IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 25, 2020, accepted June 29, 2020, date of publication July 3, 2020, date of current version July 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3006980

Robust Spherical Panorama Image Watermarking Against Viewpoint Desynchronization

JIHYEON KANG[1], JONG-UK HOU[2], (Member, IEEE), SANGKEUN JI[3], AND HEUNG-KYU LEE[3]

[1]Graduate School of Information Security, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea
[2]School of Software, Hallym University, Chuncheon 24252, South Korea
[3]School of Computing, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea

Corresponding author: Heung-Kyu Lee (heunglee@kaist.ac.kr)

This work was supported in part by NRF under Grant NRF-2019R1A2C2084569, and in part by IITP, Development of High Reliability Elementary Image Authentication Technology for Smart Media Environment, funded by the Korean Government (MSIT) under Grant 2017-0-01671.

ABSTRACT Although quantitative and qualitative growth of spherical panorama content has been achieved, watermarking methods for copyright protection are still insufficient. The spherical panorama image watermarking technique must be able to detect a watermark from the viewed images, but it is difficult to detect the watermark directly from the images due to various distortions. We propose a method for recovering the watermark-detection target viewed image to an equirectangular-formed source image and detecting the watermark after recovery. For recovery, we propose combining the scale-invariant feature transform point matching and the shift and rotation estimation method using a Euclidean transformation matrix. For watermark embedding and detection, a discrete Fourier transform magnitude coefficient is used. With the shift-invariant characteristic, our proposed method is robust against viewpoint desynchronization. Because finding an accurate viewpoint is difficult, our method offers a substantial advantage in the spherical panorama image watermarking scheme. Furthermore, it allows watermark detection on a continuous viewpoint spherical panorama system. In addition, our experiments reveal that our method is robust against signal attacks, such as JPEG compression, Gaussian filter blurring, noise addition, and histogram equalization.

INDEX TERMS 360 VR, block desynchronization, image watermarking, omni-directional, spherical panorama, viewpoint desynchronization.

I. INTRODUCTION

Spherical panorama contents provide users with the freedom of the direction the users desire to see, giving the user the experience of being in a virtual space. For this reason, spherical panorama content is also called omni-directional content or 360° virtual reality. There are two methods to create a virtual space. The first method is to create a space using three-dimensional (3D) computer graphic technology. The second method is to capture an existing space using a digital camera and render it into virtual space. In this paper, the spherical panorama uses the second method of creating virtual space. The panorama in this paper is a spherical panorama that can adjust the view both vertically and horizontally unlike a cylindrical panorama, which can only move the view from side to side.

Many people have already created, distributed, and enjoyed panorama content, and the types of content are becoming more diverse, such as concerts, famous tourist attractions, and movies. Panorama content is rapidly increasing qualitatively and quantitatively, and it offers people a new experience that the existing content cannot provide. Although a considerable amount of hardware and software support exist, producing spherical panorama content is expensive. However, compared to the panorama content industry and the prospects of the future, an adequate technique for copyright protection of the content has not yet been established.

Digital watermarking is a technique that inserts copyright information into multimedia content. When a copyright dispute occurs, a watermark can be used to check the authorship and can also be used as a fingerprint of the distribution path.

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

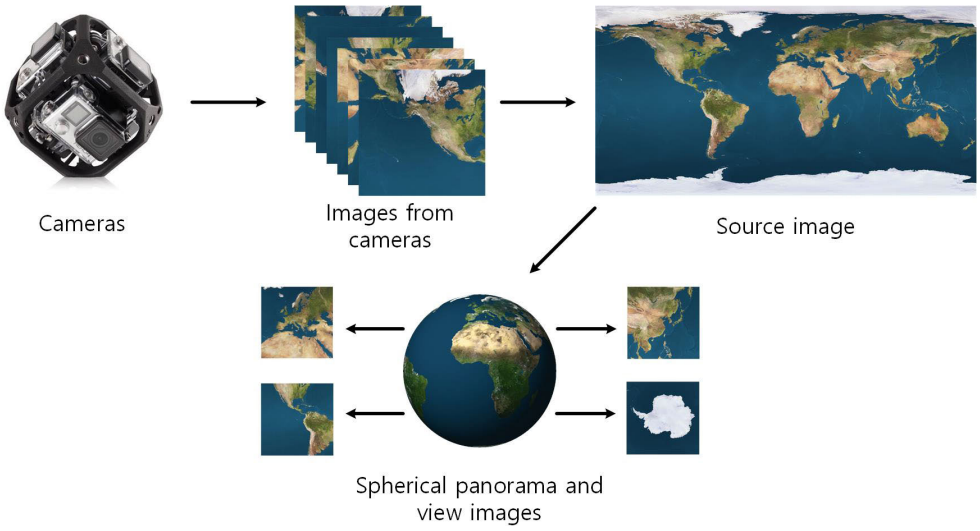


FIGURE 1. Spherical panorama contents generation process.

Digital watermarking techniques have been applied to new types of content such as stereoscopic 3D (S3D) [1], depth-image-based rendering (DIBR) [2]–[4], and 3D mesh [5]–[8]. However, the number of watermarking techniques for spherical panorama content is very low, and many of them do not even have a suitable watermarking scenario. Moreover, it is not easy to apply existing watermarking techniques to panorama content.

In panorama content, the image viewed by the user is called the view image (VI). A content provider provides a dynamically changed VI to the user according to requests. To dynamically make VIs, the content provider should have a source image (SI), which is a source of the VIs. Due to the characteristics of the panorama content, the VI that is exposed to the user is more likely to be leaked than the SI, and the VI alone can have sufficient copyright value. The distortion between the SI and VI is so severe that synchronization between the watermark embedding and detection region is difficult.

This paper proposes a spherical panorama image watermarking technique. For the synchronization between the location of the embedding and the location of detection, a watermark is embedded on the SI, and watermark detection is conducted on the recovered SI, which is obtained from the watermark-detection target VI. We propose using the use of a scale-invariant feature transform (SIFT) [9] feature-point matching algorithm and a shift and rotation estimation method using the Euclidean transformation matrix on the recovery process.

This paper is an extension of our previous work [10], in which we proposed a spherical panorama image watermarking using viewpoint detection based on the discrete cosine transform (DCT) domain. In this extension, we change to the discrete Fourier transform (DFT) domain and change the embedding and detection method for robustness against viewpoint desynchronization and block desynchronization. In addition, a more detailed explanation of spherical panorama and several experiments are added. Furthermore, we present an additional comparative experiment between other spherical panorama watermarking methods.

The main contributions of this paper are as follows. We propose:

- The prerequisites for the spherical panorama watermarking method, as well as the first method that satisfies the prerequisites to enable watermark detection from random viewpoint VI.
- A method to bypass the rendering distortion of spherical panorama in the watermark-detection process.
- A robust watermarking method against horizontal and vertical viewpoint-detection failure and block desynchronization in the watermark-detection process.
- A method that can detect watermarks not only in a discrete but also in a continuous viewpoint system of a spherical panorama image.

This paper is outlined as follows. First, we explain the spherical panorama and its watermarking scenario in Section 2. Section 3 reviews related work involving image watermarking for various content. We discuss a proposed spherical panorama image watermarking algorithm in Section 4, and Section 5 presents the experimental results. Finally, future research directions are proposed and the paper is concluded in Section 6.

II. SPHERICAL PANORAMA AND WATERMARKING

To create spherical panorama content, it is possible to directly create a SI using a dedicated camera, but to create high-quality panorama content, pictures taken with multiple cameras aimed at different places from the same location are usually used, as shown in Fig. 1. Using the stitching technique

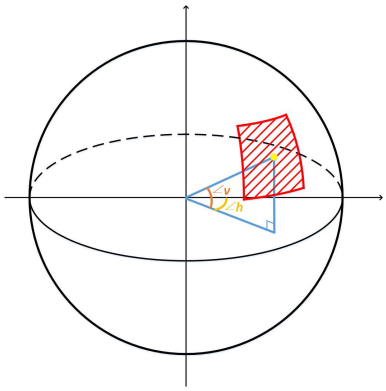


FIGURE 2. Source image to view image rendering according to vertical and horizontal viewpoint.

that connects the images, a single SI is generated from several pictures, and then the SI is rendered to create a VI that meets the user's requirement.

Rendering a spherical panorama image is equivalent to viewing the outer surface of the sphere from the center of the sphere created by the SI, as shown in Fig. 2. In other words, the SI should be able to express the spherical surface. It is possible to store the information of a sphere in a spherical coordinate of three dimensions, but primarily, 2D images representing spheres are used as SIs due to capacity and unstructured format problems.

The most frequently used SI format for spherical panoramas is the equirectangular form. In an equirectangular-formed image, the width and height are the longitude and latitude of the sphere. Therefore, the equirectangular-formed image width is always double that of its height. The world map in Fig. 1 is the typical equirectangular-formed image that represents a sphere (the Earth) in a 2D image.

