC-C&W attack can be formulated as

$$
\text{minimize} \quad ||\Delta E_{00}(x, x')||^2_2 + c \times f(x') \quad (3)
$$

Clearly, this attack substitutes the perceptual color distance calculated through $\Delta E_{00}$ for the original $L_2$ distance and then simultaneously minimizes the distance value and classification loss. $\Delta E_{00}$ is standard formula CIEDE2000 [26], [27], and it can be formulated as

$$
\Delta E_{00} = \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2} + \Delta R,
$$

where $\Delta R = R_T(\frac{\Delta C'}{k_C S_C} + \frac{\Delta H'}{k_H S_H})$.

where $\Delta L'$, $\Delta C'$, and $\Delta H'$ are the distances among the pixel values of lightness, chroma, and hue channel, respectively. $\Delta R$ is an interactive term between the difference of chroma and hue. $\Delta E_{00}$ is more similar to the perceptual features of actual humans than the $L_2$ norm.

Next, the PerC-AL attack is based on the decoupled direction and norm (DDN) [11], that is, $L_2$ norm-based iterative FGSM (I-FGSM). The DDN can succeed in the attack with fewer iterations compared with the existing C&W attack, and it can vary $\epsilon$ for each iteration. Further, it is calculated by separating the joint optimization, and it generates a powerful adversarial effect based on the gradient from the classification loss or decreases the $L_2$ loss for each iteration. PerC-AL minimizes the perceptual color distance instead of the $L_2$ loss of the existing DDN.

4) PERCEPTUAL COLOR DISTANCE (PerC) ATTACK
One study [24] proposed a new approach to realize more imperceptibility with high confidence. It perturbs an image by utilizing perceptual color distance breaks from the existing $L_p$ norm-based constraints in the RGB space. This study proposed two techniques: perceptual color distance C&W attack (PerC-C&W) and perceptual color distance alternating loss (PerC-AL). They generate an adversarial example through $L_2$ normalization in the CIELAB space [26], [27].

H. Na et al.: Adversarial Attack Based on Perturbation of Contour Region
B. STEGANALYSIS-BASED DETECTION METHOD

The steganalysis-based detection method has been applied to detect adversarial examples by using SPAM and SRM feature sets [20]. Each feature set is described follows.

1) SPAM

SPAM is a low-dimensional model with 686 features [21]. It models the pixel difference in eight directions for each pixel by using a two-dimensional Markov chain [32]. When calculating the difference between the right pixel and the pixel \( X_{i,j} \), it first calculates the difference array \( A_{i,j} = X_{i,j} - X_{i,j+1} \). At this point, \( X \) has \((m, n)\) size and \( \{i = 1, \ldots, m, j = 1, \ldots, n - 1\} \) is the pixel index. Next, it calculates the two-dimensional Markov matrix \( M_{3,4} \in \). P \((A_{i,j} = x\mid A_{i,j+1} = y, A_{i,j} = z)\), where each \( r, y, z \) belongs to \( \{-T, \ldots, T\} \), respectively. \( T \) (3 in this study) is a fixed value to reduce the dimensionality of the Markov matrix. If the difference between adjacent pixels is more than \( T \), it is clipped in \([-T, T]\). It finally results in an average sample Markov transition probability matrix \( F \) according to the calculation of the Markov matrix in eight directions, \( \{k = \frac{M^{x}+M^{x}+M^{x}+M^{x}}{4}, \ldots, k = \frac{M^{x}+M^{x}+M^{x}+M^{x}}{4}\} \), where \( k \) is the number of total features. For a two-dimensional Markov matrix, \( k = (2T + 1)^2 \) is calculated. If \( T = 3 \), the number of total features is \( 2k = 686 \).

2) SRM

SRM is a high-dimensional model with 34,671 features [22]. It can be seen as an expanded version of SPAM, and it adds nonlinearity by introducing the concept of MinMax. In SRM, a filter that produces one feature set is called a submodel. It contains 45 submodels based on either SPAM or MinMax, and it extracts a noise residual \( R \) via each submodel filter. Next, it forms a cooccurrence matrix of horizontal and vertical pixels for the extracted \( R \). Then, it eliminates the duplicate values in the cooccurrence matrix by using the symmetry of the image to reduce the dimensionality of the submodel and increase the statistical robustness. Finally, it classifies whether a feature is an adversarial example or not by training an ensemble classifier for the extracted features from each submodel. At this point, [20] is used as the ensemble classifier [33]. The detailed process of the SRM has been described in [22].

III. PROPOSED METHOD

A. MOTIVATION

In Fig.1, the RGB values of adjacent pixels within the body of the object (shark) are very similar. By contrast, the RGB values of adjacent pixels between the object and the background (sea) are very different. The regions are clipped according to threshold (i.e., maximum value) \( T \) when calculating the Markov matrix for extracting the SPAM feature sets. Therefore, the perturbations for these regions can be added without strongly influencing and thereby changing the distribution of the steganalysis features.

In addition, according to [25], adversarial perturbation has higher perceptibility in the low-variance region than in the high-variance region. However, most existing attack methods do not consider this property. Further, perturbations are often added in the low-variance region. However, we can judge the contour region in the high-variance region; therefore, if we perturbed only the contour region, the imperceptibility of these perturbations could be improved.

