Hexagonal Image Compressionusing Singular Value Decomposition in Python

Prathibha Varghese
Research Scholar,
Electronics & Communication Department,
Noorul Islam Centre for Higher Education,
Tamil Nadu , India

Dr.G.Arockia Selva Saroja
Associate Professor,
Dept. Electronics & Communication Engineering,
Noorul Islam Centre for Higher Education,
Tamil Nadu , India

2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS) | 978-1-7281-7136-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/ACCESS51619.2021.9563312

Abstract: With the advent of multimedia technologies in last two decades, there is a widespread need for efficient storage and transmission of data. Dealing with the vast information interchangein this digital era, image compression for reduction in byte size of graphics image file without loss of image quality to an acceptable level becomes the large interest area. Inspired from the biological models of human fovea, hexagonal image processing has gained a lot of attention in artificial intelligence era that deals with the application of image processing system that combines the benefits of biologically motivated structures. In this paper a singular value decomposition (SVD) over hexagonal image compression which is a missing stone in computer vision which provides higher packing density, higher angular symmetry and uniform connectivity. Due to lack of developments in hexagonal imaging devices, different resampling methods like alternate pixel shift method, half pixel shift method, pseudo hexagonal pixel method for sourcing hexagonal images. SVD is one of the powerful cutting-edge technology for image compression algorithms. SVD based image compression is performed on hexagonal grid and is compared with square grid using different parameters like compression ratio, compression size, PSNR and MSE using PYTHON's SVD function. SVD based hexagonal image achieves the goal of compression by preserving good image quality at higher compression ratios, high computational efficiency, provides low mean square error (MSE), acceptable compression size depending on application and high peak to signal ratio (PSNR).

Keywords: Image compression, SVD, Pseudo hexagonal pixel, Hexagonal lattice, Hexagonal image processing

I. INTRODUCTION

In today's world, hexagonal lattice is used various research field in due to the various benefits of compared to the conventional square approach. In recent years, machine learning techniques based on artificial neural network for hexagonal lattice is also developed [15]. Compared to the square lattice structure hexagonal lattice require only 13.4% fewer samples than the traditional square lattice, which indicates that hexagonal image processing and sampling will be definitely computationally efficient related to storage and computational costs [19]. In addition to these benefits, hexagonal structure possesses many geometric benefits including equidistant pixels and uniform connectivity. Besides, the hexagonal structure is available in nature such as honey comb grid [16], compound eyes of insects [17], photoreceptor cells of retina [18].

Hexagonal grid image processing is a field of research which is closer to human vision as the photoreceptors of human retina are more similar to hexagonal pattern. This is as shown in Fig.1 [1].

Hexagonal sampling is an alternate sampling scheme which is introduces by peterson [2] for a 2-D Euclidean space. At the same time, hexagonal structure pattern for parallel computing devices was developed by Golay in [3]. Merserau [4] has shown that hexagonal samples benefit by using less than square specimen for the same entry data from computational savings.

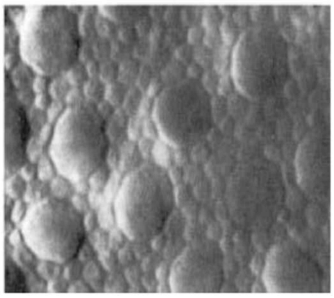

Figure 1. Photo receptors of human retina

Sheridan [20] in his advanced research in the field of hexagonal grid image processing, proposed a single dimensional addressing scheme based spiral architecture (SA). Using this architecture, different image operations like image translation and rotation is made possible by spiral addition and multiplication. But the prime disadvantage of these method is the loss of resolution and intensity accuracy. Wu et al. [21] overcome this difficulty by introducing the virtual spiral architecture concept during the processing of hexagonal image, and revert back to the original rectangular grid for viewing purpose. Since this concept is applied to the sub pixel level, image resolution is not much affected. Xiangguo Li [22] recently developed an innovative square to lattice conversion technique using pseudo hexagonal pixel in the frequency domain.

