BACK TO BASICS: TOWARDS NOVEL COMPUTATION AND ARRANGEMENT OF SPATIAL SENSORY IN IMAGES

Wei Wen*, Siamak Khatibi

Institute of Communication, Blekinge Institute of Technology, Karlskrona, Sweden

* corresponding author: wei.wen@bth.se

Abstract. The current camera has made a huge progress in the sensor resolution and the low-luminance performance. However, we are still far from having an optimal camera as powerful as our eye is. The study of the evolution process of our visual system indicates attention to two major issues: the form and the density of the sensor. High contrast and optimal sampling properties of our visual spatial arrangement are related directly to the densely hexagonal form. In this paper, we propose a novel software-based method to create images on a compact dense hexagonal grid, derived from a simulated square sensor array by a virtual increase of the fill factor and a half a pixel shifting. After that, the orbit functions are proposed for a hexagonal image processing. The results show it is possible to achieve image processing operations in the orbit domain and the generated hexagonal images are superior, in detection of curvature edges, to the square images. We believe that the orbit domain image processing has a great potential to be the standard processing for hexagonal images.

Keywords: Hexagonal pixel, Square pixel, Hexagonal sensor array, Hexagonal processing, Fill factor, Convolution, Orbit functions, Orbit transform.

1. Introduction

Nowadays, the ubiquitous influence of cameras in our life is undoubtable and this is thanks to the current camera sensory technique, which has made a huge progress on increasing the sensor resolution and the low-luminance performance [1]. The progress achievements are due to the size reduction of the sensory element (pixel), improvement in generation of the signal from collected light (quantum efficiency), and the used hardware techniques of the sensor [1, 2]. However, the image quality is not affected only by the pixel size or quantum efficiency of a sensor [3]. As the sensor pixel size becomes smaller, a smaller die size detection, gaining higher spatial resolution and obtaining lower signal-to-noise ratio are required; all in cost of a lower dynamic range and a fewer number of tonal levels. The form, arrangement, and inter-element distance of sensors play significant roles in the image quality, which is verified by a comparison between the current sensor techniques and animal, especially human, visual systems. The effect of the inter-element distance on an image quality was studied in [4] which showed that the inter-element distance could be decreased by the means of a physical modeling and thus obtaining higher quality images, higher dynamic range and greater number of tonal levels. Anatomical and physiological studies indicate that our visual quality related issues, such as high sensitivity, high speed response, high contrast sensitivity, high signal to noise ratio, and optimal sampling are related directly to the form and arrangement of the sensors in the visual system [5, 6]. In the human eye, three types of color photoreceptors, cones, are packed densely within a hexagonal pattern form and are located mostly in fovea in the center of retina [7]. Figure 1 shows a dramatic increase of the cone cross sectional area and a decrease of cone density within the eccentricity range represented in a strip of the inner segments of photoreceptors from the foveal center along the temporal horizontal meridian. Rods, another type of photoreceptors with non-color property, first appear at about 100 μm from the foveal center and are smaller than the cones.

Hexagonal photoreceptors arrays are not found only in human, but also often in compound eyes of insects and other invertebrates. Actually, such hexagonal arrays are more common for animals and plants than any other geometric arrays, such as rectilinear orthogonal arrays. This is due to the need of a motion detection, which is obtained based on the local difference of light intensity between adjacent neighboring photoreceptor cells [8–10] using the lateral inhibition process [8, 11]. Lateral inhibition is a contrast enhancement computation to exaggerate the light intensity differences of the neighboring cells; it is useful in an edge detection. The hexagonal array provides the best candidate for a contiguous neighboring-cell computation of the local light intensity difference between adjacent cells [12]. By using the contiguous neighboring cells at 60° angle, the first-order computation of light intensity differences is computed, and for the computation of finer angular differences, such as 30° angle, the second-order adjacent cells are used. Thus, this provides means for a symmetric computation, using one set of computational algorithms to compute the light intensity difference of exactly 6 contiguous neighbors. Each successive higher-order neighbor will compute the light intensity difference with the angular direction rotated by 30° degree successively. In the gradient based edge
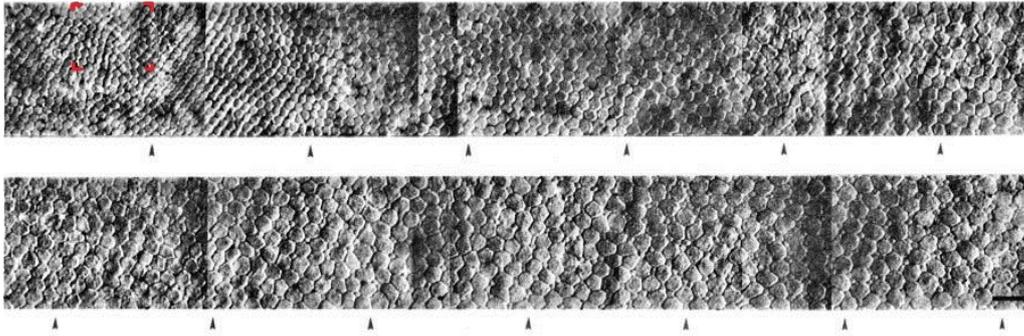
FIGURE 1. The enhanced image of inner segments of a human fovea1 photoreceptor mosaic from the original image printed in [7]. The mosaic strip is extending 575 μm from the fovea1 center along the temporal horizontal meridian; shown from upper left to lower right in the figure. Arrowheads indicate the edges of the sampling windows. Brackets, shown as red, indicate a quadrant of the first sampling window with the highest density of cones. The midpoint of the boundary of this quadrant and a quadrant adjacent to it in the temporal direction (to the right) with similar density and mean spacing was considered to be the point of 0.0 eccentricity. The strip contains profiles of only cones up to the fifth window, where the small profiles of rods begin to intrude. The large cells are cones and the small cells are rods and the bar is 10 μm.

