IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 16, 2021, accepted February 17, 2021, date of publication February 22, 2021, date of current version March 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3060735

Watermarking and Coefficient Scanning for Light Field Images in 4D-DCT Domain

FELIPE A. B. S. FERREIRA AND JULIANO B. LIMA, (Senior Member, IEEE)
Department of Electronics and Systems, Federal University of Pernambuco, Recife 50740-550, Brazil

Corresponding author: Juliano B. Lima (juliano_bandeira@ieee.org)

This work was supported in part by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) under Grant 309598/2017-6 and Grant 409543/2018-7, and in part by the Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (FACEPE) under Grant IBPG-1275-3.04/16.

ABSTRACT Light field images have emerged as an important advance in the representation of visual data; at the same time, dealing with such images has brought new challenges related to multimedia transmission and security. In this article, we introduce a robust and blind scheme to watermark lenslet light field images. The method, which is based on quantization index modulation, allows to recover the watermark even after the image is submitted to compression in the 4D-DCT (four-dimensional discrete cosine transform) domain or other attacks. To achieve this, an empirical strategy for scanning the coefficients in the referred domain is also proposed; we demonstrate that, in general, such a strategy outperforms state-of-the-art methods whose purpose is to exploit the energy compacting property of the 4D-DCT. The introduced watermarking scheme has been evaluated using 30 light field images and several embedding / extracting parameters. The obtained results indicate that the watermark is satisfactorily imperceptible and confirm its robustness against various attacks.

INDEX TERMS 4D-DCT, light fields, quantization index modulation, watermarking.

I. INTRODUCTION

Light field imaging allows to capture more visual information than conventional 2D imaging methods. This fact, associated with advances in hardware and software for signal processing, has attracted the attention of academia and industry to this kind of images [1]–[11]. Light field images are formed from the plenoptic function, which describes the light in a scene as a function of position, angle, wavelength, and time [2]. This approach provides benefits and supports improvements in operations such as segmentation [12], [13], synthetic aperture imaging [14], classification [15], [16], recognition [17], depth estimation [18], etc.

At the same time that several possibilities associated with the light field technology are investigated, techniques for protecting digital objects generated from imaging procedures continue to be developed [19]–[22]. In this context, watermarking techniques deserve special attention, being important tools to monitor unauthorized distribution and modification of digital content. Digital watermarking applications can be classified into two main categories: (i) protection of intellectual property rights and (ii) content verification. In the first, the watermark has to be robust against modifications, thus allowing its correct recovery and identification. This includes copyright protection applications [23], digital printing [24] and tracking of unauthorized content distribution [25], for example. In the context of content verification, the digital watermarking process indicates whether there has been any change in the content; in some cases, it is also possible to know the type of change applied to the object as well as the part of the object to which it was applied. Examples of typical applications in this context are authentication [26], [27] and integrity checking [28]. The requirements that a watermark is expected to fulfill depend on its application context, but they are usually related to robustness, imperceptibility, embedding capacity and security [29].

The use of light field images as a cover object for digital watermarking involves different possibilities and depends on the object one desires to protect. To be more specific, the object of interest can be the rendered image (usually displayed as a video), the image depth information, or the raw information (that is, the light field image itself) [30], for instance. In particular, when the purpose is to protect a raw light field image, the number of views results in a variety

The associate editor coordinating the review of this manuscript and approving it for publication was Szidónia Lefkovits.

of possibilities for embedding the watermark; all the views can be considered or just a subset of them. Additionally, the redundancy present in the different views, which is intrinsic to the acquisition method, can be exploited to obtain a better performance of the watermarking technique.

Most watermarking techniques for light field images have been developed in the context of protection when displaying rendered images on the so-called free-view television (FVT). In this context, the viewer can freely select the position and the angle of the view through the image-based rendering application in a multi-view video (MVV). These works aim to protect copyright and avoid illegal copying, considering that the viewer can record the transmitted video after selecting a specific view and angle. In the paper authored by Koz *et al.* [31], published in 2006, the scenario of images generated by light field rendering in an MVV is considered. The position and the rotation of the chosen rendered view are known and the four-dimensional representation of a light field image is interpreted as an array of 2D images, where the watermark is embedded; the technique is applied in the spatial domain. One year later, as an extension of the above referred work, the same authors presented a technique in which the chosen rendered view can be located at an arbitrary position and angle [32].

In 2008, two schemes based on the approach used in [32] were presented [33], [34]; however, in such papers, the assessment was made with rendering by bilinear interpolation instead of nearest neighbors. Subsequently, another technique considering a similar context was published: in [35]. The detector analyzes and estimates variations in the watermark pattern to complete the extraction. A watermarking scheme based on spectral spreading for FVT was presented in [36]; the watermark is embedded in all views and its detection is performed by exploring the coefficients of the discrete cosine transform (DCT) calculated from the rendered view. In 2015, Paudyal *et al.* proposed a method in which all views of the light field image are reorganized into a sequence of views in order to form a single image [30]. The watermark is then embedded into the Haar's wavelet transform coefficients of this single image. Besides, the degradation of the depth map was evaluated, after embedding the watermark, to identify its quality when changing the light field's focus. In [37], a semi-blind watermarking technique in the 2D-DCT domain was proposed; such a method considers the context of raw information protection for light field images. Each watermark bit is embedded into a 2D image formed by pixels arbitrarily selected from different views of the light field. In [38], the authors proposed a modified version of the semi-blind method presented in [37]. More specifically, a 4D discrete wavelet transform (4D-DWT) is included in the scheme. Apparently, such a modification leads to an improvement of the results previously obtained in [37], when considering the performance in terms of the watermark robustness; on the other hand, the embedding capacity is reduced.

In general, the above cited papers propose to adapt 2D image watermarking techniques in order to apply them to light field images. However, the most natural representation of a light field appears to be through a four-dimensional structure, with two dimensions for the view plane and two for the spatial plane of the image. In [39], for example, the authors demonstrate that using the 4D-DCT allows to explore more efficiently the redundancy present in the structure of a lenslet light field image than using its 2D version. This fact has been employed to support the development of native 4D CODECs for lenslet light field images using the 4D-DCT, exploiting the high correlation between views and the spatial plane [40], [41]. The results presented in these works suggest the 4D-DCT as a promising alternative for light field compression.

A. MAIN CONTRIBUTIONS AND PAPER STRUCTURE

In this article, we consider the above described context and introduce a blind watermarking technique for raw lenslet light field images in the 4D-DCT domain. The method is based on quantization index modulation [42] and provides high robustness against compression and other attacks. In a more specific way, our contributions can be summarized as follows:

(i) To the best of our knowledge, the proposed watermarking technique is the first blind method that embeds and extracts the watermark considering the 4D representation of light fields.[1] As mentioned, this is in line with recently introduced efficient methods to compress this type of image;

(ii) Novel coefficient scanning strategies in the 4D-DCT domain are introduced. The development of such strategies, which are used during the watermarking embedding and extraction processes, takes into account the behavior, in terms of energy compaction, of 88 light field images in the referred domain.

(iii) We demonstrate that, in terms of the area under the percent cumulative curve, the performance of the proposed scanning strategies is better than that of state-of-the-art techniques for the same purpose [41]; they have also produced results similar to those of the well-established JPEG (2D-DCT domain) zig-zag, in terms of the number of inversions of an ordered sequence. This suggests that, besides being used in the referred watermarking operations, the 4D scanning we have presented can be efficiently used in the scope of light field compression;

(iv) The proposed watermarking scheme has been evaluated for a total of 30 light field images. The obtained results have evidenced the imperceptibility of the watermark as well as its robustness, even after the image is submitted to compression with relatively low percentages of retained 4D-DCT coefficients and other attacks as well;

(v) Taking into account the extracted watermarks' bit error rate, our method generally outperforms those proposed in [37] and [38], whereas maintaining the quality of the

[1]The method presented in [38], which was published when we were finishing writing this article, also exploits the 4D representation of light fields, but it is semi-blind.

watermarked light field. The conclusion obtained from such a comparison is particularly relevant, since the methods given in the cited papers are semi-blind and, at first glance, should be more robust than the proposed (blind) scheme.

The next sections are organized as follows. Section II presents some of the main aspects regarding light field images, as well as their representation in the 4D-DCT domain and the used 4D data structure. Section III presents the proposed watermark embedding and extracting steps, and a variation which uses bit replication. Section IV reports the analysis carried out on the 4D-DCT coefficient scanning order and provides details regarding the algorithms proposed for this task. Section V presents the materials and methods used in the simulations, and Section VI presents the results. Finally, Section VII presents the conclusion and future works.

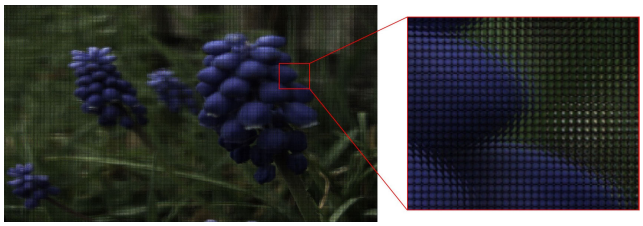
II. LIGHT FIELD IMAGES

Light field images are typically acquired by means of two different techniques [43]. The first method employs a High-Density Camera Array (HDCA) and produces an array of images captured by the camera from different views and angles of the same scene. In the second technique, a single camera with an array of microlenses captures samples of light beams emanating from different directions; this results in the so-called lenslet light field image, which, due to the capture mechanism, is formed by an array of microimages (MI) composed of the same point of the scene seen from different angles. Fig. 1a shows an example of this type of image and a specific region zoomed-in allowing the visualization of the microimages. Instead of a single two-dimensional image formed by MIs, even a lenslet light field can be represented by a two-dimensional array where each element is a two-dimensional image, as can be seen in Fig. 1b. Note that, in this representation, all possible views are visualized at once.

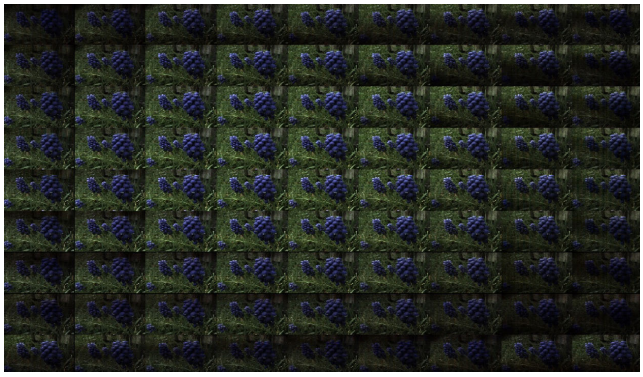
Although it is possible to use a 2D array as a data structure to represent a light field image, it is more convenient to represent it as a four-dimensional array; in this case, the captured light rays are parameterized by the coordinates of their intersection with two planes [2]. The coordinate system is denoted by (s, t) and (u, v) for the first and the second planes, respectively; a ray of light $L(s, t, u, v)$ defined in that system first intersects the st plane, in the (s, t) coordinate, and then intersects the uv plane, in the (u, v) coordinate. Fig. 2 shows an example of the coordinate structure in a light field with $s = t = u = v = 3$.

A. LIGHT FIELD IMAGE IN 4D-DCT DOMAIN

The DCT has a high capacity to concentrate the energy at low frequencies. When standard two-dimensional images are considered, the energy of the 2D-DCT spectrum is concentrated in the coefficients located in the upper left corner. Similarly, in the 4D-DCT domain, the energy is concentrated in the upper left corner for both the st and the uv planes. Naturally, this property can be exploited for compression and



(a) Array of microimages



(b) Array of images

FIGURE 1. Representations of a lenslet light field image.

