The FFT in a hexagonal-image processing framework

Article · January 2001

CITATIONS
22

READS
411

2 authors:

Lee Middleton
University of Southampton
65 PUBLICATIONS **713** CITATIONS
SEE PROFILE

Jayanthi Sivaswamy
International Institute of Information Technology, Hyderabad
120 PUBLICATIONS **1,377** CITATIONS
SEE PROFILE

The FFT in a Hexagonal-image Processing Framework

Lee Middleton
Dept. of Electrical and Electronic Engineering
University of Auckland
Private Bag 92019, Auckland
New Zealand
l.middleton@auckland.ac.nz

Jayanthi Sivaswamy
Indian Institute of Information Technology
Gachibowli
Hyderabad-500019
India
jsivaswamy@iiit.net

Abstract

Computer vision on lattices based on hexagonal pixels is considered to be advantageous due to the consistent connectivity within the lattice. This leads to more sensible definitions for neighbourhoods and positive benefits when processing curved structures. These advantages should also exist in the frequency domain as well. To this end, this paper derives a Fast Fourier Transform for the hexagonal lattice. The resulting algorithm is based upon the Cooley-Tukey approach and is a radix-7 decimation in space algorithm. Furthermore, the overall computational complexity of the algorithm is $N \log_7 N$. The use of a single index to address every point in the image also adds to the computational efficiency of the proposed algorithm. The algorithm is applied to linear filtering examples and proves to be effective. Furthermore, the utility for curved representation is effectively demonstrated.

1. Introduction

Many applications of image processing require the evaluation of discrete Fourier transforms upon the image. For example, convolutions are more computationally efficient when computed in the frequency domain. Further examples include transform coding (image compression) and filter design. Image processing of hexagonally sampled images has been shown to have utility for the representation of curved structures [10] due to the uniform connectivity within the sampling lattice. Furthermore, if the symmetry inherent in the hexagonal lattice is exploited it is possible to generate an efficient indexing scheme [6, 1]. Finally, hexagonal sampling has been shown to be the optimal sampling regime for circularly band-limited signals [8]. The central aim of this paper is to extend the work previously performed on the hexagonal-image processing (HIP) [7] framework and to implement the fast Fourier transform upon it. The approach to the fast Fourier transform employed is the vector-radix Cooley-Tukey algorithm [5].

This paper is divided into several sections. The first section describes the indexing scheme employed in the HIP framework. The following section will examine the particular ramifications of transforming the hexagonal lattice into the frequency domain. The third section will provide details of the HFFT. This section will be followed with a practical case study and then a discussion of the presented scheme.

2. Indexing in the HIP framework

The hexagonal lattice exhibits three degrees of symmetry which should be exploited in indexing schemes. A naive approach [2] would be to use two of these axes analogously to Cartesian coordinate systems. This however, ignores the third axis of symmetry which can yield many operations computationally difficult and inefficient. Another approach [3] is to use all three axes of symmetry. This exploits symmetry but, however, may result in cumbersome data structures which are inefficient. A third approach is to use a positional number system which encodes spatial location using a single digit. One such scheme is the balanced ternary [4]. The use of the balanced ternary indexing scheme is central to the HIP framework.

The indexing scheme can be constructed intuitively by using an analogy based upon tiling the image. Consider a single hexagon to be a tile at layer 0. This can then be surrounded moving anti-clockwise by a further 6 hexagons. This new structure forms a tile at layer 1, with the new tile being the super-tile of layer 0. Similarly, this process can be extended to any number of layers. An example of layer 2 is illustrated in figure 1. The number of hexagons that are contained in a given layer λ super-tile are 7^λ. Each hexagon can then be numbered uniquely as a sequence of numbers where every digit gives its position in a given tile. The highlighted hexagon in figure 1 is in the fourth tile in layer 2 and the

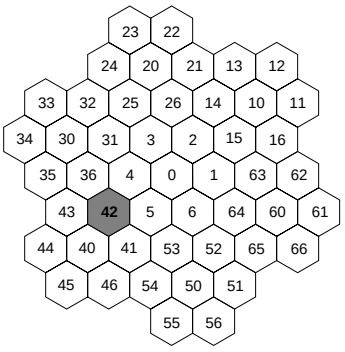


Figure 1. The spatial domain hexagonal indexing scheme.

second tile at layer 1. Thus, all the hexagons in a layer λ super-tile can be addressed uniquely by a λ-digit base 7 number. As previously stated this number encodes spatial location and as such the number can be viewed analogously to a vector. This vector property can be used to define various arithmetic operations such as hexagonal addition and hexagonal multiplication [6].