Algorithm 1 Extracting Contour Region

Input: Input Image \( X \), Threshold \( t \)

Function: \( \text{zeros}(a, b) \): Create a new array of \((a, b)\) shape filled with zeros.

Function: \( \text{clip}(x, \min, \max) \): Clip the elements of \( x \), where values \( < \min \) and \( > \max \) are replaced with \( \min \) and \( \max \), respectively.

Output: \( x_{\text{con}} \): Contour region

1: \( x_{\text{con}} = \text{zeros}(H, W) \)
2: for \( c = 0, 1, 2 \) do
3: \( M = \text{zeros}(H, W) \)
4: for \( row = 1, \ldots, H - 1 \) do
5: \( \text{for col} = 1, \ldots, W - 1 \) do
6: \( x = X[row, col, c] \)
7: \( \text{for direction} = \{\rightarrow, \leftarrow, \downarrow, \uparrow, \downarrow, \uparrow\} \) do
8: \( \text{contour} = |x - x_{\text{direction}}| > t \)
9: \( M[row, col] = \text{contour} \)
10: end for
11: if \( M[row, col] < 2 \) then
12: \( M[row, col] = 0 \)
13: else
14: \( M[row, col] = 1 \)
15: end if
16: end for
17: end for
18: \( x_{\text{con}} = \text{clip}(x_{\text{con}}, 0, 1) \)
19: return \( x_{\text{con}} \)
FIGURE 2. Pipeline of contour attacks. We input the image $x$, extract the contour region $x_{con}$, and fix it in each iteration. We calculate the perturbation $\eta$ in $L_p$ attacks. Then, we calculate $x_{con} \times \eta + x$ and iterate for the output adversarial example $x'$. 

FIGURE 3. Contour region according to threshold $t$. 

B. CONTOUR ATTACK

This study proposes a contour attack. This method adds adversarial perturbations to only the extracted contour regions via a simple algorithm to evade steganalysis-based detection. The pixel values of the original image are preserved in the other regions. This method can be applied to most existing attack methods like $L_\infty$ norm-based attacks including FGSM and BIM and $L_2$ norm-based attacks including C&W and PerC attacks. Fig.2 shows the algorithm of our method. First, the contour region $x_{con}$ for the input image $x$ is extracted. In an iterative attack, the contour region is not updated and the attack is performed on $x_{con}$ extracted in the first iteration. In each iteration, a perturbation is added only in the contour region by multiplying $x_{con}$ with the generated perturbation $\eta$ via $L_p$ attacks. Finally, the output image $x'$ is generated.

1) EXTRACT CONTOUR REGION

Algorithm 1 describes the algorithm for extracting the contour region $x_{con}$. Each pixel of an input image $x$ is three-channel RGB data with a value between [0, 255], where $H$, $W$ are the height and weight of $x$, respectively. To extract $x_{con}$, we calculate the differences with adjacent pixels in eight directions for all pixels for each channel. Here, if any difference is greater than parameter $t$, it is set as 1 and otherwise it is set as 0. Then, if the difference for two out of eight adjacent pixels is greater than $t$, it implies that the given pixel is in the contour region. Additionally, if more than one out of the three RGB channels correspond to the contour region, this pixel is in the contour region. Finally, $x_{con}$ of size of $[H, W]$ is obtained. The value of each pixel in $x_{con}$ is given as follows:

$$x_{con}(i, j) = \begin{cases} 0 & \text{if } x_{con}(i, j) \neq \text{contour region} \\ 1 & \text{if } x_{con}(i, j) = \text{contour region} \end{cases}$$

where $t$ is used as a constant to constrain the range of the contour region, and the contour region $x_{con}$ in Fig.2 is the result of applying $t = 10$. Fig.3 shows $x_{con}$ generated according to each $t$. Clearly, the larger the $t$ value, the narrower is the range of the contour region.

2) APPLY CONTOUR REGION IN $L_p$ ATTACK

To apply various $L_p$ attack methods to the extracted contour region, we can apply them to $\eta_i$ before generating an
adversarial image $x'_i = x + \eta_i$ for each iteration as follows:
\[
x'_i = x + \eta_i \times x_{\text{con}}
\]
where $\eta_i$ is the adversarial perturbation calculated for various attack methods such as $L_\infty$ attack or $L_2$ attack. In Fig.2, the perturbed image $x'_i$ is the output image of Contour FGSM. We can observe that perturbations have been added in the contour region between the face of the shark (i.e., object of $x$) and the sea (i.e., background of $x$). Further, these perturbations are in a high-variance region. Additionally, the output image $x'$ is the output image of Contour BIM; its perturbations are more imperceptible than $x'_i$.

IV. EXPERIMENT
We performed various experiments to evaluate the ability of the proposed contour attack. First, we measured the attack success rate, amount of distortion, and perturbation size to evaluate the attack ability. Second, we measured the detection rate and similarity of steganalysis feature sets to evaluate the detection evasion rate for steganalysis-based detection. Finally, we measured the similarity metric and visualized images to evaluate their imperceptibility. Besides, we additionally tested with the extended size of the ImageNet dataset to generalize the performances of our approach, and compared our proposed method with prior edge extraction methods to evaluate the effectiveness of the attack ability.

