Displays 50 (2017) 63–69

Contents lists available at ScienceDirect

Displays

journal homepage: www.elsevier.com/locate/displa

Implementation of a simulated display for hexagonal image processing

Xiangguo Li

College of Information Science and Engineering, Henan University of Technology, Zhengzhou, Henan 450001, PR China

ARTICLE INFO

Keywords:
Hexagonal image processing
Sampling lattice
Simulated display
Voronoi cell

ABSTRACT

Hexagonal image processing is theoretically superior to that based on the common square lattice, and due to the lack of practical imaging devices, this paper intends to implement a simulated display. The paper investigates the sampling lattices used in the hexagonal image processing, and finds that variable lattices occur in hexagonal discrete Fourier transform (HDFT), while the most commonly used "hyperpel" approach, due to its fixed cell pattern, cannot handle such displaying tasks. Then, the paper proposes to represent each pixel with the exact Voronoi cell (VC) according to the sampling lattice. In the paper, a simple algorithm is presented to compute the VC vertices, and each simulated pixel is constructed with the VC filled with its intensity value, and then the simulated display is implemented by the tessellation of each simulated pixel on the correct lattice position. Finally, experimental results show that the proposed simulated display can display data without geometric distortion.

1. Introduction

Practical digital signals are mainly obtained by sampling the analog ones. For images, due to the 2-D nature, the sampling points lie on a 2-D plane and form a regular 2-D lattice. According to the sampling theory, the hexagonal lattice is the optimal sampling scheme for circularly band-limited analog images [1]. Since most optical systems are circularly symmetric and thus exhibit circularly low-pass nature, the hexagonal lattice is actually the optimal sampling scheme for sampled imaging systems. Compared with the commonly used square lattice, the hexagonal lattice can provide 13.4% fewer samples [1], and it is also superior with respect to its geometric properties, such as higher degree of symmetry, equal distance and uniform connectivity with its neighbors, as shown in Fig. 1. In recent years, especially with the rise of biologically inspired image processing, hexagonal image processing has spread to applications such as edge detection [2,3], hexagonal Gabor filtering [4], ultrasound image processing [5], and adaptive beamforming [6]. Besides, hexagonal lattice is also adopted in advanced imaging techniques, such as 3-D display system [7], 3-D endoscopy [8], and integral imaging system [9].

However, people are more familiar with the Cartesian orthogonal coordinate system as well as the square lattice. Also, the square lattice has the advantage of simplicity for practical data storage and addressing [10], thus it predominates in the commercial imaging devices, including image sensors (CCD and CMOS) and displays. Therefore, for the hexagonal image processing research at the present time, it is natural to adopt the simulation approach. For the data acquisition, lattice conversion can be applied by means of interpolation under the resampling framework [11–13]. For the data displaying, each pixel can be simulated by a proper cell and the simulated display can be constructed based on current displays. This paper concerns the data displaying task and proposes to implement a simulated display for hexagonal image processing.

Consider a common square lattice image displayed on a practical square lattice display. If we zoom in the image, it is easy to recognize that each pixel is presented with a small square cell, as illustrated in Fig. 2. This has motivated the construction of simulated display for hexagonal lattice data. Firstly, Lester and Sandor [14] proposed the "brick wall" technique, in which each hexagonal pixel was approximated with four (2×2) square pixels and even rows stagger odd rows with one square pixel pitch, as shown in Fig. 3(a). Obviously, the "brick wall" approach is coarse and can incur geometric distortion. Later, Wüthrich and Stucki [15] proposed the "hyperpel" technique, in which a more fine cell was used for each hexagonal pixel, as shown in Fig. 3(b). In the regular hexagonal lattice, the vertical and horizontal pitch ratio is , and in the "brick wall" approach, the ratio is 1:1, while in the "hyperpel" approach, the ratio is $7{:}8 = 0.8750$. Therefore, the "hyperpel" approach can be treated as a good approximation and it is the most common displaying approach used in current hexagonal image research [2,3,16–18].