detection, a local process at pixel level, the edges are discriminated by comparing the light intensity gradient of neighboring cells to find sudden changes of light intensity in adjacent pixels. It is shown in numerical studies that, when using the spatial computation, the hexagonal pixel-arrays are more efficient for edge detection instead of the rectilinear arrays [12–18]. It achieves a 40 % computational efficiency by using a hexagonal edge detection operator [13, 16] and it can efficiently detect motion in six different directions at 60 degree [6]. Due to a biological inspiration and aforementioned benefitsl of hexagonal sensory arrays implementation; generally, two main approaches are followed by researchers to acquire hexagonal sampled images. The first approach is to manipulate the result of conventional acquisition sampling devices using square sensor arrays, via software, to generate a hexagonally sampled image. The second approach is to use dedicated hardware to acquire the image, such as the super CCD from Fujifilm whose sensor structure is hexagonal [19], or color filters in hexagonal shape for the image sensors [20] to improve the quality of the acquired color by the sensor.

In this paper, we propose a novel software-based method to create images on a dense hexagonal sensor grid by shifting the sensor array virtually. This is derived from a simulated camera sensor array by a virtual increase of fill factor and transferring of the shifted square pixel grid into a virtual hexagonal grid. In our method, the images are firstly rearranged into a new grid of virtual square grid composed by subpixels with the estimated value of the fill factor. A statistical framework proposed in [4], consisting of a local learning model and the Bayesian inference, is used to estimate new subpixel intensities. Then the subpixels are projected onto the square grid, shifted and resampled in the hexagonal grid. Unlike previous hexagonal image processing methods, the intensity of each hexagonal image pixel is estimated in

our method,which results to maintaining the density of the arrangement. To the best of our knowledge, our approach is the first work that tries to create a hexagonal image by a virtual increase of a fill factor. After the hexagonal image is generated, type A2 orbit function from Weyl group is used to map the image into the orbit domain. Then smoothing and gradient filtering, as typical image processing operations, are done in the orbit domain on the hexagonal images. We believe such hexagonal image processing has great potential to be the standard processing for hexagonal arrangement due to its practical usage. Such processing is surprisingly more genuine and close to the way how human brain works.

This paper is organized as follows. In Sections 2 and 3, the related researches of hexagonal resampling and the methodology are discussed more in detail. Section 4 presents the experiment setup. Then the results are shown and analyzed in Section 5. Finally, we summarized and discussed our work in this paper in Section 6.

2. RELATED WORKS TO HEXAGONAL RESAMPLING

The software based approaches rearrange the conventional square pixel grids to generate the hexagonal images. There are three most common methods for simulating hexagonal resampling [21]:

(1.) The hexagonal grid is mimicked by a half-pixel shift, which is derived from delaying the sampling by a half a pixel on the horizontal direction as it is shown in Figure 2 [22]. In Figure 2, the left and right patterns are showing the conventional square lattice and the new pseudo-hexagonal sampling structure, whose pixel shape is still square; see the six connected pixels by the dashed lines. In such sampling structure, the distance between the
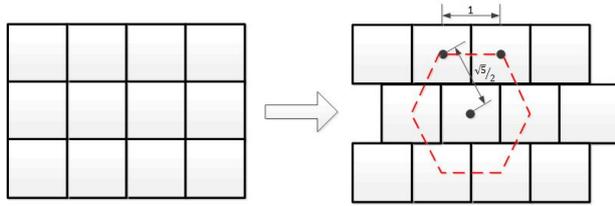
FIGURE 2. The procedure from square pixels to hexagonal pixels by half-pixel shifting method.

