*Journal of Information Hiding and Multimedia Signal Processing* ©2016 ISSN 2073-4212
**Ubiquitous International** Volume **7**, Number **3**, May **2016**

# A Robust Depth-Image-Based-Rendering 3D Image Hashing Scheme Based on Histogram Shape

Chen Cui, Shen Wang, Xia-mu Niu

School of Computer Science and Technology
Harbin Institute of Technology
Harbin 150080,China
394874486@qq.com,shen.wang@hit.edu.cn,xiamu.niu@hit.edu.cn

Received December, 2015; revised December, 2015

Abstract. *Using the approximate invariance of the image histogram shape to geometric distortions especially the DIBR process, we propose a hashing scheme for DIBR 3D images by selecting several pixel groups to construct the image histogram shape, and computing the ratios of pixel numbers in different groups to generate the final hash sequence. Meanwhile, we use an N-times searching method to improve the robustness of proposed hashing scheme. Experimental results show that the proposed method can achieve better identification performances under geometric attacks such as rotation attacks, and provide comparable performances under classical distortions such as additive noise, blurring, and compression. Furthermore, this method can ensure that the generated virtual images could be identified with the same content as the corresponding center image when we adjust the baseline distance.*

**Keywords:** Depth-image-based rendering (DIBR), 3D image hashing, Histogram, Image identification

1. **Introduction.** Virtual viewpoint rendering is a key technology of 3D video. There are three types of traditional virtual viewpoint rendering methods, Model-based Rendering (MBR), Image-based Rendering (IBR) and Depth-image-based rendering (DIBR). MBR is only suitable for to single object 3D modeling [1]. The performance of IBR is determined by the interpolation algorithm. We need a large number of viewpoints for sampling to get 3D images with high quality. Moreover, the existing video acquisition device and digital communication network are unable to meet the need of IBR [2, 3]. In the DIBR system, we can generate the virtual left and right images with the corresponding depth image [4, 23]. As a result, it is very easy to transmit and store the DIBR 3D images (including the center image and depth image), and we can easily achieve the 3D video effects with little additional information. The natural advantages of DIBR in representation of 3D video will greatly promote the development of 3D industry, and make the 3D products spreading rapidly. The rapid development of DIBR will lead to a variety of problems about content-based identification for 3D products.

Robust images hashing has been extensively studied for content-based identification of 2D images. Generally, image hashing consists of two main aspects, one is feature extraction, and the other is feature compression. Since feature extraction can directly affect the robustness and discrimination performance of image hashing, more and more approaches have been designed to find robust feature in order to make the image hashing resist standard degradation processing and malicious attacks, just like the methods using transform domain features [5-7]. In addition, some matrix analysis approaches have also been employed to extract the perceptual features for hash computation, such as singular value decomposition (SVD) [8] and non-negative matrix factorization (NMF) [9]. Most of the hashing algorithms in the aforementioned categories can perform well in alignment-preserving manipulations, while geometric distortions such as rotation are still of great challenges. As a result, some geometric-invariant features have been exploited to devise robust hashing, among which salient points are the most commonly used features. In paper [10], Lv and Wang developed a new image hashing algorithm using local feature points to overcome the concerns presented above. They used the popular scale invariant feature transform (SIFT) to detect robust feature points and incorporated the Harris criterion to select the most stable points which are less vulnerable to image processing attacks. Then based on these robust feature points, in order to characterize local information, they introduced the shape contexts [11] into hash generation to represent the geometric distribution of the detected feature points. Unfortunately, for DIBR 3D images, the method they proposed generates different hashes for center image and virtual images, which should have the same or similar hashes. Moreover, the performance degraded when the key points detected from the test image do not coincide with that of the original. In paper [12], they designed an efficient image hashing with a ring partition and a nonnegative matrix factorization (NMF), which has both the rotation robustness and good discriminative capability. The key contribution is a novel construction of rotation-invariant secondary image, which is used for the first time in image hashing and helps to make image hash resistant to rotation. In addition, NMF coefficients are approximately linearly changed by content-preserving manipulations, so as to measure hash similarity with correlation coefficient. In [13], Roy et al. proposed a scale-invariant feature transform (SIFT) points based robust hashing, and the relationships between the positions of SIFT points and hyper planes are binarized as hash value.

Although many methods have been exploited for designing 2D image hashing, they can not been directly used for DIBR 3D image hashing algorithm. For example, in order to make their methods more robust to rotation manipulation, they divide the image to several rings with the image center as the center of rings in [10] and [12]. However, virtual images are generated from the center image with the corresponding depth information, the virtual images are different from the center image with pixels moved horizontally, and the image center will change when dividing the virtual images into several rings, which could make the hashes of virtual images different from the center image. Unfortunately, this kind of 2D image hash methods proposed recently do not take into account this problem. On the other hand, the DIBR process can be seen as a partial translation along the horizontal plane, we could consider this operation as a content-preserving manipulation, and the virtual images should be identified with the same content as the center image with same hashes as shown in Fig. 1.

In this paper, a histogram shape based hashing method is proposed for DIBR 3D images. We set the number of bins in a pixel group by analyzing the difference of mean value between original and attacked images firstly, then we use a mean value based method to select the index pixel group, and select several suitable pixel groups involving the index group to generate the final hash value. As the experimental results shown, our method is much more robust to common attacks such as JPEG compression, noise added, median filtering, scaling, rotation and cropping after rotation. In addition, the proposed method is robust to baseline distance adjusting. This paper is organized as follows. Section 2 briefly reviews the DIBR operations. In Section 3, the proposed hashing scheme based on histogram shape is illustrated. In Section 4, we analyze the robustness of proposed method to geometric distortion attacks including DIBR process. Experimental results of

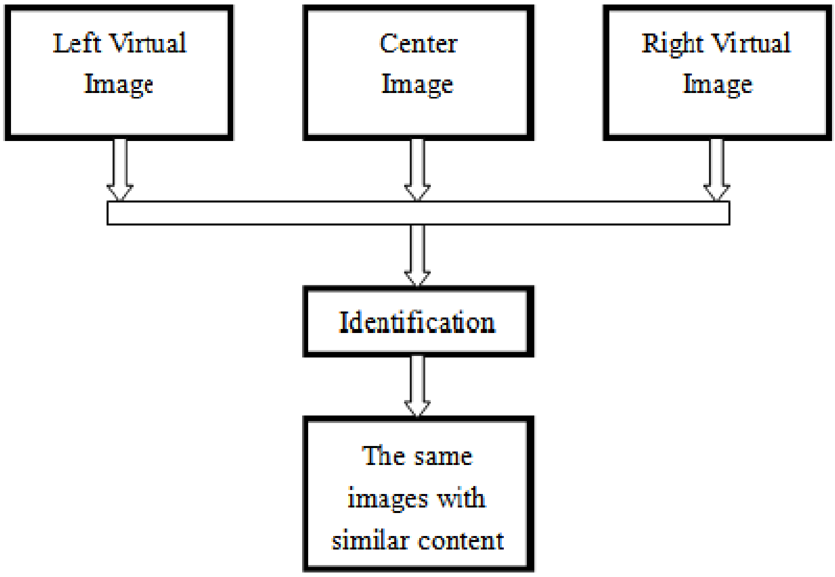


FIGURE 1. The character of image hashing for DIBR images

the proposed method are demonstrated in Section 5. Conclusions of this work are given in Section 6.