We notice that, when dealing with samples in the frequency domain, the sampling lattices will vary with several factors (fully discussed in Section 2). Then, for the displaying task of the discrete spectrum, the "hyperpel" approach will fail to provide good

E-mail address: xiangguoli@gmail.com.

http://dx.doi.org/10.1016/j.displa.2017.09.005
Received 11 February 2017; Received in revised form 1 August 2017; Accepted 18 September 2017
Available online 21 September 2017

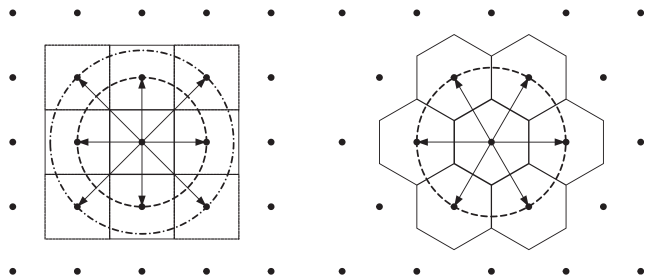

Fig. 1. Comparison of geometric properties between the square lattice (left) and the regular hexagonal lattice (right).

Fig. 2. Illustrations of displaying a square lattice image on the square lattice display and the partial enlarged views.

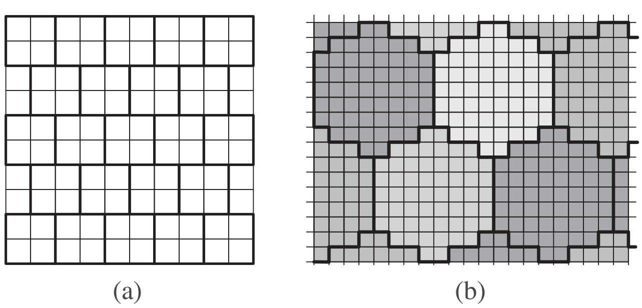

Fig. 3. Illustration of the techniques used for simulated hexagonal display. (a) the "brick wall" and (b) the "hyperpel".

approximation and due to its fixed cell pattern it can not be easily adapted either. For this reason, this paper intends to implement a simulated display that can deal with the general data displaying tasks used in hexagonal image processing. This is the main starting point of this paper.

Actually, the small squares in Fig. 2 serve as the zero-order hold reconstruction filter and the small square area is the Voronoi cell (VC) of the square lattice. Accordingly, each pixel of the hexagonal lattice data can also be represented with the VC of the hexagonal lattice [11,19–21], as shown in Fig. 4. This approach, instead of directly concerning the discrete nature of practical displays as in both the "brick wall" and the "hyperpel" approaches, is based on exact VC shape of the hexagonal lattice. Owing to the exact VC shape, the image data can be displayed without geometric distortion, and by simply linearly adjusting the VC values, the image can also be easily zoomed without need of interpolating the image data. This paper will continue to apply this motivation into the general 2-D sampling lattice, i.e., the proposed simulated display will be based on the exact VC shape with regard to the given sampling lattice and it will be able to cope with the need of data displaying tasks in the hexagonal image processing.

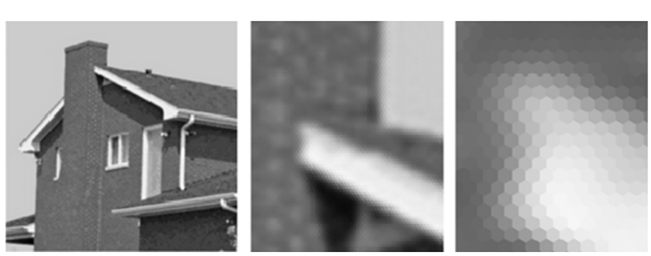

Fig. 4. Illustrations of displaying the hexagonal lattice image data based on the VC of the hexagonal lattice.