A. SETTING
For our experiments, we used the ImageNet dataset [34] that is widely used in computer vision research. It consists of 1,000 classes, in which each image is three-dimensional data resized to $299 \times 299 \times 3$ size. We attacked 11 images from 100 randomly selected classes in this dataset. As the target classification model, we used the pretrained Inception ResNet V2 model [35] when running $L_\infty$ attacks such as FGSM, Contour FGSM, BIM, and Contour BIM and the pretrained Inception V3 model [36] when running $L_2$ attacks such as C&W, Contour C&W, PerC C&W, PerC AL, and Contour PerC AL. The classification accuracies of these classification models were measured to be 95.4% and 88.2%, respectively. The detailed structures of these classification models are described in Table 1. When we attacked the images, each pixel value of the input image $x$ was normalized to $[0, 1]$, and when we extracted the contour region $x_{\text{con}}$, it was denormalized to $[0, 255]$. In our experiments, we set the threshold as $t = 10$.

\[
C_{\text{mean}} = \text{mean}(\Delta E_D(x, x'))
\]

B. ADVERSARIAL ATTACK ABILITY
For our experiments, we first set the perturbation size as $\epsilon = 0.03$ when running the $L_\infty$ attack. Further, we set the one-step perturbation size and iteration number as $\alpha = \epsilon / 10$ and $N = 20$, respectively, when running BIM. By contrast, we set the number of binary search steps, maximum number of iterations, step size, and confidence as $9$, $N = 1,500$, $\alpha = 0.001$, and $k = 0$, respectively, when running an $L_2$ attack like the C&W attack. Finally, we set the step size for minimizing the classification loss, step size for minimizing the perceptual color difference, maximum number of iterations, and confidence as $\alpha = 0.01$, $\alpha_c = 0.05$, $N = 1,500$, and $k = 40$, respectively.

The attack success rate $\text{Acc}_{\text{attack}}$ to evaluate the attack ability is the rate of adversarial example $x'$ that satisfies $C(x) \neq C(x')$. In addition, the amount of distortion $L_0$ to evaluate the attack efficiency is the count of perturbed pixels.

We have used three statistical values to measure the size of the generated perturbations. First, we calculated the $L_2$ and $L_\infty$ distances commonly used to measure the perturbation size in the RGB space. $L_2$ is the overall perturbation size calculated through the Euclidean distance between the channel of each pixel of $x$ and $x'$, and $L_\infty$ is the maximum distortion in the image, that is, the maximum difference at each index of $x$ and $x'$.

Next, we calculated the $C_{\text{mean}}$ distance to measure the perturbation size in terms of the perceptual color difference. It is calculated as follows:

\[
C_{\text{mean}} = \text{mean}(\Delta E_D(x, x'))
\]

where $\Delta E_D(x, x')$ is calculated using (4). It measures the perturbation size in terms of the human color perception more than $L_p$ norm-based values. $C_{\text{mean}}$ is the mean of the perceptual color distance, and it is used to numerically compare the imperceptibility.

To compare the abilities in our experiments, we calculated $L_0$, $L_2$, $L_\infty$, and $C_{\text{mean}}$; these are the means of $L_0$, $L_2$, $L_\infty$, and $C_{\text{mean}}$ for all images, respectively.

C. ADVERSARIAL ROBUSTNESS TO STEGANALYSIS-BASED DETECTION
To evaluate the robustness of the adversarial examples generated through the above processes to steganalysis-based detection, we extracted the SPAM and SRM features for each image. Furthermore, we performed experiments with two machine learning classification models to detect the

---

**TABLE 1.** Structures of target classification model. Inception ResNet V2(left) was used to evaluate the $L_\infty$ attacks, and Inception V3(right) was used to evaluate the $L_2$ attacks. Further details of these classification models can be referred from [35], [36].

| Layer (Input shape) | Layer (Input shape) |
|---------------------|---------------------|
| Inception ResNet V2 | Inception V3 |
| Input(299, 299, 3)  | Input(299, 299, 3)  |
| Conv(299, 299, 3)   | Conv(299, 299, 3)   |
| Conv(147, 147, 32) | Conv(149, 149, 32) |
| Stem(35, 35, 192)   | Padd(147, 147, 32) |
| Conv(35, 35, 200)   | Pool(147, 147, 64)  |
| 10 x Inception(35, 35, 200) | Conv(73, 73, 64) |
| MixedInception(17, 17, 1088) | Conv(73, 73, 80) |
| 20 x Inception(17, 17, 2088) | Conv(35, 35, 192) |
| MixedInception(8, 8, 2080) | Pool(8, 8, 1536) |
| 10 x Inception(8, 8, 2080) | 2 x Inception(8, 8, 1280) |
| Pool(8, 8, 1536)     | Linear(2048) |
| Softmax              | -  |
TABLE 2. Attack ability of different attack methods.