2. **Review of Depth-Image-Based Rendering Process.** The principle of binocular vision is to obtain the parallax which is related to the depth information of objects. The other way round, we can also generate the virtual views from center views by 3D warping with the depth information of objects. DIBR operations are consisted of pre-processing of the depth image, 3D image warping and hole-filling.

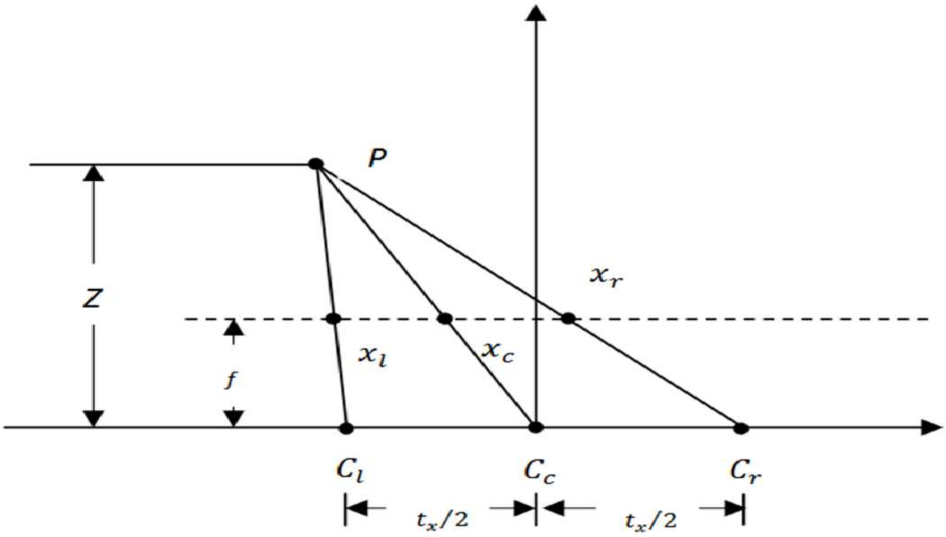


FIGURE 2. The relationship of pixel in left view, center view and right view

Zhang and Tam [14] proposed a method in their paper, by which we can use the two-dimensional image to compute the disparity and generate the virtual left and right views. As shown in Fig. 2, we can find a point $P$ in space, where $Z$ represents the depth value of the point in center view, $f$ is the focal length of the camera, $C_l$ and $C_r$ represent the left camera and the right camera respectively. $t_x$ represents the value of the baseline distance which is the distance between the left and right cameras. By the geometric relationships as shown in Fig. 2, we can conclude the following formula

$$\begin{aligned} x_l &= x_c + \tfrac{t_x}{2}\tfrac{f}{Z}, \\ x_r &= x_c - \tfrac{t_x}{2}\tfrac{f}{Z}, \\ d = x_l - x_r &= t_x\tfrac{f}{Z} \end{aligned} \tag{1}$$

$$Z(v) = Z_{far} + v \times \tfrac{Z_{near}-Z_{far}}{255} \quad v \in [0,255] \tag{2}$$

Where $(x_l, y)$, $(x_c, y)$ and $(x_r, y)$ represent the projected points in the image plane. $d$ represents the value of disparity between the left and right virtual views, the value of $f$ is set to 1 without loss of generality.

In order to compute the depth value of pixel in center view, the gray values are normalized to two main depth clipping planes, the near clipping plane $Z_{near}$ with gray level 255 and the far clipping plane $Z_{far}$ with gray level 0. According to formula (2), we can compute the depth value $Z(v)$ of $P$, where $v$ represents the gray level value.

3. **Proposed Method.** Robustness design of the proposed hashing algorithm is based on the use of the Gaussian low-pass filtering and the histogram shape invariance. As shown in Fig. 3, process of the proposed method consists of four steps: center image filtering, histogram extracting, pixel group selection, and hash generation.

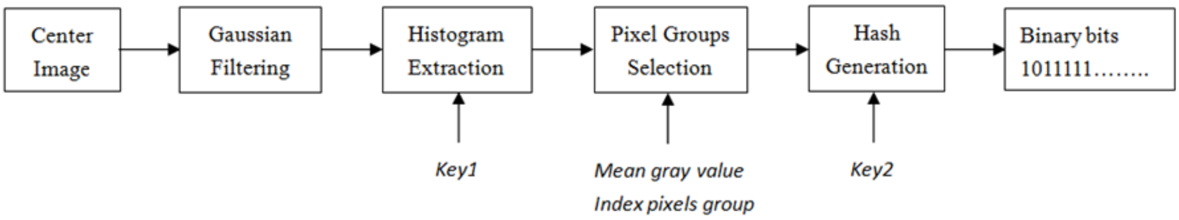


FIGURE 3. Block diagram of the image hashing

3.1. **Center image filtering.** In order to get the low frequency component $I_{low}$ suiting for hash generation, the center image is filtered with a Gaussian kernel low-pass filter $G$ for removing the high-frequency information $I_{high}$. As shown in [15], [16], low-pass filtering is an effective operation for enhancing robustness to common image processing, such as compression and filtering. The low-frequency component $I_{low}$ of image $I$ can be obtained as:

$$I_{low}(x, y) = G(x, y, \sigma) * I(x, y) \tag{3}$$

Where $*$ represents the convolution operation, and the low-pass filter Gaussian function can be represented as:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4}$$

Where $\sigma$ is the standard difference of the distribution, and usually chosen as $\sigma = 1$ [15]. In practice, the size of the Gaussian mask $G$ is often chosen as $(2{\cdot}k{\cdot}\sigma{+}1) \times (2{\cdot}k{\cdot}\sigma{+}1)$. Since 99.7% energy of the Gaussian distribution is concentrated within 3 standard differences from the mean, we can set $k = 3$. Thus the size of the Gaussian mask $G$ used in this paper is $7 \times 7$.