FIGURE 3. an example of hyperpel

```
Begin
  Is <- Select Pixel Matrix
  F <- Fill factor
  S <- Active Area Size (F)
  G <- Gaussian Distribution Estimation (I)
  In <- Introduce Gaussian Noise (G)
  P <- Active Area Size of Middle pixel
  While P <= S
        L <- Local Learning (In)
        X <- Subpixel Intensity (L)
  End while
  Ev <- Subpixel Gaussian Estimation (X)
  H <- Hexagonal grid projection (Is)
  E <- Histogram Distribution (Ev)
  Es <- Subpixel Intensity Estimation (E)
  I <- Pixel Intensity Estimation (Es)
End.
```

ALGORITHM 1. Resampling process.

two sampling points, pixels, are not the same; they are one or $\sqrt{5}/2$.

(2.) The hexagonal pixel is simulated by a set of square pixels, called hyperpel [23] which is widely used for displaying a hexagonal image on normal monitor. Figure 3 shows an example of the hyperpel, composed of 20×20 square sub-pixels.

(3.) The hexagonal pixel is generated by mimicing the Spiral Architecture method [24] by averaging the pixel grey level values of four square pixels in the structure. The method preserves the hexagonal arrangement as each hexagonal pixel is surrounded by six neighbour pixels. A reduction in the resolution of the resulted hexagonal image is expected due to averaging of four pixels for each hexagonal generated pixel. Also, in this method the distance between each of the six surrounding pixels and the central pixel is not the same as it should be in a hexagonal structure.

The spiral architecture is a common method for the hexagonal addressing and for the processing of the hexagonal images [25]. In the spiral architecture, all the hexagonal pixels are arranged on a spiral; it maps the hexagonal image into a one-dimensional vector. There are still many problems related to the hexagonal arrangement and computation that cannot be solved with all these approaches, e.g. the image resolution and pixel intensity values are changed during the resampling from the square grid to the hexagonal grid. In the case of the hexagonal computation, the hexagonal image still has to be mapped to a certain architecture form ; as far as the Cartesian indexing and coordinate is not yield in hexagonal computation. For example, in the spiral architecture, the hexagonal image is mapped into a one-dimensional vector to achieve a faster addressing and processing. However, such mapping results to a complicated computation process for each image processing operation; e.g. due to the reduction of the neighbour pixels from six to two in the spiral architecture.

3. METHODOLOGY

3.1. HEXAGONAL PIXEL

In this section, the method for generating a hexagonal image by resampling and half-pixel shifting a square-pixel image is explained. According to the resampling process in [4], the process is divided into

three steps: projecting the sampled signal to a new grid of sub-pixels; estimating the values of subpixels at the resampling positions; estimating the new pixel intensity and arranging the data to the hexagonal grid.

The three steps are elaborated in the following, as well as in Algorithm 1.

(1.) A grid of virtual image sensor pixels is designed. Each pixel is divided into 20×20 subpixels. According to the known fill factor, the size of the active area is S by S, where $S = 20 \times \sqrt{F}$. The intensity value of every pixel in the image sensor array is assigned to the virtual active area in the new grid. The intensities of subpixels in the non-sensitive areas are assigned to be zero. An example of such sensor rearrangement on sub-pixel level is presented on the left in Figure 4, where there is a 3 by 3 pixels grid, and the light and dark grey areas represent the active and non-active areas in each pixel. The active area is composed by 12 by 12 subpixels, and thereby the fill factor becomes 36% according to the above equation, and the intensities of active areas are represented by different greylevel values.

(2.) The second step is to estimate the values of subpixels in the new grid of subpixels. Considering the statistical fluctuation of incoming photons and their conversion to electrons on the sensor, there is a need for a statistical model to estimate the
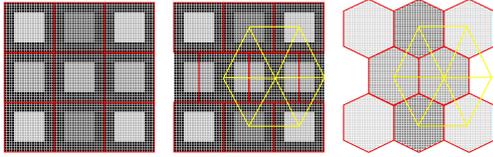
FIGURE 4. From left to right: the sensor rearrangement onto the subpixel, the projection of the square pixels onto the hexagonal grid by half pixel shifting and the pixel intensities displayed by hyperpel.

original signal. The Bayesian inference is used for the estimating every subpixel intensity, which is considered to be in the new position of resampling. Therefore, the more subpixels are used to present one pixel, the more accurate the resampling will be. By introducing the Gaussian noise into a matrix of selected pixels and estimating the intensity values of the subpixels at the non-sensitive area with different sizes of active area (local modeling), a vector of intensity values for each subpixel is created. Then the subpixel intensity will be estimated by the maximum likelihood.