In this digital era, with the rapid demand for multimedia applications, there is a great concern regarding the manipulation, storage and transfer of images. Image compression not only deal with the reduction in the byte size but also holds a major role in the resolution of an image. SVD based image compression is a robust method which is widely used in today's technological era. Image compression techniques are classified broadly into two categories namely:

1. *Lossless compression*
2. Lossy compression

Lossless compression is a reversible compression technique in which compressed image can be reconstructed.it maintains same quality as that of the original image. This technique is applied critical application like medical field and technical drawings. Lossy compression is an irreversible compression technique in which it identifies each minute details and variation that human eye cannot identify and capture. Lossy compression compromises in the image quality thereby reducing the need of storage space and in terms of compression ratio. This technique is mainly preferred in multimedia applications.

This paper proposes a hybrid approach that combines hexagonal image processing with SVD compression which is quite familiar in square lattice addressing format. By combining both the hexagonal lattice and SVD compression technique, image can be compressed with good resolution. The resulting compressed image can be further processed for different image processing and computer vision application.

II. HEXAGONAL IMAGE FRAMEWORK

Hexagonal image processing framework (HIP) was used to perform SVD image compression. Due to the abundance of the square sampling device and heavy costs of switching to the hexagonal sampling limits the practical usage of hexagonally sampled images. Due to the practical limitation for sourcing and processing the hexagonal image, the main concerns involved are:

- Square lattice to hexagonal lattice conversion.
- Addressing schemes.
- Storage and manipulation of hexagonal image.
- Displaying this hexagonal image on current square grid graphical devices.

In the preceding sections, detailed discussion about the HIP frameworks is done.

III. PSEUDO HEXAGONAL PIXEL FRAMEWORK

Wuthrich and Stucki [1] designed pseudo hexagonal structure, in which single hexagonal pixel represented known as hyperpel, which can be simulated by combining group of square pixels as shown in Fig. 2.

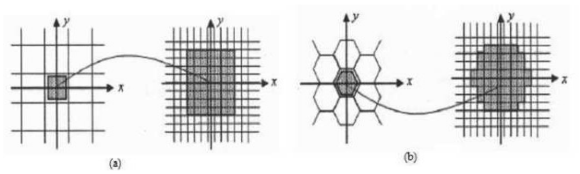

Figure 2. Creation of pseudo hexagonal pixel a) square b) hexagon

This method does not create exact hexagon shape, it creates a hexagon like shape which is having same properties as that of hexagon. Pixel selection for making cluster of hexagon pixels depends on two factors: Firstly, when tiling pixels, it should not overlap with the pixels and should not produce any gap between the neighbouring pixels. Secondly, hexagon cluster should closely resemble the hexagon geometry. Two possibility of hexagon clustering is shown in Fig. 3. The second pixel cluster is more effective in terms of length and shape [13].

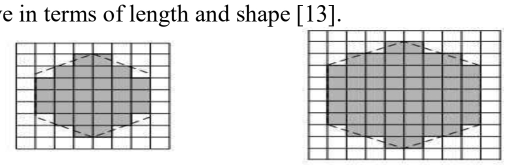

Figure 3. Two possible hexagon cluster pattern

Layering these hexagon hyper pels will produce a hexagonal image with zero gap or overlapping of pixels and assure the geometric charateristics of hexagonal based image as shown in Fig. 4.

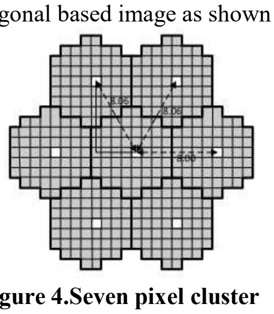

Figure 4.Seven pixel cluster

The distance measure calculated from the centre point of one pixel to centre point of other neighbouing pixel remains same.We can calculate this by using pythagorean theorem.

ie, $\sqrt{7^2 + 4^2} = 8.06 \cong 8\, units$

Almost all pixels are at equidistant fro the centre resulting in no image distortion.However,this method resulted in huge loss of resolution, as each hyper-pixel consist of original 56 pixels.Wu.et al [23] overcome this difficulty by proposing a partition method .All the pixels in an image is partioned into 7x7 group of sub pixels of identical intensity. Each hyper-pixel is the mage up of 56 sub pixels cluster as shown in Fig. 4 and average intensity is calculated from the 56-sub pixels used.

