Multimed Tools Appl (2018) 77:7811–7850
DOI 10.1007/s11042-017-4678-x

A SIFT features based blind watermarking for DIBR 3D images

Seung-Hun Nam[1] · **Wook-Hyoung Kim**[1] · **Seung-Min Mun**[1] · **Jong-Uk Hou**[1] · **Sunghee Choi**[1] · **Heung-Kyu Lee**[1]

Received: 27 September 2016 / Revised: 12 February 2017 / Accepted: 3 April 2017 /
Published online: 16 May 2017

Abstract Depth image based rendering (DIBR) is a promising technique for extending viewpoints with a monoscopic center image and its associated per-pixel depth map. With its numerous advantages including low-cost bandwidth, 2D-to-3D compatibility and adjustment of depth condition, DIBR has received much attention in the 3D research community. In the case of a DIBR-based broadcasting system, a malicious adversary can illegally distribute both a center view and synthesized virtual views as 2D and 3D content, respectively. To deal with the issue of copyright protection for DIBR 3D Images, we propose a scale invariant feature transform (SIFT) features based blind watermarking algorithm. To design the proposed method robust against synchronization attacks from DIBR operation, we exploited the parameters of the SIFT features: the location, scale and orientation. Because the DIBR operation is a type of translation transform, the proposed method uses high similarity between the SIFT parameters extracted from a synthesized virtual view and center view images. To enhance the capacity and security, we

✉ Heung-Kyu Lee
heunglee@kaist.ac.kr

Seung-Hun Nam
shnam@mmc.kaist.ac.kr

Wook-Hyoung Kim
whkim@mmc.kaist.ac.kr

Seung-Min Mun
smmun@mmc.kaist.ac.kr

Jong-Uk Hou
juheo@mmc.kaist.ac.kr

Sunghee Choi
sunghee@kaist.edu

[1] School of Computing, Korea Advanced Institute of Science and Technology (KAIST), 291 Daehak-ro Yuseong-gu, Daejeon 34141, Republic of Korea

propose an orientation of keypoints based watermark pattern selection method. In addition, we use the spread spectrum technique for watermark embedding and perceptual masking taking into consideration the imperceptibility. Finally, the effectiveness of the presented method was experimentally verified by comparing with other previous schemes. The experimental results show that the proposed method is robust against synchronization attacks from DIBR operation. Furthermore, the proposed method is robust against signal distortions and typical attacks from geometric distortions such as translation and cropping.

Keywords 3D image watermarking · Depth image based rendering (DIBR) · Scale invariant feature transform (SIFT) · Blind detection

1 Introduction

Recently, with the development of three-dimensional (3D) rendering technologies and low-cost 3D display devices, 3D content and applications have become actively used in various areas of industries. At the same time, public interest in 3D content is increasing because it offers a tremendous visual experience to viewers. These higher value-added contents can be achieved by two methods: stereo image recording (SIR) and depth image based rendering (DIBR) [5]. The SIR method, which is also referred to as stereoscopic 3D (S3D), uses two cameras horizontally located in different positions to capture left and right views for the same front scene. In a SIR based transmission system, captured stereoscopic images are transmitted to viewers, and they can experience 3D perception using a 3D display with 3D glasses. Because capturing a scene with two cameras acts like the two eyes of a human, this enables viewers to experience a high quality viewing environment. However, this conventional approach of generating stereoscopic content has numerous disadvantages as follows: 1) only one depth condition due to the fixed positions of the cameras, 2) high cost of multiple cameras, and 3) large transmission bandwidth and storage for multiple color images [5, 6, 12].

On the other hand, as shown in Fig. 1, the DIBR method generates virtual images at a different view point using a monoscopic center image and its associated per-pixel depth image [5, 6]. In a DIBR based transmission system, the content distributor transmits a center image

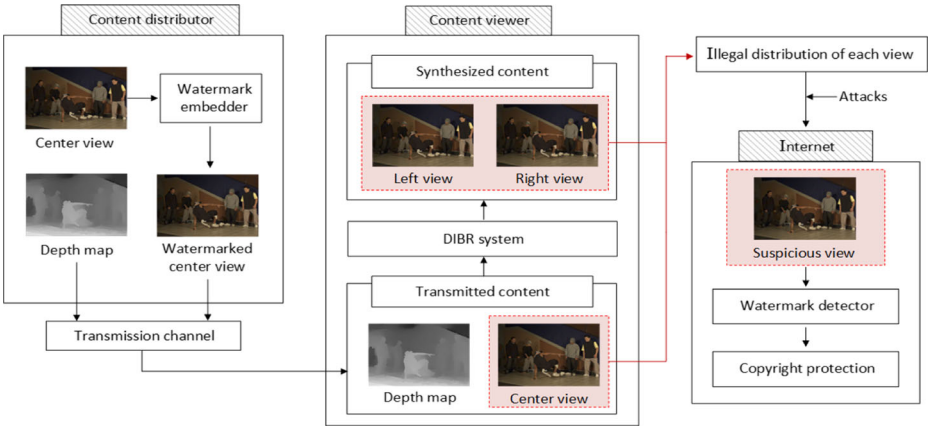

Fig. 1 Block diagram of a DIBR based transmission system and an illegal distribution scenario of a center view and the synthesized virtual view images

and its corresponding depth image to viewers. And then, on the receiver side, stereoscopic images are synthesized by the DIBR system with the transmitted images. In [5, 6, 12], the authors have shown the advantages of DIBR: 1) customized 3D experience adjusting for the depth conditions, 2) backward compatibility with two-dimensional (2D) TV systems, and 3) low-cost transmission bandwidth and data storage. Compared with traditional multi-camera based systems, this technology of extending viewpoints can reduce equipment cost and has a low-cost transmission bandwidth due to the existence of a gray-level depth map. Additionally, the DIBR system enables viewers to control the parallax of the synthesized two views to achieve the experience of 3D depth perception taking into consideration user preference [5, 6, 11]. Due to the above advantages, this depth map based rendering method has received significant attention. Furthermore, with the advances in depth acquisition and 3D rendering techniques, DIBR has received much attention in the research community. Thus, a watermarking method for DIBR 3D images will have an important role in dealing with copyright protection issues for 3D content and in promoting the 3D based industry.

Although many watermarking methods for 2D images have been proposed, these techniques cannot be directly applied to DIBR 3D images due to the inherent characteristics of the DIBR operation. To design a watermarking algorithm for DIBR 3D images, illegal distribution in DIBR based transmission systems should be considered first. As shown in Fig. 1, a malicious user can illegally duplicate a center view and the synthesized view images and then distribute the duplicated images as 2D and 3D content, respectively [4, 11, 15, 16, 27]. Therefore, a watermarking scheme for DIBR 3D images should take into account the illegal distribution of the following contents: 1) the provided center view image as 2D, 2) the synthesized virtual view image as 2D, and 3) the synthesized stereoscopic images as 3D. Thus, as shown on the right side of Fig. 1, a well-designed watermarking scheme has to detect an embedded watermark from an illegally distributed suspicious image. Second, the synchronization attack of the DIBR system is also a big challenge because some pixels of the provided watermarked center image partially move horizontally due to the following three operations on the DIBR system: 1) horizontal shifting in the 3D warping process, 2) adjustment of the baseline distance, and 3) pre-processing of the depth map [4, 11].

Taking the above characteristics of a DIBR system into consideration, some watermarking schemes for DIBR 3D images have been proposed. The authors in [8] proposed an estimation of the projection matrix based watermarking method for DIBR 3D images. A watermark pattern is embedded into a spatial domain of a center image, and the projection matrix estimation scheme is exploited during watermark detection. However, it has a disadvantage in terms of constraints in practical application because this method is non-blind watermarking, which requires the presence of the original content in the watermark detection process. In [14], Lee et al. proposed a spatial domain based perceptual watermarking scheme. This scheme embeds a watermark signal into the occlusion areas that are predicted to be occluded by the adjacent pixels after the DIBR operation. This method only protects watermarked center images and enables viewers to experience a high quality viewing environment. However, the inserted watermark cannot be detected from a synthesized virtual image because the watermarked areas are occluded by an adjacent object after the DIBR operation. Furthermore, the original center image is always needed during watermark detection.

In [27], a local feature descriptors based matching method was exploited to perform synchronization of the watermark. On the watermark embedder, the left view and right view images are synthesized using a DIBR operation with a predefined baseline distance, and then, the watermark is embedded at the location of matched feature points between the center and

synthesized left and right images using the descriptor matching algorithm. However, this method is semi-blind watermarking because it always needs pre-saved matched descriptors in the watermark extraction process. Moreover, it is not robust against geometric distortions and does not consider a change in the baseline distance. In [19], image descriptor based semi-blind watermarking was proposed. In order to compensate for the distortion produced by the DIBR operation, a side information based resynchronization process estimates the disparity between the suspected view and the original view. Thus, this method can detect an embedded watermark on arbitrary virtual views after the DIBR operation. However, this approach has a low perceptual quality of the watermarked image. In addition, because the descriptors of the original image are needed to detect the watermark, its use in wide-ranging applications is restricted. Taking the various geometric distortions into consideration, the authors in [4] proposed a DWT-based watermarking method with geometric rectification based on keypoint descriptors. Because local image descriptors are used for geometric rectification to rectify the altered image, this approach is robust against various geometric attacks. Additionally, because the DIBR operation can be considered as a translation attack, this method based on geometric rectification can detect watermarks on arbitrary virtual views. However, this method is semi-blind watermarking because it always needs pre-saved feature descriptors in the watermark extraction process. Thus, the main issue of this approach is its semi-blind nature which limits its application.

Although a non-blind watermarking scheme has better robustness than a blind one, a blind watermarking scheme has great potential in practical applications because it does not require the original work and side information [7]. Taking into account the advantage of blind watermarking, the authors in [15] proposed a horizontal noise mean shifting (HNMS) based stereoscopic watermarking scheme. Because this scheme changes the mean of the horizontal noise which is an invariant feature of the 2D-3D conversion, it is robust against the 3D warping process. However, this approach does not consider baseline distance adjustment and pre-processing of the depth map on a DIBR system. In addition, this scheme is not robust against different types of geometric distortions, such as a cropping attack and translation. Lin et al. proposed a blind watermarking scheme that takes into consideration the characteristics of the 3D warping process [16]. To deal with the synchronization attack from the DIBR operation, on the watermark embedder, this scheme estimates the virtual left and right images from the center image and its depth map by using information about the DIBR operation with a predefined baseline distance. Based on the estimated relationship, this scheme embeds three different reference patterns into the DCT domain of the center image. This approach shows robustness against common signal distortions such as noise addition and JPEG compression. However, it does not consider the synchronization attacks including the baseline distance adjustment and pre-processing of the depth map which frequently occur during the DIBR operation. Kim et al. presented a robust blind watermarking scheme by exploiting quantization on a dual tree complex transform (DT-CWT) [11]. In this scheme, the sub-bands of the DT-CWT coefficients are selected taking into consideration the characteristic of the DIBR operation and directional selectivity. Because the method by Kim is designed using the characteristics of the approximate shift invariance of the DT-CWT, it is robust against synchronization attacks from the DIBR operation. Moreover, this approach is robust for common processing in the DIBR system including a change in the baseline distance and pre-processing of the depth image. However, this approach has low imperceptibility and does not take into account frequently occurring synchronization attacks such as translation and cropping.

In this paper, we propose a scale invariant feature transform (SIFT) features based blind watermarking algorithm for DIBR 3D images. The SIFT extracts features by taking into account local properties and is invariant to signal processing distortions, translation and 3D projection [13,

18]. The proposed scheme uses location, scale and orientation which are the parameters of the SIFT features. The location and scale of the SIFT keypoints are used to select the area for watermark embedding and detection. Additionally, depending on the orientation of each keypoint, our method embeds a different watermark pattern into the adjacent pixel area within the region around the keypoints in order to enhance capacity and security. Because virtual left and right images are synthesized based on a center image and its corresponding per-pixel depth image on a DIBR system, there are subtle changes between the parameters of the SIFT features extracted from the virtual view images and the original center image. Thus, the proposed method uses the invariability of the parameters of the SIFT features after the DIBR operation. Unlike previous feature descriptor based methods that exploit the descriptor of the original image as side information, the proposed method can detect a watermark in a blind fashion without side information and complicated pre-processing. Moreover, our method uses the spread spectrum technique and perceptual masking taking into consideration the robustness and imperceptibility.

We make the following contributions in this paper: 1) blind watermarking for DIBR 3D images, 2) robustness against synchronization attacks from the DIBR system including horizontal shifting during the 3D warping process, adjustment of the baseline distance, and pre-processing of the depth map, 3) robustness against geometric distortions such as translation and cropping frequently occurring during illegal distribution of content, and 4) high imperceptibility verified by subjective and objective testing. The rest of this paper is organized as follows. A brief review of the DIBR system and SIFT features is given in sections 2 and 3, respectively. Based on the parameters of the SIFT features, a blind watermarking algorithm for DIBR 3D images is proposed in section 4. In section 5, we evaluate the performance of the proposed method. Finally, we conclude our work in the last section.

2 A brief overview of the depth image based rendering system

DIBR is a promising technique for synthesizing a number of different perspectives of the same scene. Authors in [5, 6] proposed the DIBR system with a center image and the associated gray-level depth image. Figure 2(a) shows the center image and Fig. 2(b) the corresponding depth image. A higher intensity value in the depth image means that the objects are closer from the camera. In order to synthesize the virtual view images, the DIBR operation partially moves some pixels of the center image horizontally according to the corresponding depth value of the depth image [5, 6, 28]. The DIBR system consists of three parts, and the overall DIBR process is shown in Fig. 3. Both the 3D warping process and hole filling process are exploited to

Fig. 2 *Ballet* image (1024 × 768): (**a**) center image and (**b**) associated depth image

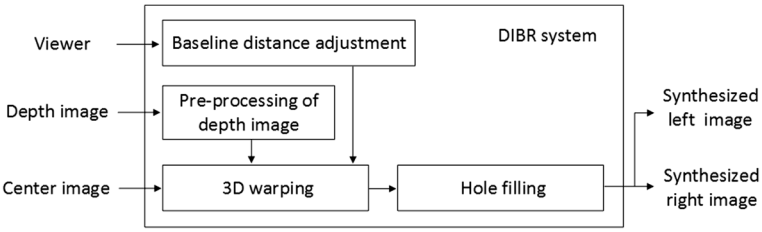

Fig. 3 Diagram of the DIBR system

synthesize virtual view images. Moreover, pre-processing of the depth map is exploited to reduce sharp depth discontinuities in the depth map [12, 28]. The baseline distance adjustment process is exploited to control depth perception.