The index can be more formally defined as follows :

$$\begin{aligned} \boldsymbol{g} &= \{g_{\lambda-1}\cdots g_1 g_0, \forall g_i \in [0,6]\} \\ &= (g_{\lambda-1} \otimes 10^{\lambda-1}) \oplus \cdots \oplus (g_1 \otimes 10) \oplus (g_0 \otimes 1) \end{aligned}$$

In the above $\otimes$ and $\oplus$ are hexagonal multiplication and hexagonal addition of the indices. Due to the positional nature of the index each digit, g_i, can be considered to be a rotation about a point 10^i as can be seen by observing figure 1 and the above equation. In fact, the rotation caused by each digit is $60°$. Finally, the set of all points in a given λ-level structure can be defined as $\mathbb{G}^\lambda$.

3. HIP in the Frequency Domain

An image is a two dimensional signal. Generally a two dimensional signal can be written as $f = f(x,y), -\infty < x,y < \infty$. These points (x,y) can be considered to be sampled upon a spatial sampling lattice $\mathbf{V}$. The columns of $\mathbf{V}$ form a basis of $\mathbb{R}^2$. f can be considered to be periodic in two dimensions if the following relationship holds :

$$\begin{aligned} f(x,y) &= f(x+N_{11}, y+N_{21}) \\ &= f(x+N_{12}, y+N_{22}) \end{aligned} \tag{1}$$

The ordered pairs $(N_{11}, N_{21})^T$ and $(N_{12}, N_{22})^T$ can be interpreted as vectors $\mathbf{N_1}$ and $\mathbf{N_2}$ which represent the displacements from any sample to the corresponding sample in other periods. Together, $\mathbf{N_1}$ and $\mathbf{N_2}$ are known as the periodicity matrix $\mathbf{N} = [\mathbf{N_1}\mathbf{N_2}]$. Obviously, for any periodic two dimensional signal the choice of $\mathbf{N}$ is not unique. Examining, figure 1 shows the image to be periodic as the central set of seven points is repeated. This should come as no surprise as it is implicit in the construction process of the image. In fact, for a layer λ structure the image is periodic using a layer $\lambda - 1$ structure as a tile. As well, it is also periodic using every layer lower than this as a tile.

The relationship between the lattices in the spatial and frequency domain can be said to reciprocal [9]. Thus, the spatial and frequency axes are orthogonal to one another. This can be formally defined for a frequency lattice $\mathbf{U}$ as [5] :

$$\mathbf{U} = \left((\mathbf{V}\mathbf{N})^{-1}\right)^T \tag{2}$$

For example the 2-layer HIP structure in figure 1 has :

$$\mathbf{V} = \begin{bmatrix} 1 & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} \end{bmatrix} \qquad \mathbf{N} = \begin{bmatrix} 3 & -2 \\ 2 & 1 \end{bmatrix}$$

$$\mathbf{U} = \frac{1}{7\sqrt{3}} \begin{bmatrix} \sqrt{3} & -2\sqrt{3} \\ 5 & 4 \end{bmatrix}$$

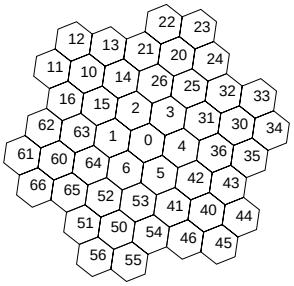


Figure 2. The frequency domain hexagonal indexing scheme.