IV. SINGULAR VALUE DECOMPOSITION: BASIC MATHEMATICS

In 1873, Beltrami and Jordan autonomously discovered the design of the own paradigm for square matrices, and in the 1930s Jordan was interplayed with the rectangular matrices by Eckart and Young. SVD has many major benefits as: the pseudoinverse calculation of the matrix and mul- tivariate analyses, resolution of the smallest-square problem, maximum energy maximumenergy packing [12, 13]. SVD is a numerical technique used in many image processing applications. The main application areas of SVD image processing are: image compression, face recognition, watermarking and tex- ture classification. The main interest behind the study of SVD of an image (MxN) is to create image approximate decompositions using least number of terms in the diagonal matrix. This ap- proximation of the matrix forms the basis of SVD image compression, as images can be viewed as numerical values called pixel values in a matrix. This matrix approach is based on SVD image compression, as images are seen in a matrix called pixel values as numerical values. The real or complex rectangular matrix to a linear algebra is a factor similar to the diagonalization by eigen vectors of the symmetrical or Hermitian square matrices. SVD means that a system is divided into several linear autonomous components, which each have their own energy contribution [14, 15] in a stable and efficient manner.

A. SVD TECHNIQUE: BASIC MATHEMATICS

Consider a digital image A of size with M>=N, SVD of rectangular matrix is a decomposition of the form

A=UΣV (5)

$$A = U \sum V^T [\underbrace{u_1 \;\; u_2 \;\; \dots \;\; u_r}_{ColA} \underbrace{u_{r+1} \;\; \dots \;\; u_m}_{NulA^T}]$$

$$\begin{bmatrix} \sigma_1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 & 0 & \dots & 0 \\ \dots & & & & & & \\ 0 & 0 & \dots & \sigma_r & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \dots & & & & & & \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} v_1^T \\ v_2^T \\ \dots \\ v_r^T \\ v_{r+1}^T \\ \dots \\ v_n^T \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a\\a\\a\\a \end{matrix}}\right\} RowA \\ \left.\vphantom{\begin{matrix} a\\a\\a \end{matrix}}\right\} NulA \end{matrix} \qquad (6)$$

where U is an MxM orthonormal matrices, Σ is an Mx N matrix with the diagonal elements shows the singular values, Si of A and V is an NxN orthogonal matrix. Singular value decomposition decomposes the image A into the product of two orthonormal matrix and a diagonal matrix. The basic procedure to find out SVD decomposition are as follows [16]:

1.The square root of the eigen values of AA^T and $A^T A$ are used to find out singular values.
2. place the diagonal matrix in the lowering order of singular value
3. U and V are calculated from the own eigenvectors of AA^T and A^TA
4.The square root of the eigen values are the singular values along the diagonal of the matrix Σ.
5. Using Σ, U, V obtain the image A.
6. Eliminate the unused singular values in matrix to get compressed image of A

V. SVD BASED IMAGE COMPRESSION IN HEXAGONAL GRID

In this research work, SVD based image compression on hexagonal framework is done. As there is a lack of standard hardware device to capture hexagonal lattice images, resampling of original square lattice images to hexagonal lattice is done. In order to obtain hexagonal framework, pseudo hexagonal pixel method or hexagonal deep learning framework discussed in section 2 can be used. The basic block diagram for the work methodology is shown in Table 1. And work flow is shown in Fig.8.

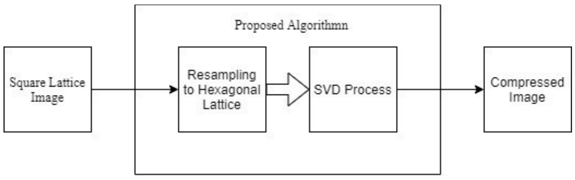

Figure 8. Methodology used.

VI. PROPOSED METHOD

Table 1. Pseudocode for hexagonal image compression

SL.NO.	HEXAGONAL IMAGE FRAMEWORK FOR IMAGE COMPRESSION USING SVD
1	**Inputs:** Read any square image of any format(*jpg,*png)
2	**Outputs**: Hexagonal compressed image
3	**Input parameters:** Pseudo hexagonal pixel neighborhood size like 6*6,7*7,9*8 Kernel size, Pixel intensity coefficients.
4	**Begin**
5	**//Resampling Phase**
6	Use pseudo hexagonal pixel framework /Alternate pixel suppressal method
7	**//Compression phase**
8	Select the required value for the rank 'k'.
9	Perform SVD to obtain U, ∑, V values.
10	Apply approximations on ∑ matrix values.
11	Regenerate the matrix values and remove the singularity
12	**//Extraction Phase**
13	Display the hexagonal compressed image
14	**End for**
15	Return hexagonal image
16	**End**

The implementation method of hexagonal structural representation and SVD compression using PYTHON is:

Step 1: Image acquisition- square image pixels is captured and stored as grey scale format.