2.1 Pre-processing of depth image

For natural virtual view generation, pre-processing of the depth image is employed. In this step, the depth image is smoothed by a Gaussian filter to reduce the occurrence of holes [12, 28]. Because this process can mitigate sharp depth discontinuity in the depth image, the quality of synthesized images can be improved. The Gaussian filter is generally used for smoothing the depth image:

$$g(u,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp\left(-u^2/\sigma^2\right), \quad for \ -\frac{w}{2} \le u \le \frac{w}{2} \tag{1}$$

$$\hat{d}(x,y) = \frac{\sum_{v=-\frac{w}{2}}^{\frac{w}{2}} \left\{ \sum_{h=-\frac{w}{2}}^{\frac{w}{2}} \left(d(x-h,y-v)\ g\left(h,\sigma_h\right) \right)\ g\left(v,\sigma_v\right) \right\}}{\sum_{v=-\frac{w}{2}}^{\frac{w}{2}} \left\{ \sum_{h=-\frac{w}{2}}^{\frac{w}{2}} g(h,\sigma_h) \right)\ g\left(v,\sigma_v\right) \right\}} \tag{2}$$

where $g(u,\sigma)$ is Gaussian function, and w is the kernel size. σ represents standard deviation, and determines the depth smoothing strength. $\hat{d}(x,y)$ and $d(x,y)$ are the blurred depth image

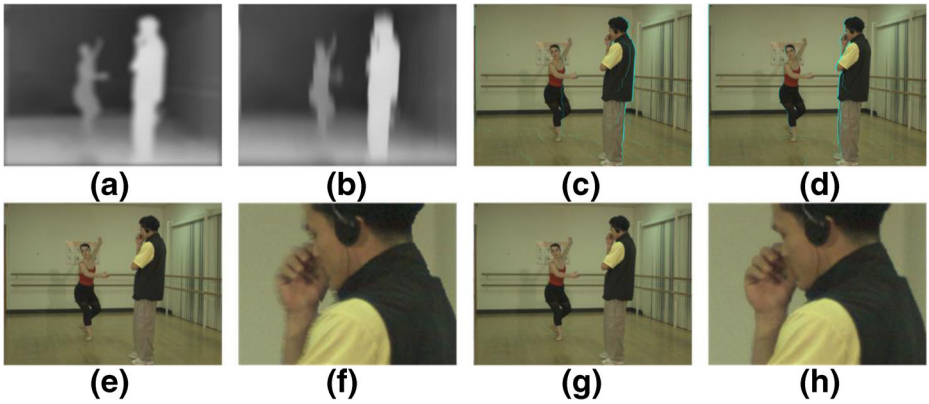

Fig. 4 (**a**) Depth image preprocessed by the symmetric smoothing filter with $\sigma_h = \sigma_v$=30, (**b**) depth image preprocessed by the asymmetric smoothing filter with σ_h = 10 and σ_v = 70, (**c**) Right image with holes, (**d**) Left image with holes, (**e**) Hole-filled left image, (**f**) Magnified regions of (**e**), (**g**) Left image after pre-processing with an asymmetric filter and hole filling, (**h**) Magnified regions of (**g**), The virtual views are synthesized with the baseline distance t_x = 5% of the image width

and original depth image, respectively. $g(h, \sigma_h)$ and $g(v, \sigma_v)$ are the Gaussian function for the horizontal and vertical directions. x and y are the pixel coordinates. σ_h and σ_v are the horizontal and vertical standard deviations, respectively. In [28], an asymmetric Gaussian filter based pre-processing method is presented, and this method can minimize texture distortions appearing in newly exposed areas of a synthesized image. The depth image after pre-processing with a symmetric Gaussian filter is shown in Fig. 4(a). The depth image after pre-processing with an asymmetric Gaussian filter is shown in Fig. 4(b).

2.2 3D warping process and calculation of the relative depth

Before the 3D warping process, the depth value of the gray-level depth image is normalized to two main depth clipping planes [5]. The far clipping plane Z_f represents the largest relative depth value Z, and the near clipping plane Z_n represents the smallest relative depth value, respectively. Therefore, the provided gray-level depth value within a range from 0 to 255 is normalized to the relative depth value within a new range from Z_n to Z_f:

$$Z = Z_f - d \cdot \frac{Z_f - Z_n}{255}, \quad for\ d \in [0, \ldots, 255] \tag{3}$$

Here, d represents the depth value from the depth image. Z_f and Z_n are the new farthest and nearest clipping planes, respectively, and Z is the relative depth value within the range from Z_n to Z_f. In the 3D warping process, pixels in a center image are horizontally moved according to the corresponding relative depth value. According to the parallel configuration approach, virtual view images can be generated from the following function [5, 12, 28]:

$$x_l = x_c + \frac{t_x}{2} \cdot \frac{f}{Z}, \quad x_r = x_c - \frac{t_x}{2} \cdot \frac{f}{Z} \tag{4}$$

where x_l, x_c and x_r denote the pixel x-coordinate of the synthesized left image, center image and synthesized right image. t_x is the baseline distance between the two cameras, and f is the focal length. The camera configuration for generation of the virtual views and the 3D warping process are shown in Fig. 5. The synthesized right view and left view images are shown in Fig. 4(c) and (d), respectively. Because these two images are synthesized by horizontal shifting of the pixels in the center image, a new exposed area, which is also referred to as a hole area, appears in the virtual view. As seen in Fig. 4(c, d), the cyan pixels are the hole area that occurred because of sharp depth changes.

2.3 Hole filling process

The last step of DIBR is the hole filling process. Due to sharp depth discontinuity in the relative depth map, new exposed areas appear in the synthesized images after the 3D warping process [28]. To get high-quality virtual images, hole-areas are filled by interpolation with adjacent pixels. The hole-filled left image without pre-processing of the depth image and the hole-filled left image with pre-processing of the depth image are shown in Fig. 4(e) and (g), respectively. By comparing the magnified images [see Fig. 4(f) and (h)], the effectiveness of the pre-processing is verified. The number of occurring holes and perceptible distortions on the synthesized image are mitigated. In order to make the watermarking method robust against the DIBR system, three characteristics of the DIBR operation, which are types of synchronization

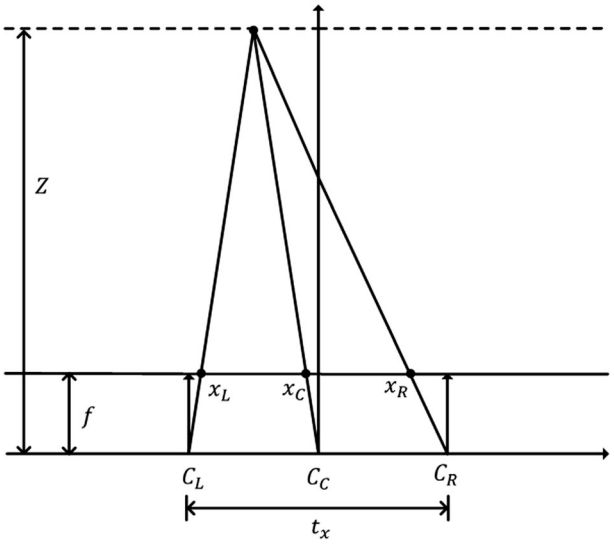

Fig. 5 Camera configuration for the generation of virtual views

attacks, should be considered: 1) horizontal shifting in the 3D warping process, 2) adjustment of the baseline distance, and 3) pre-processing of the depth map. To deal with the above synchronization issue, we exploit SIFT to detect highly distinctive and translation invariant feature points.

3 Analysis on the invariability of the parameters of the SIFT features after the DIBR operation

3.1 A brief overview of SIFT

In [18], the author proposed SIFT which transforms an image into coordinates relative to distinctive local features. Based on a scale-space approach, SIFT extracts local features using parameters such as the coordinates of keypoints (k_x, k_y), scale σ_s and orientation θ. These features are very distinctive, and SIFT is invariant to common signal distortions, translation and projection transformations. As seen in Fig. 6(a), the four steps of the local features

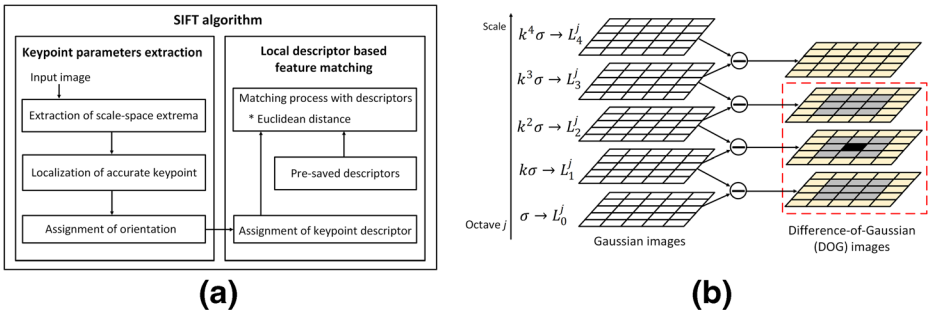

Fig. 6 (**a**) Diagram of SIFT algorithm, (**b**) Gaussian images and scale-space with the Difference-of-Gaussian function

extraction algorithm of SIFT is organized as follows [1, 13, 18, 25]: 1) extrema detection in the scale space of the Difference-of-Gaussian (DOG) function, 2) accurate features localization with measurement of stability, 3) local image gradient based orientation assignment, and 4) generation of the local image descriptor. The fundamental idea of SIFT is to extract features through a cascade filtering approach that identifies standing out points in the scale space [18, 25]. To extract keypoint candidates, the scale space is computed using the DOG function. Let $I(x, y)$ denote the input image and $G(x, y, \sigma)$ represents the 2D Gaussian function with standard deviation σ which determines the smoothing strength:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \cdot \exp\left(-\left(x^2 + y^2\right)/2\sigma^2\right) \tag{5}$$

The scale space of an input image $I(x, y)$ is defined as a function $L(x, y, \sigma)$. And, $L(x, y, \sigma)$ is a Gaussian image from the input image using Gaussian filter with standard deviation σ. In order to construct a set of images in the scale space, the input image is successively convolved with the Gaussian function:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{6}$$

where $*$ is the convolution operation. In order to the construct scale space, the SIFT algorithm repeatedly computes the Gaussian image $L(x, y, \sigma)$ while increasing the value of σ. As can be seen on the left side of Fig. 6(b), the input image is incrementally convolved with the Gaussian function to construct Gaussian images that are separated by a multiplicative constant factor k. The second Gaussian image $L(x, y, k\sigma)$ following $L(x, y, \sigma)$ is generated at $k\sigma$. $L(x, y, k\sigma)$ is the convolution of the input image $I(x, y)$ with the Gaussian function $G(x, y, k\sigma)$ at scale $k\sigma$. Let s be an integer greater than or equal to 1 and $k \geq 2^{\frac{1}{s}}$. Let σ_i be the standard deviation value in i-th Gaussian filter. Then, σ_i is defined as $\sigma_i = k^i\sigma$ where $0 \leq i < s + 3$. Here, σ is the initial standard deviation value. Under the given condition, the $i + 1$-th Gaussian image can be defined as $L_{i+1}(x, y) = G(x, y, \sigma_i) * L_i(x, y)$ where $0 \leq i < s + 3$. In this fashion, it is possible to compute the sequence composed of the Gaussian images from $L_0(x, y)$ to $L_{s+2}(x, y)$ for various scales.

This sequence of Gaussian images is called an octave. In the case shown in Fig. 6(b), we can see that there is one j-th octave. Since the octave includes five Gaussian images, it can be seen that the value of s is set to 2. Let L_i^j be the i-th Gaussian image included in the j-th octave. On the left side of Fig. 6(b), it can be seen that the Gaussian images from L_0^j to L_4^j generated at different scales from σ to $k^4\sigma$ form one octave. By repeating the method of forming one octave, octaves are additionally generated in order to construct the scale space. For efficiency, the last down-sampled Gaussian image of the previous octave can be used as the first Gaussian image of the next octave [18, 25].

The necessary process after constructing the scale space of the input image $I(x, y)$ is to compute the DOG. The SIFT algorithm uses the DOG to guarantee scale invariance [18]. The authors in [17] showed that the normalized Laplacian of Gaussian(LOG) is useful for finding edges and blobs. The scale-normalized LOG is defined as $\sigma^2\nabla^2 G(x, y)$, where the σ^2 term is exploited for normalization. And, the image filtered using LOG can be defined as $\sigma^2\nabla^2 G(x, y) * I(x, y)$. The characteristic of LOG extracting blob area provides scale invariance. The author in [20] proposed stable features to exploit the extrema of LOG. LOG provided good performance for scale invariance, but it had a disadvantage of high computational

complexity, so DOG was introduced. The DOG function provides an approximation of the scale normalized Laplacian that is used for scale invariant blob detection. The relationship between DOG and LOG can be explained by the heat diffusion equation, $\frac{\partial G}{\partial \sigma} = \sigma \nabla^2 G$ [18]. From the heat diffusion equation and the finite difference approximation, the following relationship is derived: $\sigma \nabla^2 G = \partial G / \partial \sigma \approx (G(x, y, k\sigma) - G(x, y, \sigma))/(k\sigma - \sigma)$. Finally, by summarizing the previous equation, we can derive the following equation: $\sigma^2 \nabla^2 G(k-1) \approx G(x, y, k\sigma) - G(x, y, \sigma)$. Here, the $G(x, y, k\sigma) - G(x, y, \sigma)$ term means that the DOG function nearby scales at $k\sigma$ and σ. This means that the DOG function provides an approximation of the scale normalized LOG. $D(x, y, \sigma)$ represents the difference of two nearby Gaussian images [18]:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \tag{7}$$

As seen in the red dashed box on the right side of Fig. 6(b), the adjacent Gaussian images are subtracted to construct the DOG images [18]. In order to extract the location of stable features in the scale space, scale space extrema (e.g., local maxima and minima) in the DOG images is retrieved by comparing between the sample point and its 26 neighbors which include the eight adjacent pixels in the current scale and 18 neighbors in the adjacent scales. Moreover, these local extremas determine the scale σ_s and location (k_x, k_y) of the SIFT features. After detection of the scale space extrema, detailed fitting for an accurate location and scale of features is performed because some of the keypoint candidates are unstable [18]. In this step, the keypoints that have high edge responses and low-contrast ones are eliminated to increase stability.

In the third step of the SIFT algorithm, the local image gradient directions based orientation θ is assigned to each keypoint. At first, the scale of the refined keypoints is employed to select the Gaussian image $L(x, y, \sigma_s)$ with the closest scale in the scale space. And then, the gradient magnitude $m(x, y)$ and orientation $\theta(x, y)$ of the Gaussian image sample $L(x, y)$ at scale σ_s are calculated from the following functions [18]:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \tag{8}$$

$$\theta(x, y) = \arctan((L(x, y+1) - L(x, y-1))/(L(x+1, y) - L(x-1, y))) \tag{9}$$

For pixel areas around the keypoints in the Gaussian image, the gradient magnitude and orientation are computed, and then, the orientation histogram is formed using the gradient orientations and weighted gradient magnitude. During the formation of the orientation histogram with 36 bins covering 360 degrees, each sample is added to each bin of the orientation histogram. The highest peak in this histogram corresponds to the dominant direction of the local gradients, and it is assigned to orientation θ of the keypoint. So far, the operation of constructing the parameters including the location, scale and orientation has been described [18].