All 2D images that represent a sphere have distortions. In an equirectangular-formed image, a small distortion occurs near the equator regions, but the polar regions have more distortion. Fig. 3 illustrates distortion between the SI and VI. The images in the upper row display the VIs from different vertical viewpoints using the world map as the SI. The red circles in the VIs demonstrate how the same region changes as the user's viewpoint and the scaling and rotation are included in the distortion. The images in the lower row show that the VIs from the cross stripes SI. Each VI is a small part of the SI; in other words, it has a strong cropping. Furthermore, the lower row images in Fig. 3 reveal a dynamic warping distortion.

In the case of a spherical panorama, the content distributor renders the SI to create the requested VI and provides the VI to the user. While the VI alone should be the target of copyright protection, unfortunately, malicious users may leak the VI(s). For example, if a concert is provided as panorama content, the VIs that reveal the stage must have copyright protection. In addition, nearly the same panorama content can be created with only some of the leaked VIs. Therefore, the spherical panorama watermarking technique should be able to detect the watermark in the VIs and SIs. It is possible to embed a watermark directly in real-time into the VIs before distributing them to the user. However, because motion-to-photon delay—the delay from the user's viewpoint change request to the VI change—is very critical in spherical panorama content, real-time watermark embedding on VIs is not a suitable option. In other words, it is necessary to embed a watermark in the SI and to be able to detect the watermark in the VI. However, as Fig. 3 reveals, the distortion between the SI and VI includes dynamic warping, scaling, cropping, and rotation; therefore, it is impossible to directly detect the watermark in the VI. To solve this problem, we propose to recover the VI to an equirectangular-formed image. The difficulty of spherical panorama image watermarking can be expressed in three ways.

First, because the user chooses a viewpoint to generate the VI, the VI from each viewpoint must include the watermark information. This means the watermark should be robust against extreme cropping attacks. In this paper, we propose dividing the SI into appropriate-sized blocks and embedding the same watermark signal into all blocks.

Second, viewpoint information is necessary for the recovery process. In other words, we need to know which vertical viewpoint (latitude) and horizontal viewpoint (longitude) are used in the rendering process of the SI to the VI. If the viewpoint information is known, we can obtain the form of rendering distortion, making it possible to recover part of the SI from the VI. With the recovered SI, we do not need to consider warping, scaling, or rotation to detect the watermark. However, the problem is that no viewpoint information exists in the VI. Unfortunately, comparing all the VIs that come from every vertical and horizontal viewpoint of the target VI takes too much time and is resource-consuming work. To solve this problem, we propose finding the viewpoint of the VI using the SIFT point matching and the shift and rotation estimation method using the Euclidean transformation matrix. Furthermore, we propose a watermarking method that is robust against viewpoint desynchronization. In our watermarking scheme, even if finding the complete viewpoint detection fails, watermark detection for continuous viewpoint remains possible.

Lastly, due to the effects of distortion in the process of spherical panorama rendering and in the recovery process of filling in insufficient image pixel information with interpolation, the quality of the recovered SI is much lower than the original SI. Thus, the spherical panorama watermarking method should be robust against quality degradation. To overcome the robustness problem, the frequency domain is used to embed and detect the watermark. We propose to use the DFT magnitude coefficients. Because of the DFT magnitude coefficient's shift-invariant characteristic, the robustness of the proposed watermarking scheme against viewpoint desynchronization is both achievable and offers a considerable advantage in the spherical panorama watermarking method, because finding an accurate viewpoint is otherwise challenging.

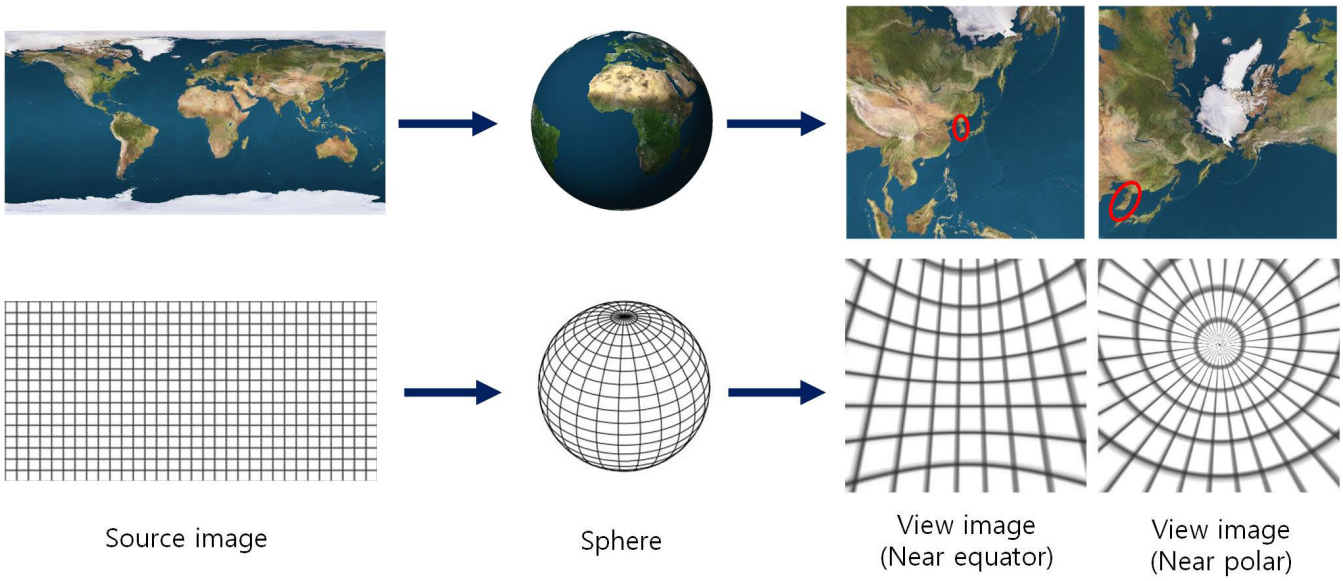


FIGURE 3. The dynamic distortion of a view image according to its viewpoint.

III. RELATED WORK

Various types of image content include 2D, S3D, DIBR, 3D mesh, and spherical panorama images. Watermarking methods have been proposed for various image contents, and because the various types of content each have specific characteristics, a watermarking technique should be designed in consideration of these characteristics.

A. 2D IMAGE WATERMARKING

To date, numerous watermarking techniques have been proposed for 2D images. Frequency domains, such as the DCT [11]–[14], DFT [15]–[17], discrete wavelet transform (DWT) [12], [18], and contourlet [19] are often used for invisibility and robustness. The spread spectrum [20] and quantization index modulation [21] methods, among others, are used. Furthermore, many 2D image watermarking techniques use a template [22] or feature points [23] for robustness against geometric distortion. Recently, 2D image watermarking techniques have been introduced that use deep learning [24]–[26].

B. STEREOSCOPIC 3D AND DIBR

For 3D images, two kinds of formats exist for 3D content distribution: S3D and DIBR. S3D uses two images (left and right image) to get one view and S3D image watermarking schemes emphasize the human visual system to reduce visual fatigue in the 3D rendering process [1]. S3D and DIBR both use two images, but DIBR uses a center image and a depth image; DIBR generates left and right images using the center and depth images. Therefore, DIBR can control the degree of depth in an image. DIBR watermarking schemes [2]–[4] should consider a variable depth depending on user preferences. Both DIBR and spherical panorama image watermarking schemes should detect the watermark in a changing image according to the setting, but the degree of change differs. DIBR rendering can only make local horizontal translations; therefore, it is not adjustable to spherical panoramas.

C. 3D MESH

A mesh is a collection of polygonal facets that represent a digital approximation of the surface of a real 3D object. Currently, 3D mesh models are widely used in 3D printing and scanning, medical imaging, virtual reality, autonomous vehicles, and so on. Since the first publication of a 3D model watermarking method in 1997, various kinds of 3D model watermarking methods have been studied [5]. The statistical features of the vertex norm [6] and spectral analysis-based watermarking [7] are commonly used techniques. Due to the advent of 3D printing applications, there are huge demands to secure the data from 3D object sharing platforms such as Thingiverse, and Pinshape [27]. For this reason, Hou *et al.* [8] proposed a 3D watermarking scheme to protect 3D data in this environment. In addition, primitives of 3D mesh watermarking have the potential to be linked with 360° content and its applications.

D. SPHERICAL PANORAMA

In the case of spherical panoramas, which have recently entered into the spotlight, watermarking techniques are uncommon. Miura *et al.* proposed a data-hiding technique for omni-directional images [28], but they did not consider robustness because they focused on hiding information. Furthermore, they only used equirectangular images and did not deal with VIs. Kang *et al.* proposed spherical panorama image watermarking using feature points [29]. They embedded watermarks into several original images before combining them into one watermarked equirectangular image. In this

way, they could detect a watermark in the equirectangular image. However, protecting only the equirectangular image can be performed using existing 2D image watermarking techniques. Several original images taken from a specific position at a specific time to make one equirectangular image are usually taken by one person. Therefore, a different watermark is not needed for each original image. Furthermore, the VI can easily be stolen, whereas the SI, which is mostly equirectangular formed image, is relatively difficult to steal due to the characteristics of the spherical panorama. Therefore, we embed watermarks into SIs and detect them in VIs. Baldoni *et al.* proposed a spherical panorama image watermarking [30] in the domain of the N-level DWT of the block. They adopted the watermarking scenario from our previous paper, embedding the watermark on an equirectangular-formed SI and detecting the watermark in the recovered SI from the target VI. However, they recovered the SI from the VI under the assumption that they knew the viewpoint information. Thus, because it is impossible to know the viewpoint information of the VI without any process, their method is impossible to use in the real world.