3.2. **Histogram extracting.** Suppose that $I_{low}$ is the filtered image with gray levels ranging from 0 to 255, the histogram $H$ indicates the number of pixels belong to every gray level. In order to increase the security of hashing algorithm, only $M$ gray levels are used for hash generation. Here, we can generate a key-based sequence $P(M) = \{p_i | i = 1...M, 0 \leq p_i \leq 255\}$. With the key-based sequence $P(M)$, we extract $M$ gray levels $g_1, g_2, ..., g_M$, where $g_i = p_i$. The shape of extracted histogram can be represented as:

$$H_M = \{h_M(g_i) | i = 1, ...., M\} \tag{5}$$

Where $h_M(g_i)$ represents the number of pixels with gray level $g_i$.

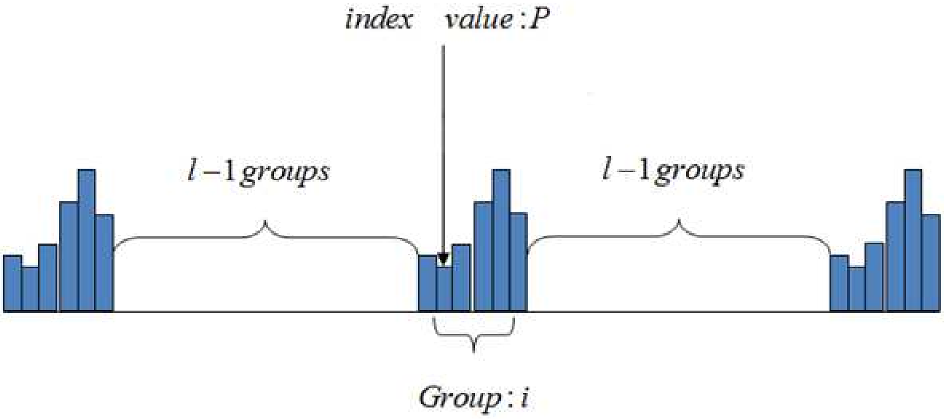


FIGURE 4. Selection of pixel groups with index value

3.3. **Selection of pixel groups.** After constructing the histogram $H_M$, we take each $L_B$ neighboring gray levels in $H_M$ to form a group. In total, one can form $M_G = \lfloor M/L_G \rfloor$ groups, where $\lfloor \bullet \rfloor$ is a floor function. It is easy to find that the number of pixels in the $ith$ group is $h_G(i) = h_M(g_{(i-1).L_G+1}) + h_M(g_{(i-1).L_G+2}) + \cdots + h_M(g_{i.L_G})$, where $i = 1, ..., M_G$. Next, we select the pixel groups that are suitable for hash generation. In this paper, we use the ratio between $h_G(i)$ and $h_G(j)$ and the number of pixels in each groups to represent the histogram shape in such a way that the effect of various attacks on the histogram shape can be objectively analyzed by observing the alteration of the ratios. Obviously, the number of pixels in a group has a crucial effect on the shape of histogram. In some groups, the number of pixels is less (zero or a few). The shape of these groups is usually unstable even if a few number of pixels in these groups change. Taking account into this problem, we only select several pixel groups suitable to construct the histogram shape. In order to find the suitable groups, we should find an index value firstly, then the pixel group involving the index value can be selected as the index pixel group, and some pixel groups nearby the index pixel group can be selected for hash generation. Here, we use the mean pixel gray value of the filtered host image as the index value, which can benefit from its outstanding robustness to content-preserving attacks [15].

As shown in Fig. 4, given a mean pixel value $P$, we can find which pixels group it belongs to as:

$$g_{(i-1).L_G+1} \leq P \leq g_{(i+1).L_G} \tag{6}$$

Then the selected pixel groups for hash generation can be modeled as:

$$G(P) = [h_G(i-l) \cdots h_G(i) \cdots h_G(i+l)] \tag{7}$$

3.4. **The final hash generation.** The insensitivity of the histogram shape plays an important role in our proposed method since it is prerequisite for withstanding geometric attacks. To enhance the robustness of the hash, the first step is to filter an image. Once the low-pass filtering process is finished, the histogram is extracted from the resulting image by referring to the mean gray value and index pixel group. Then, the hash bits are computed by comparing the population among each two different pixel groups. Let us describe the hash extraction process as follows. Denote the number of pixels in the $ith$ group as $h_G(i)$. Then we can generate one bit hash value with the two different groups $\{h_G(i), h_G(j)\}$ satisfying the condition $1 \leq i < j \leq 2l+1$. In this case, the number of groups is totally $C_{2l-1}^{2} = \frac{2(2l-1)(l-1)}{2}$. For $\{h_G(i), h_G(j)\}$, we can get a binary value by computing the ratio between $h_G(i)$ and $h_G(j)$, formulated as:

$$bit = \begin{cases} 1 \quad if \quad h_G(i)/h_G(j) \geq 1 \\ \\ 0 \quad if \quad h_G(i)/h_G(j) < 1 \end{cases} \tag{8}$$

4. **Analysis of robustness to geometric attacks.** The performance of proposed method depends on two factors, one is the computation of the index mean value and selection of index pixel group, and the other is the ratios of numbers of pixels in two groups in selected pixel groups. For a original image $I$, $I_D$ is a distorted and content-preserving versions of $I$, suppose that the index value $P$ of the original image $I$ belongs to the $ith$ group$h_G(i)$, if the index value $P'$ of $I_D$ can be estimated and the pixels group can be selected as former, and the ratios are slightly modified the both images can generate the nearly same hash values.