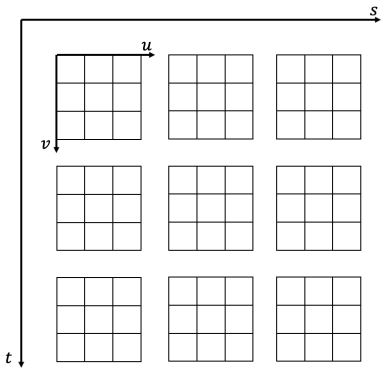


FIGURE 2. Coordinate structure example of a light field image of size $(s, t, u, v) = (3, 3, 3, 3)$.

watermarking applications, since it allows to modify several coefficients and obtaining a low level of degradation in the reconstructed images.

Due to its separability, we can calculate the 4D-DCT by computing the one-dimensional DCT for each dimension or, similarly, we can compute the 2D-DCT for a pair of dimensions and then compute the 2D-DCT for the other pair. So, considering a light field $\mathbf{L}$ of dimensions $S \times T \times U \times V$, one can compute the 4D-DCT by computing the 2D-DCT concerning the uv plane and then computing the 2D-DCT with respect to the st plane. That is, for each view $\mathbf{L}(i, j)_{st}$, $1 \leq i \leq S$ and $1 \leq j \leq T$, we calculate

$$\mathbf{L}'(i, j)_{st} = \Gamma_{\mathbf{u}} \mathbf{L}(i, j)_{st} {\Gamma_{\mathbf{v}}}^T, \tag{1}$$

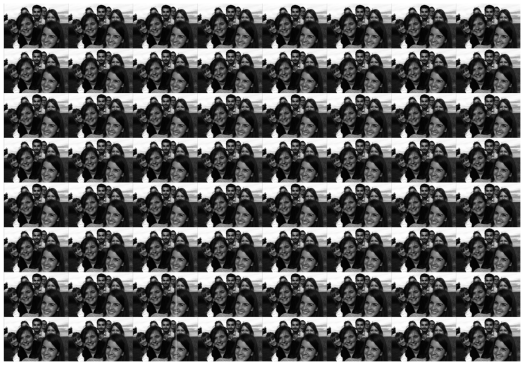


FIGURE 3. Light field image of size $(S, T, U, V) = (8, 8, 432, 624)$.

and then, for each view $\mathbf{L}'(k,l)_{uv}$, $1 \leq k \leq U$ and $1 \leq l \leq V$, we calculate

$$\tilde{\mathbf{L}}(k,l)_{uv} = \Gamma_{\mathbf{s}} \mathbf{L}'(k,l)_{uv} {\Gamma_{\mathbf{t}}}^T, \tag{2}$$

where $\Gamma_{\mathbf{n}}$ is a DCT matrix of size $n \times n$.

The inverse transform, 4D-IDCT, can be computed, performing the left and right products of all views from $\tilde{\mathbf{L}}$ by ${\Gamma_{\mathbf{u}}}^T$ and $\Gamma_{\mathbf{v}}$, respectively,

$$\begin{aligned} {\Gamma_{\mathbf{u}}}^T \tilde{\mathbf{L}}(i,j)_{st} \Gamma_{\mathbf{v}} &= {\Gamma_{\mathbf{u}}}^T \Gamma_{\mathbf{u}} \mathbf{L}'(i,j)_{st} {\Gamma_{\mathbf{v}}}^T \Gamma_{\mathbf{v}} \\ &= \mathbf{L}'(i,j)_{st}, \end{aligned} \tag{3}$$

and then compute the left and right products of $\mathbf{L}'$ by ${\Gamma_{\mathbf{s}}}^T$ and $\Gamma_{\mathbf{t}}$, respectively,

$$\begin{aligned} {\Gamma_{\mathbf{s}}}^T \mathbf{L}'(k,l)_{uv} \Gamma_{\mathbf{t}} &= {\Gamma_{\mathbf{s}}}^T \Gamma_{\mathbf{s}} \mathbf{L}(i,j)_{uv} {\Gamma_{\mathbf{t}}}^T \Gamma_{\mathbf{t}} \\ &= \mathbf{L}(k,l)_{uv}, \end{aligned} \tag{4}$$

obtaining the original light field image, $\mathbf{L}$.

Fig. 3 presents a light field of size $8 \times 8 \times 432 \times 624$, converted from the RGB color format to YCbCr and represented as a 2D array of images; only channel Y is shown in the figure and used in the following examples.

Fig. 4 shows the 4D-DCT coefficients for the light field shown in Fig. 3, computed using blocks of size $8 \times 8 \times 8 \times 8$. As expected, the 4D-DCT spectrum (viewed in two dimensions) has an aspect similar to that of the 2D-DCT, in the sense that the coefficients of greater magnitude are concentrated in the upper left corner of the two-dimensional spectrum. Fig. 4 also shows a zoom-in of one of the views; more specifically the view in the position $s = 1$ and $t = 1$. Note that the concentration of the coefficients in the upper left corner can be seen both in the st plane and in the 8×8 blocks of the uv plane.

III. PROPOSED WATERMARKING SCHEME

The embedding and extraction procedures related to the watermarking scheme proposed in this work are described in the following subsections. In both procedures, pseudo-random mechanisms are used and must be synchronized in order to provide the correct watermark recovery. The seed used to generate pseudo-random sequences is used as the key

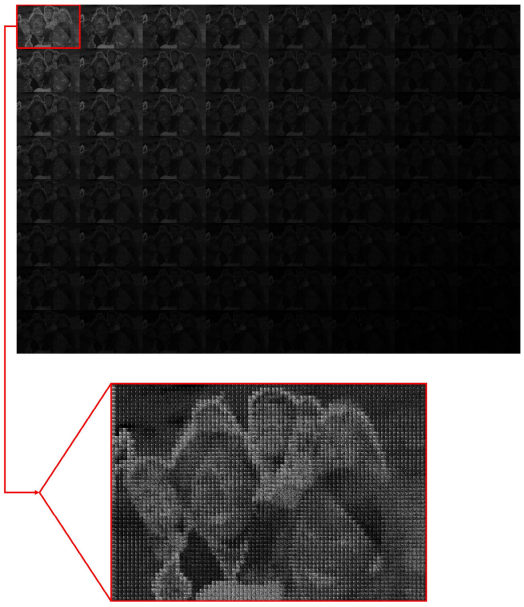


FIGURE 4. 4D-DCT spectrum (viewed in two dimensions), of a light field image, calculated with blocks of size 8 × 8 × 8 × 8 and the view in the position $(s, t) = (1, 1)$ expanded.

to the watermarking process. Fig. 5 shows the flow diagram of the proposed watermarking scheme and includes the embedding and the extraction stages. Each step is described with more details in what follows.

A. EMBEDDING

Let $\mathbf{L}$ be a light field (luminance only) of size $S \times T \times U \times V$ and let $\mathbf{W}$ be a binary watermark of size $N \times N$. In order to embed the watermark $\mathbf{W}$ into the light field $\mathbf{L}$, the following steps are performed:

- **Step 1:** Scramble the bits of $\mathbf{W}$ using the Arnold transform [44] and obtain $\mathbf{W}'$. This decorrelates the watermark pixels in the spatial domain. The Arnold transform consists of mapping a point (x, y), where $x, y \in \{1, 2, 3, \ldots, N\}$, to another point (x', y'), where $x', y' \in \{1, 2, 3, \ldots, N\}$, according to

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \pmod{N}, \tag{5}$$

  where (x, y) and (x', y') are pixel coordinates of the original watermark $\mathbf{W}$ and its scrambled version $\mathbf{W}'$, respectively;
- **Step 2:** For each block $\mathbf{B}$ of size $s \times t \times u \times v$ from the light field $\mathbf{L}$, compute the 4D-DCT and obtain blocks $\tilde{\mathbf{B}}$;
- **Step 3:** Scan the coefficients of each 4D block $\tilde{\mathbf{B}}$ and rearrange it in a 1D array (more details in Section IV);
- **Step 4:** For each bit b of $\mathbf{W}'$, perform the modulation of a coefficient C in $\tilde{\mathbf{B}}$, obtaining $\tilde{C}$ according to

$$\tilde{C} = \begin{cases} 2\Delta \times \text{round}\left(\frac{C}{2\Delta}\right) + \frac{\Delta}{2}, & \text{if } b = 1 \\ 2\Delta \times \text{round}\left(\frac{C}{2\Delta}\right) - \frac{\Delta}{2}, & \text{if } b = 0, \end{cases} \tag{6}$$

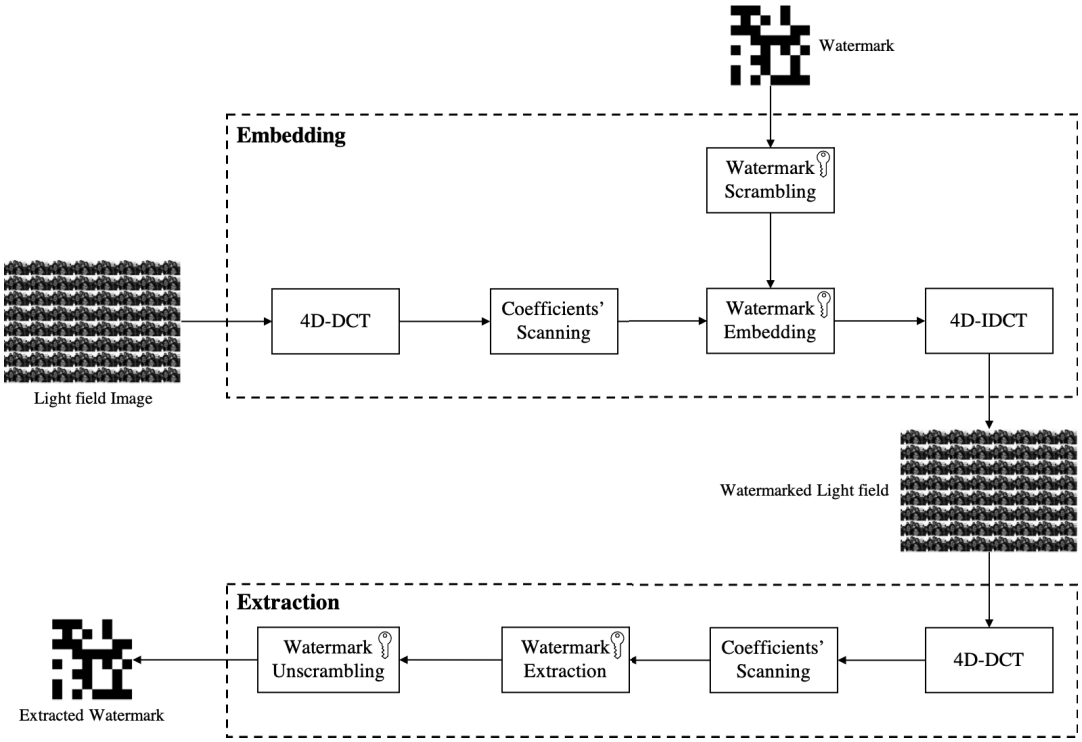


FIGURE 5. Proposed watermarking scheme flow diagram.

where Δ is a user-defined parameter to control the watermark "strength" and round$(\cdot)$ is a rounding function (to the closest integer). The bits of $\mathbf{W}'$ are embedded in a specific subset of coefficients in the transform domain regarding the scanning order mentioned in Step 3. If a bit of $\mathbf{W}'$ is embedded in a coefficient of a given transformed block, the next bit is embedded in the next block; the blocks are cyclically taken until all bits of $\mathbf{W}'$ are embedded. For the sake of safety enhancement, the order the blocks are taken are secretly scrambled;

- **Step 5:** Compute the 4D-IDCT of each block, obtaining the watermarked light field $\tilde{\mathbf{L}}$.