The next step is to generate a feature descriptor that is a 128 element vector for each feature. First, the keypoint descriptor is generated by computing the gradient orientation and magnitude of sample pixels within a region around the keypoint. And then, the coordinates of the descriptor are rotated based on the orientation of the keypoint to attain robustness against rotation. Lastly, the orientation histogram is constructed by using the precomputed magnitude and orientation values of the samples, and then, the keypoint descriptor is computed based on

the orientation histogram [18]. As shown on the right side of Fig. 6(a), the keypoint descriptor is originally used for image matching and feature matching. With the minimum Euclidean distance based matching approach, the extracted feature descriptors are matched to the nearest descriptor in the database of SIFT features extracted from the test images. During the feature matching operation, pre-computed descriptors extracted from test images are needed to compute the minimum Euclidean distance of the extracted descriptors. Because blind watermarking should detect an embedded watermark in a work without both the original work and side information, the feature descriptor based feature matching operation is not employed in the proposed blind watermarking method. The previous works in [4, 19, 27] proposed descriptor matching based semi-blind watermarking schemed. Because these semi-blind watermarking methods always need pre-saved feature descriptors in the watermark detection process, they have lower general usefulness in the real world. Thus, previous descriptor matching based watermarking schemes for DIBR 3D images have a critical issue which is its semi-blind nature limiting its application. Unlike previous feature descriptor based methods, we propose a blind watermarking scheme that uses the parameters of the SIFT features.

3.2 Analysis on the invariability of the SIFT parameters after the DIBR operation

The DIBR operation is type of horizontal shift. To synthesize the virtual view images, DIBR operation partially moves the pixels of the center image horizontally according to the corresponding depth value of the depth image [5, 6, 28]. And, this horizontal shift is performed according to formula (4). Thus, except for the sharp depth discontinuity areas in the depth image, objects in the center image can be naturally warped to a new coordinate in the horizontal direction. In other words, objects having a similar depth value in the center view are moved while maintaining the original structure. For a specific area that has a high normalized depth value Z, there is only a subtle horizontal shift when compared to the original view. And, the newly exposed areas, referred to as a hole area, can be filled by averaging textures from neighboring pixels. Furthermore, pre-processing of the depth map is employed to reduce sharp depth discontinuities. With these common processes of the DIBR system to achieve better quality virtual views, the virtual left and right images are synthesized to be similar to the center view image.

As shown in Fig. 7, for the test images *"Ballet"* and *"Breakdancers"*, there are some horizontal shift changes between the virtual view images and the original center image. Based on the DIBR operation with the center view images and their corresponding depth images, virtual images are generated. In Fig. 7, the starting point of the arrows indicates the location of the keypoints. The length of the arrows means the scale of each keypoint, and the direction of the arrows denotes the orientation of each keypoint. Without loss of generality, the baseline distance t_x is set to 5% of the center view width. The focal length f is set to 1. And, Z_f and Z_n are set to $t_x/2$ and 1, respectively. Regardless of whether pre-processing of the depth map was done, there is subtle variation between the parameters of the SIFT features extracted from the virtual images and the center image shown in Fig. 7. After the horizontal shift of the DIBR operation, the majority of the SIFT parameters including the scale and orientation suffer from subtle changes. Despite the variation in the location of the keypoints caused by the 3D warping process of the DIBR operation, the tendency of the parameters including the scale and orientation is maintained. Comparing the arrows included in the left, center, and right images in Fig. 7, we can see that the arrows are very similar in length and direction. Although some keypoints have disappeared or changed due to the 3D warping process, most of the keypoints retain their inherent parameters including scale and orientation.

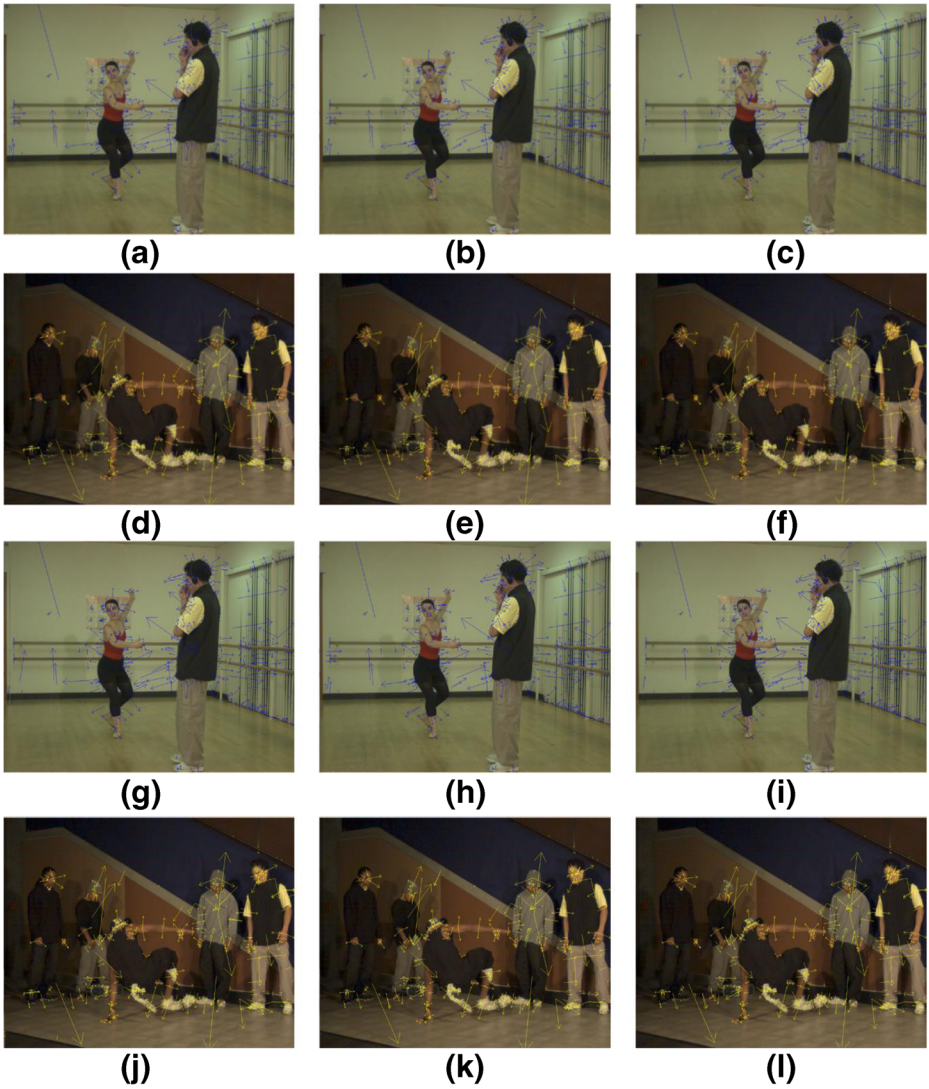

Fig. 7 Variation of the SIFT parameters after the DIBR operation: (left column) left view images with $t_x = 5\%$ for the width, (center column) center view images and (right column) right view images with $t_x = 5\%$ for the width. (**a**)–(**f**) the resultant image after the DIBR operation with pre-processing of the depth map. (**g**)–(**l**) the resultant image after the DIBR operation without the pre-processing of the depth map

In order to analyze the invariability of the SIFT parameters after DIBR, we analyzed the ratio of the variation for each parameter. Let r_m represent the average ratio of the matched features between the center and virtual left view. And r_v denotes the average ratio of the variation of the SIFT parameters after the DIBR operation. r_m and r_v are computed with the following formula (10):

$$r_m = \frac{n_m}{n_c}, \quad r_v = \frac{1}{n_m} \sum_{i=1}^{n_m} \frac{\left| p_i^c - p_i^l \right|}{p_i^c}, \qquad for\ p_i^c \in M_c\ and\ p_i^l \in M_l \tag{10}$$

where n_c denotes the number of SIFT features from the center image, and n_m is the number of matched features between the center and left images. Here, M_c and M_l are the set of matched features extracted from the center image and matched features extracted from the left image, respectively. And p_i^c and p_i^l represent the i-th SIFT parameter in the set M_c and M_l, respectively. $|\cdot|$ represents an absolute-value norm. In this analysis, the SIFT feature matching process is exploited to get accurate locations of the horizontally shifted keypoints corresponding to the keypoints of the center image. Based on this matching data, it is possible to compare the variation of the parameters among the corresponding keypoints. Table 1 shows the ratio of the matched features and the ratio of the variation of the SIFT parameters between the center image and the synthesized left images. The left images are synthesized with various baseline distances t_x. *"Ballet"* and *"Breakdancers"* are included in the Microsoft Research 3D Video

Table 1 Ratio of the matched features and ratio of the variation of the SIFT parameters between the center and left views

	Pre-processing of depth map	t_x	Average ratio of matched features: r_m	Average ratio of the variation of the SIFT parameter: r_v	
				Scale	Orientation
Ballet	with	3	0.9334	0.0109	0.0033
		4	0.9285	0.0126	0.0055
		5	0.9107	0.0149	0.0064
	without	3	0.8936	0.0198	0.0055
		4	0.8760	0.0202	0.0059
		5	0.8692	0.0204	0.0068
Break-dancers	with	3	0.9439	0.0074	0.0063
		4	0.9327	0.0077	0.0064
		5	0.9271	0.0086	0.0065
	without	3	0.9345	0.0088	0.0064
		4	0.9114	0.0093	0.0067
		5	0.9016	0.0096	0.0069
Interview	with	3	0.9975	0.0011	0.0008
		4	0.9831	0.0017	0.0014
		5	0.9642	0.0036	0.0027
	without	3	0.9930	0.0009	0.0006
		4	0.9835	0.0016	0.0019
		5	0.9769	0.0013	0.0018
Orbi	with	3	0.9942	0.0004	0.0007
		4	0.9920	0.0007	0.0006
		5	0.9811	0.0021	0.0013
	without	3	0.9855	0.0008	0.0004
		4	0.9667	0.0023	0.0026
		5	0.9861	0.0016	0.0006
Teddy	with	3	0.9975	0.0019	0.0015
		4	0.9780	0.0016	0.0009
		5	0.9718	0.0017	0.0011
	without	3	0.9505	0.0029	0.0008
		4	0.9456	0.0027	0.0023
		5	0.9443	0.0024	0.0013
Cones	with	3	0.9736	0.0015	0.0002
		4	0.9631	0.0022	0.0008
		5	0.9529	0.0025	0.0009
	without	3	0.9532	0.0029	0.0025
		4	0.9420	0.0026	0.0018
		5	0.9226	0.0041	0.0031

Datasets [29], *"Interview"* and *"Orbi"* are included in the Heinrich-Hertz-Institut Datasets [5], and *"Teddy"* and *"Cones"* are included in the Middlebury Stereo Datasets [9, 21–23]. A detailed description of each dataset is given in section 5.

The larger the t_x value, the greater the degree of horizontal movement of the pixels in the center image. As the degree of 3D warping increases, the difference between the original center image and the synthesized image increases. Thus, the r_m value tends to decrease when the t_x value increases. As shown in Table 1, the average ratio of the matched features r_m with different t_x is above 0.85. More than 85% of the keypoints extracted from the left views are matched with the corresponding keypoints of the center view. And, for various baseline distances t_x from 3 to 5, the average r_m of six test sets is 0.9517. After the horizontal shift of the DIBR operation, the keypoints similar to the keypoints extracted from the center view image are found in the synthesized view image.

Based on the matched keypoints between the center and left images, the variation ratio of the scale and orientation of the SIFT keypoints is computed by formula (10). As listed in Table 1, there is only subtle variation between the corresponding parameters regardless of the pre-processing of the depth map. When pre-processing of the depth map is performed, the average r_v for a scale of six test sets for various t_x from 3 to 5 is 0.0054. If the depth map is not pre-processed, the average r_v for the scale of six test sets for various t_x is 0.0062. After the DIBR operation, the ratio of the variation for the scale of keypoints is small. As mentioned in section 3.1, the scale of the SIFT keypoint is calculated from the extrema of the scale space. As can be seen on the right side of Fig. 6(b), the scale space extrema is retrieved by comparing between the sample point and its 26 neighbors. If the depth values corresponding to the area around the sample point are not discontinuous, neighboring pixel areas within the region around the sample point undergo a similar strength of horizontal shift attack. Therefore, the scale parameter of the keypoint that is not included in the discontinuous region of the corresponding depth image is robust against the DIBR operation.

As can be seen in Table 1, the experimental results for the orientation of SIFT keypoints are similar to the experimental results for the scale of the keypoints. The ratio of the variation of the orientation of the keypoints is relatively smaller than the ratio of the variation of the scale of the keypoints. When pre-processing of the depth map is performed, the average r_v for the orientation of the six test sets for various t_x from 3 to 5 is 0.0026. If the depth map is not pre-processed, the average r_v for the orientation of the six test sets for various t_x is 0.0032. As described in section 3.1, for pixel areas around the keypoint, the gradient orientation and magnitude are computed. Using these values of gradient orientation and magnitude, the orientation of the keypoint is determined. Like the scale parameter, because the orientation of the SIFT features is computed using their neighboring pixels, a low r_v value of the orientation means that neighboring pixel areas within the region around the keypoints undergo a similar strength of horizontal shift attack. The test results for the variation of the SIFT parameters show that each SIFT parameters including the scale and orientation has robustness against the DIBR operation. Therefore, we propose a SIFT parameters based blind watermarking method. Unlike previous local descriptor based semi-blind watermarking schemes, the proposed method that only exploits the SIFT parameters including the location, scale and orientation can detect a watermark in a blind fashion without any side information. The detailed algorithm of the proposed method is described in section 4.

4 Proposed watermarking scheme

In this section, we describe the proposed watermarking scheme based on the SIFT parameters: location, scale and orientation. In the watermark embedding process, using the location of keypoints, we select patches that are robust against common distortions and synchronization attacks. Because SIFT keypoints with both small and large scales can be eliminated by distortions, we refine the SIFT features based on the scale of the keypoints. In order to select non-overlapped patches to avoid mutual-interference, we select non-overlapped patches based on the orientation parameter. Furthermore, in order to enhance the capacity and security, we propose an orientations based watermark pattern selection method. The watermark is embedded into the selected patches in the discrete cosine transform (DCT) domain. Taking the robustness and imperceptibility into consideration, we use the spread spectrum technique [3] and the perceptual making with noise visibility function (NVF) [26]. In the watermark extraction process, using the location of the refined keypoints, we select patches. Based on the correlation-based detection algorithm, the embedded watermarks are extracted from the patches.