| Method             | Attack Ability | Method             | Attack Ability |
|--------------------|----------------|--------------------|----------------|
| FGSM [7]           |                | Contour FGSM       |                |
| $\text{Acc}_{\text{attack}}$ | 58.2%          | $\text{Acc}_{\text{attack}}$ | 58.6%          |
| $L_0$              | 263,023        | $L_0$              | 136,439        |
| $L_2$              | 15.25          | $L_2$              | 10.74          |
| $L_\infty$         | 0.03           | $L_\infty$         | 0.03           |
| $C_{\text{mean}}$  | 4.213          | $C_{\text{mean}}$  | 2.308          |
| BIM [8]            |                | Contour BIM        |                |
| $\text{Acc}_{\text{attack}}$ | 97.4%          | $\text{Acc}_{\text{attack}}$ | 94.4%          |
| $L_0$              | 222,228        | $L_0$              | 142,218        |
| $L_2$              | 33.34          | $L_2$              | 5.92           |
| $L_\infty$         | 0.03           | $L_\infty$         | 0.03           |
| $C_{\text{mean}}$  | 2.098          | $C_{\text{mean}}$  | 1.220          |
| C&W [10]           |                | Contour C&W        |                |
| $\text{Acc}_{\text{attack}}$ | 100.0%         | $\text{Acc}_{\text{attack}}$ | 100.0%         |
| $L_0$              | 16,064         | $L_0$              | 15,190         |
| $L_2$              | 0.4492         | $L_2$              | 0.5120         |
| $L_\infty$         | 0.0093         | $L_\infty$         | 0.0237         |
| $C_{\text{mean}}$  | 0.071          | $C_{\text{mean}}$  | 0.072          |
| PerC C&W [24]      |                | Contour PerC C&W   |                |
| $\text{Acc}_{\text{attack}}$ | 100.0%         | $\text{Acc}_{\text{attack}}$ | 100.0%         |
| $L_0$              | 19,880         | $L_0$              | 17,649         |
| $L_2$              | 0.5930         | $L_2$              | 0.6614         |
| $L_\infty$         | 0.0281         | $L_\infty$         | 0.0378         |
| $C_{\text{mean}}$  | 0.043          | $C_{\text{mean}}$  | 0.040          |
| PerC AL [24]       |                | Contour PerC AL    |                |
| $\text{Acc}_{\text{attack}}$ | 99.1%          | $\text{Acc}_{\text{attack}}$ | 94.616%        |
| $L_0$              | 126,406        | $L_0$              | 44,015         |
| $L_2$              | 2.9625         | $L_2$              | 4.4015         |
| $L_\infty$         | 0.1207         | $L_\infty$         | 0.2253         |
| $C_{\text{mean}}$  | 0.2940         | $C_{\text{mean}}$  | 0.3            |

D. Perturbation Imperceptibility

We used SSIM [39] to evaluate the imperceptibility of the adversarial perturbation generated through our proposed method. SSIM is a statistical value that is used to measure the perceptual quality or similarity in terms of human color perception, and it measures the degree of distortion in structural information. It is being used as an indicator to evaluate the quality of images or the imperceptibility of perturbations in various fields such as image reconstruction [40], [41] and adversarial examples [42], [43]. The $\text{SSIM}(A, B)$ between image $A$ and image $B$ is expressed as

$$\text{SSIM}(A, B) = l(A, B) \times c(A, B) \times s(A, B)$$

where $l$, $c$, and $s$ are the luminance comparison, contrast comparison, and structure comparison, respectively. They are calculated to utilize the average $\mu$ and standard deviation $\sigma$ of $A$ and $B$ as follows:

$$l(A, B) = \frac{2\mu_A \mu_B + c_1}{\mu_A^2 + \mu_B^2 + c_2}$$

$$c(A, B) = \frac{2\sigma_{AB} + c_2}{\sigma_A^2 + \sigma_B^2 + c_2}$$

$$s(A, B) = \frac{\sigma_{AB} + c_3}{\sigma_A \sigma_B + c_3}$$

V. EXPERIMENTAL RESULTS

A. Adversarial Attack Ability

Table 2 shows the measured attack abilities of each attack method. When evaluating the attack ability of the proposed method, we should first consider the change in performance when perturbations are added only to the contour region. Table 2 shows that $L_0$ of the contour attack methods was much smaller than that of existing attack methods. In particular, for $L_\infty$ attacks, $L_0$ was decreased by more than 100,000, that is, the maintainability of the original image was improved. Further, the attrition rate of $\text{Acc}_{\text{attack}}$ was less than 3% despite the large decrease in $L_0$. For Contour FGSM, the decrease in $\text{Acc}_{\text{attack}}$ was due to the additional constraint, such as the
perturbation region, relative to the existing FGSM. However, Table 3 shows the improvement when the number of iterations was $i = 100$. In other words, we can supplement the decrease in $\text{Acc}_{\text{attack}}$ owing to the contour attack through additional iterations. Further, the $\text{Acc}_{\text{attack}}$ of the contour attack was similar to that of the existing $L_2$ attack methods. In other words, the proposed contour attack maintains the attack success rate and reduces $L_0$.

where $c_1$, $c_2$, and $c_3$ are constraints. The more similar or different two images are, the closer SSIM is to 1 or 0, respectively.

As shown in Table 2, $L_2$ and $C_{\text{mean}}$ of $L_\infty$ norm-based contour attacks were reduced. In other words, the perturbation size of the contour attack was smaller; this proves the attack ability of the $L_\infty$ attack for the contour region. Further, $L_2$ and $L_\infty$ of the $L_2$ norm-based contour attacks were increased. In other words, the perturbation size in the contour region increased because of the decrease in $L_0$. Further, the perturbation size was maintained in terms of human color perception as $C_{\text{mean}}$ was maintained.