Step 2: Image resampling-Convert square image into hexagonal image using pseudo hexagonal frame work or alternate pixel suppressal method.

Step3: Hexagonally re- sampled image is compressed is using SVD technique.

Step4: Display the hexagonally sampled compressed image

VII. PERFORMANCE PARAMETERS

The evaluation of SVD based image compression on square and hexagonal lattice is performed using different evaluation parameters like MSE, PSNR, Compression Ratio , compression size and run time efficiency.

(i) **Mean Square Error (MSE):** This metric tells how much the compressed image quality deteriorates with the original image. The MSE is the cumulative difference error between the compressed and the original image and is given by

$$\mathrm{MSE} = \frac{1}{N} \sum_{i} \sum_{j} (\mathrm{Xij} - V\mathrm{ij})^2 \tag{7}$$

Where, Xij is the original image and Vij is the compressed image. A lower MSE shows that the model is better fitting with good compression quality. MSE should be as small as for effective compression. MSE measures the square of the error.

(ii) **Peak signal to noise ratio (PSNR)**:it is a ratio between the peak signal value and the noise in the image.

$$PSNR = 10log10(\frac{nmax^2}{MSE}) \tag{8}$$

Where n $_{max}$ indicates the maximum pixel value in the image. If the PSNR value tends infinity MSE approaches zero; which indicates a higher PSNR value shows a higher image quality.

(iii) **Compression Ratio**: The compression ratio is defined as the ratio of size of the original image to the size of the compressed image.

$$Compression\ ratio = \frac{size\ of\ the\ original\ image}{size\ of\ the\ compressed\ image}$$

(iv) **Size of the compressed image**: The ultimate aim of image compression is to reduce the size of the graphic image file without degrading the quality of the image. The compression in size allows more images to be stored in agiven amount of memory space. It also reduces the time required for images to be sent over the Internet for faster data transfer of information

VIII. .RESULTS AND DISCUSSIONS

We use the computer system with following specifications as shown in Table 2.

Table 2: System specification

Operating system	**Windows 10**
Processor	Intel core i5
Software	Python 3.8, Matlab 2020a
Ram	8 GB

This paper presents the results of SVD based hexagonal image compression in PYTHON. Fig.6 and Fig. 8 gives the simulated result of square and hexagonal lattice image for different values of k. The result obtained for different performance metrics is shown in Table 3. The result proves that SVD based image compression on hexagonal lattice images is superior to square lattice image. The performance is proved using both qualitatively and quantitatively performance metrics. Visual quality of the compressed image gives the qualitative measure. MSE, PSNR, compression ratio and the compressed image size are used for quantitative measure. Figure shows original and compressed images in square and hexagonal lattices for different sample images. From the error plot (Fig 7 and Fig 9), it is clear that, for hexagonal lattice has less error compared to square lattice in SVD compression.

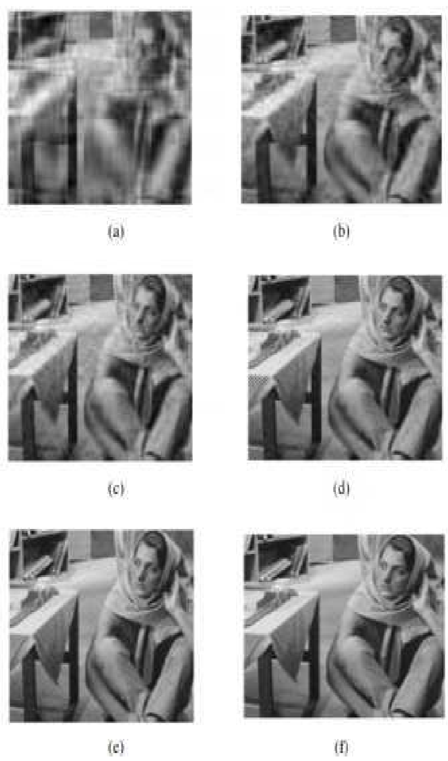

Figure 9. Square lattice compressed image for different values of k a) k=10 b) k=20 c) k=25 d) k=50 e) k=75 f) k=100.