4.1 Watermark embedding

Figure 8 shows a diagram of the proposed watermark embedding process. The overall process can be decomposed into eight steps.

- *Step 1 (SIFT keypoints extraction): I* and *D* are the center image and depth image of the same size, respectively. I_w and I_h are the width and height of *I*. The SIFT keypoints are extracted from the center image *I*. Suppose $\mathbf{S} = \{s_1, \ldots, s_L\}$ is a set of keypoints with their corresponding SIFT parameters. Here, *L* represents the number of keypoints. s_i denotes the extracted SIFT keypoint, and the SIFT parameters of s_i is described by the following information: $s_i = \{x, y, \sigma, \theta\}$, where (x, y) are the location of the keypoint; σ is the scale of the keypoint, and θ is the orientation of the keypoint. And $s_{i,x}$ and $s_{i,y}$ are the x and y coordinates of the *i*-th keypoint, respectively. $s_{i,\sigma}$ and $s_{i,\theta}$ are the scale and orientation of the *i*-th keypoint, respectively. The proposed method selects patches that are neighboring pixels within the region around the keypoints s_i for watermarking. P_w and P_h are the width and height of each patch to be watermarked, respectively.

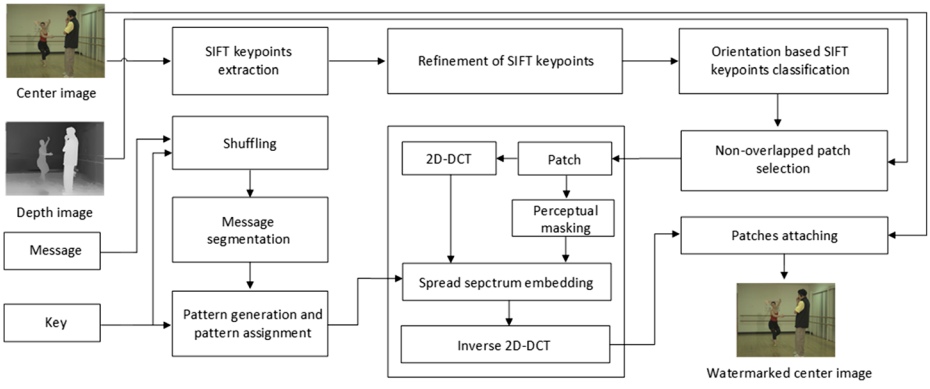

Fig. 8 Diagram of the proposed watermark embedding process

- *Step 2 (Refinement of the keypoints)*: The extracted keypoints are refined taking into consideration the robustness of the proposed watermarking scheme. First, because the SIFT keypoints with both small and large scales can be eliminated by attacks, we eliminate the keypoints whose scale is above σ_{max} or below σ_{min}. $\mathbf{E}_1$ denotes a set of keypoints that is to be eliminated due to the scale criteria.

$$\mathbf{E}_1 = \left\{ s_i \mid s_{i,\sigma} < \sigma_{min},\ \ s_{i,\sigma} > \sigma_{max} \right\} \tag{11}$$

SIFT keypoints whose scale parameter is too small are less likely to be redetected because of their low robustness against distortions. Additionally, SIFT keypoints whose scale parameter is too large are less likely to be redetected because their location parameter is easily moved to other locations [13]. In this paper, we set σ_{min} and σ_{max} as 1 and 8, respectively. Second, in order to select square patches with a defined size of P_w and P_h, keypoints located on the boundary surface of the $I(x, y)$ are eliminated. $\mathbf{E}_2$ denotes a set of keypoints to be eliminated due to location criteria:

$$\mathbf{E}_2 = \left\{ s_i \mid s_{i,x} < \frac{P_w}{2},\ s_{i,x} > I_w - \frac{P_w}{2},\ \ s_{i,y} < \frac{P_h}{2},\ \ s_{i,y} > I_h - \frac{P_h}{2} \right\} \tag{12}$$

Finally, because the proposed method assigns a reference pattern to the patch around each keypoint based on its orientation $s_{i,\theta}$, we eliminate keypoints that have multiple orientations.

- *Step 3 (Keypoints classification based on orientation)*: Suppose $\mathbf{S}' = \left\{ s_1, \ldots, s_{L'} \right\}$ is a set of refined keypoints obtained through *step 2* above. Here, L' represents the number of refined SIFT keypoints. And the SIFT keypoints from a set $\mathbf{S}'$ are divided into K distinct sections, hereafter referred to as bins, according to their orientation. The orientation θ of each SIFT keypoint varies from 0° to 360°. To enhance the capacity and security, one reference pattern is assigned to a single bin. Each bin is independently processed to embed one reference pattern. Because every bin is used for the watermark embedding process, we can embed K reference patterns to cover the work. A detailed description of the relation between the reference pattern and the message bits to be inserted is described in *step 6*. To classify the SIFT keypoints into K bins, the regular interval θ_K is computed in advance:

$$\theta_K = \frac{\theta_{max} - \theta_{min}}{K} \tag{13}$$

where K represents the number of bins. And the maximum and minimum orientations θ_{max} and θ_{min} are set to 360° and 0°, respectively. Because the range of degrees $(0, 360]$ is divided by the regular interval θ_K according to formula (13), each bin has a θ_K degree range. Additionally, the n-th bin $\mathbf{B}_n$ is defined with the following formula (14):

$$\begin{aligned} \mathbf{B_n} = \left\{ s_j^n \right\} = \{ s_i | & mod(\theta_{min} + \theta_S + \theta_K n, \theta_{max}) \\ & < s_{i,\theta} < mod(\theta_{min} + \theta_S + \theta_K (n+1), \theta_{max}) \}, \\ & for\ 0 \le n \le K-1, 0 \le i \le L'-1, 0 \le j \le M_n \end{aligned} \tag{14}$$

where s_j^n is the j-th keypoint of the n-th bin, and M_n is the number of keypoints belonging to the n-th bin. θ_S indicates the degree offset from 0°, and the classification of the bin is processed at the degree of θ_S. As shown in Fig. 9, the whole degree range of the orientation is classified into K bins, and each of the bin $\mathbf{B_n}$ is a set that includes SIFT keypoints classified by their orientation parameter $s_{i,\theta}$.

After orientation based classification, the keypoints easily deformed by attacks are removed through the refinement of θ. Because the changes in the orientation of the keypoints will adversely affect the detection of the watermark, we remove the keypoints around the border of each bin. As shown in Fig. 9, the keypoints contained in the shaded area are removed. Additionally, the orientation based refined n-th bin is defined with the following formula (15):

$$\begin{aligned}\mathbf{B}_{\mathbf{n}}^{'} = \left\{s_j^n\right\} = \{s_i |\ mod\left(\theta_{min} + \theta_S + \frac{\theta_E}{2} + \theta_K{\cdot}n, \theta_{max}\right) \\ < s_{i,\theta} < mod\left(\theta_{min} + \theta_S - \frac{\theta_E}{2} + \theta_K{\cdot}(n+1), \theta_{max}\right)\}, \\ for\ 0 \le n \le K-1, 0 \le i \le L^{'}-1, 0 \le j \le M^{'}{}_n\end{aligned} \tag{15}$$

where θ_E is the degree offset value used to eliminate unstable keypoints, and $M_n^{'}$ is the number of keypoints belonging to the n-th bin $\mathbf{B}_{\mathbf{n}}^{'}$. In addition, $s_{j,x}^n$ and $s_{j,y}^n$ are the x and y coordinates of the j-th keypoint belonging to $\mathbf{B}_{\mathbf{n}}^{'}$, respectively. $s_{j,\sigma}^n$ and $s_{j,\theta}^n$ are the scale and orientation of the j-th keypoint belonging to $\mathbf{B}_{\mathbf{n}}^{'}$, respectively.

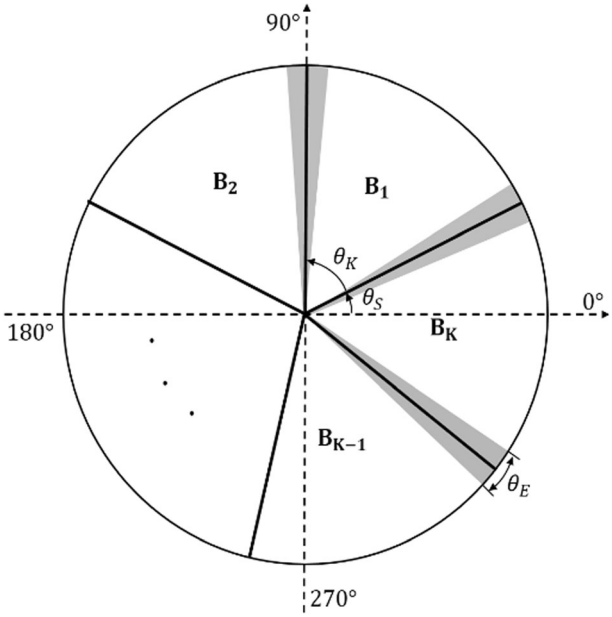

Fig. 9 Keypoints classification based on their orientation

- *Step* 4 *(Non-overlapped patch selection)*: Suppose $\mathbf{S}'' = \{s_1, \ldots, s_{L''}\}$ is a set of refined keypoints obtained through *step 3* above. Here, L'' represents the number of refined SIFT keypoints. These refined keypoints are classified into $\mathbf{B}_{\mathbf{n}}'$ through the orientation based classification process. Suppose $\mathbf{P} = \{p_1, \ldots, p_{L''}\}$ is a set of selected square patches that are pixel areas around the refined keypoints. Here, p_i denotes the i-th patch of I corresponding to s_i. And d_i represents p_i's associated depth patch of D. P_w and P_h are the width and height of each patch, respectively. Using the location parameters of $s_{i,y}$ *and* $s_{i,x}$, we obtain p_i with the following equation:

$$\begin{aligned} p_i = I[n][m], \quad d_i = D[n][m], \\ for \ \ \lfloor s_{i,y} \rfloor - \left(\frac{P_h}{2} - 1\right) \le n \le \lfloor s_{i,y} \rfloor + \left(\frac{P_h}{2} - 1\right), \\ \lfloor s_{i,x} \rfloor - \left(\frac{P_w}{2} - 1\right) \le m \le \lfloor s_{i,x} \rfloor + \left(\frac{P_w}{2} - 1\right) \end{aligned} \tag{16}$$

where $[n][m]$ represents the image pixel from the n-th row and the m-th column. When the watermark pattern is inserted into all the patches, the watermark can be noticeable to the viewer. Particularly, if the selected patches are overlapped on the coordinates, the watermark degrades the quality of the content. In order to avoid mutual interference between adjacent watermarks, we select a non-overlapped patch based on the orientation parameter. Before the process for the non-overlapped patch selection, the local mean and local variance for d_i are determined. The local area is defined as the $P_h \times P_w$ patch. The local mean and variance of d_i can be computed as follows:

$$\mu_{d_i} = \frac{1}{P_h \cdot P_w} \cdot \sum_{k=1}^{P_h} \sum_{l=1}^{P_w} d_i(k,l) \tag{17}$$

$$\sigma_{d_i}^2 = \frac{1}{P_h \cdot P_w} \cdot \sum_{k=1}^{P_h} \sum_{l=1}^{P_w} \left[d_i(k,l) - \mu_{d_i}\right]^2 \tag{18}$$

Here, $d_i(x, y)$ denotes the gray-level depth value of a pixel in the i-th depth patch d_i. The term σ_{d_i} is the local standard deviation.

In the 3D warping process in the DIBR system, pixels in a center image I are horizontally moved according to the corresponding relative depth value. Because the gray-level depth value d within a range from 0 to 255 is normalized to the relative depth value Z within a new range from Z_f to Z_n as defined by formula (3), a pixel of I with its corresponding large depth value is horizontally moved more than a pixel with its corresponding low depth value. Therefore, a pair of p_i and its associated depth patch d_i with a low μ_{d_i} is affected less by a synchronization attack from the DIBR operation than a pair of p_i and its associated depth patch d_i with a large μ_{d_i}. In addition, compared to a d_i with a low σ_{d_i}, a d_i with a large σ_{d_i} means that there are depth discontinuities in d_i. Because the sharp depth discontinuity of the depth map cause hole (new exposed areas) occurrences, a pair of p_i and its associated d_i with a low σ_{d_i} is affected less by a synchronization attack from the DIBR operation than a pair of p_i and its associated depth patch d_i with a large σ_{d_i}.

Based on the analysis of the relation between the patch and depth patch, we select the M_p non-overlapped patches from each bin $\mathbf{B}_\mathbf{n}^{'}$. Here, M_p represents the number of selected non-overlapped patches of each bin. p_j^n denotes the selected j-th patch of the n-th bin, and d_j^n is an associated depth patch of p_j^n, where $0 \le j \le M_p - 1$. $\mu_{d_j}^n$ and $\sigma_{d_j}^n$ are the local mean and local standard deviation of d_j^n, respectively. At first, the p_1^n with lowest μ_{d_i} is selected from $\mathbf{B}_\mathbf{n}^{'}$ for $0 \le n \le K\text{-}1$, $0 \le i \le M_n^{'} - 1$. If there are multiple patches with the same local mean value of depth patch, the patch selection is processed based on the local standard deviation of the depth patch. To deal with the repeatability issue, we eliminate the candidate patch that is overlapped with the other selected patch p_j^n. By repeating the above process, we can obtain non-overlapped patches for watermarking for each orientation based bin.

- *Step 5 (Perceptual masking)*: In order to enhance the imperceptibility of the watermark, the perceptual masking technique is exploited [26]. The insertion of the watermark must not be noticeable to the viewer and should not degrade the perceptual quality of the cover work. The perceptual masking technique is based on the noise visibility function (NVF) which characterizes the local image properties. Furthermore, the technique can identify particular regions where the watermark should be strongly inserted. In other words, NVF exploits the fact that the human visual system (HVS) cannot easily recognize the noise in textured and edge regions. Therefore, based on perceptual masking, the proposed watermarking method controls the embedding strength of the watermark. The NVF of the patch NVF_p is computed with the following formula:

$$NVF_p = \frac{1}{1 + \tau\sigma_p^2}, \quad \tau = \frac{D}{\sigma_{p_{max}}^2} \tag{19}$$

where σ_p^2 is the local variance of a patch, whose size is $P_h \times P_w$. τ represents the scaling parameter that should be computed for every image. $\sigma_{p_{max}}^2$ denotes the maximum local variance for a given I. $D \in [50,100]$ is a scaling constant that is experimentally determined. In the textured and edge regions, NVF_p approaches 0. On the other hand, NVF_p approaches 1 in the flat regions. And the local weighting factor of patch φ_p is computed as follows:

$$\varphi_p = \beta + (\gamma - \beta) \cdot NVF_p \tag{20}$$

where β and γ are set to 1 and 0.8, respectively. Using this content adaptive perceptual masking approach, we control the level of the watermark strength taking into consideration the fidelity.