### TABLE 3. Attack success rate of BIM-based attacks and detection rate of SPAM feature-based detector with increasing number of iterations.

| Method           | BIM ($i = 20$) | BIM ($i = 100$) | Contour BIM ($i = 20$) | Contour BIM ($i = 100$) |
|------------------|----------------|----------------|-----------------------|------------------------|
| $\text{Acc}_{\text{attack}}$ | 97.4           | 99.6           | 94.4                  | 98.7                   |
| $\text{Acc}_{\text{detect}}$  | 99.9           | 99.9           | 98.8                  | 98.6                   |

**FIGURE 4.** Visualization of SPAM feature set. The graphs (bottom) are the visualizations of the SPAM features of the original image and adversarial images generated by BIM and Contour BIM. The horizontal axis represents each feature index and the vertical axis represents the value corresponding to the feature index.

As shown in Table 2, $L_2$ and $C_{\text{mean}}$ of $L_\infty$ norm-based contour attacks were reduced. In other words, the perturbation size of the contour attack was smaller; this proves the attack ability of the $L_\infty$ attack for the contour region. Further, $L_2$ and $L_\infty$ of the $L_2$ norm-based contour attacks were increased. In other words, the perturbation size in the contour region increased because of the decrease in $L_0$. Further, the perturbation size was maintained in terms of human color perception as $C_{\text{mean}}$ was maintained.

### B. ADVERSARIAL ROBUSTNESS TO STEGANALYSIS-BASED DETECTION

We compared the adversarial robustness to steganalysis-based detection. Table 4 shows the detection rate measured using SPAM features. LDA and EC denote linear discriminant analysis and ensemble classifier, respectively. The result showed that the detection evasion rate of all contour attack methods was improved relative to that of the existing attack methods. For Contour BIM with the EC, the detection rate was 82.1%; this was 17.5% lower than the detection rate of 99.6% of BIM. Further, for Contour PerC AL with the EC, the detection rate was improved by 19.9%. Contour C&W had a detection rate of 51.7%. Binary classification with a classification accuracy of 50% cannot completely classify each class. Therefore, the detection rate of Contour C&W is very meaningful.

**Fig.4** shows visualization graphs extracted by SPAM for adversarial images generated by BIM and Contour BIM attacks for the original image. When compared with the features of the original image, the distribution of features in the image generated by BIM is very different whereas that in the image generated by Contour BIM is very similar.

| Method            | LDA : EC | Method            | LDA : EC |
|-------------------|----------|-------------------|----------|
| FGSM [7]          | 99.3 : 98.3 | Contour FGSM     | 95.4 : 83.1 |
| BIM [8]           | 99.9 : 99.6 | Contour BIM      | 98.8 : 82.1 |
| C&W [10]          | 92.7 : 57.6 | Contour C&W      | 91.2 : 51.7 |
| PerC C&W [24]     | 94.5 : 64.2 | Contour PerC C&W | 91.0 : 52.6 |
| PerC AL [24]      | 98.8 : 91.4 | Contour PerC AL  | 93.2 : 71.3 |

**TABLE 4. Detection rate of SPAM-based detector.**
Therefore, we can evade the SPAM-based detector through the contour attack. Furthermore, Table 5 shows the cosine similarity of SPAM features measured for all images. The cosine similarity is calculated as follows:

\[
similarity(a, b) = \frac{\sum_{i=1}^{n} a_i \times b_i}{\sqrt{\sum_{i=1}^{n} (a_i)^2} \times \sqrt{\sum_{i=1}^{n} (b_i)^2}}.
\]

(11)

It calculates the similarity between vectors \( a \) and \( b \). If their directions are more similar or more different, the cosine similarity is closer to 1 or \( -1 \), respectively. In this experiment, we measured the cosine similarity of the SPAM feature for the original and the adversarial image. As a result, we can verify that contour attack methods have a cosine similarity of almost 1; therefore, they were better than existing attack methods.

Table 6 shows the detection rate measured using SRM features. The detection evasion rate was improved through contour attacks, as in the case of experiments with SPAM features. In particular, with the EC, the detection rate of the Contour C&W attack was 7.1% lower than that of the C&W attack. Table 7 shows the cosine similarity calculated for SRM features in a manner similar to Table 5. The cosine similarities of the contour attack methods were very close to 1, indicating that they were better than existing attack methods.

### C. PERTURBATION IMPERCEPTIBILITY

Table 8 shows a comparison of SSIM for each attack method. All contour attack methods showed higher SSIMs (very close to 1.000) than existing attack methods. In other words, the contour attack improves the perceptual quality of adversarial images and increases the similarity between the original and the adversarial image. Additionally, the SSIM of Contour FGSM was 0.2912 higher than that of FGSM; thus, the contour attack affords high imperceptibility through a one-step attack.

Fig.5 shows the perturbed images via \( L_\infty \) attack methods. We compared the differences with existing attack methods through a very large perturbation size \( \epsilon \) to clearly identify the improved imperceptibility of contour attacks from the viewpoint of human eyes. These images were generated using \( L_\infty \) attacks with \( \epsilon = 0.2 \). The perturbed images generated using Contour FGSM and Contour BIM were clearly cleaner than those generated using FGSM and BIM, respectively. Similarly, Fig.6 shows perturbed images generated using \( L_2 \) attack methods with step size \( \alpha = 0.2 \) and confidence \( k = 100 \) for C&W and PerC C&W-based attacks and \( \alpha_l = 0.1 \), \( \alpha_c = 1 \), and \( k = 70 \) for PerC AL-based attacks. The perturbed images generated using Contour C&W were much cleaner than those generated using C&W. Further, the sea (i.e., background region) in the perturbed image generated using PerC C&W had low quality, whereas that in the perturbed image generated using Contour PerC C&W was improved. Additionally, the perturbations in the left
nostril region of the shark in the perturbed image generated using PerC AL were notably unnatural, whereas those in the perturbed image generated using Contour PerC AL were improved.