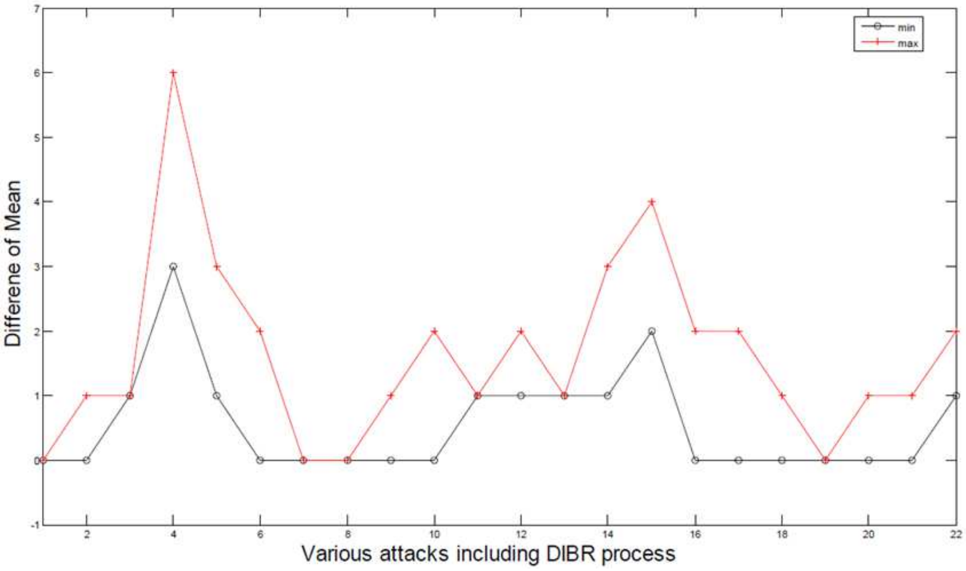


FIGURE 5. Differences between the mean value of attacked images and original images

4.1. **Robustness of the mean value and index group.** As described in section 3, we compute the mean pixel gray value and decide which index pixel group the mean value belongs to. Finally, we can select the $2l+1$ pixel groups to generate the hash value. Obviously, the computation of mean value and selection of index pixel group may affect the robustness of hashing method. Note that after geometric attacks, an interpolation process is generally needed for images [15]. The interpolation process will introduce interpolation errors such as aliasing, blocking and blurring. However, in order to maintain the perceptual quality of the attacked image to an acceptable level, the errors introduced by interpolations are often small and thus will be ignored in the following analysis.

Assume that the original image $I$ has $N = RC$ pixels, where $R$ and $C$ denote the number of rows and columns of the image, respectively. The mean pixel gray value $P$ can be computed as:

$$P = \frac{1}{RC} \sum_{i=0}^{255} h_M(g_i) \cdot g_i \tag{9}$$

Where $I(i, j)$ represents the gray level value of a pixel in the original image.

We first consider scaling attack with scaling factor $\alpha$ and $\beta$ in vertical and horizontal directions, respectively. After scaling, $R^{'} = \alpha R \text{and} C^{'} = \beta C$, which lead to:

$$N^{'} = R^{'}C^{'} = \alpha\beta N = \alpha\beta RC \tag{10}$$

$$h_M^{'}(g_i) = \alpha\beta h_M(g_i) \tag{11}$$

$$P^{'} = \frac{1}{R^{'}C^{'}} \sum_{i=0}^{255} h_M^{'}(g_i) \cdot g_i \tag{12}$$

$$\begin{aligned} P^{'} &= \tfrac{1}{R^{'}C^{'}} \sum_{i=0}^{255} h_M^{'}(g_i) \cdot g_i \\ &= \tfrac{1}{\alpha\beta RC} \sum_{i=0}^{255} \alpha\beta h_M(g_i) \\ &= \tfrac{1}{RC} \sum_{i=0}^{255} h_M(g_i) \cdot g_i \\ &= P \end{aligned} \tag{13}$$

Then we consider rotation and affine attacks. It is obvious that rotation only makes the pixels change their positions. Similarly, affine transform also only causes pixel position displacements as it can be regarded as a combination of scaling, rotation, translation and shearing attacks. So the mean pixel gray value is robust to rotation and affine attacks.

Strictly speaking, the robustness of mean pixel gray value under cropping attacks depends on the image, the cropped area and the function for selecting the gray levels. So the invariance property of the mean value of an image to cropping attacks is an approximate invariance. Similarly, in a DIBR process, the virtual images is generated from the center image with a small parts removed and the holes filled, so the mean value of virtual images is most close to the value of host center image.

We compute the mean pixel gray value of 16 test images under several kinds of attacks including DIBR process, then the difference of the mean between attacked images and original images can be computed. As shown in Fig. 5, the maximum and the minimum differences are 6 and 0, respectively. Although the mean is robust to content-preserving processing to some extent, the mean value with nonzero difference may led to wrong index pixel group selection ,and the robustness of hashing method will be affected.

4.2. **Robustness of the histogram shape.** Assume that the original image $I$ has $N = RC$ pixels, where $R$ and $C$ denote the number of rows and columns of the image, respectively. For one bit hash value, it can be generated by computing the ratio of numbers of pixels in two pixel groups (Group1 with $h_G(i)$ pixels and Group2 with $h_G(j)$ pixels). For scaling attacks with factors $\alpha$ and$\beta$, the ratio $r$ can be compute as:

$$r = \frac{\alpha\beta h_G(i)}{\alpha\beta h_G(j)} = \frac{h_G(i)}{h_G(j)} \tag{14}$$

Obviously, the histogram shape is robust to scaling attacks. For rotation and affine attacks, it is obvious that rotation and affine attacks only make the pixels change their positions. So the shape of histogram is robust to rotation and affine attacks.

The resistance of the histogram shape to cropping is described as follows. Suppose that the size of an image is $R \times C$, and the number of pixels in the cropped regions is $N$. Referring to (5), we denote the histogram of the original and cropped images as $H_M$ and $H_M^{'}$, respectively. In the event the data distribution of the cropped regions is different from the original image, the shape of $H_M^{'}$ will be different from that of $H_M$. Strictly speaking, the robustness of histogram shape under cropping attacks depends on the image, the cropped area and the function for selecting the gray levels. So the invariance property of the histogram shape of an image to cropping attacks is an approximate invariance. Similarly, in a DIBR process, the virtual images is generated from the center image with a small parts removed and the holes filled, so the histogram shape of virtual images also is an approximate invariance.

5. **Experimental Results.** In this section, we plan to evaluate the robustness of proposed image hashing algorithm from two aspects. The first one is their perceptual robustness against content-preserving manipulations, which is important for content-based image identification and retrieval. It is desired that perceptually identical images under distortions would have similar hashes. The other is the perceptual robustness against DIBR process. This is desired that the virtual images generated from center image at any baseline distance would have similar hashes with the corresponding center image.