IV. SPHERICAL PANORAMA IMAGE WATERMARKING

This paper presents a watermarking algorithm for spherical panorama images. As mentioned in the introduction, a spherical panorama watermarking algorithm should be able to detect a watermark in both the SI and VI. To overcome the difficult synchronization between the SI and VI, we propose to recover the detection-targeted VI to an equirectangular formed image.

Unlike the DCT domain used in our previous method [10], the DFT magnitude coefficient is used for watermarking in this paper. Although DCT domain-based watermarking is better in non-attack or signal attack cases, the shift-invariant characteristics of the DFT magnitude bring a great advantage in spherical panorama image watermarking. Using shift-invariant characteristics, a watermark can be robust against viewpoint desynchronization. Finding a rough viewpoint rather than a completely accurate viewpoint means that watermark detection is possible even in a continuous viewpoint.

A. WATERMARK EMBEDDING

1) MULTIPLE BLOCKS

We propose to embed a watermark into the equirectangular formed SI and to detect watermarks from recovered SIs from the VI. This is similar to defending against a random extreme cropping attack. We want at least one complete watermark pattern in the VI regardless of the viewpoint, which we achieve by dividing the SI into $n \times 2n$ blocks and embedding the same watermark into each of the blocks. If n is too small, the VI cannot include all of the watermark information; if n is too large, it is difficult to make the watermark robust. Therefore, the block size must not be too big or too small. In the experiments of this paper, the SI (1024×2048) is divided into 4×8 blocks so that each block has a size of 256×256. The size of each block should be determined in consideration of the resolution of the SI.

2) EMBEDDING A WATERMARK PATTERN

In the previous work [10], we used a basic DCT-based spectrum watermarking method. However, in this paper, we propose to use the magnitude coefficient of the Fourier domain. We used the DFT domain because of the shift-invariant characteristics of the magnitude coefficient. As mentioned in the previous section, because we embedded the same watermark into each block and because of the shift-invariant characteristics of the DFT, if the block size is maintained, the watermark detection is possible even if the block location synchronization is incorrect. We primarily applied the method from [16] among the DFT domain based watermarking methods because of its flexibility concerning the watermark frequency range selection and length of the insertion information.

When a block is selected, we transform the color space from RGB to YCbCr. Then, we obtain only the luminance (Y) image of the block. After that, the Y-channel image is transformed into the Fourier domain using DFT. Then, we obtain the magnitude coefficient of the Fourier domain and the low frequency magnitude coefficient, which are moved to the center. With this, we can control the frequency range of the watermark.

A secret key (K) is used to make a watermark binary pattern (b) that is a random binary sequence. The security of the watermark can be improved by using a secure chaotic cryptography method [31] and information theory-based analysis and method [32]. The length of b is proportional to the size of a block and the radius of the circle r that control the frequency of the watermark. Equation 1 indicates the elements of the calculation of the watermark matrix.

$$W(x_i, y_i) = b(j)\left[\frac{1}{9}\sum_{s=-1}^{1}\sum_{t=-1}^{1} M(x_i + s, y_i + t)\right] \quad (1)$$

In Eq. (1), $W(X_i, Y_i)$ is an element of the watermark matrix, and $b(j)$ denotes the j'th element of the binary vector b. The $M(x_i, y_i)$ is an element of the magnitude of the Y-channel of the DFT coefficient of the block image.

The coordinates (x_i, y_i) are calculated using the following equations (Eqs. (2) and (3)):

$$x_i = \left(\frac{m}{2} + 1\right) + \left\lfloor r \cdot \cos\left(\frac{j \cdot \pi}{l}\right)\right\rfloor \quad (2)$$

$$y_i = \left(\frac{n}{2} + 1\right) + \left\lfloor r \cdot \sin\left(\frac{j \cdot \pi}{l}\right)\right\rfloor \quad (3)$$

where m and n denote the size of the matrix M, r is implementation radius of the circle that determines the frequency range, and l denotes the length of the random binary vector b. The elements of the watermark are placed at equal intervals from the center of the W. Due to the characteristic of the DFT, half of the circle is used, and the other half is symmetrically

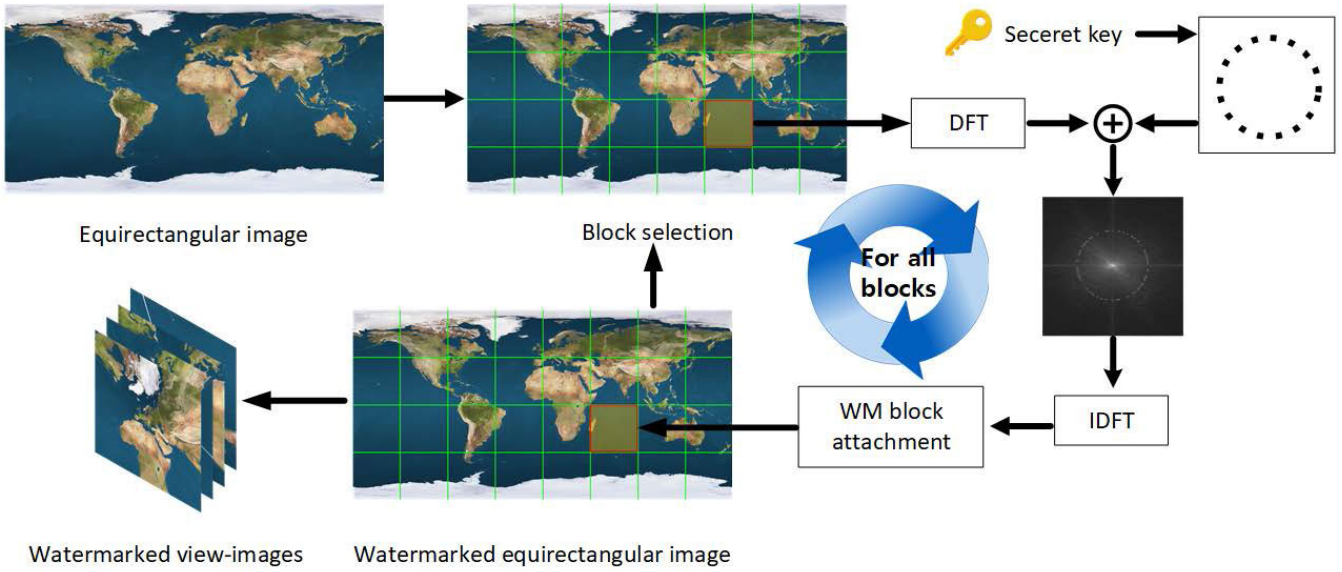


FIGURE 4. Watermark embedding process.

inserted with the same information. Then, W is embedded in the M using Eq. (4).

$$M_w(x, y) = M(x, y) + \alpha \cdot W(x, y) \tag{4}$$

In Eq. (4), α denotes the strength of the watermark and controls the trade-off between invisibility and robustness. The *alpha* value can be adjusted differently according to the complexity of each block. Afterward, the watermarked magnitude coefficients (M_w) are combined with the unchanged phase coefficients and are transformed using the inverse-DFT. Finally, the watermarked color image is created by combining the unaltered chrominance components. If the above process is performed for all blocks, the watermark embedded SI, can be obtained. The overall process of watermark embedding is illustrated in Fig. 4.

B. WATERMARK DETECTION

Spherical panorama watermarking schemes should be able to detect watermarks in SIs and VIs. Detecting a watermark in the SI is the same as detecting a watermark in a 2D image in this scheme. We propose a method to detect watermarks in a VI by recovering the VI to the SI. However, the recovery process requires viewpoint information from the VI; therefore, we need a viewpoint-detection process. We divide the viewpoint-detection process into two steps: a near-viewpoint (NV) detection step and a precise-viewpoint (PV) detection step. These are explained in detail in the following sections. The overall process of watermark detection is illustrated in Fig. 5.

1) NEAR-VIEWPOINT DETECTION

A spherical panorama VI is part of a spherical surface. Therefore, the center of each VI can be expressed as two spherical viewpoint variables: yaw (horizontal, 0° to 360°) and pitch (vertical, −90° to 90°). Obtaining the viewpoint of a VI by comparing each of the VIs is almost impossible; when the interval between VIs is 1°, a total of 64,800 comparisons exist. We propose a method to determine the NV using the SIFT matching technique.