- *Step 6 (Message encoding and assignment of the reference pattern)*: M represents the original message which consists of N bits. As shown in Fig. 8, the original message goes through the shuffling process using the secret key. M stands for the shuffled message to be inserted which consists of N bits represented as $b_1, \ldots, b_N$. The value of the i-th bit b_i is 1 or 0. In order to assign different reference patterns to K bins, the shuffled message M is divided into K segemeted-messages. m_i denotes the i-th segmented-message which

consists of N/K bits, where $0 \le i \le K-1$. Additionally, $2^{N/K}$ reference patterns are generated using a secret key. The reference pattern w_i follows a Gaussian distribution with a zero mean and constant variance for $0 \le i \le 2^{N/K}-1$. L_w is the vector length of the reference pattern. Suppose $D(\cdot)$ is a function for converting a binary number into a decimal number. We assign $D(m_i)$-th reference pattern to $\mathbf{B}_i^{'}$ for $0 \le i \le K-1$.

- *Step 7 (DCT and spread spectrum embedding)*: Through *steps 1–6*, the selected patch p_j^n and reference pattern w^n are assigned to the n-th bin $\mathbf{B}_n^{'}$, where $0 \le j \le M_p-1$, $0 \le n \le K-1$. Here, w^n denotes the reference pattern that is assigned to the $\mathbf{B}_n^{'}$. Taking robustness and imperceptibility into consideration, the reference pattern is embedded into the selected patch by spread spectrum embedding [3, 10]. We apply 2D–DCT to the selected patches. Then, we exploit the spread spectrum embedding scheme to insert a reference pattern into the DCT coefficients. The reference pattern is inserted into the middle band of the DCT domain. The coefficients from the (L_s+1)th to the (L_s+L_w)th in the zigzag scan ordering of the DCT domain are watermarked, according to the following formula (21):

$$s_{L_s+i}^{'} = s_{L_s+i} + (\alpha |s_{L_s+i}| w_i)\varphi_p, \; for \; 1 \le i \le L_w \tag{21}$$

where $s^{'}$ and s denote the watermarked DCT coefficients and the original DCT coefficients, respectively. w and φ_p represent the vector of the reference pattern and the local weighting factor of the patch, respectively. And α adjusts the strength of the watermark. We can adaptively adjust the embedding level for each patch according to the HVS characteristic.

- *Step 8 (Inverse DCT and patch attaching)*: The watermarked patches are reconstructed by inverse zigzag scan ordering and the inverse DCT transform. Then, based on the original coordinates of the patch, each reconstructed patch is attached to the original center image in order to generate a watermarked center image.

4.2 Watermark extraction

Figure 10 shows a diagram of the proposed watermark extraction process. The overall process can be decomposed into six steps.

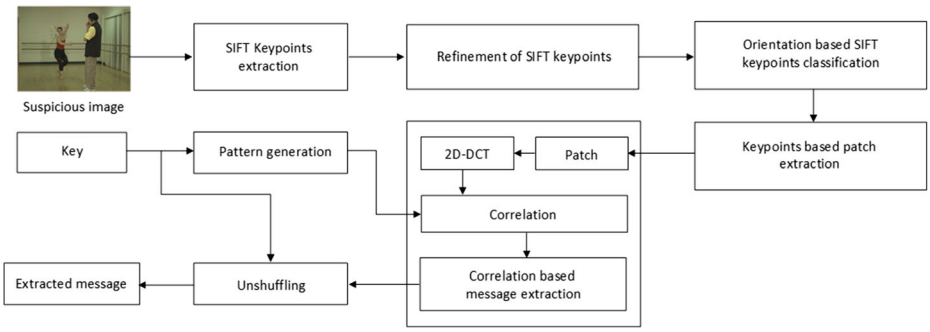

Fig. 10 Diagram of the proposed watermark extraction process

- *Steps 1, 2 and, 3 (SIFT keypoints extraction, Refinement of keypoints and Keypoints classification based on the orientation)*: As shown in Fig. 10, the first three steps are the same as those of the embedding process. Here, S is the suspicious image, and the SIFT keypoints are extracted from S. S_w and S_h are the width and height of S, respectively. Just as in the watermark embedding process, the extracted SIFT keypoints are refined. Then, the refined SIFT keypoints are classified into K bins with different degree offset values for θ_E^*. Suppose $\mathbf{S}^* = \{s_1, \ldots, s_{L^*}\}$ is a set of refined keypoints obtained. Here, L^* represents the number of refined SIFT keypoints. $\mathbf{B_n^*}$ denotes the classified n-th bin, where $0 \le n \le K-1$. M_n^* is the number of keypoints belonging to the n-th bin $\mathbf{B_n^*}$. In order to deal with the change in the orientation parameter of the keypoints from the DIBR operation, we set θ_E^* to a value less than θ_E.

- *Step 4 (Keypoints based patch extraction)*: Because we do not know which keypoints are used for watermarking, we extract patches using all the classified SIFT keypoints. Unlike the watermark embedding process, the depth image is not used in the watermark detection process taking into consideration the illegal distribution scenario. Therefore, the patch extraction processing proceeds only using the classified keypoints and formula (16). Suppose $\mathbf{P}^* = \{p_1, \ldots, p_{L^*}\}$ is a set of square patches that are pixel areas around the classified keypoints. Here, P_w and P_h are the width and height of each patch, respectively.

- *Step 5 (Correlation)*: Just as in the watermark embedding process, $2^{N/K}$ reference patterns are generated using a secret key. We apply 2D-DCT to the patches generated through *step 4*. Then, we calculate the correlation between the DCT coefficients of one of the patches and all the generated reference patterns in order to determine whether the reference pattern is present [3]. The DCT coefficients of a patch are reordered into a zigzag scan, and the coefficients from the $(L_s + 1)$ th to the $(L_s + L_w)$ th are selected. In the proposed method, we compute the correlation between the coefficients of the middle band of the DCT domain and the reference pattern, according to the following formula (22):

$$c = \frac{1}{L_w}\sum_{i=1}^{L_w} w_i s^*_{L_s+i} \quad , \quad T_c = \frac{\alpha}{\rho L_w}\sum_{i=1}^{L_w}\left|s^*_{L_s+i}\right| \tag{22}$$

  where s^* denotes the DCT coefficients of a patch in S. w represents the vector of the reference pattern, and c represents the correlation value. Here, L_w is the vector length of the reference pattern. ρ is the predefined constant value.

- *Step 6 (Correlation based message extraction)*: Through *step 5*, the correlation results between the classified patches and the reference patterns are computed. Suppose $c_{i,j}^n$ is the correlation between the i-th patch belonging to the n-th bin $\mathbf{B_n^*}$ and the j-th reference pattern w_j, where $0 \le n \le K-1$, $0 \le i \le M_n^* - 1$, $0 \le j \le 2^{N/K} - 1$. The computed correlation value is compared to a predefined threshold T_c. For each bin, the number of correlation values exceeding the threshold is counted based on the reference patterns:

$$C_j^n = \begin{cases} C_j^n + 1 & \text{if } c_{i,j}^n \ge T_c \\ C_j^n & \text{if } c_{i,j}^n \le T_c \end{cases} \tag{23}$$

$$for \;\; 0 \le n \le K-1, 0 \le i \le M_n^* - 1, 0 \le j \le 2^{N/K} - 1$$

where the initial count value C_j^n is set to 0. After that, we choose the index j with the largest count value for each bin. The target index j^n for each bin is found by maximizing the following function:

$$j^n = arg \ \max_j \left(C_j^n \right) \tag{24}$$

where $0 \le n \le K$, $0 \le j \le 2^{N/K} - 1$. In the proposed method, based on the correlation results, we conclude that the j^n -th reference pattern is embedded into the patches belonging to the n-th bin, where $0 \le n \le K-1$. In order to decode the message, we convert the index of the reference pattern into a segmented-message for each bin. Suppose $B(\cdot)$ is a function for converting a decimal number into a binary number. We can conclude that $B(j^n)$ is the segmented-message for $\mathbf{B}_\mathbf{n}^*$, where $0 \le n \le K-1$. m_n^* denotes the n-th segmented-message which consists of N/K bits. K segmented-messages are merged to generate the estimated message. The merged message goes through the un-shuffling process using the secret key. After that, we can determine the estimated message M^* which consists of N bits represented as $b_1^*, \ldots, b_N^*$. To show the effectiveness of the presented method, we compute the bit error rate (BER) in the following experiment section. The BER for the original message M and estimated message M^* is defined as follows:

$$BER\left(M, M^*\right) = \text{number of}\left(b_i \text{ in } M \neq b_i^* \text{ in } M^*\right)/N \ \ for \ 0 \le i \le N-1 \tag{25}$$

5 Experimental results

In this section, we show the performance of the proposed watermarking method in terms of robustness and fidelity to various attacks. In order to substantiate the effectiveness of our method, a series of experiments were done on 15 pairs of center and depth images. The color images and their corresponding depth images available in the Heinrich-Hertz-Institut Datasets [5], Middlebury Stereo Datasets [9, 21–23] and Microsoft Research 3D Video Datasets [29] were used in the experiments. Figure 11 shows the pairs of center and depth images, and the depth images are 8 bit gray-scale images. As listed in Table 2, for the Heinrich-Hertz-Institut Datasets, the resolution of the pairs of the center and depth image is 720×576. And, for the Middlebury Stereo Datasets, the resolution of the pairs of center and depth images ranged from 620×555 to 1800×1500. In particular, the Middlebury Stereo Datasets consist of 3D images taken under three different illuminations and with three different exposures. For the Microsoft Research 3D Video Datasets, the resolution of the pairs of the center and depth image is 1024×768, and the test image pairs contained in the Microsoft Research 3D Video Datasets are (d) and (g). The resolutions of the three test sets are different, and the size and number of objects in the image are also different. As can be seen in Fig. 11, for a fair experiment, we have chosen test sets containing objects of various sizes and numbers. And, for a diversity of stochastic properties of the test sets, we have selected 3D images with planar regions and 3D images with textured regions as test sets. Also, considering the 3D depth perception in the 3D viewing environment, DIBR 3D images with various types of depth values are selected as test sets.

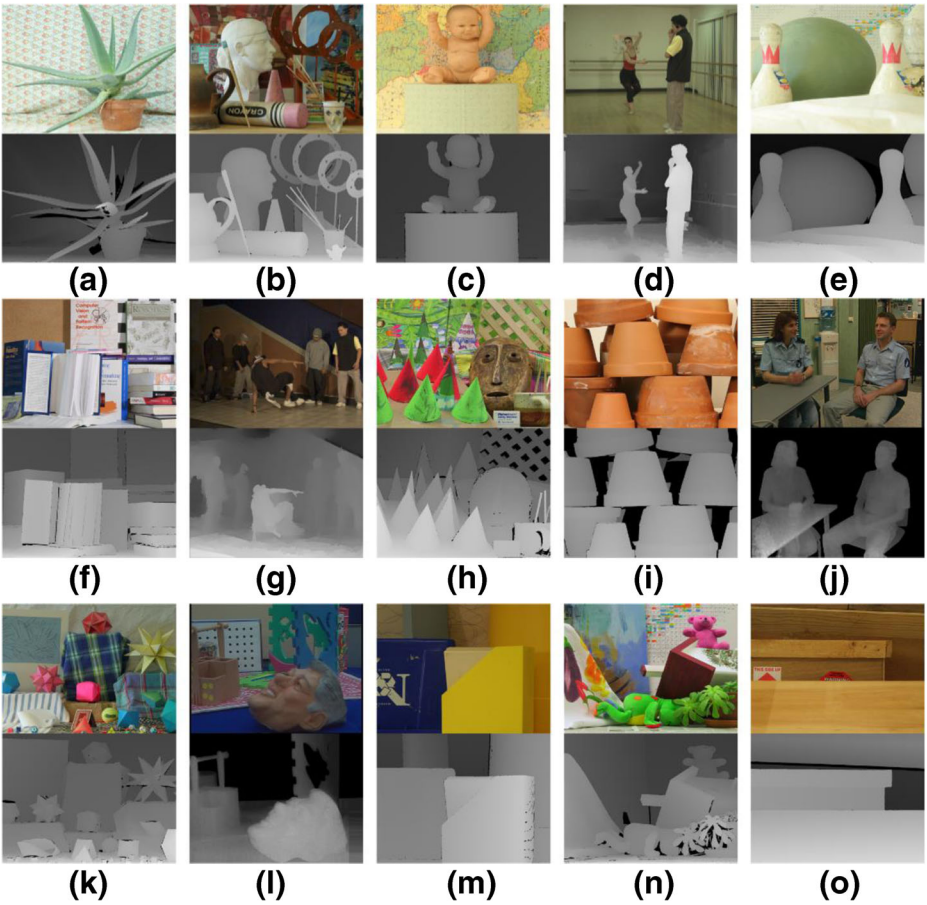

Fig. 11 Test image pairs of center and depth images: (**a**) *Aloe*, (**b**) *Art*, (**c**) *Baby*, (**d**) *Ballet*, (**e**) *Bowling*, (**f**) *Books*, (**g**) *Breakdancers*, (**h**) *Cones*, (**i**) *Flowerpots*, (**j**) *Interview*, (**k**) *Moebius*, (**l**) *Orbi*, (**m**) *Plastic*, (**n**) *Teddy*, and (**o**) *Wood*

As a comparative experiment, Lin's method in [16] and Kim's method in [11] were also applied to these test images. The two compared methods used to extract the watermark in a blind fashion are denoted as Lin's method and Kim's method. To evaluate the robustness of the watermarking methods, the BER is calculated by formula (25). Additionally, to evaluate the fidelity of the watermarking methods, objective and subjective assessment methods were exploited. The experiments were implemented in Matlab R2014a. We used the open-source software the Stirmark benchmark tool [24], which contains a number of typical attacks.

Table 2 Test sets used in experiments and their properties

Sets	Test image pair	Resolution	Image format
Heinrich-Hertz-Institut datasets	(j), (l)	720×576	BMP
Middlebury stereo datasets	(h), (n)	1800×1500, 900×750	PNG
	(a), (c), (e) (i), (m), (o)	1240×1110, 620×555	
	(b), (f), (k)	1390×1110, 695×555	
Microsoft research 3D Video datasets	(d), (g)	1024×768	JPG

5.1 Parameter decision

The maximum baseline distance t_x for the DIBR operation was set to 5% of the center image width for comfortable viewing. A t_x within a range from 3% to 5% of the image width offers a comfortable viewing experience to viewers [5, 6, 16, 28]. Without loss of generality, the focal length f was set to 1. Z_f and Z_n were set to $t_x/2$ and 1, respectively. Based on these DIBR parameters, the experiments were conducted. In the case of Lin's method, corresponding to the watermarking scenario in [16], we used two different settings for the watermarked sub-block size. In Lin's method*, the watermarked sub-block size was set to 8×8. The length of the watermarked DCT coefficients was set to 20, and the length of the skipped DCT coefficients was set to 9. In Lin's method**, the watermarked sub-block size was set to 16×16. The length of the watermarked DCT coefficients was set to 80, and the length of the skipped DCT coefficients was set to 39. α and λ were set to 1and 1, respectively. In the case of Kim's method, corresponding to the watermarking scenario in [11], *errMin, maxBit,* and *W* were set to 450, 8 and 2, respectively. The size of the sub-block was set to $(w/8 \times h/8)$ pixels. Here, w and h are the width and height of the image. The two compared methods embed the watermark into the y channel of the center image.