5.1. **Test images database and attack manipulations.** We construct a dataset with about 5757 images. In this database, there are 19 pairs of center and depth images from Middlebury Stereo Datasets [17-19] and Microsoft Research 3D Video Datasets [20], having various resolutions from 450×375 to 1390×1110. The depth images are gray scale images of 8-bit level. For each gray center or generated virtual image which is converted from the original color one, we generate 101 distorted versions by manipulating the original image according to 10 classes of content-preserving distortions, which include additive noise, blurring, JPEG compression, and geometric attacks. The motivation to construct such a database is to simulate possible quality distortions of digital images due to the noise in transmission channel, lossy quantization, and geometric manipulations. The details are given in Table 1. For the additive noise and blurring attacks, the distortions are introduced based on an acceptable quality range. The noise addition and blurring operations are implemented with Matlab, the geometric distortion and JPEG compression are implemented with Stirmark benchmark tool [21].

5.2. **Identification accuracy.** Suppose $I = \{I_i, 1 \leq i \leq N\}$ be the set of original images without distorted in the database, and we generate corresponding hash vector $hash(I_i)$ for each original image. We apply the Hamming distance as the performance metric to measure the discriminating capability between two hash vectors $hash(I_1)$ and$hash(I_2)$. Given a query image $I_Q$, we first generate its hash vector and calculate its distance to

Table 1. Content-preserving manipulations and parameters setting

| Manipulation | Parameters Setting | Copies |
|---|---|---|
| Additive Noise | | |
| Gaussian Noise | $variance \in (0.0005\tilde{}0.005)$ | 10 |
| Salt&Paper Noise | $variance \in (0.001\tilde{}0.01)$ | 10 |
| Speckle Noise | $variance \in (0.001\tilde{}0.01)$ | 10 |
| Blurring | | |
| Gaussian Blurring | filter size:3 $\sigma \in (0.5\tilde{}5)$ | 10 |
| Circular Blurring | radius $\in (0.2\tilde{}2)$ | 10 |
| Motion Blurring | len=1,2,3 $\theta = 0°, 45°, 90°$ | 9 |
| Geometric Attacks | | |
| Rotation | $\theta \in (-10°\tilde{} + 10°)$ | 12 |
| Cropping&Rotation | $\theta \in (-10°\tilde{} + 10°)$ | 12 |
| Scaling | $factor \in (0.5\tilde{}2.0)$ | 6 |
| JPEG Compression | $QF \in (15\tilde{}100)$ | 12 |

$hash(I_i)$ of each original image in database. Intuitively, the query image is identified as the $ith$ original image as following:

$$i = \underset{i}{argmin}\{dis(hash(I_Q), hash(I_i))\} \tag{15}$$

According to section 4.1, the selection of index pixels group for attacked image can affect the performance of identification accuracy. In fact, there are certain relationships among the mean value $P$ of original image, mean value $P'$ of corresponding attacked image and width $L_G$ of pixels group $h_G$. Suppose $G(P)$ represents the pixel groups of original image and $G'(P')$ represents the pixel groups for attacked image, the relationships can be represented as:

$$\begin{cases} G(P) = G'(P') \quad if \quad P = P' \quad or \quad g_{(i-1).L_G+1} \leq P \neq P' \leq g_{(i+1).L_G} \\ G(P) = G'(P' - nL_G) \quad if \quad P \leq g_{(i+1).L_G} < P' \\ G(P) = G'(P' + mL_G) \quad if \quad P' < g_{(i-1).L_G+1} \leq P \end{cases} \tag{16}$$

Obviously, we can find the right pixel groups of attacked image for hash generation within $m+n+1$ times, and the relationship between the maximum mean value differences $D$ and $L_B$ will decide the searching times.

Based on the analysis and extensive testing above, we propose a method to find the right index pixel groups with three times searching. According to Fig.8, we can see if we set $L_G = D$, the searching times $m + n + 1 = 3$. Suppose $hash(I_Q) = \{hash_1(I_Q), hash_2(I_Q), hash_3(I_Q)\}$ are the three final hashes by three times searching of a query image $I_Q$, and $hash(I_i)$ is the hash of a original image, we can compute the distance between $I_Q$ and $I_i$ following the blow steps as shown in Fig. 6.

1) Step 1: Compute the mean value $P_1$ of the query image $I_Q$, and select the pixel groups $G(P_1)$ to generate the hash value $hash_1(I_Q)$.

2) Step 2: Compute the estimated mean value $P_2 = P_1 - L_G$ of the query image $I_Q$, and select the pixel groups $G(P_2)$ to generate the hash value $hash_2(I_Q)$.

3) Step 3: Compute the estimated mean value $P_3 = P_1 + L_G$ of the query image $I_Q$, and select the pixel groups $G(P_3)$ to generate the hash value $hash_3(I_Q)$.

4) Step 4: Compute the distance between $I_Q$ and $I_i$ in reference to the following formula:

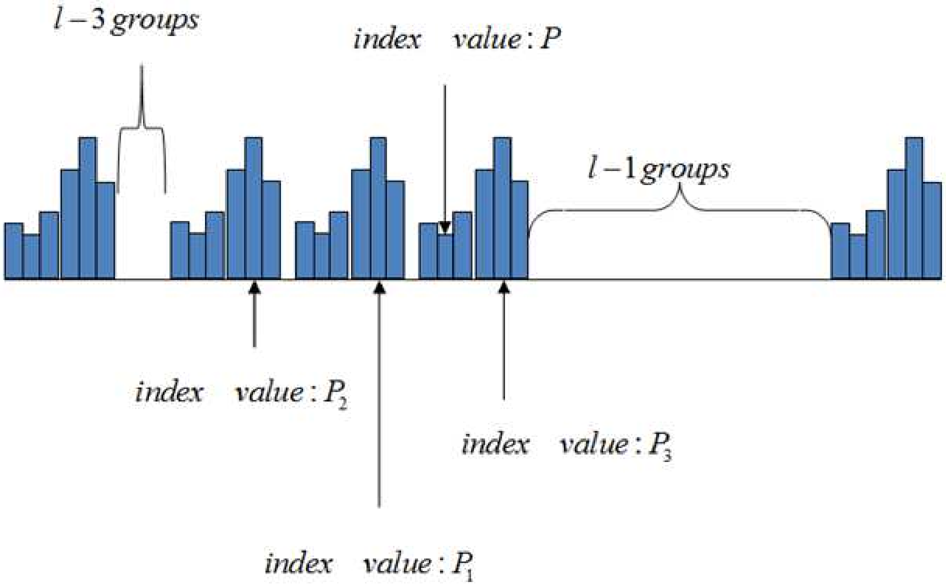


FIGURE 6. Selection of index pixel group by three times searching

$$dis(hash(I_Q), hash(I_i)) = dis(hash(I_i), \mathop{\arg\min}_{hash_j(I_2)} (dis(hash(I_i), hash_j(I_Q)))) \tag{17}$$