VC is an essential property of lattice. In digital image processing, VC is a fundamental tool in the sampling and reconstruction analysis. VC has also been used in hex-splines [22], Voronoi splines [23], screen reconstruction analysis [24], etc. However, computing the VC of a lattice, i.e., determining the vertices, is not direct and the task may be quite complicated for general lattices [25,26]. This paper concerns the 2-D sampling lattices in the hexagonal image processing, and owing to the regular nature, only the VC vertices with regard to one lattice point is needed. In the paper, an intuitive algorithm is presented to compute the VC vertices, and then each pixel cell as well as the whole simulated display can be constructed.

The remainder of the paper is organized as follows. For the better understanding the motivation, Section 2 discusses the sampling lattices used in the hexagonal image processing. Section 3 presents the algorithm to compute the VC vertices. Then, Section 4 implements a simulated display. Finally, Section 5 summarizes this paper.

2. Lattices in hexagonal image processing

First of all, the (virtually) hexagonally sampled digital images correspond to a common regular hexagonal lattice in the spatial domain. Meanwhile, the periodic sampling process in the spatial domain causes the original spectrum to be periodically replicated in the frequency domain, for which the extension pattern is defined by the reciprocal lattice that is also a regular hexagonal lattice but with different orientation. Given the sampling matrix $\mathbf{V}$ for a given spatial sampling lattice, and the corresponding reciprocal lattice $\mathbf{U}$ is given by $\mathbf{U} = 2\pi\mathbf{V}^{-\mathrm{T}}$. Specifically, these two basic lattices in the hexagonal sampling are shown in Fig. 5, in which $\mathbf{V} = [\mathbf{v}_1|\mathbf{v}_2] = [1,-0.5;0,\sqrt{3}/2]$ and $\mathbf{U} = [\mathbf{u}_1|\mathbf{u}_2] = 2\pi[1,0;1/\sqrt{3},2/\sqrt{3}]$.

On the other hand, spectrum analysis and frequency domain image processing are fundamental from both the theoretical and the practical perspectives. However, although a digital image can be obtained from either practical imaging sensor or any resampling approach, the spectrum of the obtained digital image is still continuous because it is simply the periodic extension of the original continuous spectrum. To enable the digital processing techniques in the frequency domain, it is often assumed to construct the periodic sequence in the spatial domain, which corresponds to sampling the periodic spectrum in the frequency domain. This is the basis for the discrete Fourier series (DFS) as well as the discrete Fourier transform (DFT). With the spatial sampling matrix $\mathbf{V}$ and the spatial periodic extension pattern $\mathbf{N}$, the frequency domain sampling matrix is given as $\mathbf{S} = 2\pi(\mathbf{VN})^{-\mathrm{T}}$ [27].

In the commonly used square lattice case, the spatial sampling matrix $\mathbf{V}$ is diagonal, and the extension pattern $\mathbf{N}$ is usually selected as

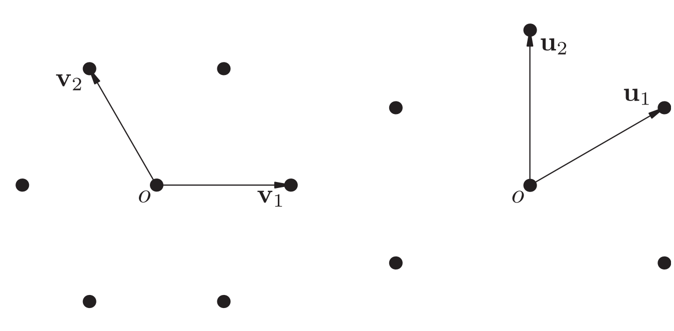

Fig. 5. Illustration of the hexagonal sampling lattice in the spatial domain (left) and the corresponding reciprocal lattice in the frequency domain (right).