In the proposed method, the watermark embedding strength α has a significant effect on the robustness and imperceptibility of the watermarking scheme. Embedding watermarks will cause a perceptual distortion in the cover work. Moreover, the robustness of the watermarking scheme increases when we increase the embedding strength of the watermark. Figure 12 shows the average BER and peak signal-to-noise ratio (PSNR) of the center image with different watermark embedding strengths. As shown in Fig. 12(a), when we increase α, the robustness of the watermarking scheme increases. In particular, when the value of α is set to 0.8, the average BER nearly converges to zero. On the other hand, when we increase α, the imperceptibility of the watermarking scheme decreases shown in Fig. 12(b). The average PSNR for a value of α less than 1.2 is more than 45 dB. Table 3 shows the re-detection ratio of the keypoints between the original center image and watermarked center image with different watermark embedding strengths. The re-detection ratio of the keypoints shows the similarity between the keypoints extracted from the original center image and the keypoints extracted from the watermarked center image. Because embedding the watermark will cause a perceptual distortion to the original center image, α contributes to extract the keypoints that are

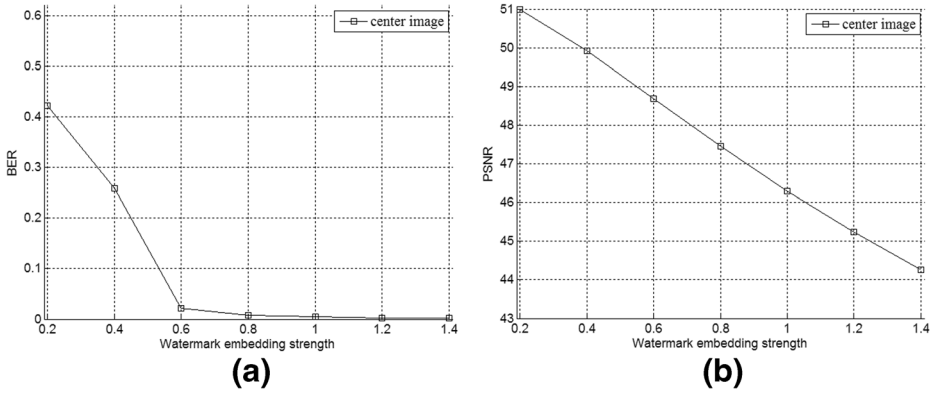

Fig. 12 (**a**) Average BER of the center image with different watermark embedding strength α, (**b**) Average PSNR between the center image and watermarked center image with different watermark embedding strength α

Table 3 Re-detection ratio of the keypoints between the original center image and watermarked center image with different watermark embedding strength α

α	0.2	0.4	0.6	0.8	1.0	1.2	1.4
Re-detection ratio	0.9356	0.9224	0.9122	0.9047	0.8981	0.8914	0.8855

slightly different from the keypoints extracted from the original center image. The re-detection ratio of the keypoints is calculated by formula (10). Here, n_m is the number of matched features between the center and watermarked center images. When we increase α, the re-detection ratio of the keypoints decreases. As shown in Table 3, for a value of α less than 0.8, the similarity between the keypoints extracted from the original image and the keypoints extracted from the watermarked image is above 90%. Therefore, considering the robustness, imperceptibility and re-detection ratio of the feature points, the parameter α of the proposed method is set to 0.8.

In order to determine the effective number of bin K, we made a histogram of the orientations of the SIFT keypoints extracted from the test sets. Figure 13 shows the histogram of orientation θ obtained from 15 pairs of DIBR 3D images. The dashed vertical lines of the histogram indicate the border of each bin, and K bins cover the 360 degrees. As the value of K increases, the capacity increases because the number of reference patterns inserted in the image increases. On the other hand, when the value of K increases, the robustness of the watermarking scheme decreases. As the number of bin increases, the degree area assigned to each bin becomes narrower, and so changes in the orientation of the keypoints due to a malicious attack can degrade robustness. Therefore, histogram analysis was performed to find the optimal K that could be used to consider robustness and capacity.

As seen in Fig. 13, the histogram has high peaks at specific degree ranges (0°, 90°, 180° and 270°). It also shows that many of the keypoints have an orientation parameter belonging to specific angle ranges. The local gradient within the region area of the keypoints has a dominant direction in the horizontal and vertical directions. This means that the keypoints extracted from the center images of the test sets have horizontal and vertical orientation parameters. When we set the number of bins to 4, we can see in Fig. 13 that the dominant orientations are stably

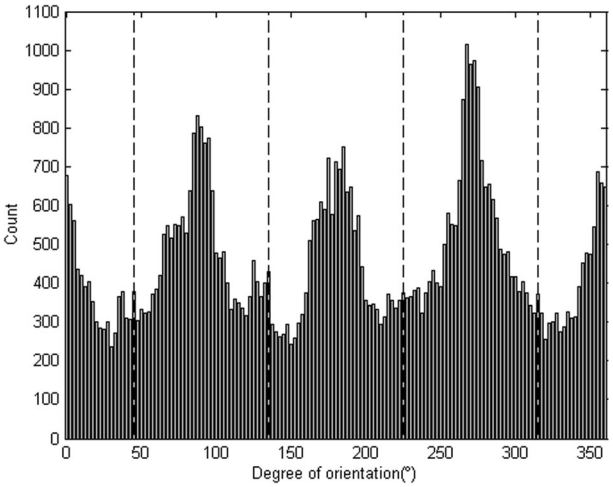

Fig. 13 Histogram of the orientation parameter of keypoints

contained in the bin. Here, θ_S is set to 45°. In the proposed method, based on the keypoints contained in each bin, M_p non-overlapped patches are obtained. If the number of keypoints allocated to the bin is not sufficient, the probability of extracting fewer than M_p non-overlapped patches increases. This affects the robustness of the watermarking technique. Thus, in the experiments, K, representing the number of bins, is empirically set to 4 by taking into consideration the analysis of the orientation of keypoints.

And, the size of each patch ($P_h \times P_w$) is set to 32×32 pixels. The number of non-overlapped patches of each bin M_p is set to 15. θ_S is set to 45°, and θ_K is set to 90°. θ_E is set to 2°, and θ_E^* is set to 1.5°. The length of the reference pattern L_w is set to 320, and we embed the reference pattern in the 120-th position of the zigzag scan ordering of the DCT domain. The constant value ρ is set to 2. In the experiments, we embed 12 bits of the watermark into the y channel of the center image considering the tradeoff between the robustness and the imperceptibility. Additionally, comparative experiments were done in the same conditions as the 12 bits of capacity.

5.2 Fidelity test

Based on the parameter decision, objective and subjective assessment methods for image quality were exploited. In order to evaluate the objective perceptual quality of the watermarked content, we calculated the PSNR and structure similarity (SSIM) between the watermarked center image and original center image. Table 4 shows the experimental results of the objective fidelity test. As shown in Table 4, the proposed method showed higher quality measures than that of the other methods for the average PSNR and SSIM. Because our method embeds the watermark into some of the areas around the classified keypoints, only parts of the original image are altered unlike the other methods that modify the overall original image. Since, for robustness, Kim's method strongly quantizes the sub-bands of the DT-CWT coefficients, Kim's method in PSNR and SSIM measurement experiments showed the worst performance among the three methods. The average PSNR and SSIM of Lin's method* and Lin's method** are 42.27 dB and 0.995, respectively. Lin's method has a higher PSNR than that of Kim's method but a lower PSNR than that of the proposed method. In the Lin method, the fidelity of a watermarked image is degraded since the watermarks are inserted into all blocks after dividing the original image into blocks.

On the other hand, the proposed method has a high fidelity because it inserts the watermarks only in the patches obtained based on the extracted refined keypoints. The average PSNR of the proposed method is 46.89 dB, which is higher than the results of the comparison methods. Furthermore, the average SSIM of the proposed method for the test set arrived to 0.998, which is higher than that of the comparison methods. As a result, the proposed method achieved a higher average PSNR and SSIM value than that of Lin's method and Kim's method. In terms of the objective perceptual quality, the proposed method showed good performance relative to the other methods.

Table 4 Average PSNR and SSIM for the proposed method, Lin's method*, Lin's method** and Kim's method

	PSNR	SSIM
Proposed method	46.89 dB	0.998
Lin's method*	42.17 dB	0.994
Lin's method**	42.36 dB	0.996
Kim's method	41.84 dB	0.990

For the subjective quality analysis, two types of experimental systems were used: 1) a passive 3D based experimental system and 2) an active 3D based experimental system. The passive 3D based experimental system consisted of a 27-in. LG Cinema 3D Smart TV 27MT93D, a SAPPHIRE RADEON R9 290 Tri-X D5 4GB, and Polarized 3D Glasses. The active 3D based experimental system consisted of a 23-in. LG Platron full HD 3D, a NVIDIA GeForce GTX 460, and 3D Vision active shutter glasses. The default refresh rate setting of the active 3D based monitor was 120 Hz. Based on the Double Stimulus Continuous Quality Scale (DSCQS) method recommended by the ITU-R [2], the subjective quality scores, which indicate the similarity between the original and marked images were evaluated. The left side of Fig. 14 shows the grading scale for the mean opinion score (MOS), and the right side of Fig. 14 shows the stimulus presentation structure in the subjective fidelity test. In the DSCQS method, shown in Fig. 14, the similarity of a pair of images consisting of the watermarked center image and the original center image was evaluated with a five-grade continuous scale where 1 = Bad, 2 = Poor, 3 = Fair, 4 = Good, and 5 = Excellent. The test images were presented in random order. Twenty subjects participated in the experiment and blindly evaluated the subjective quality of 15 test images by measuring the MOS.

Table 5 shows the results of the subjective fidelity test of the 2D and 3D views. Like the objective fidelity test, the result shows that the proposed method can produce good performance relative to the other methods in terms of subjective perceptual quality. Additionally, the results show that both the proposed method and all comparison methods received higher scores for a 3D viewing experience than for a 2D viewing experience. Furthermore, for the "*Teddy*" image, the subjective perceptual quality of the proposed method is shown in Fig. 15. It was observed that there is no perceptual difference between the original image and the watermarked image. In the magnified images at the bottom of Fig. 15, there is no visual artifact caused by watermark embedding.

5.3 Robustness test: DIBR operation with a predefined baseline

In this paper, BER for the original message M and the estimated message M^* is used to measure the robustness of a watermarking method against various attacks. In comparative robustness test experiments, a watermark is embedded into a center image and left and right images are then synthesized by means of DIBR operation. To deal with the illegal distribution of DIBR 3D images, the watermark should be extracted from the center, the synthesized left

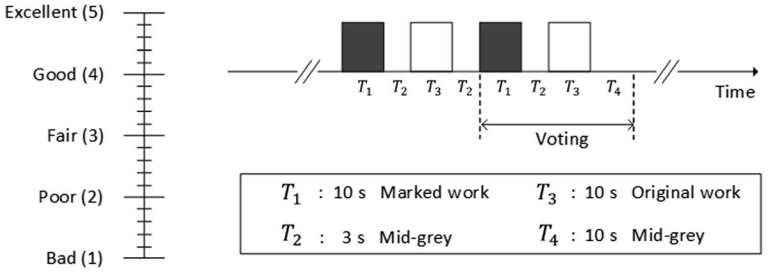

Fig. 14 Grading scale of the MOS and stimulus presentation structure in the DSCQS method

Table 5 Average MOS for the proposed method, Lin's method*, Lin's method** and Kim's method

	Watermarked monoscopic view	Watermarked stereoscopic view
Proposed method	4.55	4.73
Lin's method *	4.34	4.58
Lin's method **	4.37	4.60
Kim's method	4.23	4.54

and the synthesized right images. The left and right images were synthesized by a DIBR system with a predefined baseline distance t_x, which was set to 5% of the center image width. A detailed description of the DIBR operation is given in section 2.

As listed in Table 6, without distortion, the proposed method and all comparative methods showed a low BER for the center image. For the center image, the proposed method showed the lower BER, i.e., 0.002 in this case. The BER of Lin's method was the lowest among the three methods. On the other hand, Kim's method showed the worst performance with a BER of 0.007. For left and right images, without distortion, the proposed method showed lower BER value than both Kim's method and Lin's method. The average BER values for the left and right images of the proposed method are 0.008 and 0.009, respectively. In the robustness test, Lin's method showed excellent performance for the center image, but showed the worst performance for the left and right images. Kim's method showed the highest BER for the center image, but showed better performance than that of Lin's method for the left and right images. Thus, the proposed method demonstrated the stronger robustness against DIBR operation with a predefined baseline distance as compared to both Lin's method and Kim's method.

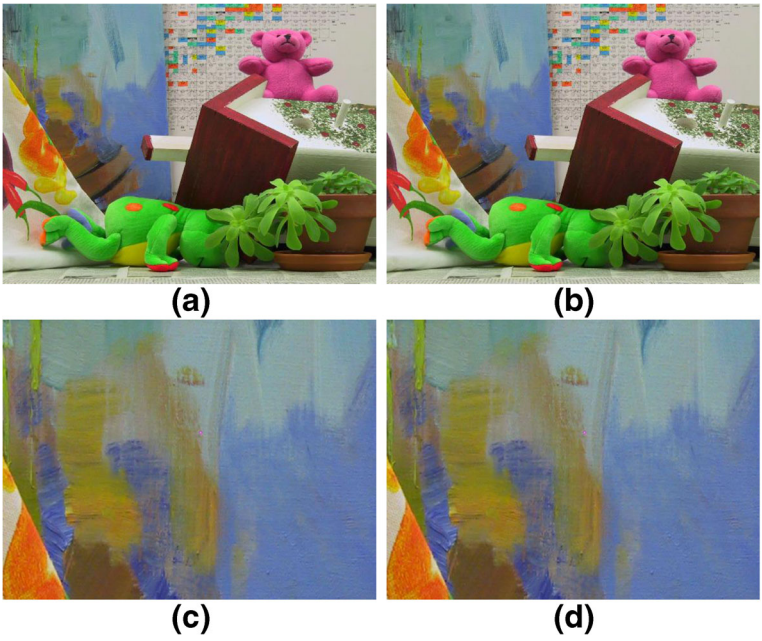

Fig. 15 Subjective performance evaluation of the watermarked center image of the "*Teddy*" image: (**a**) Original image, (**b**) Proposed method, (**c**) Magnified regions of (**a**), (**d**) Magnified regions of (**b**)

Table 6 Average BER values of center, left and right images for the proposed method, Lin's method*, Lin's method** and Kim's method without distortion

	Center image	Left image	Right image
Proposed method	0.002	0.008	0.009
Lin's method *	0	0.053	0.050
Lin's method **	0	0.062	0.058
Kim's method	0.007	0.018	0.022

5.4 Robustness test: baseline distance adjustment and pre-processing of a depth image

In the above section, virtual view images, in this case left and right images, are synthesized by a DIBR system with a predefined baseline distance t_x. One of the advantages of a DIBR system is that they provide a customized 3D experience by adjusting for different depth conditions. In other words, the DIBR system enables viewers to control the parallax of two synthesized views to achieve the experience of 3D depth perception taking into consideration user preferences. This baseline distance adjustment can be regarded as a synchronization attack, as it affects pixels which are horizontally warped to a new coordinate according to the corresponding depth. If the baseline distance t_x is large, the amount by which the pixels in the center image are horizontally moved is also greater. In this experiment, t_x was set to range from 3% to 7% of the image width.