B. EXTRACTION

As mentioned before, for the correct watermark extraction, it is necessary to use the same key employed to generate the pseudo-random numbers used in the embedding step. In what follows, we describe the steps for extracting the watermark:

- **Step 1:** Compute the block-based 4D-DCT of $\tilde{\mathbf{L}}$;
- **Step 2:** Scan the coefficients of each 4D block and rearrange them into 1D arrays (see Step 3 of the embedding procedure);
- **Step 3:** For each modulated coefficient $\tilde{C}$, extract the bits of the scrambled watermark $\mathbf{W}'$, according to

$$b = \underset{\theta\in\{0,1\}}{\arg\min} \left(|C_\theta - C|\right), \tag{7}$$

where

$$C_\theta = \begin{cases} 2\Delta \times \text{round}\left(\frac{\tilde{C}}{2\Delta}\right) + \frac{\Delta}{2}, & \text{if } \theta = 1 \\ 2\Delta \times \text{round}\left(\frac{\tilde{C}}{2\Delta}\right) - \frac{\Delta}{2}, & \text{if } \theta = 0; \end{cases} \tag{8}$$

- **Step 4:** Apply the inverse Arnold transform to $\mathbf{W}'$, recovering the original watermark $\mathbf{W}$:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \pmod N. \tag{9}$$

C. WATERMARKING USING BIT REPLICATION

As a possible variation for the proposed watermarking technique, we consider the replication of the watermark bits before embedding them. Each bit is replicated c times, so that the replicated version of a sequence of bits $\mathbf{S} = (s_1, s_2, s_3, \ldots, s_n)$, corresponds to

$$\mathbf{S}' = \Big(\overbrace{s_1, s_1, \ldots, s_1}^{c \text{ times}}, \overbrace{s_2, s_2, \ldots, s_2}^{c \text{ times}}, \ldots, \overbrace{s_n, s_n, \ldots, s_n}^{c \text{ times}}\Big). \tag{10}$$

In the extraction step, the original sequence is recovered by

$$s_i = \begin{cases} 1, & \text{if } \sigma \geq 0{,}5 \\ 0, & \text{otherwise,} \end{cases} \forall\, i \leq n, \tag{11}$$

where

$$\sigma = \frac{\sum_{j=1}^{c} s_{(i-1)c+j}}{c}. \tag{12}$$

It is important to remark that this variation potentially increases the robustness against attacks due to the bit replication but, on the other hand, it decreases the imperceptibility because it affects more intensely the image being watermarked. As will be demonstrated, such a tradeoff can be controlled by carefully adjusting the strength of the watermark during the embedding procedure.

IV. 4D-DCT COEFFICIENTS SCANNING ORDER

At the time this work was being developed, coefficient scanning algorithms in the 4D-DCT domain have not been found in the literature. This led us to develop the methods described in this section. Initially, we created an empirical strategy based on the coefficients' average magnitude in the transform domain of a set of light field images. More specifically, the 4D-DCT was applied in a block basis ($8 \times 8 \times 8 \times 8$) to 88 images from the EPFL (École Polytechnique Fédérale de Lausanne) light field image database [45]. The average of the absolute value of each coefficient was then computed according to its position in the blocks. This process resulted in a single block of size $8 \times 8 \times 8 \times 8$ containing the average of each coefficient in its respective position. The referred single block was then vectorized, resulting in a sequence of 4096 values. The indexes in such a sequence were then ordered according to the absolute value of their respective elements in decreasing order. More specifically, we consider n blocks, $i = 1, \ldots, n$, of coefficients[2]. In this case, since each image has dimension $8 \times 8 \times 432 \times 624$ and 88 images were used, we have $n = 370656$. From all blocks $\mathbf{B}_i$, the block

$$\widehat{\mathbf{B}} = \begin{bmatrix} \frac{\sum_{i=1}^{n} c_{1,1}^{i}}{n} & \frac{\sum_{i=1}^{n} c_{1,2}^{i}}{n} & \cdots & \frac{\sum_{i=1}^{n} c_{1,64}^{i}}{n} \\ \frac{\sum_{i=1}^{n} c_{2,1}^{i}}{n} & \frac{\sum_{i=1}^{n} c_{2,2}^{i}}{n} & \cdots & \frac{\sum_{i=1}^{n} c_{2,64}^{i}}{n} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\sum_{i=1}^{n} c_{64,1}^{i}}{n} & \frac{\sum_{i=1}^{n} c_{64,2}^{i}}{n} & \cdots & \frac{\sum_{i=1}^{n} c_{64,64}^{i}}{n} \end{bmatrix}, \tag{13}$$

containing the average of each coefficient is obtained. The block $\widehat{\mathbf{B}}$ is then vectorized following the lexicographic order of the indexes of its elements, that is,

$$\begin{aligned} \mathbf{V} = (&\widehat{B}_{1,1}, \widehat{B}_{1,2}, \cdots, \widehat{B}_{1,64}, \\ &\widehat{B}_{2,1}, \widehat{B}_{2,2}, \cdots, \widehat{B}_{2,64}, \\ &\cdots, \\ &\widehat{B}_{n,1}, \widehat{B}_{n,2}, \cdots, \widehat{B}_{n,64}). \end{aligned} \tag{14}$$

The above coefficients are placed in the descending order of their absolute value and the final order of their indexes is used to scan the coefficients of the lexicographic sequence of a 4D-DCT block. We have provided a file containing de referred final scanning order at `https://github.com/felipebsferreira/4D-DCT-Scanning` [46]. From this point forward, the empirical scanning approach, using the indexes of the decreasing order of the coefficients' average energy in the transform domain, will be referred to as Average Scanning (AS).

[2]The blocks $\mathbf{B}_i$ are four-dimensional structures and are represented here in two dimensions as an 8×8 array of blocks 8×8.

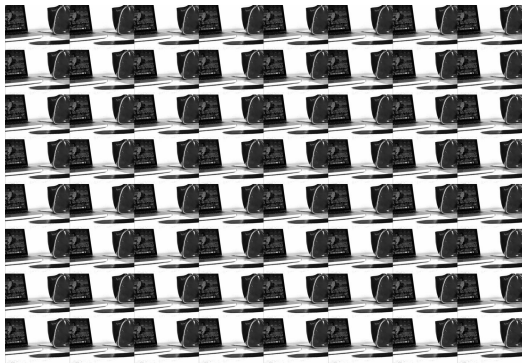


FIGURE 6. A light field image from the test set called "Desktop."

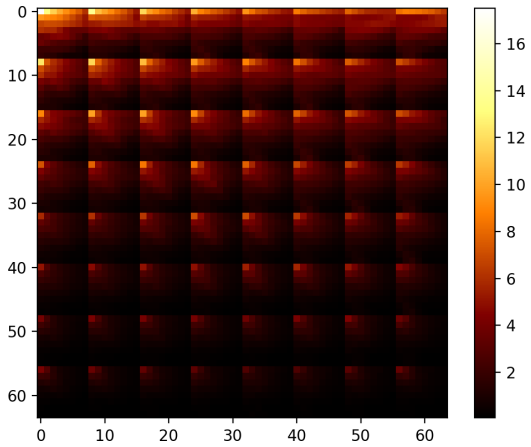
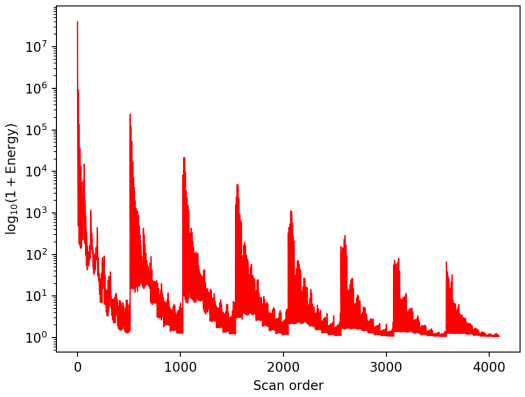


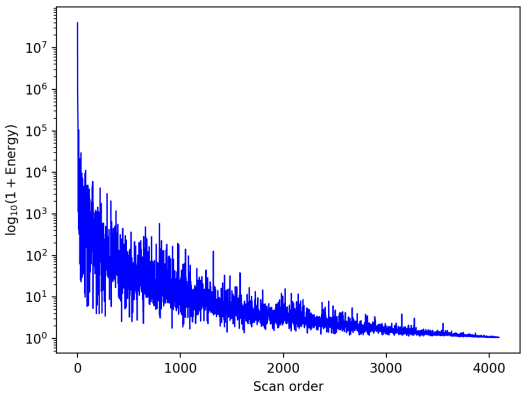
FIGURE 7. Heat map of desktop's energy distribution.

At the EPFL's database, there are 118 light field images divided into ten categories such as buildings, grids, light, nature, among others. In addition to the 88 images used to generate the scanning order, the remaining 30 were used as "test" images to evaluate the proposed scanning technique. The choice of test images followed two criteria. The first criterion is to ensure the maximum number of images in each category. The second criterion is to include the images that are part of the JPEG Pleno Call for Proposals on Light Field Coding [47] test set, since one of the objectives of this work is to propose a watermarking method robust to compression in the 4D-DCT domain.

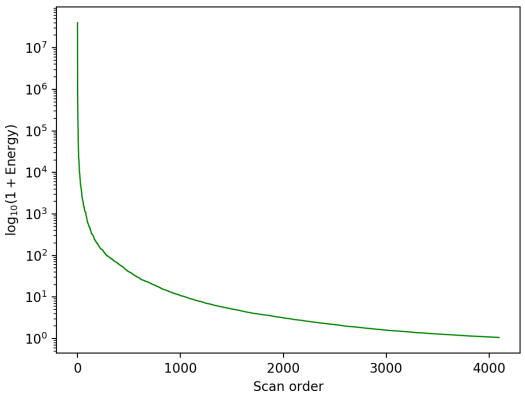
In Fig. 6, we present the image "Desktop", contained in the test set; in Fig. 7, we present the energy distribution of its 4D-DCT coefficients. The estimated energy of each 4D-DCT coefficient corresponds to the variance between all four-dimensional blocks of a light field for each coefficient position [48].



(a) Lexicographical order



(b) Average Scanning



(c) Real sorting

FIGURE 8. Estimated energy of the light field image "desktop" presented on a logarithmic scale for better visualization.

In Fig. 8, the energy distribution is sequentially plotted according to the lexicographical order of the four coordinates that identify them (Fig. 8a) and according to the above described AS scan (Fig. 8b). Note that the energy distribution in Fig. 8b is somewhat concentrated on what would be the "low frequencies" whereas in Fig. 8a the energy looks more disperse in the sequence. For reference, in Fig. 8c, the energy distribution is plotted according to an order defined by a traditional real sorting algorithm (in this case, the quicksort algorithm [49] is used).

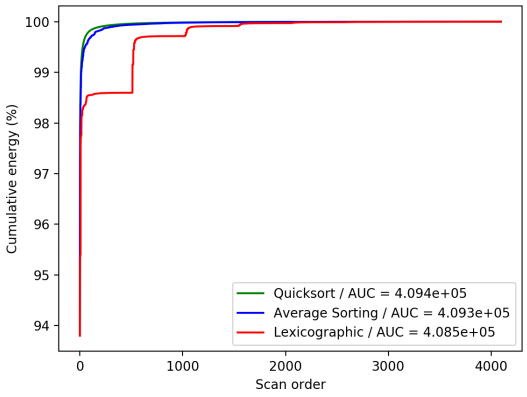


FIGURE 9. Percent cumulative energy of sequences scanned by different methods.