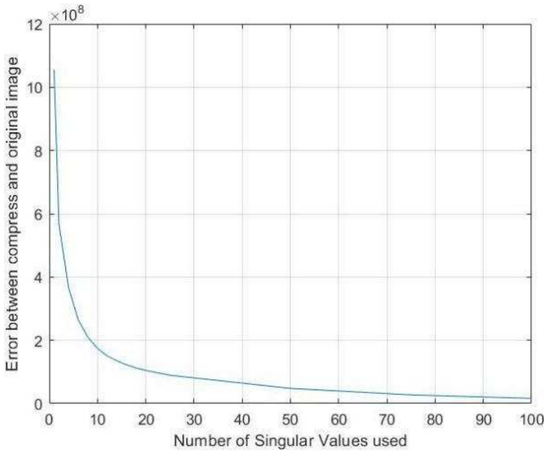

Figure 10. Error plot between compressed and original image in square lattice.

Figure 11. Hexagonal framework image compressed for different values of k a) k=10 b) k=20 c) k=25 d) k=50 e) k=75 d) k=100.

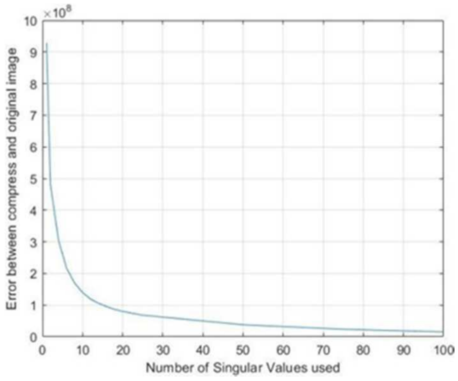

Figure 12. Error plot between hexagonal compressed image and hexagonal image.

Table 3. Performance analysis hexagonal v/s square image SVD compression

Sample images	sampling	MSE	PSNR	Compression ratio (CR)
Barbara.jpg (50.6KB)	square	25.05	34.14	1.01
	Hexagonal	23.14	36.32	1.13
Mandrill.jpg (73.8KB)	square	24.17	37.80	1.13
	Hexagonal	21.78	39.91	1.27

Table 3.presents the comparitive analysis of proposed hexagonal image compression method with the SVD technique. Here the best performing two sample images were discussed and is very clear that the proposed hexagonal image compression has overall best performance result.It achieved the high performance with the values 36.32dB,39.91dB for PSNR,1.13,1.27 for Compression ratio and with compression size 44.5 KB and 57.7KB.

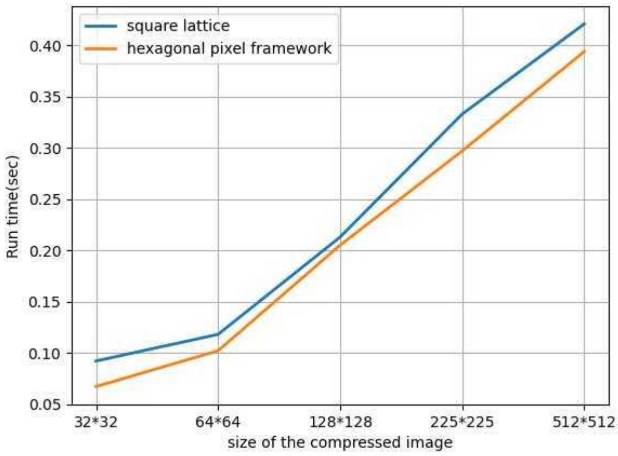

Figure 13. Run time('sec') requirement for various sized square and hexagonal pixel framework image for SVD compression

From the run time graph (Fig.13), it can be seen that run time speed required for hexagonal pixel framework is less compared to the square pixel's framework. Since the hexagonal framework require 13.4% pixels less compared to the square lattice.

IX. CONCLUSION

This work has implemented the technique of linear algebra "singular value decomposition (SVD)" in hexagonal lattice and is compared with square lattice image. We have achieved the compression level depending upon the application by appropriate values of k. In the SVD image compression technique the appropriate selection of K values plays an important role, and the output images shown in Fig. 9and10. Performance comparison using different metrics, error plot and run time requirements shows clearly the SVD based image compression gives better results for hexagonal lattice compression. SVD technique is robust, powerful, fast and have less computational complexity for hexagonal lattice images.it can perform well in a constrained environment also. In nutshell, hexagonal domain SVD image compression outperforms square domain image compression.