In Lin's method, to deal with a synchronization attack from the DIBR operation, on the watermark embedder, this scheme estimates the virtual left and right images from the center image and its depth map using information about the DIBR operation with a predefined baseline distance. In Lin's method, a predefined baseline distance t_x was set to 5% of the image width during the embedding procedure. As shown in Fig. 16 (a), when the baseline distance ratio is close to 5%, Lin's method* shows the lowest BER, in this case 0.053. However, when baseline distance ratio was changed from 5%, the BER in Lin's method increased. In Kim's method, to deal with a synchronization attack from a baseline distance adjustment, the authors exploit the characteristic of an approximate shift invariance of the DT-CWT domain. Therefore, Kim's method showed lower BER for various baseline distance ratios. With consideration of baseline distance adjustments, the proposed method exploits the invariability

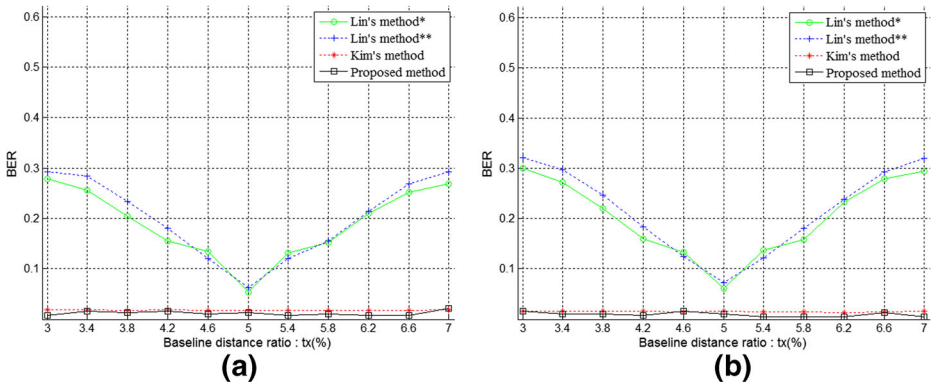

Fig. 16 (**a**) Average BER values of the proposed method, Lin's method and Kim's method for various baseline distance ratios, and (**b**) average BER values of the proposed method, Lin's method and Kim's method with pre-processing of a depth map for various baseline distance ratios

of the SIFT parameters after the DIBR operation. The average BER of the proposed method for various baseline distance ratios from 3.0 to 7.0 is 0.012. The average BER by the proposed method is slightly higher than that in Kim's method but is much less than that by Lin's method for various baseline distance ratios.

In the DIBR system, pre-processing of the depth map is employed for the generation of a natural virtual view. During the pre-processing of the depth map, the depth map is smoothed by a Gaussian filter to reduce the occurrence of holes. In addition, the depth value of the filtered depth map effects the DIBR operation. In this experiment, the depth map is pre-processed by an asymmetric smoothing filter for which $\sigma_h = 10$ and $\sigma_v = 70$. Fig. 16(b) shows the average BER values of the proposed method, Lin's method, and Kim's method with the pre-processing of the depth map for various baseline distance ratios. Like in Fig. 16(a), when the baseline distance ratio is close to 5%, Lin's method* showed the lowest BER, in this case 0.062. The average BER of the proposed method for various baseline distance ratios from 3.0 to 7.0 is 0.013. For Lin's method with the pre-processing of the depth map, the average BER is higher than the results of the Lin's method without the pre-processing of the depth map. Due to the effect of the pre-processed depth map, Lin's method showed higher BER. However, both the proposed method and Kim's method demonstrated robustness against a pre-processing depth map. Both the proposed method and Kim's method showed lower BERs than that by Lin's method, as pre-processing with the asymmetric filter can reduce artifacts and distortions in the synthesized image.

5.5 Robustness test: signal distortion and geometric distortion

In the sections above, without distortion, the proposed method successfully extracts an embedded message from a center image and a synthesized image. For a DIBR-based broadcasting system, however, a malicious adversary can illegally distribute both a center image and a synthesized virtual image as 2D and 3D content, respectively. These illegally distributed images can be distorted by the typical attacks and malicious attacks. These common attacks, such as signal processing distortion and geometric distortion, can degrade the watermarked image and desynchronize the synchronization of the watermark. In order to demonstrate the effectiveness of the proposed method, we attempted to extract a watermark from synthesized left images after applying various attacks, in this case additive noise, JPEG compression, median filtering, Gaussian filtering, cropping and translation. In the experiments, we used the Stirmark benchmark tool [24] and Matlab functions in order to apply various types of distortion to synthesized images generated from the watermarked center image. In this experiment, t_x was set to 5% of the image width.

As shown in Fig. 17, for the additive noise, the proposed method showed a lower BER value than both Lin's method* and Lin's method**. When the variance of noise is 5.0×10^{-4}, the BER value of the proposed method is 0.082. For different variances of noise, Kim's method demonstrated robustness against additive noise. The average PSNR for variance of noise exceeding 7.0×10^{-4} is less than 25 dB, indicating serious degradation of the watermarked image. For additive noise attack, the performance of the presented method is unstable but acceptable. Figure 17(b) shows the average BERs of distorted synthesized images under JPEG compression. When the JPEG quality is 75, the BER value of the proposed method is 0.029. Although the proposed method showed a slightly higher BER value than Kim's method, it demonstrated stronger robustness than Lin's method. The average PSNR for JPEG quality of less than 50 is less than 34 dB. When the JPEG quality is lower than 100, the

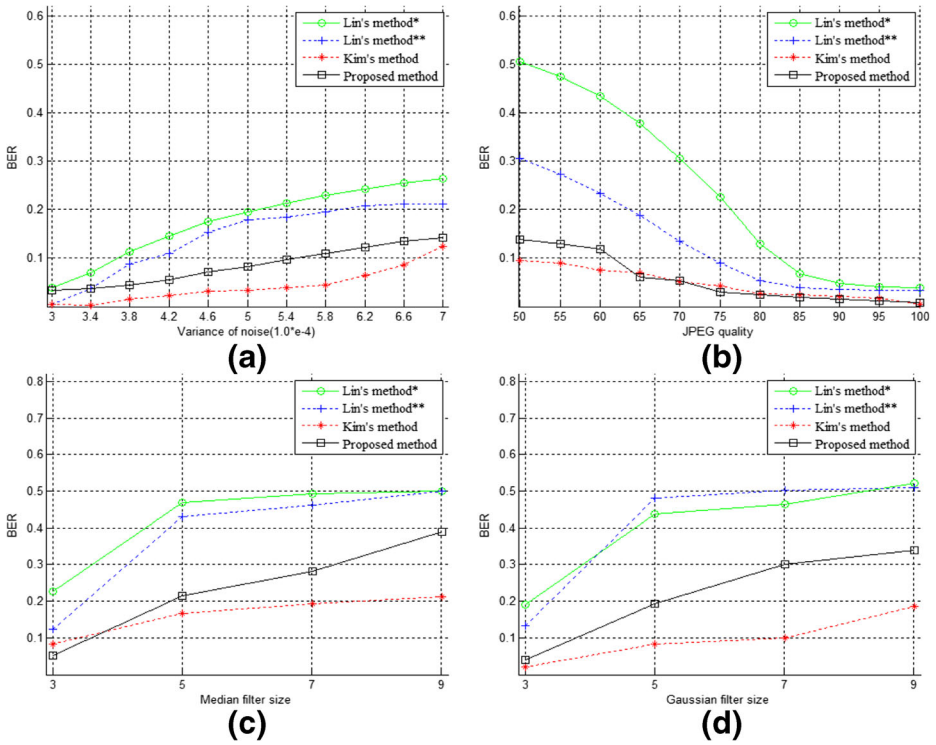

Fig. 17 Average BER of distorted left images for various types of signal distortion: (**a**) additive noise, (**b**) JPEG compression, (**c**) median filtering and (**d**) Gaussian filtering

performance of the proposed method is better than both Lin's method* and Lin's method**. For additive noise and JPEG compression, the proposed method showed sufficient robustness against the level of attacks that can be applied in the real world.

For filtering attacks, the average PSNR when the median filter size exceeds 7 is less than 29 dB, and the average PSNR when the size of the Gaussian filter exceeds 7 is less than 30 dB. When the filter size is 3, the BER values of the proposed method for the Median filter and the Gaussian filter are 0.051 and 0.042, respectively. Under the same conditions, for Kim's method, the BER for the Median filter is 0.083 and the BER for the Gaussian filter is 0.025. For median filtering, both the proposed method and Kim's method showed better performance than Lin's method for a moderate filter size. As shown in Fig. 17(d), the robustness of the proposed method against Gaussian filtering is demonstrated for a moderate filter size within a range of 3 to 5. For filtering attacks, the performance of the presented method is unstable but acceptable. For signal processing distortion, Kim's method showed the best performance because the technique strongly quantized the DT-CWT coefficients in the watermarking process. Due to the trade-off between imperceptibility and robustness, Kim's method is robust against signal processing attacks, but shows the worst performance in the fidelity test. Since the proposed method is designed with a consideration of the trade-off between imperceptibility and robustness, it shows robustness against signal processing attacks at a level that can be applied in the real world and shows the best performance in the fidelity test, as shown in section 5.2.

Internet websites such as YouTube which provide new types of content sharing have received much attention by users who seek to find content which interests them. However, malicious users degrade the contents and then illegally distribute the distorted contents without the consent of the copyright holder. Figure 18 shows distorted images after a cropping attack and a translation attack, respectively. The cropping attack and the translation attack frequently occur in relation to instances of illegal distribution. These geometric attacks desynchronize the synchronization of watermarks. For geometric attacks, the proposed method showed better performance than both Kim's method and Lin's method. Because the proposed method embeds a watermark into the patches that are neighboring pixels within the region around refined keypoints, only parts of the original image are altered, unlike other methods which modify the overall original image. Therefore, the proposed method is robust against synchronization attacks such as cropping and translation. As shown in Fig. 19, the proposed method showed much lower BERs than both Kim's method and Lin's method during a cropping attack and a translation attack, respectively. While the proposed method maintained a low BER for various cropping factors and translation factors, Kim's method and Lin's method showed large increases in BER as the factors increased.

Moreover, in order to verify the robustness the proposed method, affine transformation, which is a general type of geometric distortion, is considered. In this experiment, we exploit the affine transformation formula and eight matrices, as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} p_1 & p_2 \\ p_3 & p_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{aligned} M_1 &= \begin{bmatrix} 1.00 & 0.00 \\ 0.01 & 1.00 \end{bmatrix}, M_2 = \begin{bmatrix} 1.00 & 0.00 \\ 0.02 & 1.00 \end{bmatrix}, M_3 = \begin{bmatrix} 1.00 & 0.01 \\ 0.00 & 1.00 \end{bmatrix}, \\ M_4 &= \begin{bmatrix} 1.00 & 0.02 \\ 0.00 & 1.00 \end{bmatrix}, M_5 = \begin{bmatrix} 1.00 & 0.015 \\ 0.015 & 1.00 \end{bmatrix}, M_6, = \begin{bmatrix} 1.010 & 0.013 \\ 0.009 & 1.011 \end{bmatrix}, \\ M_7 &= \begin{bmatrix} 1.007 & 0.010 \\ 0.010 & 1.012 \end{bmatrix}, M_8 = \begin{bmatrix} 1.013 & 0.008 \\ 0.011 & 1.008 \end{bmatrix} \end{aligned} \tag{26}$$

Fig. 18 (**a**) Distorted image (618 × 821) after a cropping attack with a cropping factor of 20, (**b**) Distorted image (1024 × 768) after a translation attack with a translation factor of 10

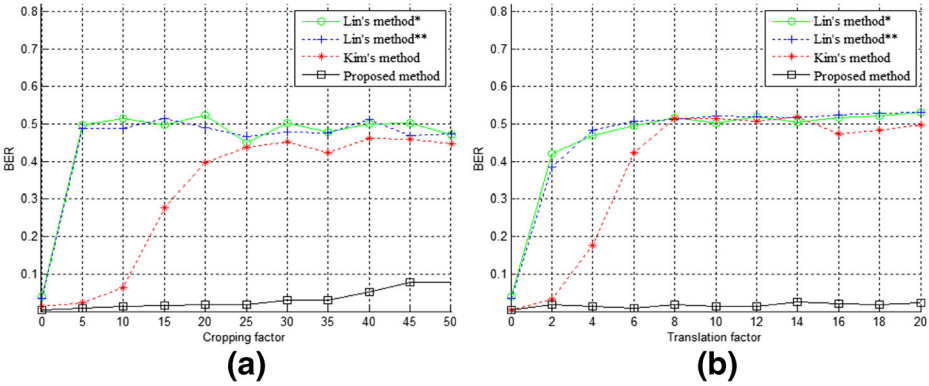

Fig. 19 Average BER of distorted left images for geometric distortion: (**a**) cropping and (**b**) translation

Here, x and y are the pixel coordinates, and x' and y' are the new pixel coordinates according to the affine transformation. The five matrices from M_1 to M_5 represent the shearing attack. The matrices from M_6 to M_8 indicate a combined attack of the shearing attack and the scaling attack. As shown in Fig. 20, the proposed method and Kim's method outperform Lin's method for the eight samples of affine transformation. The average BER of the proposed method for 8 types of affine transform attacks is 0.037, and the scheme has a fairly good performance compared to Lin's method. On the other hand, Lin's method showed high BER values for all types of affine transform attacks. Lin's method divides the original image into small blocks and inserts the watermark into each block. Therefore, this scheme cannot extract the watermark properly if the synchronization of watermarked blocks is broken. Since the affine transform adversely affects the synchronization of the watermarked blocks, Lin's method has a higher BER than those of the other two schemes. Kim's method showed the best performance among the three methods. Although the proposed method shows slightly poorer performance than Kim's method, its robustness against combined geometric distortions is acceptable.