TABLE 2. Identification accuracy performances for center and virtual images by proposed method, NMF, RSCH and Ring Partition and NMF hashing algorithms under different attacks Manipulation Proposed

| Manipulation | Proposed | Ring partition [12] | NMF [9] |
|---|---|---|---|
| Additive Noise | | | |
| Gaussian Noise | 100% | 87.78% | 100% |
| Salt&Paper Noise | 100% | 88.15% | 100% |
| Speckle Noise | 100% | 84.81% | 100% |
| Blurring | | | |
| Gaussian Blurring | 100% | 88.89% | 100% |
| Motion Blurring | 100% | 86.83% | 100% |
| Circular Blurring | 100% | 88.89% | 100% |
| Geometric Attacks | | | |
| Rotation | 96.60% | 82.10% | 55.56% |
| Cropping&Rotation | 100% | 82.41% | 63.19% |
| Scaling | 100% | 87.04% | 92.36% |
| JPEG Compression | 100% | 85.19% | 100% |

For the comfortable viewing experience, the maximum baseline distance $t_x$ is recommended to be 5% of the width of the image at a viewing distance of four times of the image height [4]. In order to show the performance about identification of proposed hashing method, we compare the proposed method with some traditional hash schemes designed for 2D image, such as the current state-of-the-art image hashing algorithm NMF-NMF-SQ [9], SVD hashing [8], RSCH hashing [10] and Ring partition hashing [12]. First, we illustrate the identification accuracy of different hashing methods in Table 2 and Table

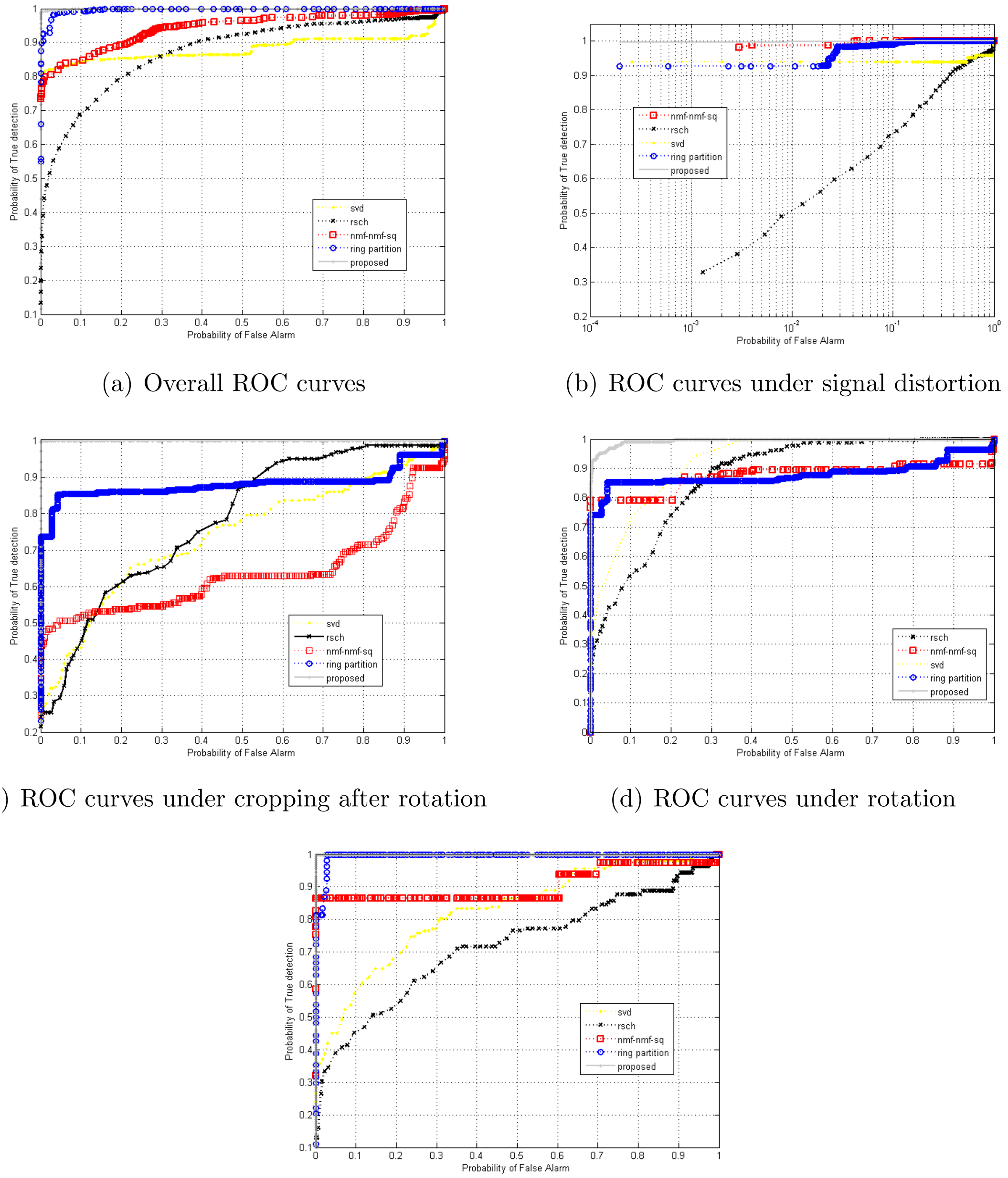
Table 3. Identification accuracy performances for center and virtual images by proposed method, NMF, RSCH and Ring Partition and NMF hashing algorithms under different attacks Manipulation Proposed

| Manipulation | Proposed | RSCH [10] | SVD [8] |
|---|---|---|---|
| Additive Noise | | | |
| Gaussian Noise | 100% | 70.37% | 100% |
| Salt&Paper Noise | 100% | 77.89% | 100% |
| Speckle Noise | 100% | 80.74% | 100% |
| Blurring | | | |
| Gaussian Blurring | 100% | 70.74% | 100% |
| Motion Blurring | 100% | 79.21% | 100% |
| Circular Blurring | 100% | 77.41% | 100% |
| Geometric Attacks | | | |
| Rotation | 96.60% | 62.35% | 60.19% |
| Cropping&Rotation | 100% | 71.60% | 59.88% |
| Scaling | 100% | 64.20% | 66.05% |
| JPEG Compression | 100% | 86.73% | 90.77% |

3. It is desired that the images with content-preserving distortions (including the copies of virtual images) can still be correctly classified to the corresponding original center image, no matter what kinds of manipulations are taken.