Table 2 shows that the $\ell_0$ of the Contour C&W and Contour PerC C&W attacks was decreased in comparison with that of existing attack methods; however, the difference between them was small. Furthermore, $\ell_2$ and $\ell_\infty$ were increased;
for Contour C&W, $C_{mean}$ was increased to 0.001. Fig. 7 explains why the imperceptibility of the perturbation was increased. This figure shows the perturbation $\delta$ generated by $L_2$ attack methods. C&W-based attack methods destroy the object in an image by adding relatively small perturbations to a smooth region to induce misclassification. In particular, PerC C&W added many perturbations in the abdomen part of the shark, thereby destroying the smooth region. Our contour attack prevents this problem. In principle, it is relatively hard to induce a misclassification by adding a perturbation to a high-variance region than to a low-variance region. However, doing so is advantageous from the viewpoint of imperceptibility. Overall, this explains why $L_2$, $L_\infty$, and $C_{mean}$ of $L_2$ norm-based contour attacks were decreased relative to those of existing attack methods and why the imperceptibility was improved.

### D. COMPARISON ON EXPANSION DATASET

In this subsection, we discuss the verification of the generality of the result presented in the above section. We used 5 images from 1,000 classes in the ImageNet dataset. In other words, the number of attack images increased over two times when compared with the above experiments. Furthermore, we attacked these images by utilizing only C&W attacks with Contour C&W attacks owing to our restricted computing resource. Table 9 shows the measured attack abilities in our additional experiments. We first evaluated the attack success rate $Acc_{attack}$. The result showed that $Acc_{attack}$ of C&W attack was slightly decreased from 100% to 99.8%. However, $Acc_{attack}$ of the Contour C&W attack was maintained at 100%. Next, we detected these adversarial images using steganalysis-based detection with LDA. Based on the results of the experiment, we can recognize that the contour attack is able to evade the steganalysis-based detection more effectively than the existing attack method. Especially, even when most detection rates were increased, the contour attack could evade detection better than the existing method.

### E. COMPARISON WITH OTHER CONTOUR REGION DETECTION METHODS

We additionally compared with some existing contour region detection methods such as Sobel filter [44], Scharr filter [45], and Canny filter [46]. The Sobel filter was calculated using particular matrices based on the principle of approximate derivative. Furthermore, the Scharr filter can calculate a more accurate derivative than the Sobel filter. These matrices have a kernel of size $3 \times 3$. We generated a contour region by overlaying the horizontal-based matrix with the vertical-based matrix. The Canny filter is a multi-step algorithm to extract a more adequate edge. First, the noise is refined using a Gaussian filter, and then the edge region is extracted using a Sobel filter. Finally, a threshold is constrained to remove
The visualization of the examples of the contour region using the various contour extraction methods is shown in Fig. 8. The difference between them and our contour region is that our approach is based on the SPAM such as calculation by comparing between eight directions for each pixel, and is very simple. Table 10 shows the results of our comparison. We evaluated the performances of $L_\infty$ attacks. Our approach shows lower attack accuracy in Contour FGSM than the Canny filter. However, it can overcome this drawback through the iterative version. On the other hand, the Canny filter-based contour attack shows the best evasion success rate for SPAM-based detection due to the least amount of distortion. Additionally, the Sobel filter-based contour attack can evade the SRM-based detection to a greater extent than the other methods. Consequently, we can conclude that the attacker should determine the most appropriate method to generate a contour region according to each situation.

VI. DISCUSSION
The proposed contour attack has some benefits and limitations. First, we successfully improved the local smoothness of the smooth region. However, the local smoothness between adjacent contour regions may be destroyed. We added
perturbations in the contour region without requiring any sophisticated mathematical calculations by simply combining existing attack methods with our method. However, this occasionally resulted in differences in RGB values between adjacent perturbations.

Second, we performed experiments with other datasets such as MNIST [47], which contains 28 × 28 gray-scale images of handwritten digits that are commonly used for training various image classification tasks. These image data are smaller than those of ImageNet. We performed experiments similar to those described above to evaluate the success of the contour attack for small-sized data. Specifically, we performed comparisons for \( L_\infty \) attacks with a perturbation size of \( \epsilon = 0.3 \). Fig.9 shows the experimental results. Our contour attacks seem cleaner than existing attack methods. In other words, the contour attack can improve the imperceptibility of adversarial perturbations even for small-sized data. However, in our experiments, the attack ability was decreased too much for small perturbations with \( \epsilon < 0.03 \). This was because the contour region became too cramped and overlapped the original image, as shown in Fig.9. Further mathematical research needs to be conducted to solve this problem.