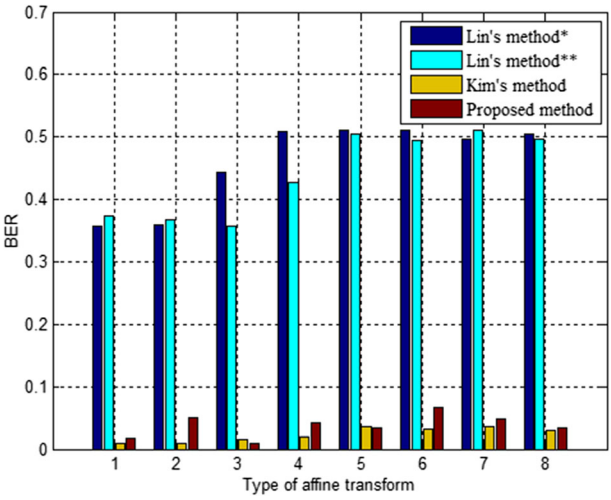

Fig. 20 Average BER of the proposed method, Lin's method and Kim's method for various affine transforms

5.6 Computational complexity

The computational complexity of the proposed watermarking technique is as follows. Suppose the resolution of the center image $I(x, y)$ is $N \times M$. When the size of the patch used in the watermarking process is $n \times m$, the computational complexity of 2D DCT and watermark embedding for each patch is calculated as $O(nm(n+m))$. In the same way, the computational complexity of watermark extraction for each patch is calculated as $p \times O(nm(n+m)) \approx O(pnm(n+m))$. Here, p means the number of reference patterns to be compared. In the case of the SIFT algorithm, the computational complexity is analyzed for three aspects [4, 18, 25]: 1) scale space construction, 2) extrema detection and keypoint detection, and 3) local image gradient based orientation assignment.

The Gaussian pyramid of $I(x, y)$ is composed of k octaves, and each octave contains $s+3$ Gaussian images. The computational complexity when a single Gaussian image is generated is $O(w^2NM)$. Here, w represents the size of the Gaussian filter. The computational complexity for computing all $s+3$ Gaussian images over one octave is $O(w^2NM(s+3)) \approx O(w^2NMs)$. Therefore, the computational complexity that produces a scale space consisting of k octaves is $O\left(\sum_{j=0}^{k-1} \frac{s}{2^j} w^2 NM\right) \approx O(w^2NMs)$. And, the computational complexity for computing all $s+2$ differences of the Gaussian images in octave j is calculated as $O\left(\frac{(s+2)NM}{2^j}\right) \approx O\left(\frac{NMs}{2^j}\right)$. Thus, the computational complexity that produces the differences of Gaussian images across all k octaves is $O\left(\sum_{j=0}^{k-1} \frac{1}{2^j} NMs\right) \approx O(NMs)$. And, the computational complexity for detecting the extremas using the DOG images generated earlier is $O\left(\sum_{j=0}^{k-1} \frac{(s+2)}{2^j} NM\right) \approx O(NMs)$. Suppose the number of extremas extracted is αNM. After the elimination of unstable extremas, the extremas that remain become keypoints. The computational complexity for detecting the keypoints is $O(\alpha NMs)$.

Suppose the number of keypoints is L, where $L \ll NM$. In this case, the computational complexity of the orientation assignment is $O(Ls)$. And, the computational complexity of the SIFT algorithm is $O(w^2NMs) + O(NMs) + O(NMs) + O(\alpha NMs) + O(Ls) \approx O(NMs)$. Since the process of extracting refined keypoints from all keypoints is a relatively small operation, it is excluded from the time complexity analysis. We assume that the number of refined keypoints is L', where $L' < L$. Therefore, the computational complexity of watermark embedding is $O(NMs) + L' \times O(nm(n+m)) \approx O(NMs) + O(L'nm(n+m))$. And, the computational complexity of watermark extraction is $O(NMs) + L' \times O(pnm(n+m)) \approx O(NMs) + O(L'pnm(n+m))$.

Also, in order to analyze the computational complexity between the proposed method and the two compared methods, we conducted a computation time measurement experiment. The measurement experiments were implemented in Matlab R2014a, and we conducted the experiment on a computer with a 4.00 GHz Intel Core(TM) i7-4790 K with 16 GB RAM. The measurement results of the average computation time are listed in Table 7. For Lin's method*, the average watermark embedding and extraction times are 13.085 s and 6.722 s, respectively. The average watermark embedding and extraction times of Lin's method** are 7.076 s and 3.714 s, respectively. In the watermark embedding and extraction process, Lin's method** has a smaller average computation time than that of Lin's method* because Lin's method** exploits larger blocks than does Lin's method*. For Kim's method, the average watermark embedding and extraction times are 4.245 s and 3.276 s, respectively. The Kim's method showed the best performance in the computation time measurement experiment.

Table 7 Average computation times of the proposed method, Lin's method, and Kim's method

	Watermark embedding (s)	Watermark extraction (s)
Proposed method	4.542	6.123
Lin's method *	13.085	6.722
Lin's method **	7.076	3.714
Kim's method	4.245	3.276

The average watermark embedding time of the proposed method, including time for the SIFT algorithm, is 4.542 s. And, the average watermark extraction time of the proposed method, including time for the SIFT algorithm, is 6.123 s. The average computation time of the SIFT algorithm included in the watermark embedding and extraction process is 3.592 s. The proposed method requires additional computation time to extract the SIFT keypoints in the watermarking process, and it is confirmed that the proposed method shows similar performance to that of Kim's method for the computation time of watermark embedding. For the proposed method, the watermark extraction process consumes more time than does the watermark embedding process because extraction is performed using patches obtained from the refined feature points and multiple patterns.

6 Conclusion

In this paper, we proposed a local keypoint-based blind watermarking scheme for DIBR 3D images. DIBR is a technique which is used to extend viewpoints with a monoscopic center image and an associated per-pixel depth map. In the DIBR operation, pixels in a center image are horizontally warped to a new coordinate according to the corresponding depth value. To design the proposed method robust against synchronization attacks from DIBR operation, the proposed method exploits the SIFT parameters. We showed high similarity between the SIFT parameters extracted from a synthesized virtual view and center view images. Based on patches that are neighboring pixels within the region around refined keypoints and an extended spread spectrum method, the proposed method can extract watermarks from the center image and synthesized view images. Unlike previous methods based on a local descriptor that exploit the descriptor of the original image, the proposed method can detect a watermark in a blind fashion without side information. Moreover, the experimental results show the effectiveness of the proposed method with respect to typical processes in a DIBR system, such as baseline distance adjustments and the pre-processing of a depth map. The proposed technique shows low BER values for a typical signal processing attack and geometric distortion processes such as translation and cropping. The effectiveness of the fidelity in terms of objective and subjective testing is verified through comparisons with other watermarking schemes. The future work will be mainly dedicated to apply the proposed method to different types of local features, such as SURF and ORB. Because the standard of the DIBR and 3D video coding is still being studied, future work will be also dedicated to investigating how to extend the proposed method to a depth-map-based 3D video coding standard. Furthermore, we plan to focus on improving the robustness of the proposed method against various types of distortions.

Acknowledgments This research project was supported by Ministry of Culture, Sports and Tourism(MCST) and from Korea Copyright Commission in 2017.

References

1. Amerini I, Ballan L, Caldelli R, Del Bimbo A, Serra G (2011) A sift-based forensic method for copy–move attack detection and transformation recovery. IEEE Trans Inf Forensics Secur 6(3):1099–1110
2. ASSEMBLY, ITU Radiocommunication (2003) Methodology for the subjective assessment of the quality of television pictures. International Telecommunication Union
3. Barni M, Bartolini F, Cappellini V, Piva A (1998) A DCT-domain system for robust image watermarking. Signal Process 66(3):357–372
4. Cui, Chen, Shen W et al (2017) "A novel watermarking for DIBR 3D images with geometric rectification based on feature points." Multimedia Tools and Applications 76.1: 649–677
5. Fehn C (2004, May) Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV. In Electronic Imaging 2004, International Society for Optics and Photonics, pp 93–104
6. Fehn C, De La Barré R, Pastoor S (2006) Interactive 3-DTV-concepts and key technologies. Proc IEEE 94(3):524–538
7. Feng X, Zhang W, Liu Y (2014) Double watermarks of 3D mesh model based on feature segmentation and redundancy information. Multimedia tools and applications 68(3):497–515
8. Halici E, Alatan AA (2009, November) Watermarking for depth-image-based rendering. In 2009 16th IEEE International Conference on Image Processing (ICIP), IEEE, pp 4217–4220
9. Hirschmuller H, Scharstein D (2007, June) Evaluation of cost functions for stereo matching. In 2007 I.E. Conference on Computer Vision and Pattern Recognition, IEEE, pp 1–8
10. Hou JU, Park JS, Kim DG, Nam SH, Lee HK (2014, June) Robust video watermarking for MPEG compression and DA-AD conversion. In Proceedings of the 1st international workshop on Information hiding and its criteria for evaluation, ACM, pp 2–8
11. Kim HD, Lee JW, Oh TW, Lee HK (2012) Robust DT-CWT watermarking for DIBR 3D images. IEEE Trans Broadcast 58(4):533–543
12. Lee PJ (2011) Nongeometric distortion smoothing approach for depth map preprocessing. IEEE Transactions on Multimedia 13(2):246–254
13. Lee HY, Kim H, Lee HK (2006) Robust image watermarking using local invariant features. Opt Eng 45(3): 037002–037002
14. Lee MJ, Lee JW, Lee HK (2011, October) Perceptual watermarking for 3D stereoscopic video using depth information. In Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2011 Seventh International Conference on, IEEE, pp 81–84
15. Lee JW, Kim HD, Choi HY, Choi SH, Lee HK (2012, February) Stereoscopic watermarking by horizontal noise mean shifting. In IS&T/SPIE Electronic Imaging, International Society for Optics and Photonics, pp 830307–830307
16. Lin YH, Wu JL (2011) A digital blind watermarking for depth-image-based rendering 3D images. IEEE Trans Broadcast 57(2):602–611
17. Lindeberg T (1994) Scale-space theory: a basic tool for analyzing structures at different scales. J Appl Stat 21(1–2):225–270
18. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
19. Miao H, Lin YH, Wu JL (2014, October) Image descriptor based digital semi-blind watermarking for DIBR 3D images. In International Workshop on Digital Watermarking, Springer International Publishing, pp 90–104

20. Mikolajczyk K, Schmid C (2002) An affine invariant interest point detector. In Proceedings of the 7th European Conference on Computer Vision-Part I (ECCV '02), pp 128–142
21. Scharstein D, Pal C (2007, June) Learning conditional random fields for stereo. In 2007 I.E. Conference on Computer Vision and Pattern Recognition, IEEE, pp 1–8
22. Scharstein D, Szeliski R (2003, June) High-accuracy stereo depth maps using structured light. In Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 I.E. Computer Society Conference on Vol. 1, IEEE, pp I–195
23. Scharstein D, Szeliski R, & Zabih, R (2001). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In Stereo and Multi-Baseline Vision, 2001.(SMBV 2001). Proceedings. IEEE Workshop on (pp. 131–140). IEEE.
24. Petitcolas FA, Anderson RJ, & M. G. Kuhn, (1998) "Attacks on copyright marking systems," in International workshop on information hiding. Springer, pp. 218–238
25. Vinukonda P (2011) A study of the scale-invariant feature transform on a parallel pipeline. Diss. Louisiana State University
26. Voloshynovskiy S, Herrigel A, Baumgaertner N, Pun T (1999, September) A stochastic approach to content adaptive digital image watermarking. In International Workshop on Information Hiding, Springer Berlin, Heidelberg, pp 211–236
27. Wang S, Cui C, Niu X (2014) Watermarking for DIBR 3D images based on SIFT feature points. Measurement 48:54–62
28. Zhang L, Tam WJ (2005) Stereoscopic image generation based on depth images for 3D TV. IEEE Trans Broadcast 51(2):191–199
29. Zitnick CL, Kang SB, Uyttendaele M, Winder S, Szeliski R (2004, August) High-quality video view interpolation using a layered representation. In ACM Transactions on Graphics (TOG) Vol. 23, No. 3, ACM, pp 600–608

Seung-Hun Nam is received the B.S. degree in Information Communication Engineering from Dongguk University, Seoul, Republic of Korea, in 2013, and the M.S. degree in School of Computing from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea, in 2015. He is currently pursuing the Ph.D. degree in Multimedia Computing Lab., School of Computing, KAIST. His research interest include digital watermarking and image forensics.

Wook-Hyoung Kim received his B.S. degree in electrical engineering from Hanyang University, Seoul, Korea in 2012, and M.S. degree in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejon, Republic of Korea in 2014. He is currently working toward his Ph.D. degree in Multimedia Computing Lab., School of Computing, KAIST. His current research interest include multimedia security.

Seung-Min Mun received the B.S. degree in Department of Mathematical Sciences from Korea Advanced Institute of Science and Technology (KAIST), Korea, in 2014. He received his M.S. degree in School of Computing from KAIST, in 2016. He is currently working toward his Ph.D. degree in Multimedia Computing Lab., School of Computing, KAIST. His research interests include digital watermarking for 3D mesh models and stereoscopic image.

Jong-Uk Hou received his B.S. degree in Information and Computer Engineering from Ajou University, Korea, in 2012. He received his M.S. degree in Web Science and Technology from Korea Advanced Institute of Science and Technology, Korea, in 2014. He is currently working toward his Ph.D. degree in Multimedia Computing Lab., School of Computing, KAIST. He was awarded a Global Ph.D. Fellowship from National Research Foundation of Korea in 2015. His major interests include various aspects of information hiding, multimedia signal processing, and computer vision.

Sunghee Choi received the B.S. degree in computer engineering from Seoul National University in 1995 and the M.S. and Ph.D. degrees in computer science from the University of Texas at Austin in 1997 and 2003, respectively. She is currently an associate professor of the School of Computing at Korea Institute of Science and Technology (KAIST). Her research interests include computational geometry, geometric modeling, computer graphics, and visualization.

Heung-Kyu Lee received a BS degree in electronics engineering from Seoul National University, Seoul, Korea, in 1978, and MS and PhD degrees in computer science from Korea Advanced Institute of Science and Technology, Korea, in 1981 and 1984, respectively. Since 1986 he has been a professor in the Department of Computer Science, KAIST. He has authored/coauthored over 200 international journal and conference papers. He has been a reviewer of many international journals, including Journal of Electronic Imaging, Real-Time Imaging, and IEEE Trans. on Circuits and Systems for Video Technology. His major interests are digital watermarking, digital fingerprinting, and digital rights management.