The second layer frequency domain HIP structure is illustrated in figure 2. It is a rotated, reflected, and scaled version of the spatial domain structure as can be evidenced by both the figure and equation (2). To choose $\mathbf{V}$ any two vectors could be chosen so long as they are a multiple of $60°$ apart. The periodicity matrix $\mathbf{N}$ is the required transformation to translate from the first layer HIP structure to the second. Furthermore, successive multiplication of $\mathbf{N}$ will give the periodicity matrices at subsequent layers. This is a byproduct of the way in which the indexing scheme was constructed. Thus for a λ layer structure equation (2) can be rewritten as :

$$\mathbf{U}_\lambda = \left((\mathbf{V}\mathbf{N}^\lambda)^{-1}\right)^T \tag{3}$$

4. HIP Fast Fourier Transform

There are many different fast Fourier transform algorithms. However, the one chosen for this work is the Cooley-Tukey (decimation in space) algorithm. Before describing the details of the algorithm with respect to the HIP framework some definitions will first be provided. A two dimensional sampled signal $f(\mathbf{n})$ can be periodically sampled with a region of support $I_{\mathbf{N}}$. The frequency domain analog with corresponding region of support $J_{\mathbf{N}}$ is $F(\mathbf{k})$. The relationship between these (the discrete Fourier transform or DFT) can be written [5]:

$$F(\mathbf{k}) = \sum_{n \in I_{\mathbf{N}}} f(\mathbf{n}) e^{-2\pi \jmath \mathbf{k}^T \mathbf{N}^{-1} \mathbf{n}} \quad (4)$$

$$f(\mathbf{k}) = \frac{1}{|\det \mathbf{N}|} \sum_{n \in J_{\mathbf{N}}} F(\mathbf{n}) e^{2\pi \jmath \mathbf{k}^T \mathbf{N}^{-1} \mathbf{n}} \quad (5)$$

In general, Cooley-Tukey type algorithms are defined whenever the periodicity matrix $\mathbf{N}$ can be factored into a non-trivial product of integer matrices. This can be expressed as :

$$\mathbf{N} = \mathbf{P}\mathbf{Q} \quad (6)$$

Furthermore, two integer vectors are congruent to one another with respect to the matrix modulus $\mathbf{N}$ if :

$$\mathbf{a} = \mathbf{b} + \mathbf{N}\mathbf{r} \quad (7)$$

Here $\mathbf{r}$ is an arbitrary integer vector. Given $\mathbf{N}$ satisfies equation (6) then using equation (7) the following relationships hold :

$$\begin{aligned} \mathbf{n} &= \mathbf{p} + \mathbf{P}\mathbf{q} \\ \mathbf{k} &= \mathbf{r} + \mathbf{Q}^T \mathbf{s} \end{aligned} \quad (8)$$

where $\mathbf{p} \in I_{\mathbf{P}}$, $\mathbf{q} \in I_{\mathbf{Q}}$, $\mathbf{r} \in J_{\mathbf{Q}}$, and $\mathbf{s} \in J_{\mathbf{P}}$. Any pair of vectors, one from $I_{\mathbf{P}}$ and one from $I_{\mathbf{Q}}$, form a unique vector $\mathbf{n}$ in the set $I_{\mathbf{N}}$. Also, any pair of vectors, one from $J_{\mathbf{P}}$ and one from $J_{\mathbf{Q}}$, form a unique vector $\mathbf{k}$ in the set $J_{\mathbf{N}}$. Using equation (8) the DFT sum in (4) can be written :

$$F(\mathbf{Q}^T\mathbf{r} + \mathbf{s}) = \sum_{\substack{\mathbf{p} \in I_{\mathbf{P}} \\ \mathbf{q} \in I_{\mathbf{Q}}}} \Big(f(\mathbf{P}\mathbf{q} + \mathbf{p}) \; e^{-2\pi \jmath (\mathbf{r}^T + \mathbf{s}^T \mathbf{Q}) \mathbf{N}^{-1} (\mathbf{P}\mathbf{q} + \mathbf{p})} \Big) \quad (9)$$

Expansion of the exponent can decompose the sum into two parts which represent the first level of decomposition for the Cooley-Tukey algorithm :

$$\begin{aligned} C(\mathbf{p}, \mathbf{l}) &= \sum_{\mathbf{q} \in I_{\mathbf{Q}}} f(\mathbf{P}\mathbf{q} + \mathbf{p}) e^{-2\pi \jmath (\mathbf{r}^T + \mathbf{s}^T \mathbf{Q}) \mathbf{Q}^{-1} \mathbf{q}} \\ F(\mathbf{Q}^T\mathbf{r} + \mathbf{s}) &= \sum_{\mathbf{p} \in I_{\mathbf{P}}} \Big(C(\mathbf{p}, \mathbf{l}) e^{-2\pi \jmath \mathbf{l}^T \mathbf{N}^{-1} \mathbf{p}} \\ &\qquad e^{-2\pi \jmath \mathbf{s}^T \mathbf{P}^{-1} \mathbf{p}} \Big) \end{aligned} \quad (10)$$