We need reference view images (RIs) to obtain the NV of a detection-targeted VI. The RIs are generated from the original (or watermarked but undamaged) SI. Each of the RI represents a near VIs that has a similar viewpoint. The sum of the RIs should cover the whole spherical panorama image, and an overlapping region should exist between the neighboring RIs. An overlapping region that is too large increases the number of required RIs and causes lower efficiency. In this paper, we used 26 RIs for one SI. twenty-four RIs were created with a combination of vertical (−45°, 0°, and 45°) and horizontal (0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°) viewpoints with 45° intervals and the remaining two are top (vertical 90°, horizontal 0°) and bottom (vertical −90°, horizontal 0°). Fig. 6 illustrates the RI of the world map.

If any two VIs have a similar viewpoint, then they will likely contain several of the same objects. If no severe distortion exists, SIFT, which is robust to rotation, scaling, and translation can match the same objects between them. In other words, if many matched SIFT feature points exist between two VIs, then they have a similar viewpoint in the spherical panorama image. An NV can be obtained using this characteristic.

Initially, the SIFT matching method is performed between a detection-targeted VI and each RI. Then, the number of matching points becomes a measure of the viewpoint similarity. If the maximum SIFT feature points matching number is not over a specific threshold, the targeted VI is not derived from the corresponding spherical panorama image. Conversely, if the maximum SIFT matching number exceeds

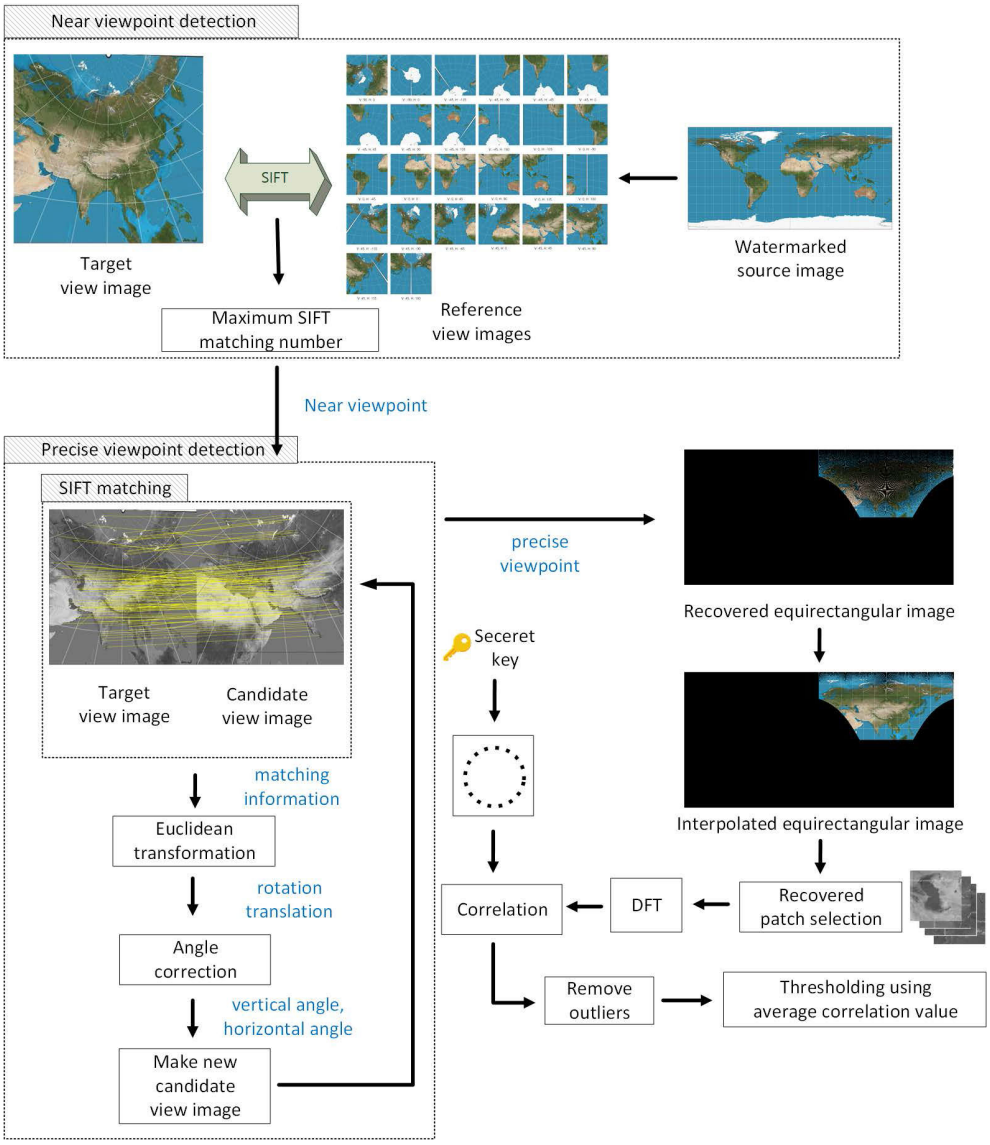


FIGURE 5. The overall watermark detection process.

a threshold value, the viewpoint of the RI is determined as the NV.

The threshold value varies depending on the characteristics of SI and the resolution of VI. The NV-threshold value can be obtained through the following method. First, we obtain the distribution of the SIFT matching number values between RIs and a VIs whose viewpoint are slightly different from the RIs. Second, we obtain the distribution of the SIFT matching number values between the RIs and VIs, which comes from different a SI. Lastly, through the obtained two distributions, the threshold value that minimizes the sum of false positives and false negatives is determined.

2) PRECISE-VIEWPOINT DETECTION

The PV can be obtained by searching the surroundings of the NV. In addition, the SIFT feature point matching method and Euclidean transformation are used to search the surroundings. Initially, we set the candidate-viewpoint (CV) as the NV that was obtained in the previous step. The CV is the last guessed viewpoint that could be a PV. After that, the peak signal noise ratio (PSNR) value between the VI from the CV and a detection-targeted VI is obtained. We can use the PSNR value to determine whether the viewpoints of the two VIs are the same. In other words, if the PSNR value exceeds a threshold, the viewpoint of the detection-targeted VI will have been found, and the CV becomes the PV. Then, we can move on to the next step. The threshold value for determining the PV varies depending on watermark strength and target of signal attacks. The PV-threshold value can be obtained through the following method. First, we obtain distribution of PSNR values between VIs that have the same viewpoint but are different due to embedded watermark and signal attacks. Second, we obtain the distribution of PSNR values between VIs that have different viewpoints. As in the case

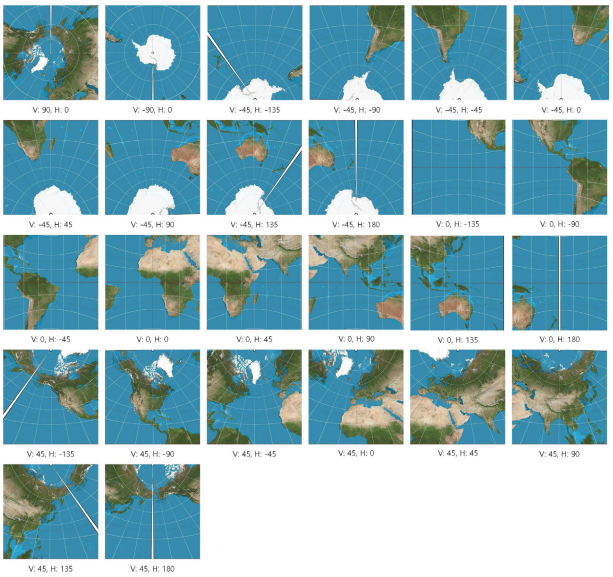


FIGURE 6. Example of 26 reference view images. V: vertical viewpoint (pitch), H: horizontal viewpoint (yaw).

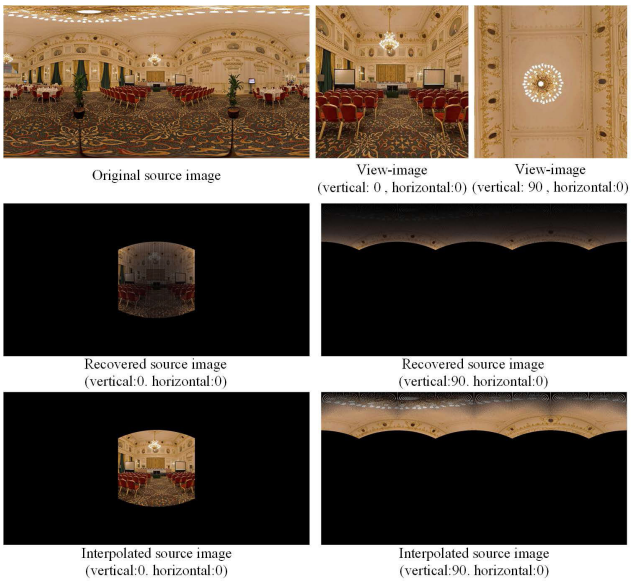


FIGURE 7. Examples of recovery from the view image to the source image and its interpolation.

of NV-threshold, through the obtained two distributions, the threshold value that minimizes the sum of false positives and false negatives is determined.