Fig. 9 shows the percent cumulative energy (PCE) of the 4D-DCT coefficients of "Desktop" image using the lexicographical order, AS and real sorting, according to the scanning order. An energy-compact PCE sequence tends to have a greater slope and consequently a larger area under it; we can observe this characteristic in the curve obtained when the coefficients are scanned using AS, which contrasts with the curve obtained when the lexicographical order is used. Comparing the sequence scanned by AS and the sequence ordered by the real sorting algorithm, we observe that the former is relatively close to a real ordering.

A. ANALYZING THE AVERAGE SCANNING APPROACH

With the purpose of obtaining some insights about the scanning order provided by AS and investigating the possible correlation between this approach and the energy decay along the (s, t, u, v) axes of a light field image, the analysis described in this subsection is performed. Such an analysis gives us support to propose a systematic scanning method, which is explained in Section IV-B.

To understand how the coefficients are scanned, a sequence of images containing a four-dimensional block of size $8 \times 8 \times 8 \times 8$, presented as a 2D block of size 64×64, was created. In the first image, this block is filled with black pixels (gray level 0) and, in the next image, the position of the corresponding next coefficient, according to the Average Scanning, is "painted" in white (gray level 255). In this manner, it is possible to visualize the formed pattern as the positions are scanned. Fig. 10 shows some of these images, where the pattern formed as more coefficients are scanned can be viewed. Considering a block of size $(s, t, u, v) = (8, 8, 8, 8)$, the AS scanning leads to the gradual appearance of a pattern in the inter-frame columns, for each position in the s coordinate, which grows from left to right and from top to bottom, as more positions are scanned, until the image is completely filled in white. Note that this pattern is similar to the energy distribution plotted in Fig. 7. In total, we have 8 columns of size (8, 64) in s coordinate, that is, considering

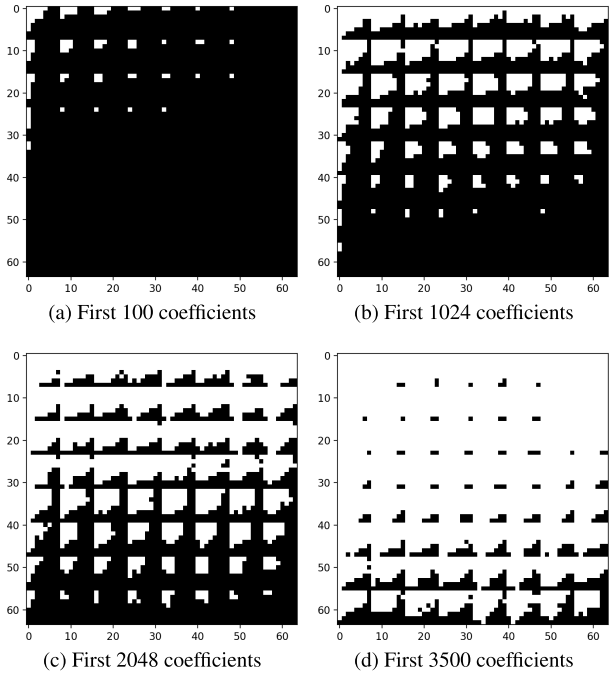


FIGURE 10. Images of blocks 8 × 8 × 8 × 8 where the position of the first *n* coefficients (according to the average scanning) are highlighted in white.

a four-dimensional image, the s coordinate is fixed whereas the others are scanned.

We can see how the energy decays from a column to the next column in Fig. 11, where the total energy of each column is plotted. The total energy of each (64, 8) line, when the t coordinate is fixed and the others are scanned, is also plotted. Note that the energy decays from left to right and from top to bottom. Additionally, the energy decays more quickly along the lines than along the columns, where it has a higher concentration. This pattern is interesting and can be used to develop an algorithm to scan 4D-DCT coefficients for a wide range of lenslet light field images, analogously to the well-known JPEG zig-zag algorithm used to scan 2D-DCT coefficients of conventional 2D images. We emphasize that a similar behavior also occurs for all evaluated light field images.

B. COLUMN-BASED SCANNING ALGORITHM

In this section, we consider the analysis carried out in Section IV-A and, exploiting the pattern formed in the columns of the s coordinate of the 4D-DCT coefficients (see Fig. 12), we propose a column-based scanning algorithm. The approach consists of individually scanning each column from left to right. Thus, instead of scanning a 4D block, we are scanning multiple 3D blocks, as shown in Fig. 13.

In Algorithm 1, we give the procedure based on the above mentioned approach, considering a 4D block of dimensions $(s, t, u, v) = (S, T, U, V)$. Essentially, the referred approach is based on a well-defined pattern for scanning the coefficients on a column. The same pattern is replicated for the

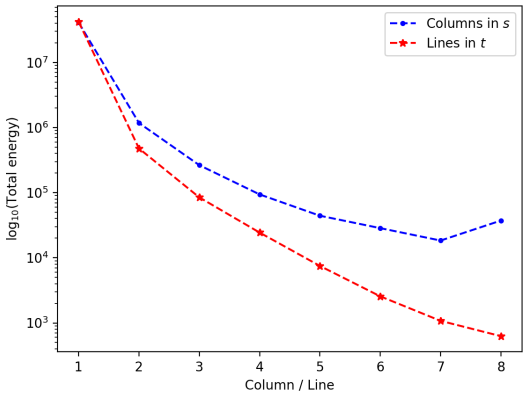


FIGURE 11. Total energy per column in the *s* coordinate and per line in the *t* coordinate. The columns are numbered from 1 to 8 from left to right and the lines from top to bottom.

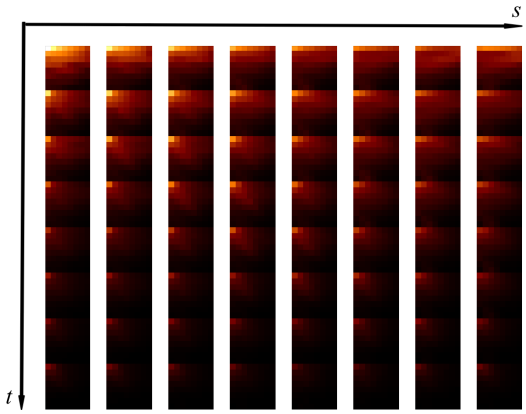


FIGURE 12. Energy distribution of the 4D-DCT coefficients of each column in *s* coordinate of "desktop" light field.

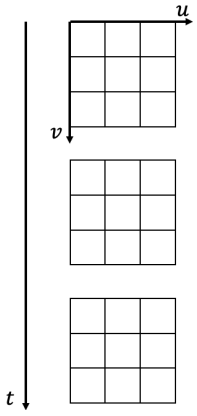


FIGURE 13. A column represented as a 3D data structure of size $(t, u, v) = (3, 3, 3)$.

other columns, which are alternately and cyclically scanned, from left to right. On line 5, SCAN( ) is a procedure that receives a three-dimensional block (that is, a column in s coordinate) and returns its scanned elements. Details of the referred procedure are given in Algorithm 2 for a single column; in such an algorithm, which has been identified by

Algorithm 1 4D array scan according to the column pattern

1: **procedure** column_scan($block, S, T, U, V$)
2: $scan \leftarrow [\,]$
3: $column\_scan \leftarrow [\,]$
4: **for** $sin[1, S]$ **do**
5: $column\_scan[s] \leftarrow \text{SCAN}(block[:, s, :, :])$ (S times)
6: $column\_index \leftarrow [\overbrace{1, 1, 1, \ldots, 1}^{S \text{ times}}]$
7: $s \leftarrow 1$
8: **for** $iin[1, S \times T \times U \times V]$ **do**
9: $scan[i] \leftarrow column\_scan[s, column\_index[s]]$
10: $column\_index[s] \leftarrow column\_index[s] + 1$
11: $s \leftarrow s + 1$
12: **if** $s > S$ **then**
13: $s \leftarrow 1$
14: **return** $scan$

Algorithm 2 3D Diagonal Column Scan

1: **procedure** 3d_diagonal_scan($block, T, U, V$)
2: $scan\_order \leftarrow [\,]$
3: $sum \leftarrow T + U + V$
4: $i \leftarrow 1$
5: **for** $current\_sumin[1, sum]$ **do**
6: **for** $tin[1, T]$ **do**
7: **for** $vin[1, V]$ **do**
8: **for** $uin[1, U]$ **do**
9: **if** $t + v + u = current \sum$ **then**
10: $scan\_order[i] \leftarrow block[t, v, u]$
11: $i \leftarrow i + 1$
12: **return** $scan\_order$

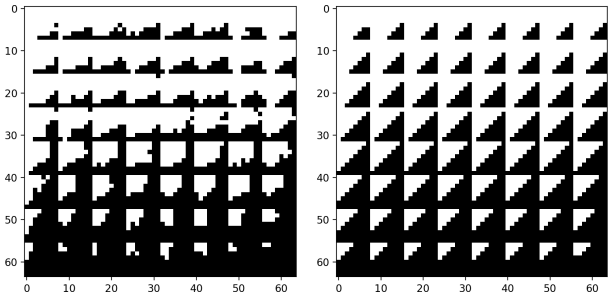
3D Diagonal Column Scan (C-3DD), the coefficients of a column are scanned by the diagonals in the order t, v, u of the coordinates.[3]

In Fig. 14, we present $8 \times 8 \times 8 \times 8$ block images where the position of the first 2048 coefficients are highlighted in white, considering the AS and C-3DD scanning. The similarity of such two images is remarkable.

V. MATERIALS AND METHODS

In this section, we present the materials and methods used to carry out the simulations of the watermarking technique and the proposed scanning algorithms. Originally, the light field images from the EPFL database [45] are $15 \times 15 \times 434 \times 625$. In this work, we used versions of these images with dimensions $8 \times 8 \times 432 \times 624$ (views and pixels in the central positions have been preserved) so that it is possible to use blocks of size $8 \times 8 \times 8 \times 8$. This has been performed because the DCT is more efficiently computed, in terms of the number of arithmetic operations, if dimensions of the

[3]Throughout the development of this work, we have created other algorithms that could be used to replace the procedure SCAN( ) on line 5 of Algorithm 1. However, we have decided to document here the C-3DD scanning only, since it has been the technique providing the best result in terms of PCE.



(a) AS (b) C-3DD

FIGURE 14. Images of $8 \times 8 \times 8 \times 8$ blocks where the position of the first 2048 coefficients, according to AS and C-3DD scanning, are highlighted in white.

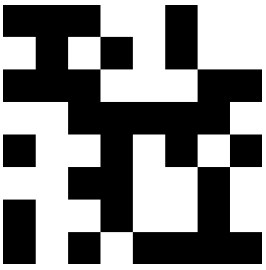
(a) Buildings (b) Grids (c) ISO and Color Charts
(d) Landscapes (e) Light (f) Mirrors and Transparency
(g) Nature (h) People (i) Studio
(j) Urban

FIGURE 15. Central view of 10 "test" light field images used in this article and their respective categories.

form 2^n are used. As previously mentioned, the database has 118 images, 88 of which were used to generate the Average Scanning sequence and 30 were used to test the proposed algorithms. Fig. 15 shows the central view of 10 test images, one from each category.

The tested watermarks are shown in Fig. 16. Fig. 16a was obtained from the Signal and Image Processing Institute database of the University of Southern California [50]. The grayscale image *Male* with size 1024×1024 was selected; it was also scaled to 512×512 and 2048×2048. These images were then binarized, resulting in images as those

(a) Binarized *Male* image



(b) Pseudo-random 8 × 8 binary block

FIGURE 16. Watermark images used in tests.

shown in Fig. 16a. Fig. 16b is a binary block of size 8 × 8, pseudo-randomly generated.