(3.) In the third step, the subpixels are projected back to the original grid and then transformed onto a hexagonal grid shown as red grids on the left and in the middle of Figure 4 respectively. The red hexagonal grid, which is presented in the middle of Figure 4, is used for estimation of the pixel intensities of the hexagonal image. As the middle row of pixels is shifted to the right by a half a pixel, on subpixel level the sampling position is also shifted by a half a pixel as the method in [22]. The yellow grid represents the pixel connection in the new hexagonal sampling grid. In comparison to the method in [22], in our method the subpixels in each square area are estimated with respect to the virtual increase of the fill factor. The intensity value of a pixel in the hexagonal grid is the intensity value which has the strongest contribution in the histogram of belonging subpixels. The corresponding intensity is divided by the fill factor for removing the FF effect to obtain the hexagonal pixel intensity; as illustrated on the right in Figure 4 by means of hyperpels.

3.2. Hexagonal computation

The discrete transform of Lie group provides a possibility for frequency analysis of discrete functions defined on a triangle or hexagonal grids [26]. Figure 5 shows an example of fundamental domain F, the fundamental weights (ω_1, ω_2) of the orbit function in the form of a hexagon. The similarity of the grid in the fundamental domain of a certain orbit function from Weyl group to the grid of hexagonal images makes the orbit functions interesting for the hexagonal computation in which it becomes possible to process hexagonal images without any further transformation. In [27], the discrete orbit functions convolution was defined and was
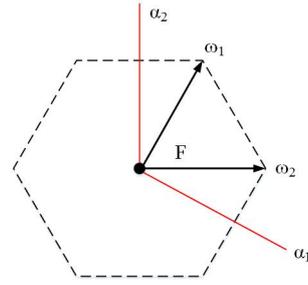
FIGURE 5. The fundamental domain F, fundamental weights (ω_1, ω_2), and simple roots (α_1, α_2).
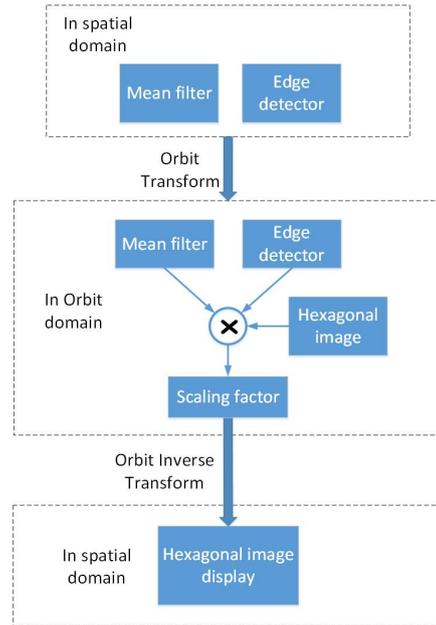
FIGURE 6. The flow chart of hexagonal computation of image processing operations.

used for image processing in the spatial domain. Here we propose hexagonal computational for the image processing operation using directly the orbit domain. The flow chart of such a hexagonal computation for typical operations is shown in Figure 6. A hexagonal image, generated by the proposed method in §3.1, is transformed to an orbit domain using the hexagonal grid related to the image. Then two filter kernels, constructed in the spatial domain, are transformed to the orbit domain using the same hexagonal grid related to the image. The process of the convolution in the orbit domain is a multiplication operation, shown as X in the middle dash box in Figure 6. According to the convolution theory in the orbit domain, the multiple convolutions can be combined to one by multiplication operations, which implies that all the image processing operations remain in the orbit domain and use the same hexagonal grid. The hexagonal image is obtained by the inverse transformation of results in the orbit domain in which the scaling factor plays a key role and it is defined by S^n, where the S is the constant scaling factors for each operation and the n is the number of operations.