If the CV does not equal the viewpoint of the targeted VI, the CV should be changed to a reasonable guess. The SIFT matching information is obtained from the VI from the CV and the targeted VI. This information is used to guess the vertical, horizontal, and rotational differences between the two VIs using the Euclidean transformation matrix. The general Euclidean transformation matrix considers the vertical translation, horizontal translation, and rotation and can be expressed as follows in Eq. (5).

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s \cdot \cos\theta & -s \cdot \sin\theta \\ s \cdot \sin\theta & s \cdot \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c \\ d \end{bmatrix} \tag{5}$$

In Eq. (5), x and y refer to the positions before the conversion, and x' and y' refer to the positions after conversion. Moreover, c represents the degree of translation in the x-axis, and d represents the degree of translation in the y-axis. In addition, s refers to the scale, and θ represents the degree of rotation based on the origin considering the scale (s) change. After replacing $s \cdot \cos\theta$ with a and $s \cdot \sin\theta$ with b, we can expand and rewrite it with the determinants a, b, c, and d. After that, the n SIFT feature point matching pairs between the two VIs can be substituted, and it can be expressed as follows in Eq. (6):

$$\begin{bmatrix} x_1 & -y_1 & 1 & 0 \\ y_1 & x_1 & 0 & 1 \\ x_2 & -y_2 & 1 & 0 \\ y_2 & x_2 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & -y_n & 1 & 0 \\ y_n & x_n & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ \vdots \\ x'_n \\ y'_n \end{bmatrix} \tag{6}$$

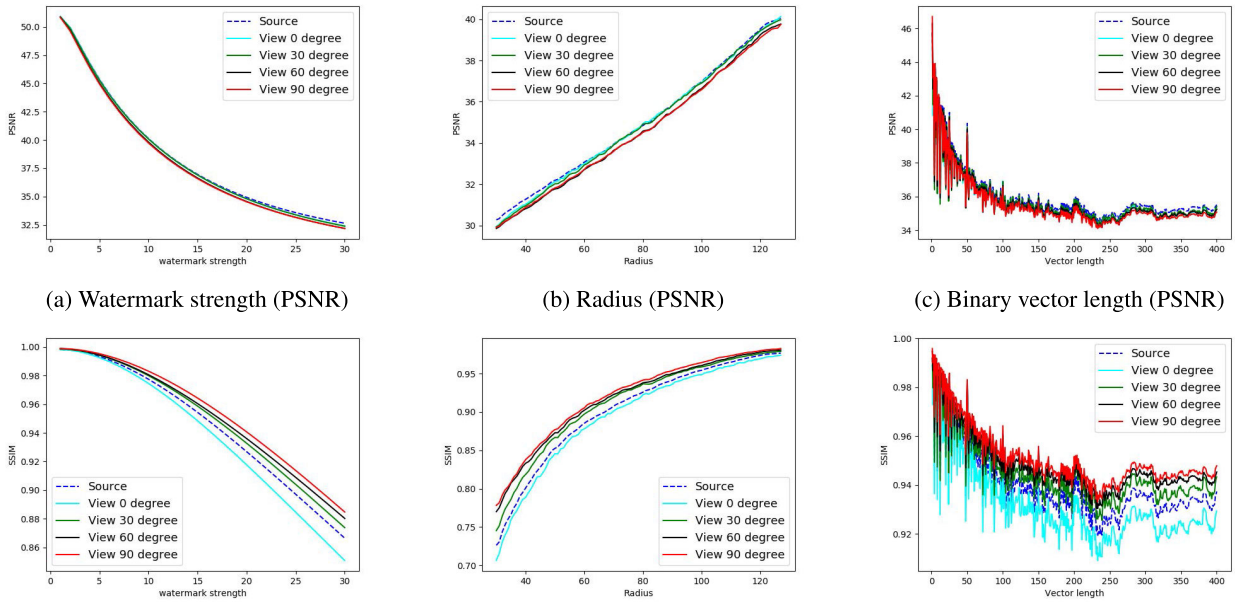
In Eq. (6), the pseudo inverse can be used to obtain the best approximate values for a, b, c, and d. Using c and d, we can estimate the degree of parallel translation of both the x and y-axis. In other words, we estimate the degree of vertical and horizontal translation. Furthermore, we estimate the degree of rotation (θ) using Eq. (7).

$$s = \sqrt{a^2 + b^2}, \quad \cos\theta = \frac{a}{s} \tag{7}$$

After obtaining the guessed vertical, horizontal, and rotational transformation information, the information is used to estimate the viewpoint of the detection targeted VI. In other words, the new CV can be obtained by adjusting the transformation information. The newly obtained CV is checked to determine whether it is the correct PV and it is verified using the PSNR value between the detection targeted VI and VI from the CV. This method is the same as before. If the PSNR value exceeds a threshold, the CV becomes the PV, and we move on to the next step. Otherwise, we should repeat this step until we find the PV. During the calculations, if the CV history information is retained, the PV can be found faster.

3) RECOVERY TO THE SOURCE IMAGE

The obtained PV is used to recover the detection-targeted VI to the equirectangular-formed image for synchronization between embedding and detection image form. The synchronization is required to avoid various distortions of spherical panorama rendering. The recovery process is equal to the inverse of spherical panorama rendering process. However, it is impossible to recover the entire SI; only the part of the SI can be recovered through the VI. This is because only the information used to create the VI remains in the VI. Furthermore, due to the asymmetry of information, holes exist even in the recovered area. Therefore, after reconstruction,



(a) Watermark strength (PSNR) (b) Radius (PSNR) (c) Binary vector length (PSNR)

(d) Watermark strength (SSIM) (e) Radius (SSIM) (f) Binary vector length (SSIM)

FIGURE 8. Influence of the watermark properties on peak signal noise ratio (PSNR) and structural similarity (SSIM).

an interpolation process is needed to reduce the hole effect, which makes a strong signal with periodicity. Since our method embeds watermark in the frequency domain, it interferes detection.

Fig. 7 presents examples of VIs and the corresponding reconstructed SIs and interpolated SIs.

4) WATERMARK DETECTION

In our previous work and in [30], completely synchronized block locations are needed to detect the watermark because of the domain characteristics. However, because we use the DFT magnitude domain, we just need to select any area that is restored well to detect the watermark. If we use multiple blocks to detect the watermark, blocks that have overlapping areas are possible to use. This feature has one substantial advantage, namely that we do not need to consider the horizontal viewpoint in the recovery process because the form of distortion is determined only by the vertical viewpoint. In addition, a small failure in the vertical viewpoint detection is acceptable in our method.

The DFT magnitude coefficients are obtained from the Y-channel of each well-restored block. Then, we extract the vector from the circle points calculated in Eqs. (2) and (3). Afterward, the correlation values between the extracted vector and binary vector (b) from the secret key (K) are calculated. One correlation value can be obtained from one recovered block.

After the average value of the remaining correlation is obtained, the average value is compared to the threshold value. If the average correlation value exceeds the threshold value, it is determined that the watermark is detected; otherwise, a watermark either does not exist or the embedded watermark binary and the watermark binary that the process aimed to detect are different.

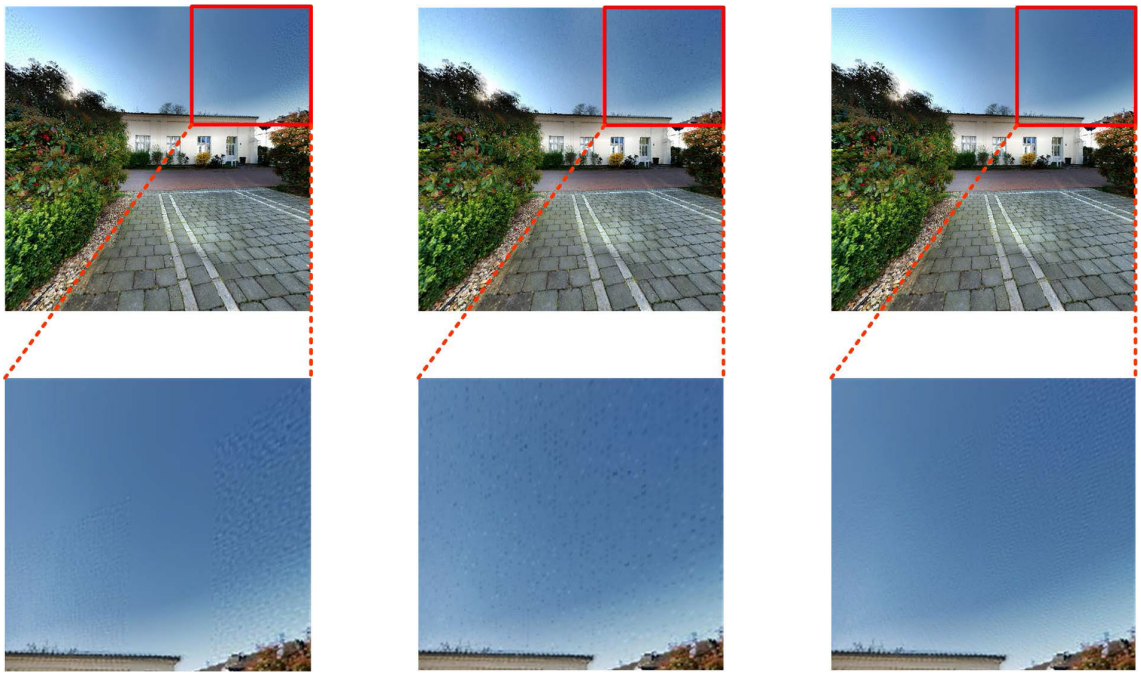
V. EXPERIMENTS

The experiments used 10 equirectangular formed SIs [33]. The resolution of the SI is 1024×2048 and the resolution of the VI is 512×512. The horizontal and vertical field-of-view values of the VI were set to 90°. The number of blocks of a SI is 4×8. The length of the radius (r) that determines the embedding frequency is 100, and the watermark strength is 15.