It is clearly that the proposed hashing, NMF hashing and SVD hashing are superior to the RSCH and Ring partition hashing algorithm under additive noise, blurring, scaling, and compression attacks, although these two kinds of hashing algorithms can achieve comparable identification performances without taking into account DIBR system. For RSCH hashing, there are two main aspects which affect the identification performance. The first one is that the local features extracted within the neighborhood of key-points are not so stable though they introduced the Harris criterion to select the most stable key-points. In addition, they considered that all perceptually insignificant distortions and malicious manipulations on a digital image would not lead to viewpoint changes, and the center of an image is generally preserved (except for attacks like cropping) and thus relatively stable under geometric attacks such as rotation, shearing in [10] and [12]. In fact, in the DIBR process, virtual images are generated from center image by pixels shifting. The center of virtual images and center image are different, so the matching SIFT feature points are divided into different hash bins and generate different hashes. Hence, these two kinds of method loses the advantages of generating robust hashes based on rotation-invariant, when applied for DIBR 3D image. Meanwhile, the proposed hashing method is much more robust to geometric attacks, such as rotation and cropping after rotation, when compared with NMF and SVD hashing method.

5.3. **Receiver operating characteristics analysis.** We also study the ROC curve [10, 22] to illustrate the identification performances of the proposed image hashing algorithms and compare them with the NMF-NMF-SQ, SVD, RSCH hashing and Ring partition hashing. The ROC curve depicts the relative trade-offs between benefits and cost of the identification and is an effective way to compare the performances of different hashing approaches. To obtain ROC curves, we define the probability of true identification $P_T(\xi)$ and the probability of false alarm $P_F(\xi)$ as formula (18), where $\xi$ is the identification threshold. Images $I$ and $I'$ are two distinct original images and the images $I_D$ and $I'_D$ are



(a) Overall ROC curves

(b) ROC curves under signal distortion

(c) ROC curves under cropping after rotation

(d) ROC curves under rotation

(e) ROC curves under scaling

FIGURE 7. ROC curves of the proposed image hashing approach compared with NMF, SVD, Ring partition ,and RSCH hashing approaches

distorted and content-preserving versions of $I$ and $I'$. Ideally, we hope that the hashes of the original image $I$ and its manipulated version $I_D$ should be similar and thus be identified accurately, while the distinct images $I$ and $I'_D$ should have different hashes. In other words, given a certain threshold $\xi$, a better hashing should provide a higher $P_T(\xi)$ with a lower $P_F(\xi)$. Based on all the distances between manipulated images and original images, we could generate an ROC curve by sweeping the threshold $\xi$ from the minimum value to the maximum value.

$$\begin{aligned} P_T(\xi) &= \Pr(dis(hash(I), hash(I_D)) < \xi) \\ P_F(\xi) &= \Pr(dis(hash(I), hash(I'_D)) < \xi) \end{aligned} \tag{18}$$

Then we present a statistical comparison of different image hashing approaches by studying the corresponding ROC curves. We divide the content-preserving manipulations listed in Table 1 into two categories: Signal distortion (additive noise, blurring, and compression) and geometric distortion (rotation, cropping after rotation and scaling). First, we show the overall ROC curves under all of these listed manipulations, and ROC curves under signal distortion attacks and geometric distortion attacks will be presented, respectively. With the same probability of false alarm $P_F(\xi)$, a better hashing approach could achieve a higher probability of true identification $P_T(\xi)$. In other words, the ROC curve is to measure the similarity of hashes between true query images and original images under a selected false classification rate. ROC curves provide a trade-off between the true retrieval and misclassification to select, when a uniform threshold is applied for identification. As shown in Fig. 7, it is clearly that the proposed hashing approach achieves the best overall robustness under all manipulations listed in Table 2 and Table 3, which is consistent with the reported identification accuracy.

Table 4. Identification accuracy performances by proposed method with different baseline distance in DIBR process

| Manipulation | baseline (5%) | baseline (6%) | baseline (7%) |
|---|---|---|---|
| Additive Noise | | | |
| Gaussian Noise | 100% | 100% | 100% |
| Salt&Paper Noise | 100% | 100% | 100% |
| Speckle Noise | 100% | 100% | 100% |
| Blurring | | | |
| Gaussian Blurring | 100% | 100% | 100% |
| Motion Blurring | 100% | 100% | 100% |
| Circular Blurring | 100% | 100% | 100% |
| Geometric Attacks | | | |
| Rotation | 100% | 100% | 100% |
| Cropping&Rotation | 96.60% | 96.60% | 96.30% |
| Scaling | 100% | 100% | 100% |
| JPEG Compression | 100% | 100% | 100% |

5.4. **Robustness against baseline distance adjustment.** As illustrated in Section 2, the virtual images can be generated with a proper baseline distance in DIBR process, where $t_x$ represents the baseline distance. Usually the parameter $t_x$ is different in order to be suitable for different people's visual. Since $t_x$ is not fixed in DIBR rendering process, the performance of identification for virtual images may be affected. In order to show the robustness of proposed hashing method against baseline distance adjustment, the baseline distance is ranged from 5% to 7% of the image width. As shown in Table 4, the identification accuracy is almost unchanged with different baseline distance. From the ROC curves we can also find that the proposed hashing method is robust to the baseline distance adjustment in DIBR process as shown in Fig. 8.

6. **Conclusions.** In this paper, we propose a novel hashing scheme for DIBR 3D images with consideration of both virtual images and their copies should be classified to the corresponding original center image, when they are attacked by the content-preserving manipulations. With the approximate invariance of the image histogram shape to geometric distortions especially the DIBR process, the robustness of hashing method is achieved.