In the above equations the indexing terms $\mathbf{n}$ and $\mathbf{k}$ are two dimensional vectors. However, as has been mentioned in section 2 the HIP framework uses a single index. The formulation of the DFT should also exploit this. To do this an intermediate representation was chosen. The 3-coordinate system of Her [3] was chosen for this purpose though other systems could also be used. A function c is defined that converts from the single index to the 3-coordinate system. From this, a further pair of functions are defined :

$$h(\boldsymbol{g}) = \frac{1}{3} \sum_{j=1}^{\lambda} \mathbf{N}_{j-1} \begin{bmatrix} 1 & 1 & -2 \\ -1 & 2 & -1 \end{bmatrix} c(g_j)$$

$$H(\boldsymbol{g}) = \frac{1}{3} \sum_{j=1}^{\lambda} (\mathbf{N}_{j-1})^T \begin{bmatrix} -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} c(g_j)$$

Here $\boldsymbol{g}$ is a λ-level index, g_j corresponds to the j-th digit of the index, and $\mathbf{N}_{j-1}$ is equivalent to $\mathbf{N}^{j-1}$. Notice that if $\boldsymbol{p} \in \mathbb{G}^{\alpha}$ and $\boldsymbol{q} \in \mathbb{G}^{\beta}$, where $\alpha + \beta = \lambda$, then $\mathbf{N}_{\alpha}\mathbf{q} + \mathbf{p} = h(\boldsymbol{qp})$. Similarly, ${\mathbf{N}_{\beta}}^T \mathbf{m} + \mathbf{l} = H(\boldsymbol{ml})$. Using these relationships, equations (4) and (5) can be rewritten in terms of HIP indices giving the HIP DFT (HDFT) :

$$F(\boldsymbol{k}) = \sum_{\boldsymbol{n} \in \mathbb{G}^{\lambda}} f(\boldsymbol{n}) e^{-2\pi \jmath H(\boldsymbol{k})^T {\mathbf{N}_{\lambda}}^{-1} h(\boldsymbol{n})} \quad (11)$$

$$f(\boldsymbol{n}) = \sum_{\boldsymbol{k} \in \mathbb{G}^{\lambda}} F(\boldsymbol{k}) e^{2\pi \jmath H(\boldsymbol{k})^T {\mathbf{N}_{\lambda}}^{-1} h(\boldsymbol{n})} \quad (12)$$

In the above, $\boldsymbol{n}$ and $\boldsymbol{k}$ are indices in the HIP indexing scheme. The objective is to rewrite equation (10) to give the Cooley-Tukey decomposition for the HIP framework. To do this first note that the periodicity matrix can be logically decomposed as $\mathbf{N}^{\lambda} = \mathbf{N}^{\lambda-1}\mathbf{N}$. This reduces the order of the DFT by 7. Thus :

$$\begin{aligned} C(\boldsymbol{p}, \boldsymbol{s}) &= \sum_{\boldsymbol{q} \in \mathbb{G}^1} f(\boldsymbol{qp}) e^{-2\pi \jmath H(\boldsymbol{s})^T \mathbf{N}^{-1} h(\boldsymbol{q})} \\ F(\boldsymbol{rs}) &= \sum_{\boldsymbol{p} \in \mathbb{G}^{\lambda-1}} C(\boldsymbol{p}, \boldsymbol{s}) e^{-2\pi \jmath H(\boldsymbol{s})^T \mathbf{N}^{-1} h(\boldsymbol{p})} \\ &\qquad e^{-2\pi \jmath H(\boldsymbol{r})^T \mathbf{P}^{-1} h(\boldsymbol{p})} \end{aligned} \quad (13)$$