A. INVISIBILITY

In the proposed method, the influence of the watermark on invisibility depends on the watermark strength, radius of the watermark circle, and length of the embedding binary vector. To determine how each parameter affects the quality, we fixed the other parameters while one parameter varies. We set the watermark strength to 15, the radius to 100, and the binary vector length to 100. The invisibility of the watermark was evaluated using the PSNR and structural similarity (SSIM) values between the watermarked image and original image. We measured the average PSNR and SSIM values between the 10 original SIs and watermarked SIs. Then, we measured the average PSNR and SSIM values between the VIs from the watermarked SI and from the original SI. We conducted experiments on four different vertical viewpoints (0°, 30°, 60° and 90°) of VIs. Fig. 8 presents the results.

A stronger watermark with a shorter radius and longer binary vector results in lower PSNR and SSIM values.



(a) Our previous work [10] (b) Baldoni [30] (c) Proposed method

FIGURE 9. Practical watermark invisibility difference for each model when structural similarity (SSIM) values are similar. The SSIM values for the original image: (a) 0.9313, (b) 0.9303, (c) 0.9208.

A longer radius results in higher PSNR and SSIM values. This is because the farther from the center of the DFT, the higher the frequency. If the watermark is inserted at a high frequency, the invisibility may be improved, but it is weak in terms of signal attacks.

In the case of PSNR, little difference exists according to the vertical viewpoint, but in the case of SSIM, a larger effect of the watermark indicates a lower the SSIM value at the lower vertical viewpoint of the VI. A higher vertical viewpoint indicates that the VI is generated from a larger area of the SI, as shown in Fig. 7. Because we embedded the watermark with the same frequency into each blocks of the SI, as the vertical viewpoint of the VI approaches the polar area, the frequency of the watermark itself becomes higher after rendering. The difference in the SSIM value, which is affected by the variance in the image, between the VIs from the polar and equator regions increases as the effect of the watermark increases.

In this paper, Baldoni's DWT-based method proposed in [30] and the method proposed in our previous work [10] were compared with the proposed method. When we set the watermark strength at 15, the radius at 100, and the binary vector length at 100, the average SSIM value between the watermarked and original SIs was 0.9543. We adjusted the watermark strength of the other methods to obtain similar SSIM values. For the technique in [30], the average SSIM value was adjusted to 0.9540, and our previous method using DCT had an SSIM value of 0.9533. Although the SSIM values of the three watermarking methods are similar, a difference in practical invisibility is shown in Fig. 9 because we embedded the watermark using a higher frequency than that of the other methods.

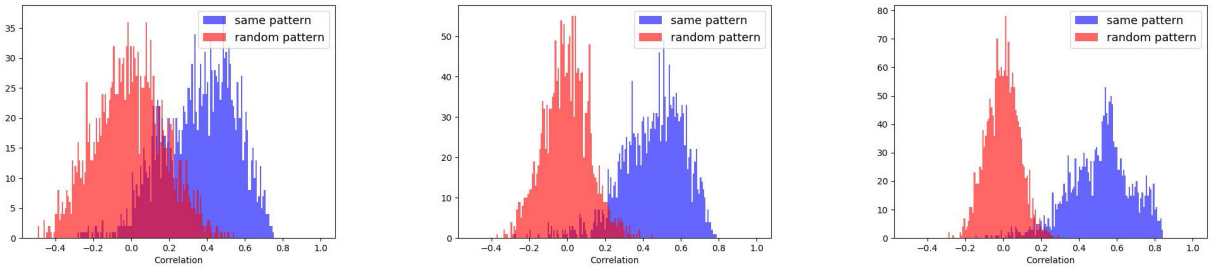
B. WATERMARK ROBUSTNESS

1) BLOCK SIZE

We experimented to determine the watermark robustness according to the block size. Comparative experiment was conducted with 64×64, 128×128, and 256×256 of block sizes. For fair comparison, the radius and binary vector length of each method were adjusted to have similar SSIM values (64×64: 0.9508, 128×128: 0.9533, 256×256: 0.9543). Fig. 10 shows the experimental result. The larger the block size, the stronger the robustness. When the block size is small, because the watermark length is small, the variance of the correlation of the random pattern becomes large. Therefore, the false positive rate and false negative rate are small in the method of larger block size.

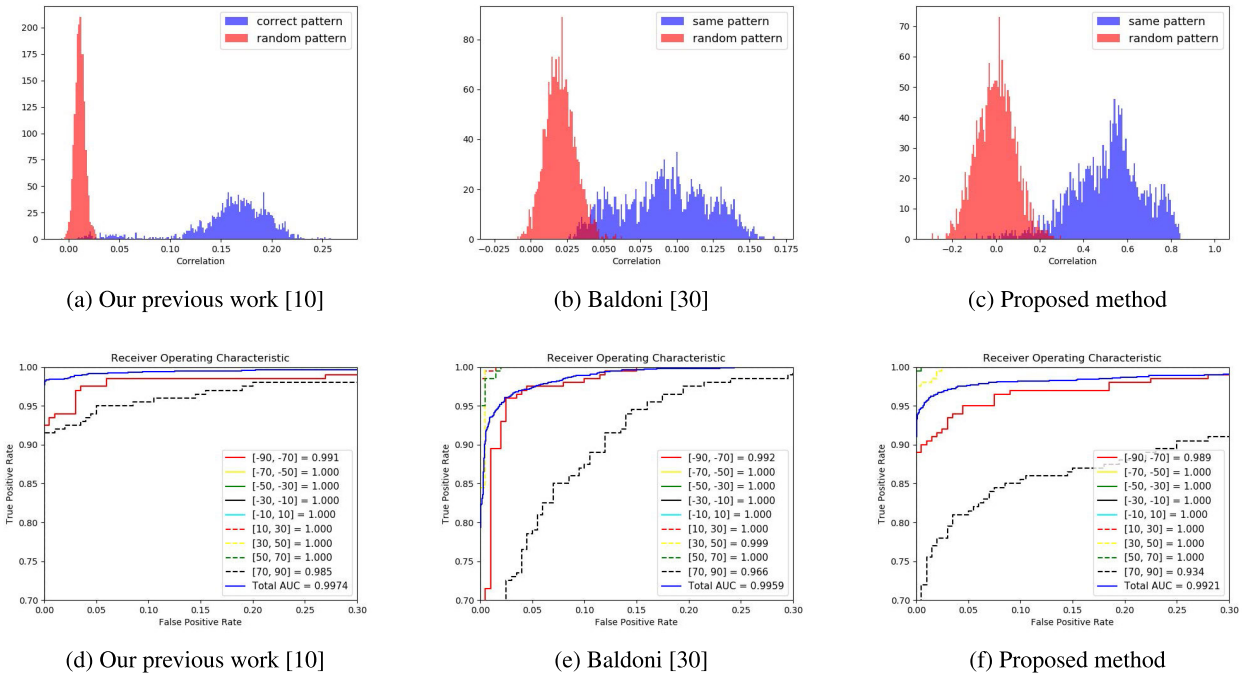
2) NON-ATTACK

After embedding a watermark into each SI without an attack, rendering it as a VI, and then recovering it to the SI, the correlation values were measured using each watermarking detection method. We divided the vertical viewpoint into 9 classes ($[-90°, -70°]$, $[-70°, -50°]$, $[-50°, -30°]$, $[-30°, -10°]$, $[-10°, 10°]$, $[10°, 30°]$, $[30°, 50°]$, $[50°, 70°]$, and $[70°, 90°]$). For each class, 20 VIs were generated for each SI. In total, we used 1800 VIs with 10 SIs. Fig. 11 reveals the results of the experiments.