REFERENCES

1. Wüthrich, Charles A., and Peter Stucki. "An algorithmic comparison between square-and hexagonal-based grids." *CVGIP: Graphical Models and Image Processing* 53.4 : 324-339,1991.
2. Peterson, D. P., and D. Middleton. "Sampling and reconstruction of wavenumber-limited functions in N-

dimensional Euclidean spaces." *Information and control* 5.4 : 279-323,1962.

3. Golay, Marcel JE. "Hexagonal parallel pattern transformations." *IEEE Transactions on computers* 100.8: 733-740,1969.
4. Mersereau, Russell M. "The processing of hexagonally sampled two-dimensional signals." *Proceedings of the IEEE* 67.6 : 930-949,1979
5. Wolberg, George. *Digital image warping*. Vol. 10662. Los Alamitos, CA: IEEE computer society press, 1990.
6. Horn, B. "Robot Vision MIT Press." *Cambridge, Mass* 1986.
7. Staunton, Richard C. "The design of hexagonal sampling structures for image digitization and their use with local operators." *Image and vision computing* 7.3 : 162-166,1989.
8. Middleton, Lee, and Jayanthi Sivaswamy. "Edge detection in a hexagonal-image processing framework." *Image and Vision computing* 19.14 : 1071-1081,2001.
9. Middleton, Lee, and Jayanthi Sivaswamy. "Framework for practical hexagonal-image processing." *Journal of Electronic Imaging* 11.1 : 104-114,2002
10. He, Xiangjian. *2D-object recognition with Spiral Architecture*. Diss. University of Technology, Sydney, 1999.
11. Wu, Qiang, Xiangjian He, and Tom Hintz. "Virtual spiral architecture." *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA'04*. 2004.
12. Moonen, Marc, Paul Van Dooren, and Joos Vandewalle. "A singular value decomposition updating algorithm for subspace tracking." *SIAM Journal on Matrix Analysis and Applications* 13.4 : 1015-1038,1992.
13. Taro Konda and Yoshimasa Nakamura, “A new algorithm for singular value decom- position and its parallelization”, *Parallel Computing*, Vol. 35, No. 6, pp. 331–344,2009.
14. Ashin, R., Akira Morimoto, and Remi Vaillancourt. "Image compression with multiresolution singular value decomposition and other methods." *Mathematical and Computer Modelling* 41.6-7 : 773-790,2005.
15. Schlosser, Tobias, Michael Friedrich, and Danny Kowerko. "Hexagonal Image Processing in the Context of Machine Learning: Conception of a Biologically Inspired Hexagonal Deep Learning Framework." *2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2019.
16. https://www.sciencefriday.com/educational-resources/why-do-bees-build-hexagonal-honeycomb-cells.
17. Vanhoutte, Kurt JA, Kristel FL Michielsen, and Doekele G. Stavenga. "Analyzing the reflections from single ommatidia in the butterfly compound eye with Voronoi diagrams." *Journal of neuroscience methods* 131.1-2 : 195-203,2003.
18. A. Curcio, K. R. Sloan, R. E. Kalina, and A. E. Hendrickson, “Human photoreceptor topography,” The Journal of Comparative Neurology, vol. 292, no. 4, pp. 497–523, 1990.
19. He, Xiangjian, and Wenjing Jia. "Hexagonal structure for intelligent vision." *2005 International Conference on Information and Communication Technologies*. IEEE, 2005.
20. Sheridan, Phillip, T. Hintz, and W. Moore. *Spiral Architecture for machine vision*. Diss. 1996.
21. Wu, Qiang, Xiangjian He, and Tom Hintz. "Virtual spiral architecture." *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA'04*. 2004.
22. Li, Xiangguo, Bryan Gardiner, and Sonya A. Coleman. "Square to hexagonal lattice conversion in the frequency domain." *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017.
23. Wu, Q., et al., “Bi-lateral filtering based edge detection on hexagonal architecture,” Proc IEEE Int Conf Acoustics, Speech, and Signal Processing, pp. 713- 716, 2005.