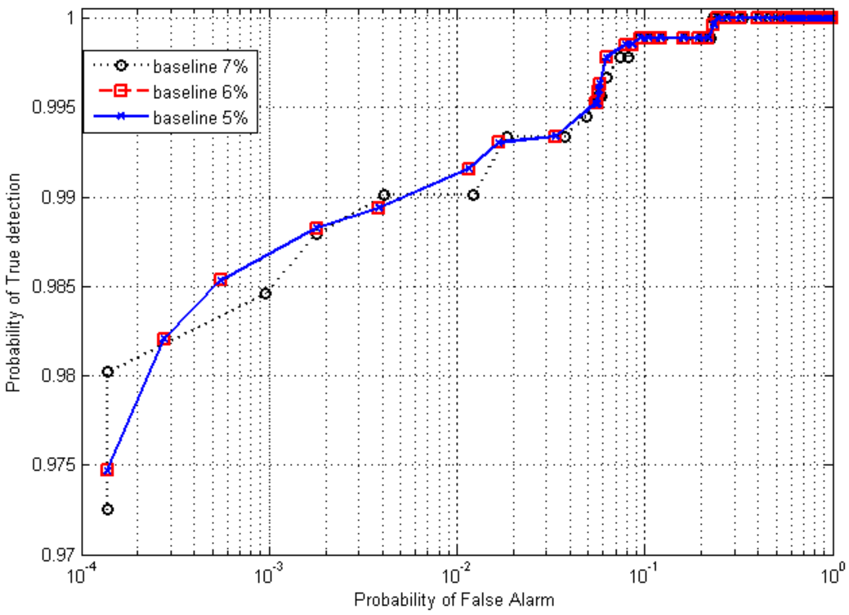


FIGURE 8. ROC curves of the proposed image hashing approach with different baseline in DIBR process

Meanwhile, we use an N-times searching method to improve the robustness of proposed hashing scheme. Experimental results show that the proposed method can achieve better identification performances in terms of geometric attacks such as rotation attacks, and provide comparable performances under classical distortions such as additive noise, blurring, and compression. Furthermore, this method can ensure that the generated virtual images could be identified with the same content as the corresponding center image when we adjust the baseline distance. However, the proposed hashing method is not robust against gamma correction, brightness adjustment, and contrast adjustment. We will combine other features to solve these problems in the future.

**Acknowledgment.** We would like to thank the helpful comments and suggestions from the anonymous reviewers. This work is supported by the National Natural Science Foundation of China (Grant number: 61100187, 61301099 and 61361166006), and Laboratory of Network and Information Security Technology, CAEP (No.J-2014-KF-01).

## REFERENCES

[1] S. C. Chan, H. Y. Shun, and K. T. Ng, *Image-based rendering and synthesis*, IEEE Signal Processing Magazine, vol. 24, pp. 22-33, 2007.

[2] M. Levoy and P. Hanrahan, *Light field rendering*, Proc. SIGGRAPH- '96, pp .31-42, 1996.

[3] T. Fujii and M. Tanimoto, *Free-Viewpoint TV System Based on Ray-Space Representation*, Proc. SPIE Three-Dimensional TV, Video, and Display, vol. 4864, pp. 175-189, 2002.

[4] C. Fehn, E. La Barre, and S. Pastoor, *Interactive 3-DTV: Concepts and key technologies*, Proc. IEEE, vol. 94, no. 3, pp. 524-538, 2006.

[5] F. Lefebvre, J. Czyz, B. Macq, *A robust soft hash algorithm for digital image signature*, Proc. IEEE Int. Conf. Image Processing (ICIP), 2003, vol. 2, pp. 495-498.

[6] C. Lin, S. Chang, *A robust image authentication method distinguishing jpeg compression from malicious manipulation*, IEEE Trans. Circuits Syst. Video Technol, vol. 11, no. 2, pp. 153-168, Feb. 2002.

[7] A. Swaminathan, Y. Mao, M. Wu, *Robust and secure image hashing*, IEEE Trans. Inf. Forensics Security, vol. 1, no. 2, pp. 215-230, Jun. 2006.

[8] S. S. Kozat, R. Venkatesan, M. K. Mihcak, *Robust perceptual image hashing via matrix invariants*, Proc. of IEEE International Conference on Image Processing, pp.3443-3446, 2004.

[9] V. Monga, MK Mhcak, *Robust and secure image hashing via non-negative matrix factorizations*, IEEE Transactions on Information Forensics and Security, vol. 2, no. 3, pp. 376-390, 2007.

[10] X. Lv, Z. J. Wang, *Perceptual image hashing based on shape contexts and local feature points*, IEEE Trans. Inf. Forensics Security, vol. 7, no. 3, pp. 1081-1093, Jun. 2012.

[11] S. Belongie, J. Malik, J. Puzicha, *Shape matching and object recognition using shape contexts*, IEEE Trans. Pattern Anal. Mach. Intell, vol. 24, no. 4, pp. 509-522, Apr. 2002.

[12] Z.J. Tang, X. Q. Zhang, S. C. Chao, *Robust Perceptual Image Hashing Based on Ring Partition and NMF*, IEEE Transactions on knowledge and data engineering, vol. 26, no. 3, Mar. 2014.

[13] S. Roy, Q. Sun, *Robust hash for detecting and localizing image tampering*, Proc. IEEE Int. Conf. Image Processing (ICIP), 2007, vol. 6, pp. 117-120.

[14] L. Zhang, W. Tam, *Stereoscopic image generation based on depth images for 3d TV*, IEEE Trans. Broadcasting, vol. 51, no. 2, pp. 191-199, Jun. 2005.

[15] S. Xiang, H. J. Kim, J. Huang, *Invariant image watermarking based on statistical features in the low-frequency domain*, IEEE Transactions on Circuits and Systems for Video Technology, vol. 18, pp. 777-790, 2008.

[16] S. Xiang, H.-J. Kim, J. Huang, *Histogram-based image hashing robust against geometric deformations*, Proceedings of the 9th workshop on Multimedia and security pp. 121-128, 2007.

[17] D. Scharstein, C. Pal, *Learning conditional random fields for stereo*, IEEE Conf. Comput. Vis. Pattern Recog, pp. 1-8, 2007.

[18] D. Scharstein, R. Szeliski, *High-accuracy stereo depth maps using structured light*, IEEE Comput. Soc. Conf. Comput. Vis. PatternRecog, vol. 1, pp. 195-202, 2003.

[19] H. Hirschmller, D. Scharstein, *Evaluation of cost functions for stereo matching*, IEEE Conf. Comput. Vis. PatternRecog, 2007.

[20] C. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, R. Szeliski, *High-quality video view interpolation using a layered representation*, ACM Trans on Graphics, vol. 23, no. 3, pp. 600-608, 2004.

[21] Stirmark Benchmark 4.0 May 2004 [Online]. Available: http://www. petitcolas.net/ fabien/watermarking/stirmark/

[22] T. Fawcett, *An introduction to roc analysis*, Pattern Recognit. Lett, vol. 27, no. 8, pp. 861-874, 2006.

[23] C. Fehn, *Depth-image-based rendering (dibr), compression, and transmission for a new approach on 3d-tv*, SPIE Stereoscopic Displays Virtual Reality Syst. XI, vol. 5291, no. 1, pp. 93-104, 2004.