(a) 64×64 (b) 128×128 (c) 256×256

FIGURE 10. **Histogram of the correlation values according to the block size.**



(a) Our previous work [10] (b) Baldoni [30] (c) Proposed method

(d) Our previous work [10] (e) Baldoni [30] (f) Proposed method

FIGURE 11. **Histogram and receiver operating characteristic (ROC) curve of the correlation values for each spherical panorama watermarking method (without attack).**

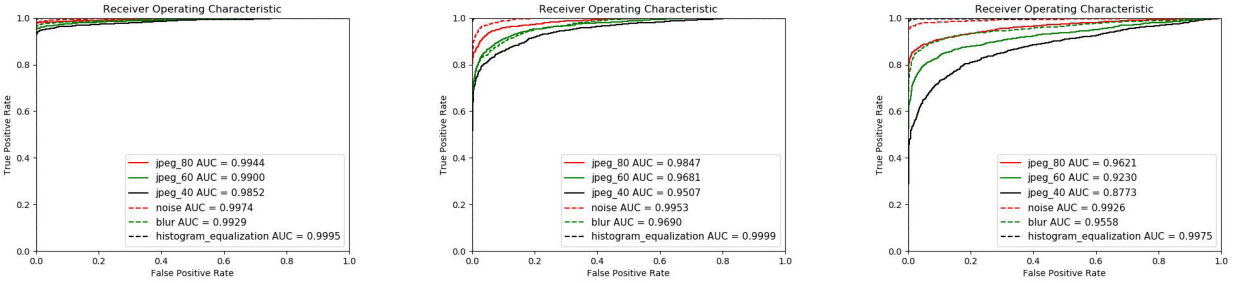
As in our previous experiments, the DCT-based domain exhibits the best performance in a non-attack scenario. The DWT domain-based [30] method and proposed DFT domain-based proposed method demonstrate a similar performance. Owing to the considerable information loss during the recovery process, as the viewpoint approaches the polar area, the correlation exhibits lower values for all methods.

3) SIGNAL ATTACKS

To confirm the robustness of the watermark against signal attacks, experiments were conducted on six signal attacks (i.e., JPEG compression with a quality of 80, 60, and 40, Gaussian noise, Gaussian filter blurring, and histogram equalization, respectively). For each attack, VIs with 1,800 viewpoints were used. After adjusting the attacks on the VIs, the VIs were recovered, and the correlation values were obtained. For the Gaussian noise addition attack, Gaussian noise with mean of 0 and standard deviation of 1 was added to the VI. For the blurring attack, a 3×3 Gaussian filter with a standard deviation of 0.6 was applied.

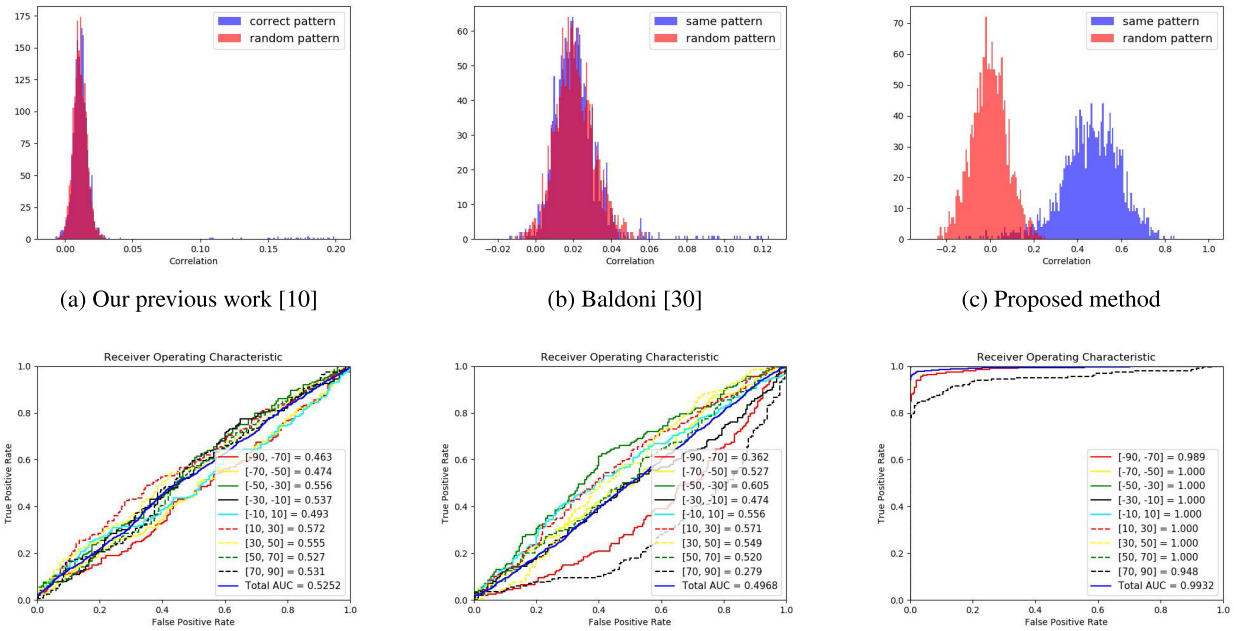
The proposed method embedded a watermark into the magnitude coefficient of the DFT for the robustness against viewpoint-detection failure. Magnitude coefficients have been used for robustness against cyclic translation because they have a shift-invariant characteristic [34]. However, because we did not exploit the phase component, the watermark signal itself is relatively weak compared to the other techniques. In contrast, the DCT or DWT frequency domain contains all the energy of the typical image, they show relatively high robustness against signal attacks.

In this experiment, we set the variables of the compared spherical panorama image watermarking techniques to have similar SSIM values. However, as shown in Fig. 9, the SSIM values cannot fully represent personal invisibility. Even



(a) Our previous work [10] (b) Baldoni [30] (c) Proposed method

FIGURE 12. The receiver operating characteristic (ROC) curve of the correlation values by signal attack: JPEG compression (80, 60, and 40 quality factor), Gaussian noise addition, Gaussian filter blurring, and histogram equalization.



(a) Our previous work [10] (b) Baldoni [30] (c) Proposed method

(d) Our previous work [10] (e) Baldoni [30] (f) Proposed method

FIGURE 13. Histogram and receiver operating characteristic (ROC) curve of the correlation values by horizontal viewpoint desynchronization.

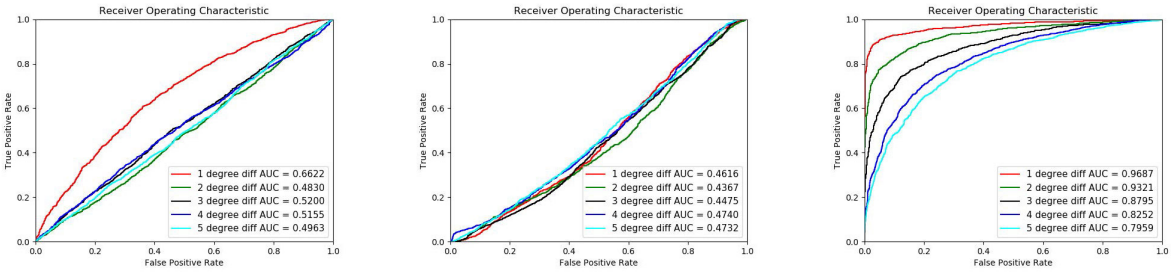
the proposed method has similar SSIM values with other methods, it shows better real invisibility. Because the invisibility and robustness of a watermark are in a trade-off relationship, it is expected that the actual gap of robustness in signal attacks is less than the experimental results.

4) VIEWPOINT DETECTION FAILURE

Unlike other methods, our proposed watermarking method based on the DFT domain is robust against viewpoint-detection failure. This robustness is critical in spherical panorama image watermarking because finding completely accurate viewpoint information is difficult. Furthermore, the viewpoint angle is not discrete but continuous. For example, a difference of 0.1° in the viewpoint can make a difference in the rendering process for high-resolution images.

When rendering from the equirectangular-formed SI to the VI, the vertical viewpoint affects both the form of distortion and vertical position in the SI, whereas the horizontal viewpoint affects only the horizontal position in the SI. We divided the experiment into a vertical viewpoint desynchronization case and a horizontal one. In the horizontal viewpoint desynchronization experiment, the watermarked VI was recovered to the SI with the correct vertical viewpoint and a random horizontal viewpoint. Because the horizontal viewpoint only affects the location in the SI, this experiment has the same effect as block desynchronization. Fig. 13 presents the experimental results.

Our proposed method exhibits almost the same results between the non-attack case and the horizontal viewpoint detection failure case. It is robust against block desynchronization because the DFT magnitude coefficient has



(a) Our previous [10] (b) Baldoni [30] (c) Proposed

FIGURE 14. The receiver operating characteristic (ROC) curve of correlation values by vertical viewpoint desynchronization.

shift-invariant characteristic. However, in the case of the DCT and DWT, if the block synchronization is not complete, the watermark detection fails. Because we set the horizontal viewpoint randomly, several accidentally fitted viewpoints exist, and they demonstrate high correlation values in the DCT and DWT.

When an incorrect vertical viewpoint is used in the inverse-rendering process, both the location and form of distortion change. However, if the difference in the vertical viewpoints is small, the difference of distortion is also small. Thus, we experimented with the differences from 1° to 5° of the vertical viewpoint. Fig. 14 illustrates the experimental results for the vertical viewpoint desynchronization.

Our proposed method exhibits more robustness against vertical viewpoint desynchronization than the other methods. Owing to this robustness, watermark detection is possible even for continuous viewpoints, and a small viewpoint-detection failure is acceptable in the proposed method.