4. Experimental setup

A group of optical images with assumption of known fill factor were simulated using our own codes and Image Systems Evaluation Toolbox (ISET) [28] in MATLAB. The ISET is designed to evaluate how image capturing components and algorithms influence image quality, and has been proved to be an effective tool for simulating the sensor and image capturing [29]. The fill factor value of 36 % was chosen for the simulated image sensor, having the resolution of 128×128 and 8-bits quantization levels. The sensor had a pixel area of 8×8 square μm, with well capacity of $10000\,\mathrm{e^-}$. The read noise and the dark current noise were set to $1\,\mathrm{mV}$ and $1\,\mathrm{mV/pixel/sec}$ respectively. The image sensor was rearranged to a new grid of virtual square sensor pixels, each of which was composed of 20×20 subpixels. All the image simulation setup is the same as in [4]. The optical images are from COIL-20 (Columbia University Image Library) [30], which is a database of grayscale images of 20 objects. For generation of sensor images, the luminance of optical images was set to $200\,\mathrm{cd/m^2}$ and the diffraction of the optic system was considered limited to ensure that the brightness of the output is as close as possible to the brightness of the scene image. The exposure time was also set to what is used for the sensor with 100 % fill factor correspondingly for the simulated sensor. For capturing images, the sensor exposure time is set to $1\,\mathrm{ms}$. All the processing is programmed and done by Matlab2015 on a HP laptop with an Intel i7-5600U CPU and a 16GB RAM memory to keep the process stable and fast.

5. Results and discussion

Three of the hexagonal images generated according to the proposed method mentioned in § 3.1 are shown next to the bottom row of Figure 7, where the corresponding optical images, simulated sensor images, recovered square-pixel images are shown from top to bottom rows respectively. The bottom row in Figure 7 shows the images that are zoomed in the red rectangle area. For visualization purposes each pixel in the hexagonal images is mapped according to the hyperpel method in [23] and is composed of 20×20 subpixels. The corresponding logarithm of histograms of the hexagonal and square-pixel images in Figure 7 is shown in Figure 8, which indicates that in both recovered square-pixel images and generated hexagonal images the tonal levels are extended, and also the tonal ranges are wider in comparison to the simulated camera sensor image. The generated hexagonal images are quite similar to the square-pixel images in intensity values according to their histograms shown in Figure 8. This is reasonable as far as the hexagonal images are obtained by half-pixel shifting of the square-pixel image recovered with the enhanced fill factor.

In our test, the type A2 orbit function from Weyl group is used for the transformation of the hexago-
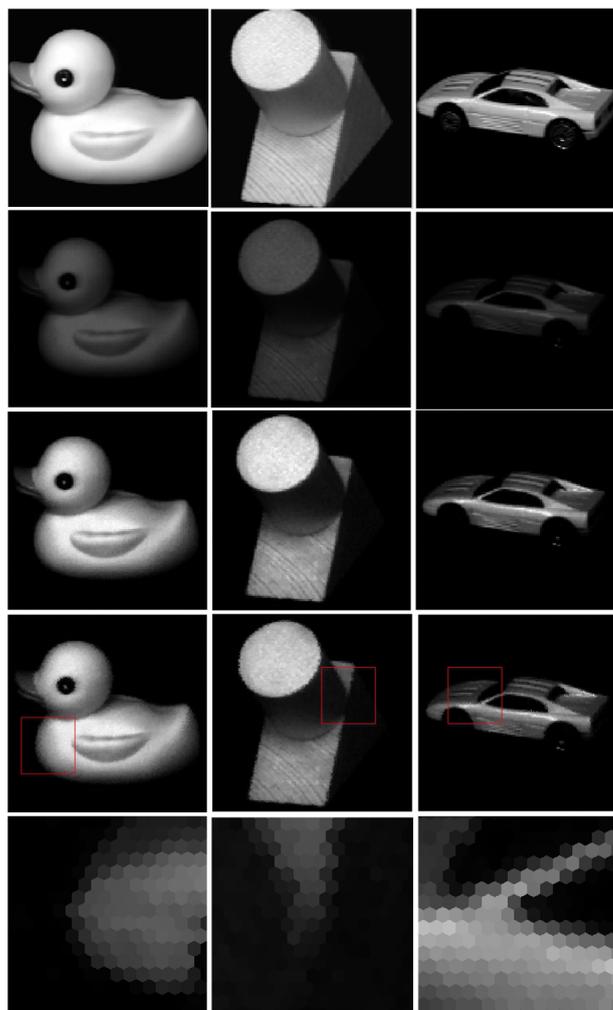
FIGURE 7. From top to bottom, the optical images, simulated sensor images, recovered square-pixel images, hexagonal images and the zoomed regions, shown with red rectangles, in hexagonal images.

nal image to the orbit domain. By using the orbit functions, there is no need to map the hexagonal image to certain coordinates system or an addressing system. We believe such hexagonal image processing has great potential to be the standard processing for hexagonal arrangement due to its practical usage. Such processing is surprisingly more genuine and close to the way how human brain works [31]; considering certain pathways of mapping of the information from the hexagonal grid in the visual system.