Prior to the watermark embedding, the light field was converted from RGB to YCbCr, so that all simulations were performed using only the Y channel. In order to evaluate the watermark imperceptibility, we computed the average structural similarity index, $\mathrm{SSIM}_{\mathrm{avg}}$, by computing the SSIM for each view, that is

$$\mathrm{SSIM}_{\mathrm{avg}} = \frac{1}{ST}\sum_{i=1}^{S}\sum_{j=1}^{T}\mathrm{SSIM}(i,j), \tag{15}$$

where

$$\mathrm{SSIM}(i,j) = \frac{(2\mu_I\mu_R + Z_1)(2\sigma_{IR} + Z_2)}{(\mu_I^2 + \mu_R^2 + Z_1)(\sigma_I^2 + \sigma_R^2 + Z_2)} \tag{16}$$

stands for the SSIM of the view (i, j), I and R respectively correspond to the evaluated and reference images of the view in question, and $Z_1 = (0.01 \times F)^2$ and $Z_2 = (0.03 \times F)^2$, where F is the light field dynamic range (e.g., 255 for 8 bpp). Besides $\mathrm{SSIM}_{\mathrm{average}}$, the average PSNR of each of the individual views were also calculated for the imperceptibility evaluation, that is

$$\mathrm{PSNR}_{\mathrm{average}} = \frac{1}{ST}\sum_{i=1}^{S}\sum_{j=1}^{T}\mathrm{PSNR}(i,j), \tag{17}$$

where

$$\mathrm{PSNR}(i,j) = 10\log_{10}\frac{255^2}{\mathrm{MSE}(i,j)}, \tag{18}$$

$$\mathrm{MSE}(i,j) = \frac{1}{NM}\sum_{k=1}^{N}\sum_{l=1}^{M}(I(k,l) - R(k,l))^2 \tag{19}$$

and $I(k, l)$ and $R(k, l)$ are the k-th *pixel* of the l-th line of the original and watermarked view (i, j), respectively.

In order to evaluate the robustness, we have considered the normalized correlation (NC) and the bit error rate (BER). The NC is defined as

$$\mathrm{NC} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} I(i,j)R(i,j)}{\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{M} I(i,j)^2}\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{M} R(i,j)^2}}, \tag{20}$$

where N and M are, respectively, the number of rows and columns of the watermark, and $I(i, j)$ and $R(i, j)$ are the j-th pixel of the i-th row of the watermark under evaluation (extracted watermark) and the reference one, respectively. The BER is defined as the ratio between the number of bits recovered with error and the total amount of bits in the watermark.

VI. RESULTS

In the first part of this section, we present the results related to the scanning algorithms. We then present the results of our experiments, with respect to the imperceptibility of the watermark and its robustness against attacks.

A. COMPARISON BETWEEN SCANNING ALGORITHMS

In [41], some scanning methods for four-dimensional blocks are presented, two of which performed better. The first technique is a four-dimensional diagonal scanning pattern, identified here as 4D Diagonal Scanning (4DD Scan); the other method is a two-dimensional scanning pattern with double diagonals, one internal at *vu* coordinates and one external at *ts* coordinates. This method is identified here as 2D Double Diagonal Scan (2DDD Scan). Such scanning algorithms consider the entire 4D block, without performing the separation by columns. In this subsection, we compare the performance of the scanning method proposed by us in Section IV-B with that of the two above referred methods.

To compare the scanning methods, the percent cumulative energy (PCE) curve for the "Desktop" image is shown in Fig. 17, with the evaluated scanning patterns. The curve also shows the PCE considering a real ordering procedure (quicksort) and the Average Scanning for reference. Among the evaluated methods, the one which obtained the greatest area under the curve is C-3DD, followed by 4DD. Fig. 18 presents a zoomed-in version of these curves. Note that the C-3DD and 4DD curves are relatively close to the Average Scanning curve. Fig. 19 shows a boxplot chart with the area under the curve considering all 30 test images. As can be seen, the C-3DD method obtained higher median and average values.

In the percent accumulated energy curve of the C-3DD, in Fig. 18, although this is the method that has the largest area under the curve, it is possible to notice that, in the first coefficients, the curve of the 4DD Scan method has higher energy compaction. In fact, the C-3DD method starts to provide higher energy compaction from the 75th coefficient, which remains higher up to the 4096th coefficient. Thus, for compression purposes, for example, if we intend to preserve at least the 75 first coefficients of each 4D block, using the 3D Diagonal Column Scan would contribute for a higher quality image reconstruction than using the 4D Diagonal Scan.

It is also possible to compare the degree of ordering obtained by the proposed scanning methods with the degree of ordering obtained by a well-established scan for two-dimensional images, the zigzag used in the JPEG standard. This can be performed by evaluating the measure of disor-

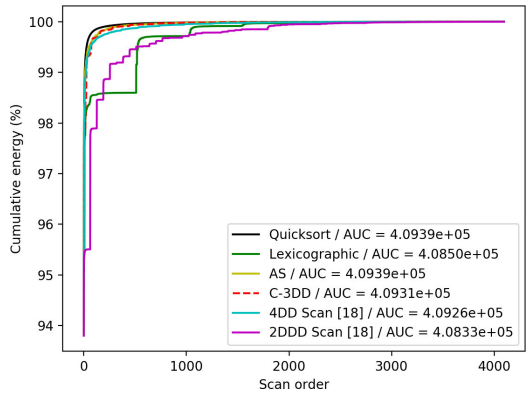


FIGURE 17. Percent cumulative energy curve for the *desktop* image after scanning the coefficients.

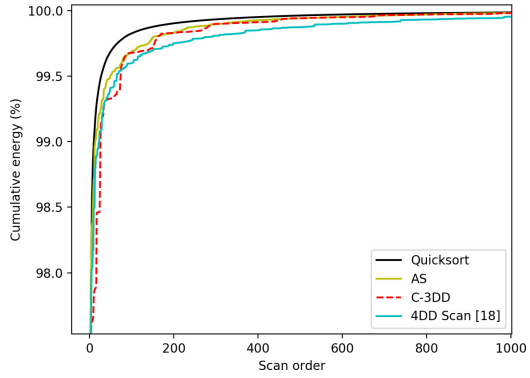


FIGURE 18. Zoomed-in curve of the percent cumulative energy after scanning the first 1000 coefficients of image *desktop*.

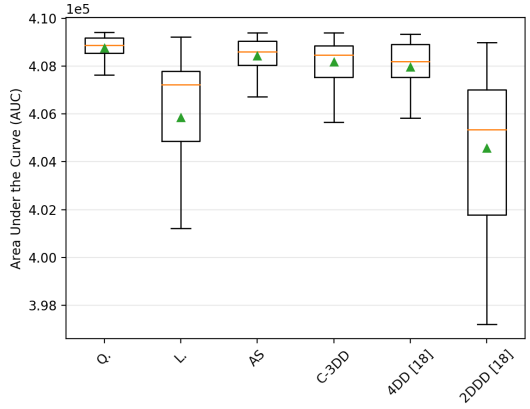


FIGURE 19. Boxplot chart of the area under the curve (percent cumulative energy) considering 30 test images. The orange lines mark the median values and the green triangles mark the average values.

der of a permutation or numerical sequence [51], [52]. The metric used with this purpose is the number of inversions (NI) [52]. For a descending order, an inversion can be defined as follows: let $\mathbf{S}$ be a numerical sequence; if $i < j$ and $S_i < S_j$, then the pair (S_i, S_j) is an inversion of $\mathbf{S}$. In this work, the number of inversions per combination (NIC) was

Algorithm 3 Number of Inversions per Combination

procedure NIC($\mathbf{S}, n$) ▷ $\mathbf{S}$ → numerical sequence; n → length of $\mathbf{S}$.
  $qty\_comb \leftarrow 0$
  $qty\_inv \leftarrow 0$
  for $iin[1, n-1]$ **do**
    for $jin[i+1, n]$ **do**
      if $|S[i]| < |S[j]|$ **then**
        $qty\_inv \leftarrow qty\_inv + 1$
      $qty\_comb \leftarrow qty\_comb + 1$
  return $qty\_inv \div qty\_comb$

TABLE 1. Average and standard deviation of the number of inversions per combination (NIC).

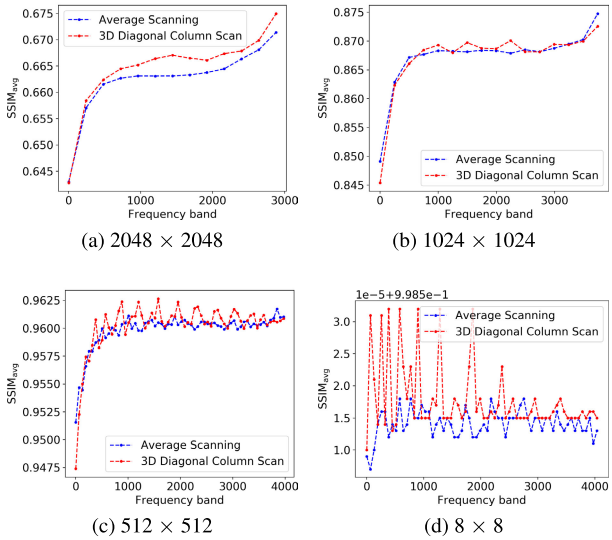
Method	NIC Average	NIC Standard Dev.
JPEG 2D Zigzag	0.3331	0.0470
Average Scanning	0.2877	0.0344
3D Diagonal Column Scan	0.2951	0.0312
4D Diagonal Scan	0.3235	0.0259
2D Double Diagonal Scan	0.3912	0.0198

calculated considering all possible combinations with $i < j$; this metric is described in Algorithm 3.

In this work the average NIC was calculated for all 8×8 blocks in the 2D-DCT domain for 210 images from the image database of the Institute of Signal and Image Processing of the University of Southern California [50] and compared with the average NIC calculated for all $8 \times 8 \times 8 \times 8$ blocks in the 4D-DCT domain for the 30 test images obtained from the EPFL's database [45], considering the proposed scanning methods and the two methods given in [41]. Table 1 presents the average and standard deviation NIC results; a completely ordered sequence has a NIC value equals to zero. The first scanning method is the conventional zigzag used in the JPEG standard. Note that the methods proposed in this article produce an average NIC value lower than that produced by the method used in JPEG. This suggests that the ordering performed by the proposed methods is satisfactory in terms of energy compaction at the "low frequencies". Moreover, the proposed methods obtain better results than the 4DD and 2DDD scanning methods. The most prominent scanning modes were the Average Scanning and 3D Diagonal Column Scan.