Third, we discussed a contour region. Recently, the contour region is paid attention in the area of adversarial example. For example, [30] proposed FineFool which generates adversarial example using spatial attention map. Their goal is similar with this paper, i.e., they focus on contour region to reduce the amount of perturbation and to improve the imperceptibility of perturbations. However, the FineFool needs to calculate the channel-spatial attention map and the pixel-spatial attention map so that they should have computation cost more than our approach because our contour attack has only cropped the existing attack method with contour region. Additionally, our approach can apply with variable attack methods. Next, [48] proposed RCA-SOC to defend the adversarial examples via refocusing on critical areas and strengthening object contours. They used a feature map created based on the shallow layer of classification model to reconstruct the input image. Specifically, they extracted a pixel channel attention map to focus on the critical feature area and extracted pixel plane attention map to strengthen the contour of the object. Additionally, they combined the RCA-SOC with some input modification methods to enhance the performance of defense. We need to take an interest in this method since the focus of their defense is the contour region that is similar with our approach.

VII. CONCLUSION

This study proposed a methodology to improve adversarial attacks by applying existing attack methods to contour regions extracted using a simple algorithm. Several existing attack methods greatly disrupt an image or add easily detectable perturbations in a region. Our experiments indicate that the proposed contour attack can solve these problems. We also demonstrated that the contour region can serve as a new perturbation space for performing effective adversarial attacks. Our proposed method can evade steganalysis-based detectors that defend against various attack methods by using the local smoothness property. Our method demonstrated an up to 19.9% higher detection evasion rate. Furthermore, it could reduce the amount of distortion, maintain the attack ability, and improve the imperceptibility of adversarial perturbations relative to existing attack methods. Nonetheless, our contour attack method has some limitations, as mentioned in Section VI. We will aim to overcome these in future studies.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.
[2] V. Albiero, K. W. Bowyer, K. Vangara, and M. C. King, “Does face recognition accuracy get better with age? Deep face matchers say no,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Mar. 2020, pp. 261–269.
[3] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, “A survey of deep learning techniques for autonomous driving,” J. Field Robot., vol. 37, no. 3, pp. 362–386, Apr. 2020.
[4] M. M. A. Monshi, J. Poon, and V. Chung, “Deep learning in generating radiology reports: A survey,” Artif. Intell. Med., vol. 106, Jun. 2020, Art. no. 101878.
[5] A. Serban, E. Poll, and J. Visser, “Adversarial examples on object recognition: A comprehensive survey,” ACM Comput. Surv., vol. 53, no. 3, pp. 1–38, Jul. 2020.
[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” 2013, arXiv:1312.6199. [Online]. Available: http://arxiv.org/abs/1312.6199
[7] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” 2014, arXiv:1412.6572. [Online]. Available: http://arxiv.org/abs/1412.6572
[8] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial machine learning at scale,” 2016, arXiv:1611.01236. [Online]. Available: http://arxiv.org/abs/1611.01236
[9] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, “DeepFool: A simple and accurate method to fool deep neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 2574–2582.
[10] N. Carlini and D. Wagner, “Towards evaluating the robustness of neural networks,” in Proc. IEEE Symp. Secur. Privacy (SP), May 2017, pp. 39–57.
[11] I. J. Goodfellow, L. G. Hafemann, L. S. Oliveira, L. B. Aced, R. Sabourin, and E. Granger, “Decoupling direction and norm for efficient gradient-based L2 adversarial attacks and defenses,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 4322–4330.
[12] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” 2017, arXiv:1706.06083. [Online]. Available: http://arxiv.org/abs/1706.06083
[13] A. Raghunathan, J. Steinhardt, and P. Liang, “Certified defense against adversarial examples,” 2018, arXiv:1801.09344. [Online]. Available: http://arxiv.org/abs/1801.09344
[14] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, “Distillation as a defense to adversarial perturbations against deep neural networks,” in Proc. IEEE Symp. Secur. Privacy (SP), May 2016, pp. 582–597.
[15] J. Buckman, A. Roy, C. Raffel, and I. Goodfellow, “Thermometer encoding: One hot way to resist adversarial examples,” in Proc. Int. Conf. Learn. Represent., 2018, pp. 1–22.
[16] A. Demontis, M. Melis, M. Finesso, and F. Roli, “Why do adversarial attacks transfer? Explaining transferability of evasion and poisoning attacks,” in Proc. 28th USENIX Secur. Symp. (USENIX Secur.), 2019, pp. 321–338
[17] F. Tramer, N. Carlini, W. Brendel, and A. Madry, “On adaptive attacks to adversarial example defenses,” 2020, arXiv:2002.08347. [Online]. Available: http://arxiv.org/abs/2002.08347
[18] B. Liang, H. Li, M. Su, X. Li, W. Shi, and X. Wang, “Detecting adversarial image examples in deep neural networks with adaptive noise reduction,” IEEE Trans. Depend. Secur. Comput., vol. 18, no. 1, pp. 72–85, Jan. 2021.
[29] J. Fridrich and J. Kodovský, “Rich models for steganalysis of digital images,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 3, pp. 868-882, Jun. 2012.

[30] J. Lu, T. Issaranon, and D. Forsyth, “Safetynet: Detecting and rejecting adversarial examples robustly,” in Proc. IEEE Int. Conf. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 1039-1048.

[31] B. Luo, Y. Liu, L. Wei, and Q. Xu, “Towards imperceptible and robust adversarial example attacks against neural networks,” in Proc. AAAI Conf. Artif. Intell., 2018, vol. 32, no. 1, pp. 1652-1659.

[32] P. Alessi, M. Brill, J. C. Acosta, E. Carter, R. Connelly, J. F. Decarreau, R. Harold, R. Hirschler, B. Jordan, C. Kim, and D. McDowell, “Colorimetry—Part 6: CIEde2000 colour-difference formula,” in Proc. ISO/CIE, 2014, Art. no. 11664.