In our experiment, three basic filter kernels in the spatial domain are used to operate on the generated hexagonal images. These filters are a mean filter, a gradient or an edge detector filter, and the combination of the two mentioned filters. According to the orbit convolution definition and a hexagonal grid, the mean and edge detector filter kernels are as follows:

$$f_{\mathrm{mean}} = 1/3 \begin{pmatrix} 1 & \\ & 1 \\ 1 & \end{pmatrix}$$
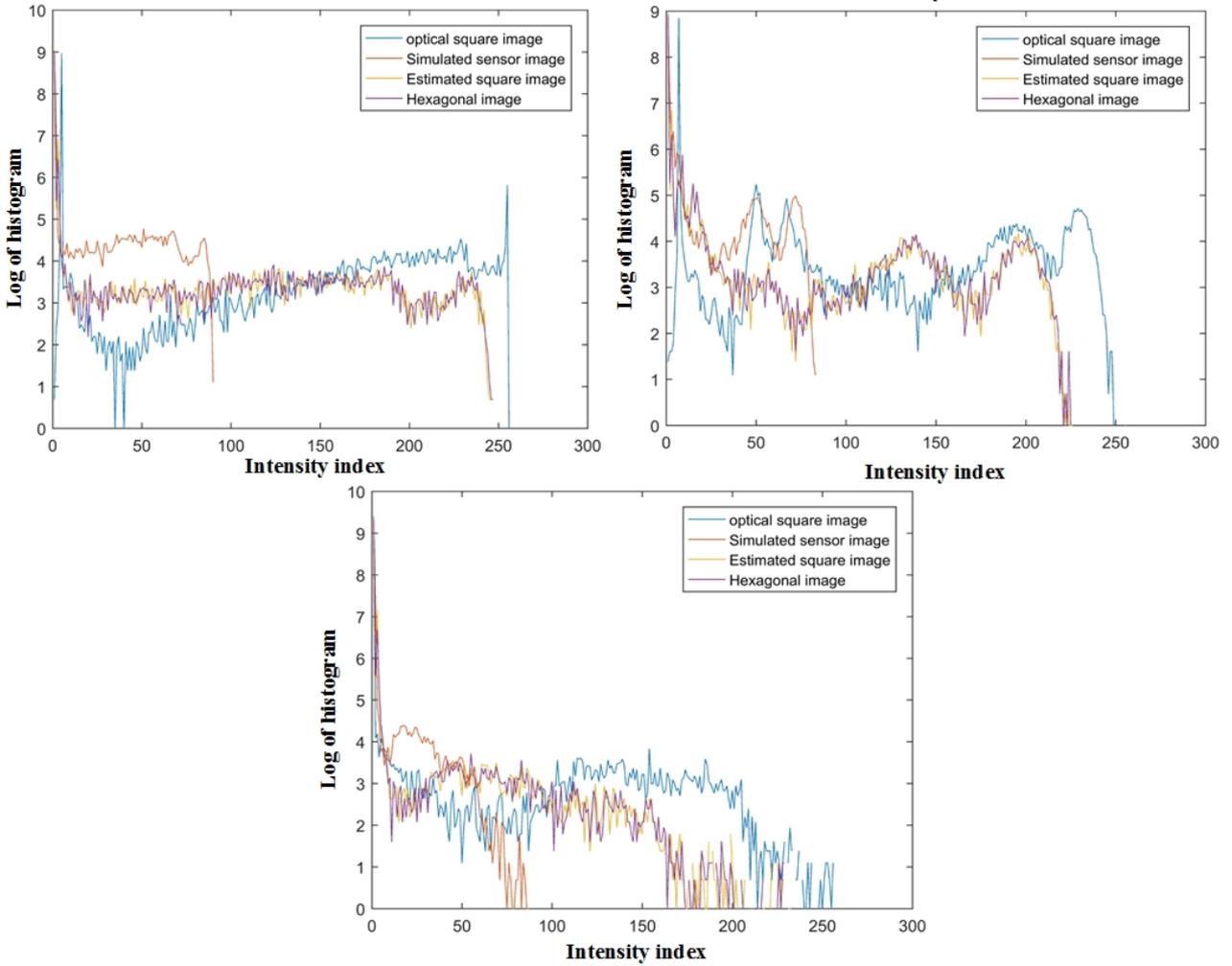
FIGURE 8. Log of histograms of the three object images which are shown in Figure 7: Bird (left), Cylinder (right), Car (bottom).

and

$$f_{\text{edge}} = \begin{pmatrix} 0 & \\ & -1 \\ 3 & \end{pmatrix},$$

where the two filter kernels in the spatial domain are

$$f_{\text{mean}} = 1/9 \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

and

$$f_{\text{edge}} = \begin{pmatrix} 0 & 0 & -1 \\ -1 & 3 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

The filtering results according to the hexagonal computation, see § 3.2, are shown in Figure 9. From the left to right, the filtered images by the mean, the edge detector, and the combination filters are shown, where the edges are shown as a binary image. The results of the edge detector filter are improved by applying the smoothing filter duo to the used edge detector filter which is a first order gradient computation.