B. IMPERCEPTIBILITY

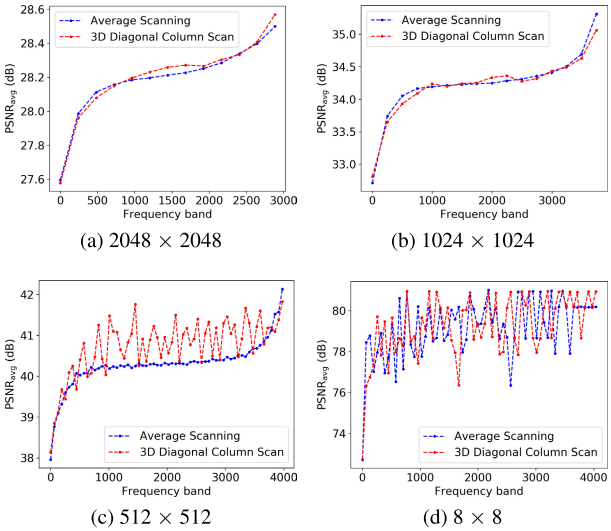
Fig. 20 shows the SSIM_{avg} results according to the "frequency range" used to embed the watermark without bit replication. For each watermark size, a different range of the frequency spectrum was used according to the number of coefficients required for embedding. For the 2048×2048 watermark, 995 bits were embedded per block, whereas for the 1024×1024 watermark, it was necessary to change 248 coefficients per block; for the 512×512 watermark, it was necessary to embed 62 bits per block and, for 8×8 size watermark, 1 coefficient of 64 different blocks has been



(a) 2048 × 2048 (b) 1024 × 1024

(c) 512 × 512 (d) 8 × 8

FIGURE 20. Curve with the $SSIM_{avg}$ values for the light field *bikes* with different watermark sizes and different frequency ranges used for embedding them.



(a) 2048 × 2048 (b) 1024 × 1024

(c) 512 × 512 (d) 8 × 8

FIGURE 21. Curve with the $PSNR_{avg}$ values for the light field *bikes* with different watermark sizes and different frequency ranges used for embedding them.

changed.[4] An important point is that, in all tests, the DC component was not modified. For example, for the watermark of size 2048 × 2048, the first range of coefficients used was from 2 to 995 (along the vectorized version of the block with dimensions $8\times8\times8\times8$ in the 4D-DCT domain, obtained after applying the proposed scanning algorithm). As expected, note that, as higher frequencies are used, the greater are the $SSIM_{avg}$ results, since these regions have lower energy concentration and modifying them degrades the light field with less intensity. The lowest values are obtained when the coefficient range used to embed the watermark occupies low-frequency regions. This behavior is more evident in cases where more bits are embedded per block (such as for watermarks of sizes 2048 × 2048, 1024 × 1024, and 512 × 512). For an 8 × 8 watermark, where only 1 bit is embedded in some blocks, the referred behavior is not so clear; in this case, the $SSIM_{avg}$ variation is low, ranging from 0.998510 to 0.998532, for example, when the 3D Diagonal Column Scan is used. The maximum, minimum, and average values are shown in Table 2. Despite the results obtained when scanning the coefficients with the AS and C-3DD methods are similar, we observe that the growth trend of the curve is less oscillating when using the Average Scanning method. The results presented are for the image *Bikes*, however, the same behavior is exhibited, if other light field images are considered.

Fig. 21 presents the $PSNR_{avg}$ results, for the *Bikes* light field image, according to the frequency range used to embed the watermark. The results were obtained using the same simulation configuration to obtain the $SSIM_{avg}$ results. As expected, the $PSNR_{avg}$ results behave similarly to the $SSIM_{avg}$ results. As we use higher frequency bands we get higher $PSNR_{avg}$ values. The maximum, minimum, and average values are shown in Table 2. Similar results are obtained for the other light field images.

C. ROBUSTNESS TO 4D-DCT-BASED COMPRESSION

We consider a compression process which basically consists of calculating in a block basis the 4D-DCT of the watermarked light field and then, for each block, retaining the $P\%$ coefficients of the highest absolute value and zeroing the other $(100 - P)\%$. In Fig. 22, the NC results of the original watermark are presented concerning the watermarks extracted after the light field *Black Fence* was compressed with different percentages of retained coefficients for each watermark size. Observing the color scale, we note the reduction in the NC value as fewer coefficients are retained. Also, as we use higher frequency ranges, NC results decrease, as embedding the watermark at high frequencies makes them more likely to be degraded by the compression process. The lowest NC results obtained were 0.62, 0.64, 0.68 and 0.79 for watermarks of sizes 2048 × 2048, 1024 × 1024, 512 × 512 and 8 × 8, respectively, and with retention of only 1% of the coefficients. The highest NC results obtained for watermarks of sizes 2048 × 2048, 512 × 512 and 8 × 8 are 1; for watermarks of size 1024 × 1024, the value is 0.99. In short, as fewer coefficients are retained and higher frequency bands are used, the higher the BER values are, as expected. The PSNR and SSIM results behave similar to the NC results: the values of such metrics are higher for higher percentages of retention and lower frequency ranges. In all cases, similar results are obtained for the other light field images.

In Fig. 23, the extracted watermarks with the lowest NC values are shown for each percentage of retained coefficients (these watermarks were embedded in the higher frequency range). As expected, the compression process makes difficult

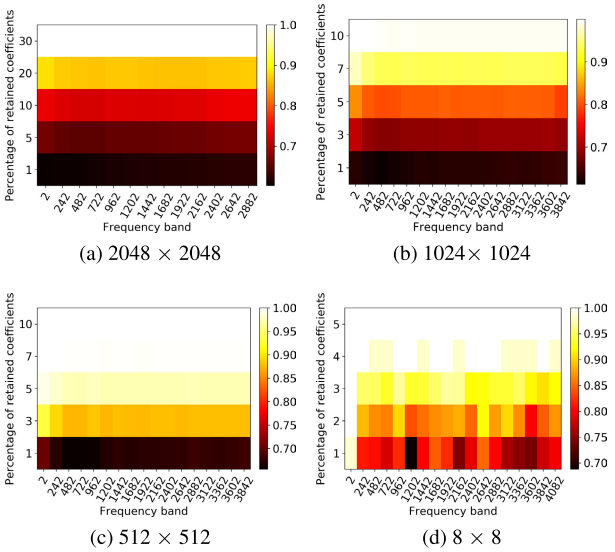
[4]Considering a light field image with dimensions 8 × 8 × 432 × 624, we have 4212 blocks of size 8 × 8 × 8 × 8.

TABLE 2. Maximum, minimum, and average $SSIM_{avg}$ values obtained when scanning the coefficients with the average scanning (AS) and 3D diagonal column scan (C-3DD) methods.

	2048×2048		1024×1024		512×512		8×8	
	AS	C-3DD	AS	C-3DD	AS	C-3DD	AS	C-3DD
Maximum	0.671	0.675	0.875	0.873	0.962	0.963	0.998518	0.998532
Minimum	0.643	0.643	0.849	0.845	0.952	0.947	0.998507	0.998510
Average	0.662	0.665	0.867	0.867	0.960	0.960	0.998514	0.998518

TABLE 3. Maximum, minimum, and average $PSNR_{avg}$ values obtained when scanning the coefficients with the average scanning (AS) and 3D diagonal column scan (C-3DD) methods.

	2048×2048		1024×1024		512×512		8×8	
	AS	C-3DD	AS	C-3DD	AS	C-3DD	AS	C-3DD
Maximum	28.500	28.569	35.311	35.062	42.131	41.828	81.011	80.959
Minimum	27.596	27.579	32.715	32.815	37.959	38.139	72.706	72.733
Average	28.189	28.201	34.230	34.209	40.298	40.730	79.195	79.252



(a) 2048×2048 (b) 1024×1024 (c) 512×512 (d) 8×8

FIGURE 22. Colormap for NC (normalized correlation) values for watermarks of different sizes, extracted from the light field *black fence* after compression with different percentages of retained coefficients. The NC values are indicated in the sidebar according to the color intensity.

to recover the watermark bits, since zeroed coefficients (of lower absolute value) tend to concentrate at high frequencies. The images present noise artifacts that increase in quantity as fewer coefficients are retained. On the other hand, we observe that, for images not generated in a pseudo-random way, even in critical cases, in which only 5%, 3% or 2% of the coefficients were retained, it is still possible to visually identify the watermark without difficulties.

D. IMPACT ANALYSIS OF THE WATERMARK EMBEDDING STRENGTH

In Figures 24, 25, 26 and 27, we show NC results for extracted watermarks, and $SSIM_{avg}$ for the watermarked images. We consider the embedding of the watermark in different frequency ranges and with different embedding strengths. The higher the strength, the better the quality of the extracted watermark and the worse the quality of the watermarked image. A similar tradeoff also exists for the frequency bands; the higher the frequency range used for embedding, the worse the quality of the extracted watermark and the better the quality of the watermarked image. For this reason, it is important to perform a tradeoff analysis to identify the range of coefficients and the strength parameters that allow to meet the specific requirements of the application. In Fig. 24, for example, where the results for watermarks of size 2048 $\times$ 2048 are shown, we note that it is possible to obtain reasonable NC values (greater than 0.85), even with only 20% of retained coefficients, with the strength parameter equal to 120. However, this parameter degrades the watermarked image, leading to $SSIM_{avg}$ less than 0.4.

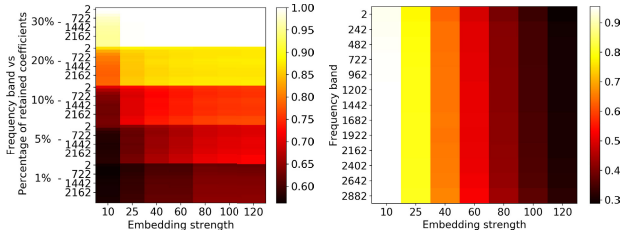
An alternative would be to use the strength parameter with a value of 10 and the frequency range limited to 242. With these values, one can obtain watermarks extracted with NC values greater than 0.8 when retaining 20% of the coefficients and greater than 0.9 when retaining 30% or more coefficients. Remember that, as can be seen in Fig. 23, extracted watermarks with NC values greater than 0.8 are still easily visualized. In contrast, with these values, it is possible to obtain a $SSIM_{avg}$, for the watermarked image, greater than 0.9, which represents a high degree of similarity with the original image. The same analysis can be performed for other watermark sizes (1024×1024, 512×512, and 8×8). In general, we understand that using the strength parameter with a value of 40 and performing the embedding in a limited frequency range between 2 and 240, seems to be adequate for most cases. The presented figures were obtained using the *Black Fence* image, but similar results are obtained for other images.

E. COMPARATIVE ANALYSIS AND OTHER ATTACKS

In order to evaluate the proposed watermark embedding and extraction method, we have computed the BER after simulating several possible attacks that an image could suffer under different conditions. The simulations were performed for the 30 test light field images. The proposed method was evaluated with and without bit replication and the scanning methods used were Average Scanning (AS) and 3D Diagonal

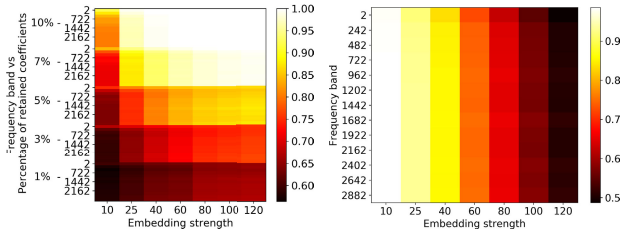
FIGURE 23. Watermarks extracted with different percentages of retained coefficients. All of these watermarks were embedded in the last coefficient range (high frequencies). In the subfigures, RC stands for retained coefficients.