VI. CONCLUSIONS

A spherical panorama image watermarking method should be able to detect a watermark in the VI. We proposed a solution to the problem by embedding a watermark into the equirectangular-formed SI and detecting the watermark in the recovered SI from the watermark target VI. The SIFT feature matching information and Euclidean transformation matrix were used for the inverse-rendering process, and the DFT circular watermarking method was used for watermark detection. In the experiments, although the proposed watermarking method exhibits less robustness against signal attacks, it demonstrates overwhelming robustness against viewpoint desynchronization compared to the other techniques. This is a substantial advantage in spherical panorama image watermarking, where it is difficult to accurately predict the viewpoint of the VI. Furthermore, due to this robustness, watermark detection is possible for continuous viewpoints. Our future work is to find a viewpoint from a VI without any information concerning the original SI and to obtain robustness against SI shape conversion attacks.

REFERENCES

[1] S.-K. Ji, J.-H. Kang, and H.-K. Lee, "Perceptual watermarking for stereoscopic 3D image based on visual discomfort," in *Proc. Int. Conf. Inf. Sci. Appl.*, Springer, 2017, pp. 323–330.

[2] Y.-H. Lin and J.-L. Wu, "A digital blind watermarking for depth-image-based rendering 3D images," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 602–611, Jun. 2011.

[3] S. Wang, C. Cui, and X. Niu, "Watermarking for DIBR 3D images based on SIFT feature points," *Measurement*, vol. 48, pp. 54–62, Feb. 2014.

[4] S.-H. Nam, W.-H. Kim, S.-M. Mun, J.-U. Hou, S. Choi, and H.-K. Lee, "A SIFT features based blind watermarking for DIBR 3D images," *Multimedia Tools Appl.*, vol. 77, no. 7, pp. 7811–7850, Apr. 2018.

[5] K. Wang, G. Lavoue, F. Denis, and A. Baskurt, "A comprehensive survey on three-dimensional mesh watermarking," *IEEE Trans. Multimedia*, vol. 10, no. 8, pp. 1513–1527, Dec. 2008.

[6] J.-W. Cho, R. Prost, and H.-Y. Jung, "An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms," *IEEE Trans. Signal Process.*, vol. 55, no. 1, pp. 142–155, Jan. 2007.

[7] R. Ohbuchi, A. Mukaiyama, and S. Takahashi, "A frequency-domain approach to watermarking 3d shapes," in *Computer Graphics Forum*, vol. 21, no. 3. Hoboken, NJ, USA: Wiley, 2002, pp. 373–382.

[8] J.-U. Hou, D.-G. Kim, and H.-K. Lee, "Blind 3D mesh watermarking for 3D printed model by analyzing layering artifact," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2712–2725, Nov. 2017.

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

[10] J. Kang, S.-K. Ji, and H.-K. Lee, "Spherical panorama image watermarking using viewpoint detection," in *Proc. Int. Workshop Digit. Watermarking*. Springer, 2018, pp. 95–109.

[11] M. Barni, F. Bartolini, V. Cappellini, and A. Piva, "A DCT-domain system for robust image watermarking," *Signal Process.*, vol. 66, no. 3, pp. 357–372, May 1998.

[12] A. Zear, A. K. Singh, and P. Kumar, "A proposed secure multiple watermarking technique based on DWT, DCT and SVD for application in medicine," *Multimedia Tools Appl.*, vol. 77, no. 4, pp. 4863–4882, Feb. 2018.

[13] S. Liu, Z. Pan, and H. Song, "Digital image watermarking method based on DCT and fractal encoding," *IET Image Process.*, vol. 11, no. 10, pp. 815–821, Oct. 2017.

[14] S. Roy and A. K. Pal, "A blind DCT based color watermarking algorithm for embedding multiple watermarks," *AEU Int. J. Electron. Commun.*, vol. 72, pp. 149–161, Feb. 2017.

[15] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1123–1129, Jun. 2000.

[16] A. Poljicak, L. Mandic, and D. Agic, "Discrete Fourier transform–based watermarking method with an optimal implementation radius," *J. Electron. Imag.*, vol. 20, no. 3, Jul. 2011, Art. no. 033008.

[17] M. Urvoy, D. Goudia, and F. Autrusseau, "Perceptual DFT watermarking with improved detection and robustness to geometrical distortions," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 7, pp. 1108–1119, Jul. 2014.

[18] X.-G. Xia, C. G. Boncelet, and G. R. Arce, "A multiresolution watermark for digital images," in *Proc. Int. Conf. Image Process.*, Oct. 1997, pp. 548–551.

[19] M. A. Akhaee, S. M. E. Sahraeian, and F. Marvasti, "Contourlet-based image watermarking using optimum detector in a noisy environment," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 967–980, Apr. 2010.

[20] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[21] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inf. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.
[22] W. Lu, H. Lu, and F.-L. Chung, "Feature based watermarking using watermark template match," *Appl. Math. Comput.*, vol. 177, no. 1, pp. 377–386, Jun. 2006.
[23] H.-Y. Lee, H. Kim, and H.-K. Lee, "Robust image watermarking using local invariant features," *Opt. Eng.*, vol. 45, no. 3, Mar. 2006, Art. no. 037002.
[24] S.-M. Mun, S.-H. Nam, H.-U. Jang, D. Kim, and H.-K. Lee, "A robust blind watermarking using convolutional neural network," 2017, *arXiv:1704.03248*. [Online]. Available: http://arxiv.org/abs/1704.03248
[25] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "Hidden: Hiding data with deep networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 657–672.
[26] H. Zarrabi, A. Emami, P. Khadivi, N. Karimi, and S. Samavi, "BlessMark: A blind diagnostically-lossless watermarking framework for medical applications based on deep neural networks," 2019, *arXiv:1911.00382*. [Online]. Available: http://arxiv.org/abs/1911.00382
[27] J.-U. Hou, D. Kim, W.-H. Ahn, and H.-K. Lee, "Copyright protections of digital content in the age of 3D printer: Emerging issues and survey," *IEEE Access*, vol. 6, pp. 44082–44093, 2018.
[28] Y. Miura, X. Li, S. Kang, and Y. Sakamoto, "Data hiding technique for omnidirectional JPEG images displayed on VR spaces," in *Proc. Int. Workshop Adv. Image Technol. (IWAIT)*, Jan. 2018, pp. 1–4.
[29] I. Kang, Y. H. Seo, and D. W. Kim, "Blind digital watermarking methods for omni-directional panorama images using feature points," *J. Broadcast Eng.*, vol. 22, no. 6, pp. 785–799, 2017.
[30] S. Baldoni, M. Brizzi, M. Carli, and A. Neri, "A watermarking model for omni-directional digital images," in *Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA)*, Sep. 2019, pp. 240–245.
[31] L. Ma, L. Chen, and S. Wang, "Security analysis of a reversible watermarking algorithm for encrypted images in wavelet domain," *Multimedia Tools Appl.*, vol. 78, no. 8, pp. 9827–9843, Apr. 2019.
[32] F. Cayre, C. Fontaine, and T. Furon, "Watermarking security: Theory and practice," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3976–3987, Oct. 2005.
[33] *Flickr/equirectangular*. Accessed: May 20, 2020. [Online]. Available: https://www.flickr.com/groups/equirectangular/
[34] D. Kirovski, *Multimedia Watermarking Techniques and Applications*. Berlin, Germany: Auerbach Publications, 2006.

JIHYEON KANG received the B.S. degree from the School of Computer Science and Electrical Engineering, Handong Global University, South Korea, in 2015, and the M.S. degree from the Graduate School of Information Security, Korea Advanced Institute of Science and Technology (KAIST), South Korea, in 2017. He is currently pursuing the Ph.D. degree with the Multimedia Computing Laboratory, Graduate School of Information Security, KAIST. His current research interests include spherical panorama content watermarking and deepfake content detection.

JONG-UK HOU (Member, IEEE) received the B.S. degree in information and computer engineering from Ajou University, South Korea, in 2012, and the M.S. and Ph.D. degrees from KAIST, South Korea, in 2014 and 2018, respectively. He has been an Assistant Professor with the School of Software, Hallym University, since 2019, and the Principal Investigator of the Multimedia Security and Application Laboratory. His major research interests include various aspects of information hiding, point cloud processing, computer vision, machine learning, and multimedia signal processing.

SANGKEUN JI received the B.S. degree from the Department of Computer and Software Engineering, Kumoh National Institute of Technology, South Korea, in 2013, and the M.S. degree from the Department of Computer Science, Korea Advanced Institute of Science and Technology (KAIST), South Korea, in 2015. He is currently pursuing the Ph.D. degree with the Multimedia Computing Laboratory, School of Computing, KAIST. His current research interests include multimedia security and image processing.

HEUNG-KYU LEE received the B.S. degree in electronics engineering from Seoul National University, Seoul, South Korea, in 1978, and the M.S. and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 1981 and 1984, respectively. Since 1986, he has been a Professor with the School of Computing, KAIST. He has authored/coauthored over 200 international journal articles and conference papers. His major research interests include information hiding and multimedia forensics. He has been a Reviewer of many international journals, including the *Journal of Electronic Imaging*, *Real-Time Imaging*, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.