Each pixel in the hexagonal image has six contiguous neighbors which results to more effective and
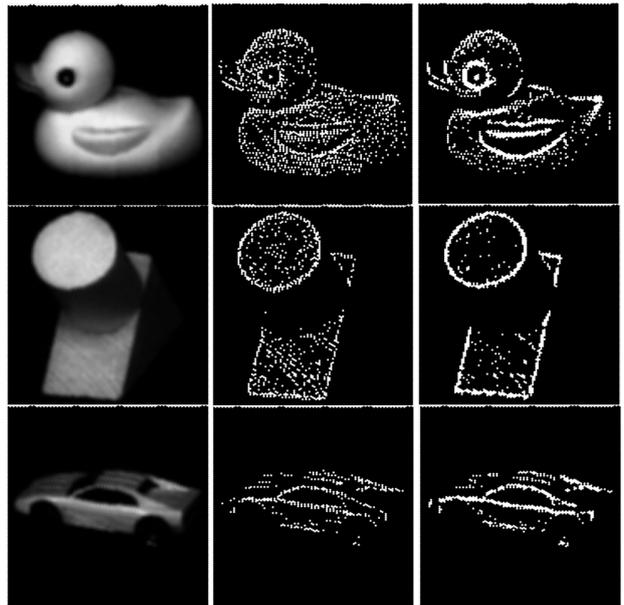
FIGURE 9. From the left to right, the filtered images by the mean filter, the edge detector filter, and the combination filter.

Images	Object 1 Bird	Object 2 Cylinder	Object 3 Car
Optical square	4858	3528	1998
Recovered square	3169	2357	4858
Hexagonal	5135	3295	3436

TABLE 1. The result of ratio images for different type of images.

efficient responsivity to gradient-based edge detection. To show this responsivity, two different mean filters, whose sizes are 3×3 and 5×5, are used for smoothing of the optical square-pixel images, recovered square-pixel images and hexagonal images, then the edge detector is used on the smoothed images to detect the edges in the images. The two resulted edge-detected images, using the two smoothing filters on each of the optical square-pixel images or the recovered square-pixel images or the hexagonal images, are gray level images. The pixel-based intensity ratio of each pair of the edge detected images is computed and finally the sum of such ratio images is obtained. Apparently, the blurring effect of more smoothed images reduces the detected edges. In Table 1, the result of the ratio images for different type of images is shown. According to the table, although it is impossible to detect the vertical edges due to the hexagonal form of our hyperpel, the hexagonal images still can preserve the edges superior in comparison to the square-pixel images.

6. CONCLUSION

In this paper, a novel approach is proposed for generating the hexagonal images from a simulated sensor based on a known fill factor of the sensor. The results show that the generated hexagonal images have the same tonal range and tonal levels as the square-pixel image. Also, their histogram distribution is very similar, indicating that our approach keeps the source information as much as possible during the generation. The orbit functions from Weyl group are used for hexagonal computation. Using orbit functions, it is possible to process hexagonal images with multiple operators, and at the same time, in the orbit domain without further need of mapping to a different coordinate system or addressing system. Although the processing speed of the orbit function convolution is still very slow, it provides a special domain for the hexagonal computation and for the image processing operations. The results also show that the edge detection on the hexagonal images preserves the edges superiorly, in comparison to the square-pixel images; with a significant curvature detection ability. In the future, our work would be to improve the processing speed of the orbit functions and to test our hexagonal images generation method with real cameras, to develop a way to create a hexagonal image from a conventional rectangle image sensor.

REFERENCES

[1] D. B. Goldstein. Physical limits in digital photography. *Northlight Images* 2009.

[2] B. Burke, P. Jorden, P. Vu. Ccd technology. *Experimental Astronomy* **19**(1-3):69–102, 2005.

[3] T. Chen, P. B. Catrysse, A. El Gamal, B. A. Wandell. How small should pixel size be? In *Electronic Imaging*, pp. 451–459. International Society for Optics and Photonics, 2000.

[4] W. Wen, S. Khatibi. Novel software-based method to widen dynamic range of ccd sensor images. In *Image and Graphics*, pp. 572–583. Springer, 2015.

[5] T. D. Lamb. Evolution of phototransduction, vertebrate photoreceptors and retina. *Progress in retinal and eye research* **36**:52–119, 2013.