Column Scan (C-3DD). The terms AS 10× and C-3DD 10× refer to the respective variations of the proposed method replicating the bits 10 times. The results of these variations were compared to those obtained using the methods proposed in [37] and [38], which also aim at watermarking the light field in the format in which it is captured. At this time, expanding such a comparison by including other methods is somewhat unfeasible. This is due to the fact that, although the literature has archived several light field watermarking techniques, most of these methods consider the image rendering scenario for Free-View-Television. On the other hand, the literature is scarce when referring to watermarking light fields in their native structure; being more specific, to the best of our knowledge, the only technique published in the literature that applies a 4D transform to the light field before embedding the watermark is that described in [38]. In any case, it is important to reiterate that our approach and those proposed in [37]and [38] lie in different watermark categories. Our work introduces a blind watermarking scheme; this means that only the watermarked image and the secret key used to embed the watermark are needed to perform its extraction. The cited works lie in the semi-blind watermark category: in addition to the watermarked image and the secret key, some additional information is required. In [37] and [38], such an information is a numerical sequence with the same size of the embedded watermark. In general, this restricts the scope



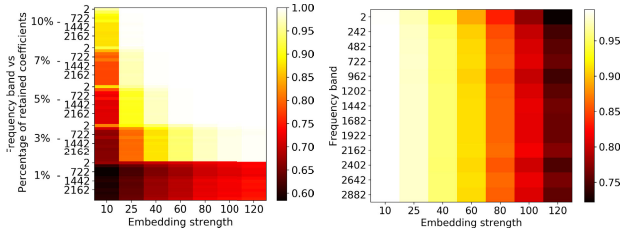
(a) NC (b) $SSIM_{avg}$

FIGURE 24. Colormap with NC (normalized correlation) values of extracted watermarks of size 2048 × 2048 and $SSIM_{avg}$ (structural similarity) of watermarked images for different ranges of coefficients and strength values. The watermarks were extracted from the light field *black fence* after compression with different percentages of retained coefficients. The values of NC and $SSIM_{avg}$ are indicated in the sidebar according to the color intensity.



(a) NC (b) $SSIM_{avg}$

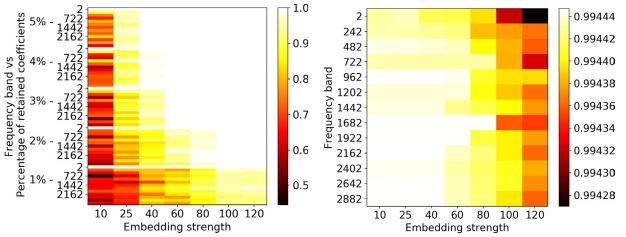
FIGURE 25. Colormap with NC (normalized correlation) values of extracted watermarks of size 1024 × 1024 and $SSIM_{avg}$ (structural similarity) of watermarked images for different ranges of coefficients and strength values. The watermarks were extracted from the light field *black fence* after compression with different percentages of retained coefficients. The values of NC and $SSIM_{avg}$ are indicated in the sidebar according to the color intensity.



(a) NC (b) $SSIM_{avg}$

FIGURE 26. Colormap with NC (normalized correlation) values of extracted watermarks of size 512 × 512 and $SSIM_{avg}$ (structural similarity) of watermarked images for different ranges of coefficients and strength values. The watermarks were extracted from the light field *black fence* after compression with different percentages of retained coefficients. The values of NC and $SSIM_{avg}$ are indicated in the sidebar according to the color intensity.

of possible applications; on the other hand, it can facilitate the correct extraction of the watermark and consequently increase the robustness against attacks. Due to a limitation related to the number of bits that can be embedded, in the method described in [37], only tests with watermarks of size 8 × 8 and 512 × 512 were performed. Another characteristic of this method is that it does not exploit the four-dimensional information on a light field; the embedding method uses a scheme based on the 2D-DCT of blocks of pixels randomly



(a) NC (b) $SSIM_{avg}$

FIGURE 27. Colormap with NC (normalized correlation) values of extracted watermarks of size 8 × 8 and $SSIM_{avg}$ (structural similarity) of watermarked images for different ranges of coefficients and strength values. The watermarks were extracted from the light field *black fence* after compression with different percentages of retained coefficients. The values of NC and $SSIM_{avg}$ are indicated in the sidebar according to the color intensity.

TABLE 4. Embedding strength values used in simulations for watermarks of size 8 × 8 and their impact on image quality in terms of PSNR (dB).

PSNR	AS	AS 10×	C-3DD	C-3DD 10×	[37], [38]
66.9	200	60	150	50	60
66.4	270	80	200	60	80
62.4	340	110	260	80	100
60.9	400	130	300	100	120
59.5	470	150	360	110	140

TABLE 5. Embedding strength values used in simulations for watermarks of size 512 × 512 and their impact on image quality in terms of PSNR (dB).

PSNR	AS	AS 10×	C-3DD	C-3DD 10×	[37]
30.7	120	40	120	40	60
28.2	160	50	160	50	80
26.3	210	60	200	60	100
24.7	250	80	240	80	120
23.9	270	90	270	90	140

chosen from the entire light field image. In the method presented in [38], the embedding capacity is reduced even more. In this case, we were only able to perform tests with a watermark of size 8 × 8. The method in [38] is similar to the one in [37] but the former first computes the 4D-DWT to exploit some of the 4D information and then applies the 2D-DCT to blocks of randomly chosen 4D-DWT coefficients.

The simulations were carried out for 14 different attacks. Also, to compare the methods, their embedding strengths were selected so that their impacts on the watermarked image are close in terms of PSNR (dB). Tables 4 and 5 show the embedding strengths used in each method and their impact on the image quality, in terms of PSNR (dB), after the embedding process.

The average BER values (containing the average of the results for 30 images, all attack parameters, and embedding strengths) are presented in Table 6, for watermarks of size 8 × 8, and in Table 7, for watermarks of size 512 × 512. In most results, as expected, one verifies an improvement in the BER results when using the version with bit replication instead of the version without replication. For watermarks of size 8 × 8, the method that obtained the best result for most attacks was the C-3DD 10×, which obtained, on average,

TABLE 6. Average BER results for watermarks of size 8 × 8 extracted after attacks under different conditions and with variations in the embedding strength. The simulations were performed in 30 test images. AS refers to average scanning and C-3DD to 3D diagonal column scan; the methods were applied without and with replication (10×). The best results are highlighted in bold and the second-best results are underlined.

Attack	AS	AS 10×	C-3DD	C-3DD 10×	[37]	[38]
Average Filter	0.0016	0.0023	**0.0000**	**0.0000**	0.3617	0.2219
Gaussian Filter	0.3086	0.0719	0.1633	**0.0000**	0.1398	0.0836
Median Filter	0.2219	0.1492	0.0016	**0.0000**	0.4359	0.2563
Gaussian Noise	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
Salt and Pepper Noise	**0.0000**	**0.0000**	**0.0000**	**0.0000**	0.0023	**0.0000**
Histogram Equalization	0.2906	0.1000	0.3000	**0.0594**	0.4844	0.4844
Sharpening	0.2906	0.1711	0.2742	0.1766	0.2484	**0.0688**
UV Crop	0.0938	**0.0430**	0.0938	**0.0430**	0.3438	0.4516
4D-DCT Compression	**0.0000**	0.0063	0.0039	0.0055	0.0641	0.0141
JPEG	0.1698	0.2823	0.1365	0.1615	0.1000	**0.0521**
JPEG 2000	**0.0000**	**0.0000**	**0.0000**	**0.0000**	0.1031	0.4844
Shuffled Views	0.4844	0.4844	0.4531	**0.3937**	0.4781	0.4219
Rotation + Cropping	0.4375	0.4062	0.2500	**0.0469**	0.4844	0.2431
Shearing + Cropping	0.0312	**0.0000**	**0.0000**	**0.0000**	0.2812	0.0766

TABLE 7. Average BER results for watermarks of size 512 × 512 extracted after attacks under different conditions and with variations in the embedding strength. The simulations were performed in 30 test images. The methods were applied with and without bit replication (10×). The best results are highlighted in bold and the second-best results are underlined.

Attack	AS	AS 10×	C-3DD	C-3DD 10×	[37]
Average Filter	0.3939	0.4398	0.4269	0.4291	**0.3859**
Gaussian Filter	0.4270	0.3759	0.4363	**0.3704**	0.3751
Median Filter	**0.3805**	0.4279	0.4123	0.4150	0.4369
Gaussian Noise	0.0009	0.0053	0.0037	0.0059	**0.0000**
Salt and Pepper Noise	**0.0000**	0.0002	**0.0000**	0.0002	0.0010
Histogram Equalization	0.2885	**0.1426**	0.2961	0.1436	0.4803
Sharpening	0.3878	0.2828	0.3865	0.2913	**0.1759**
UV Crop	0.0836	**0.0291**	0.0868	0.0293	0.3754
4D-DCT Compression	0.0665	0.2030	0.0653	0.2020	**0.0644**
JPEG	0.1667	0.1994	0.1901	0.1917	**0.1213**
JPEG 2000	0.0001	**0.0000**	0.0003	**0.0000**	0.0918
Shuffled Views	**0.4794**	**0.4797**	0.4814	0.4970	0.4851
Rotation + Cropping	0.4974	0.4881	0.4969	**0.4868**	0.4924
Shearing + Cropping	0.1647	**0.0243**	0.1578	0.0254	0.3356

a lower BER among the others for 11 out of 14 attacks. The method in [38] obtained the best result in 4 out of 14 attacks. One of these attacks is the JPEG compression, for which the method in [37] also produces a result slightly better than those provided by the proposed methods; this was predictable, since the methods given in [37] and [38] embed the watermark in the coefficients of a 2D transform.

For watermarks of size 512 × 512, the method proposed in [37] obtained, on average, the lowest BER results in 5 out of 14 attacks. In any case, the only attack for which it achieves expressively better results is the sharpening attack (the difference from the results of the variations of the proposed method is of order 10^{-1}). In other attacks where Ansari's first method achieves a better result, the biggest difference is of order 10^{-2} (JPEG compression attack). For other attacks, the best results are almost equally distributed among the variations of the proposed method. In general, for the evaluated watermark sizes, even though the proposed method is blind, it obtains better results, in comparison to Ansari's methods, which are semi-blind.

VII. CONCLUSION

Light fields have significantly contributed to the development of new image capture and visualization technologies, since they allow the same scene to be viewed under different perspectives. The 4D-DCT seems to be a promising candidate for compression of this type of image, providing the energy compaction of lenslet light field images in the transform domain; this is the scenario that has motivated the research documented in this article. The proposed light field watermarking technique has demonstrated to meet requirements of imperceptibility and robustness against to compression in 4D-DCT domain and many other attacks, suggesting that the method and, more specifically, the developed coefficient scanning strategies can be successfully applied to a broad range of images.

In this context, it is also relevant to emphasize that the proposed scheme is highly flexible and may have its parameters adjusted so that the imperceptibility-robustness tradeoff of the algorithm is controlled and its performance is improved. Besides the possibilities already discussed in the previous sections, the referred adjustments can be performed by using error-correcting codes to embed / extract the watermark, for instance. In this case, we hope that more watermark bits can be embedded without significant loss of robustness, when compared to the bit replication procedure described in Section III-C. We can also try to improve the performance of the algorithm by combining the quantization index modulation technique with other techniques for embedding / extracting the watermark; we could use, for example, variable

parameter chaotic mapping to increase the security and the robustness of the scheme [53], automorphism systems to scramble the binary image used as watermark [54] and fuzzy inference systems to model human visual system and also increase the watermarking robustness [55]. At the same time we have been evaluating the feasibility of some of the above mentioned alternatives, we have made efforts to evaluate the performance of using the proposed scanning procedures in the scenario of compression of light fields.

REFERENCES

[1] I. Ihrke, J. Restrepo, and L. Mignard-Debise, "Principles of light field imaging: Briefly revisiting 25 years of research," *IEEE Signal Process. Mag.*, vol. 33, no. 5, pp. 59–69, Sep. 2016.

[2] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field image processing: An overview," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 926–954, Oct. 2017.

[3] L. Mignard-Debise and I. Ihrke, "A vignetting model for light field cameras with an application to light field microscopy," *IEEE Trans. Comput. Imag.*, vol. 5, no. 4, pp. 585–595, Dec. 2019.