[33] M. R. Luo, G. Cui, and B. Rigg, “The development of the CIE 2000 colour-difference formula: CIEDE2000,” Color Res. Appl., vol. 26, no. 5, pp. 340-350, 2001.

[34] X. Dong, J. Han, D. Chen, J. Liu, H. Bian, Z. Ma, H. Li, X. Wang, W. Zhang, and N. Yu, “Robust superpixel-guided attentional adversarial attack,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 12895-12904.

[35] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, “Boosting adversarial attacks with momentum,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 9185-9193.

[36] J. Chen, H. Zheng, H. Xiong, R. Chen, T. Du, Z. Hong, and S. Ji, “FineFool: A novel DNN object contour attack on image recognition based on the attention perturbation adversarial technique,” Comput. Secur., vol. 104, May 2021, Art. no. 102220.

[37] X. Dong, D. Chen, J. Bao, C. Qin, L. Yuan, W. Zhang, N. Yu, and D. Chen, “GreedyFool: Distortion-aware sparse adversarial attack,” 2020, arXiv:2010.13773. [Online]. Available: http://arxiv.org/abs/2010.13773

[38] K. Sullivan, U. Madhow, S. Chandrasekaran, and B. S. Manjunath, “Steganalysis of spread spectrum data hiding exploiting cover memory,” in Proc. SPIE, vol. 5681, pp. 38-46, Mar. 2005.

[39] J. Kodovský, J. Fridrich, and V. Holub, “Ensemble classifiers for steganalysis of digital media,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 432-444, Apr. 2012.

[40] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248-255.

[41] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, Inception-ResNet and the impact of residual connections on learning,” in Proc. AAAI Conf. Artif. Intell., vol. 31, no. 1, 2017, pp. 4278-4284.

[42] 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. 2819-2828.

[43] S. Balakrishnan and A. Ganapathiraju, “Linear discriminant analysis—A brief tutorial,” Inst. Signal Inf. Process., vol. 18, pp. 1-8, Mar. 1998.

[44] R. A. Fisher, “The use of multiple measurements in taxonomic problems,” Ann. Eugenics, vol. 7, no. 2, pp. 179-188, 1936.

[45] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600-612, Apr. 2004.

[46] Y. Gao, L. Gao, and X. Li, “A generative adversarial network based deep learning method for low-quality defect image reconstruction and recognition,” IEEE Trans. Ind. Informat., vol. 17, no. 5, pp. 3231-3240, May 2021.

[47] J. Zhang, Z. Liu, S. Zhang, H. Zhang, P. Spincemaille, T. D. Nguyen, M. R. Sabuncu, and Y. Wang, “Fidelity imposed network edit (FINE) for solving ill-posed image reconstruction,” Neuroimage, vol. 211, May 2020, Art. no. 116579.

[48] P. Quan, R. Guo, and M. Srivastava, “Towards imperceptible query-limited adversarial attacks with perceptual feature fidelity loss,” 2021, arXiv:2102.00449. [Online]. Available: http://arxiv.org/abs/2102.00449

[49] Y. Xiao, C.-M. Pun, and B. Liu, “Adversarial example generation with adaptive gradient search for single and ensemble deep neural network,” Inf. Sci., vol. 502, pp. 147-167, Aug. 2020.

[50] O. R. Vincent and O. Folorunso, “A descriptive algorithm for Sobel image edge detection,” in Proc. Inf. Sci. ET Educ. Conf. (InSITE), vol. 40, 2009, pp. 97-107.

[51] H. Scharf, “Optimal operators in digital image processing,” Ph.D. dissertation, Interdisciplinary Center Sci. Comput., Heidelberg Univ., Heidelberg, Germany, 2000.

[52] J. Camny, “A computational approach to edge detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986.

[53] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.

[54] J. Chen, H. Zheng, R. Chen, and H. Xiong, “RCA-SOC: A novel adversarial defense by refocusing on critical areas and strengthening object contours,” Comput. Secur., vol. 96, Sep. 2020, Art. no. 101916.

---

**HYUNSIK NA** received the B.S. degree in applied mathematics and the M.S. degree in computer science from Konju National University, South Korea, in 2016, respectively. He is currently pursuing the Ph.D. degree in software convergence with Soongsil University, South Korea. His research interests include adversarial attacks, information security, and edge AI.

**GWONSSANG RYU** received the B.S. degree in applied mathematics and the M.S. degree in convergence science from Kongju National University, South Korea, in 2016 and 2018, respectively. He is currently pursuing the Ph.D. degree in software convergence with Soongsil University, South Korea. His research interests include adversarial attacks, adversarial defense, user authentication, and anomaly detection.

**DAESEON CHOI** (Member, IEEE) received the B.S. degree in computer science from Dongguk University, South Korea, in 1995, the M.S. degree in computer science from Pohang Institute of Science and Technology, South Korea, in 1997, and the Ph.D. degree in computer science from Korea Advanced Institute of Science and Technology (KAIST), South Korea, in 2009. He was a Professor with the Department of Medical Information, Konju National University, South Korea, from September 2015 to August 2020. He is currently a Professor with the Department of Software, Soongsil University, South Korea. His research interests include identity management and information security.

---

**VOLUME 9, 2021**

122321