[6] N. D. Tam. Hexagonal pixel-array for efficient spatial computation for motion-detection pre-processing of visual scenes. *Advances in image and video processing* **2**(2):26–36, 2014.

[7] C. A. Curcio, K. R. Sloan, O. Packer, et al. Distribution of cones in human and monkey retina: individual variability and radial asymmetry. *Science* **236**(4801):579–582, 1987.

[8] B. D. Coleman, G. H. Renninger. Theory of delayed lateral inhibition in the compound eye of limulus. *Proceedings of the National Academy of Sciences* **71**(7):2887–2891, 1974.

[9] A. Schultz, M. Wilcox. Parallel image segmentation using the l4 network. *Biomedical sciences instrumentation* **35**:117–121, 1998.

[10] M. V. Srinivasan, G. D. Bernard. The effect of motion on visual acuity of the compound eye: a theoretical analysis. *Vision research* **15**(4):515–525, 1975.

[11] S. Laughlin. A simple coding procedure enhances a neuron's information capacity. *Zeitschrift für Naturforschung c* **36**(9-10):910–912, 1981.

[12] X. He, W. Jia, N. Hur, et al. Bilateral edge detection on a virtual hexagonal structure. In *Advances in Visual Computing*, pp. 176–185. Springer, 2006.

[13] R. C. Staunton. The design of hexagonal sampling structures for image digitization and their use with local operators. *Image and vision computing* **7**(3):162–166, 1989.

[14] E. Davies. Optimising computation of hexagonal differential gradient edge detector. *Electronics Letters* **27**(17):1526–1527, 1991.

[15] S. Abu-Baker, R. Green. Detection of edges based on hexagonal pixel formats. In *Signal Processing, 1996., 3rd International Conference on*, vol. 2, pp. 1114–1117. IEEE, 1996.

[16] R. C. Staunton. Hexagonal image sampling: A practical proposition. In *1988 Robotics Conferences*, pp. 23–27. International Society for Optics and Photonics, 1989.

[17] B. Gardiner, S. Coleman, B. Scotney. Multi-scale feature extraction in a sub-pixel virtual hexagonal environment. In *Machine Vision and Image Processing Conference, 2008. IMVIP'08. International*, pp. 111–116. IEEE, 2008.

[18] E. Davies. Low-level vision requirements. *Electronics & communication engineering journal* **12**(5):197–210, 2000.

[19] Z. T. Jiang, Q. H. Xiao, L. H. Zhu. 3d reconstruction based on hexagonal pixelâĂŹs dense stereo matching. In *Applied Mechanics and Materials*, vol. 20, pp. 487–492. Trans Tech Publ, 2010.

[20] D. Schweng, S. Spaeth. Hexagonal color pixel structure with white pixels, 2008. US Patent 7,400,332.

[21] X. He, W. Jia. Hexagonal structure for intelligent vision. In *Information and Communication Technologies, 2005. ICICT 2005. First International Conference on*, pp. 52–64. IEEE, 2005.

[22] B. Horn. *Robot vision.* MIT press, 1986.

[23] C. A. Wüthrich, P. Stucki. An algorithmic comparison between square-and hexagonal-based grids. *CVGIP: Graphical Models and Image Processing* **53**(4):324–339, 1991.

[24] X. He. 2-d object recognition with spiral architecture. 1999. *University of Technology, Sydney* .

[25] P. Sheridan, T. Hintz, W. Moore. *Spiral Architecture for machine vision.* University of Technology, Sydney, 1996.

[26] A. Akhperjanian, A. Atoyan, J. Patera, V. Sahakian. Application of multi-dimensional discrete transforms on lie groups for image processing. *Data Fusion for Situation Monitoring, Incident Detection, Alert and Response Management* **198**:404, 2005.

[27] G. Chadzitaskos, L. Háková, O. Kajínek. Weyl group orbit functions in image processing, 2014. `arXiv:1404.0566`.

[28] J. E. Farrell, F. Xiao, P. B. Catrysse, B. A. Wandell. A simulation tool for evaluating digital camera image quality. In *Electronic Imaging 2004*, pp. 124–131. International Society for Optics and Photonics, 2003.

[29] J. Farrell, M. Okincha, M. Parmar. Sensor calibration and simulation. In *Electronic Imaging 2008*, pp. 68170R–68170R. International Society for Optics and Photonics, 2008.

[30] S. A. Nene, S. K. Nayar, H. Murase, et al. Columbia object image library (coil-20). Tech. rep., Technical Report CUCS-005-96, 1996.

[31] A. B. Watson, A. J. Ahumada. A hexagonal orthogonal-oriented pyramid as a model of image representation in visual cortex. *IEEE Transactions on Biomedical Engineering* **36**(1):97–106, 1989.