[4] L. Si, G. Jiang, X. Hu, and B. Liu, "Retina 3D perception reconstruction algorithm based on visual light field image," *IEEE Access*, vol. 8, pp. 196804–196812, 2020.

[5] B. Wang, Q. Peng, E. Wang, K. Han, and W. Xiang, "Region-of-interest compression and view synthesis for light field video streaming," *IEEE Access*, vol. 7, pp. 41183–41192, 2019.

[6] X. Min, J. Zhou, G. Zhai, P. Le Callet, X. Yang, and X. Guan, "A metric for light field reconstruction, compression, and display quality evaluation," *IEEE Trans. Image Process.*, vol. 29, pp. 3790–3804, 2020.

[7] H. Zhang, C.-J. Zhu, X. Tang, N. He, Y. Zeng, Q. Liu, and S. Xiang, "A light field sparse and reconstruction framework for improving rendering quality," *IEEE Access*, vol. 8, pp. 209308–209319, 2020.

[8] X. Jiang, M. Le Pendu, R. A. Farrugia, and C. Guillemot, "Light field compression with homography-based low-rank approximation," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 1132–1145, Oct. 2017.

[9] A. Ivan, W. Williem, and I. K. Park, "Joint light field spatial and angular super-resolution from a single image," *IEEE Access*, vol. 8, pp. 112562–112573, 2020.

[10] Y. Tian, H. Zeng, J. Hou, J. Chen, and K.-K. Ma, "Light field image quality assessment via the light field coherence," *IEEE Trans. Image Process.*, vol. 29, pp. 7945–7956, 2020.

[11] J. Du, X. Sang, X. Yu, X. Gao, C. Gao, B. Liu, L. Liu, P. Wang, and B. Yan, "Large viewing angle floating three-dimensional light field display based on the spatial data reconstruction (SDR) algorithm," *Opt. Commun.*, vol. 475, Nov. 2020, Art. no. 126229.

[12] Y. Xu, H. Nagahara, A. Shimada, and R.-I. Taniguchi, "TransCut: Transparent object segmentation from a light-field image," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3442–3450.

[13] J. Fiss, B. Curless, and R. Szeliski, "Light field layer matting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 623–631.

[14] T. Yang, Y. Zhang, X. Tong, X. Zhang, and R. Yu, "A new hybrid synthetic aperture imaging model for tracking and seeing people through occlusion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 9, pp. 1461–1475, Sep. 2013.

[15] N. Li, J. Ye, Y. Ji, H. Ling, and J. Yu, "Saliency detection on light field," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2806–2813.

[16] M. Schambach and M. Heizmann, "A multispectral light field dataset and framework for light field deep learning," *IEEE Access*, vol. 8, pp. 193492–193502, 2020.

[17] T.-C. Wang, J.-Y. Zhu, E. Hiroaki, M. Chandraker, A. A. Efros, and R. Ramamoorthi, "A 4D light-field dataset and CNN architectures for material recognition," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 121–138.

[18] J. Y. Lee and R.-H. Park, "Depth estimation from light field by accumulating binary maps based on foreground–background separation," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 955–964, Aug. 2017.

[19] S. Jiao, C. Zhou, Y. Shi, W. Zou, and X. Li, "Review on optical image hiding and watermarking techniques," *Opt. Laser Technol.*, vol. 109, pp. 370–380, Jan. 2019.

[20] A. K. Singh, "Data hiding: Current trends, innovation and potential challenges," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 16, no. 3s, pp. 1–16, Jan. 2021.

[21] A. Nikolaidis, S. Tsekeridou, A. Tefas, and V. Solachidis, "A survey on watermarking application scenarios and related attacks," in *Proc. Int. Conf. Image Process.*, vol. 3, 2001, pp. 991–994.

[22] A. K. Singh and C. Kumar, "Encryption-then-compression-based copyright protection scheme for E-governance," *IT Prof.*, vol. 22, no. 2, pp. 45–52, Mar. 2020.

[23] H. Chen and W. Xu, "Secure and robust color image watermarking for copyright protection based on lifting wavelet transform," in *Proc. 25th Int. Conf. Mechatronics Mach. Vis. Pract. (M2VIP)*, Nov. 2018, pp. 1–5.

[24] X. Chen, G. Qu, and A. Cui, "Practical IP watermarking and fingerprinting methods for ASIC designs," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2017, pp. 1–4.

[25] F. Hartung and F. Ramme, "Digital rights management and watermarking of multimedia content for m-commerce applications," *IEEE Commun. Mag.*, vol. 38, no. 11, pp. 78–84, Nov. 2000.

[26] A. Anand and A. K. Singh, "Watermarking techniques for medical data authentication: A survey," *Multimedia Tools Appl.*, Apr. 2020.

[27] P. W. Wong and N. Memon, "Secret and public key image watermarking schemes for image authentication and ownership verification," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1593–1601, 2001.

[28] G. Coatrieux, H. Huang, H. Shu, L. Luo, and C. Roux, "A watermarking-based medical image integrity control system and an image moment signature for tampering characterization," *IEEE J. Biomed. Health Informat.*, vol. 17, no. 6, pp. 1057–1067, Nov. 2013.

[29] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*. San Mateo, CA, USA: Morgan Kaufmann, 2007.

[30] P. Paudyal, F. Battisti, A. Neri, and M. Carli, "A study of the impact of light fields watermarking on the perceived quality of the refocused data," in *Proc. 3DTV-Conf. True Vis. Capture, Transmiss. Display 3D Video (3DTV-CON)*, Jul. 2015, pp. 1–4.

[31] A. Koz, C. Cigla, and A. A. Alatan, "Free-view watermarking for Free-view television," in *Proc. Int. Conf. Image Process.*, Oct. 2006, pp. 1405–1408.

[32] A. Koz, C. Çığla, and A. A. Alatan, "Watermarking for light field rendering," in *Proc. 15th Eur. Signal Process. Conf.*, 2007, pp. 2296–2300.

[33] E. E. Apostolidis and G. A. Triantafyllidis, "Watermark selection for light field rendering in FTV," in *Proc. 3DTV Conf. True Vis. Capture, Transmiss. Display 3D Video*, May 2008, pp. 385–388.

[34] A. Koz, C. Cigla, and A. A. Alatan, "Watermarking for image based rendering via homography-based virtual camera location estimation," in *Proc. 15th IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 1284–1287.

[35] A. Koz, C. Cigla, and A. A. Alatan, "Watermarking of free-view video," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1785–1797, Jul. 2010.

[36] H. Tian, Z. Wang, Y. Zhao, R. Ni, and L. Qin, "Spread spectrum-based multi-bit watermarking for free-view video," in *Proc. Int. Workshop Digit. Watermarking*. Atlantic City, NY, USA: Springer, 2011, pp. 156–166.

[37] A. Ansari, S. Hong, G. Saavedra, B. Javidi, and M. Martinez-Corral, "Ownership protection of plenoptic images by robust and reversible watermarking," *Opt. Lasers Eng.*, vol. 107, pp. 325–334, Aug. 2018.

[38] A. Ansari, G. Saavedra, and M. Martinez-Corral, "Robust light field watermarking by 4D wavelet transform," *IEEE Access*, vol. 8, pp. 203117–203133, 2020.

[39] G. Alves, M. P. Pereira, M. B. de Carvalho, F. Pereira, C. L. Pagliari, V. Testoni, and E. A. B. da Silva, "A study on the 4D sparsity of JPEG pleno light fields using the discrete cosine transform," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 1–5.

[40] M. B. de Carvalho, M. P. Pereira, G. Alves, E. A. B. da Silva, C. L. Pagliari, F. Pereira, and V. Testoni, "A 4D DCT-based lenslet light field codec," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 435–439.

[41] G. De Oliveira Alves, M. B. De Carvalho, C. L. Pagliari, P. G. Freitas, I. Seidel, M. P. Pereira, C. F. S. Vieira, V. Testoni, F. Pereira, and E. A. B. Da Silva, "The JPEG pleno light field coding standard 4D-transform mode: How to design an efficient 4D-native codec," *IEEE Access*, vol. 8, pp. 170807–170829, 2020.

[42] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inf. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.

[43] F. Pereira, E. A. da Silva, and G. Lafruit, "Plenoptic imaging: Representation and processing," in *Academic Press Library in Signal Processing*, vol. 6. Amsterdam, The Netherlands: Elsevier, 2018, pp. 75–111.
[44] F. Svanstrom, "Properties of a generalized Arnold's discrete cat map," M.S. thesis, Dept. Math., Linnaeus Univ., Växjö, Sweden, Jun. 2014.
[45] M. Rerabek and T. Ebrahimi, "New light field image dataset," in *Proc. 8th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, 2016, pp. 1–2. [Online]. Available: http://www.epfl.ch/labs/mmspg/EPFL-light-field-image-dataset
[46] (2020). *4D-DCT Scanning Repository*. Accessed: Dec. 6, 2020. [Online]. Available: https://github.com/felipebsferreira/4D-DCT-Scanning
[47] *JPEG Pleno Call for Proposals on Light Field Coding*, Standard ISO/IEC JTC 1/SC29/WG1N74014, Geneva, Switzerland, Jan. 2017.
[48] E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Process.*, vol. 9, no. 10, pp. 1661–1666, 2000.
[49] C. Knessl and W. Szpankowski, "Quicksort algorithm again revisited," *Discrete Math. Theor. Comput. Sci.*, vol. 3, no. 2, pp. 43–64, 1999.
[50] (2019). *The USC-SIPI Image Database*. Accessed in: Sep. 4, 2019. [Online]. Available: http://sipi.usc.edu/database/database.php
[51] H. Mannila, "Measures of presortedness and optimal sorting algorithms," *IEEE Trans. Comput.*, vol. C-34, no. 4, pp. 318–325, Apr. 1985.
[52] V. Estivill-Castro and D. Wood, "A survey of adaptive sorting algorithms," *ACM Comput. Surv.*, vol. 24, no. 4, pp. 441–476, Dec. 1992.
[53] R. Wang, H. Shaocheng, P. Zhang, M. Yue, Z. Cheng, and Y. Zhang, "A novel zero-watermarking scheme based on variable parameter chaotic mapping in NSPD-DCT domain," *IEEE Access*, vol. 8, pp. 182391–182411, 2020.
[54] Z. Zhou, S. Chen, and G. Wang, "A robust digital image watermarking algorithm based on dct domain for copyright protection," in *Smart Graphics*, Y. Chen, M. Christie, and W. Tan, Eds. Cham, Switzerland: Springer, 2017, pp. 132–142.
[55] C. Agarwal, A. Mishra, and A. Sharma, "Digital image watermarking in DCT domain using fuzzy inference system," in *Proc. 24th Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2011, pp. 000822–000825.

FELIPE A. B. S. FERREIRA was born in Recife, Pernambuco, Brazil, in 1988. He received the B.S. degree in computer science from the Catholic University of Pernambuco, in 2011, the M.S. degree in systems engineering from the University of Pernambuco, in 2017, and the Ph.D. degree in electrical engineering from the Federal University of Pernambuco, in 2020.

He is currently an Assistant Researcher with the Digital Signal Processing Research Group, Federal University of Pernambuco. His research interests include signal and image processing and computational intelligence.

JULIANO B. LIMA (Senior Member, IEEE) was born in Brazil. He studied electrical engineering in Brazil. He received the M.S. and Ph.D. degrees in electrical engineering from the Federal University of Pernambuco, Brazil, in 2004 and 2008, respectively.

He is currently an Associate Professor with the Department of Electronics and Systems, Federal University of Pernambuco. His main research interests include discrete and number-theoretic transforms, and their applications in digital signal processing, communications, and cryptography.