For the two layer HIP structure illustrated in figure 1 the above factorisation is the complete hexagonal fast Fourier transform (HFFT). For an arbitrary λ-level HIP structure the full HFFT can be written :

$$C_1(\boldsymbol{t}, \boldsymbol{r}) = \sum_{\boldsymbol{s} \in \mathbb{G}^1} f(\boldsymbol{st}) e^{-2\pi \jmath H(\boldsymbol{r})^T \mathbf{N}^{-1} h(\boldsymbol{s})}$$

where $\boldsymbol{t} \in \mathbb{G}^{\lambda-1}, \boldsymbol{n} \in \mathbb{G}^1$

$$C_{\gamma+1}(\boldsymbol{t}, \boldsymbol{rn}) = \sum_{\boldsymbol{s} \in \mathbb{G}^1} e^{-2\pi \jmath H(\boldsymbol{r})^T \mathbf{N}^{-1} h(\boldsymbol{s})} C_\gamma(\boldsymbol{st}, \boldsymbol{n}) \; e^{-2\pi \jmath H(\boldsymbol{st})^T \mathbf{N}_{\gamma-\mathbf{1}}{}^{-1} h(\boldsymbol{n}_{\lambda-\gamma})}$$

where $\boldsymbol{t} \in \mathbb{G}^{\lambda-\gamma-1}, \boldsymbol{r} \in \mathbb{G}^1, \boldsymbol{n} \in \mathbb{G}^\gamma$. The last term $e^{-2\pi \jmath H(\boldsymbol{st})^T \mathbf{N}_{\gamma-\mathbf{1}}{}^{-1} h(\boldsymbol{n}_{\lambda-\gamma})}$ is the *twiddle* factor for the HFFT. It serves to reorder the rows and scales the inputs. An analogous formula can be derived for the inverse hexagonal fast Fourier transform viz :

$$K_1(\boldsymbol{t}, \boldsymbol{r}) = \frac{1}{7} \sum_{\boldsymbol{s} \in \mathbb{G}^1} f(\boldsymbol{st}) e^{2\pi \jmath H(\boldsymbol{r})^T \mathbf{N}^{-1} h(\boldsymbol{s})}$$

where $\boldsymbol{t} \in \mathbb{G}^{\lambda-1}, \boldsymbol{n} \in \mathbb{G}^1$

$$K_{\gamma+1}(\boldsymbol{t}, \boldsymbol{rn}) = \frac{1}{7} \sum_{\boldsymbol{s} \in \mathbb{G}^1} e^{2\pi \jmath H(\boldsymbol{r})^T \mathbf{N}^{-1} h(\boldsymbol{s})} C_\gamma(\boldsymbol{st}, \boldsymbol{n}) \; e^{2\pi \jmath H(\boldsymbol{st})^T \mathbf{N}_{\gamma-\mathbf{1}}{}^{-1} h(\boldsymbol{n}_{\lambda-\gamma})}$$

where $\boldsymbol{t} \in \mathbb{G}^{\lambda-\gamma-1}, \boldsymbol{r} \in \mathbb{G}^1, \boldsymbol{n} \in \mathbb{G}^\gamma$. Much of the computation can be performed using lookup tables as is usual for a FFT algorithm. Furthermore, the two twiddle factors exhibit a large amount of redundancy and this reduces the amount of memory required to compute the HFFT and IHFFT.

For a λ-layer HIP image the HFFT can be performed in $\log_7 \lambda$ stages. Computation of these stages requires $7^{\lambda+1}$ complex multiplications. At each stage there is a HFFT requiring $(\lambda - 1)$ stages and 7^λ multiplications. Thus the overall number of complex multiplications are $\lambda 7^{\lambda+1} + (\lambda - 1)7^\lambda$. Due to the redundancy many of the multiplications can be removed resulting in $\lambda 7^\lambda$ actual calculations or complexity $N \log_7 N$.

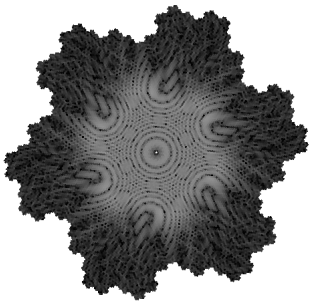
5. Case study - Linear filtering

To illustrate the HFFT a simple case study using filtering is now performed. Linear filtering in the frequency domain is advantageous as it leads to intuitive filter design methodologies. Generally, the frequency domain filtering operation can be written as :

$$G(\boldsymbol{g}) = H(\boldsymbol{g}) F(\boldsymbol{g}) \tag{14}$$

Where $\boldsymbol{g}$ is a coordinate in the HIP indexing scheme, F is the original image, H is the filter transfer function, and G is the filtered image. A variety of different operations can be carried out by varying H. This case study will concentrate on low pass and high pass filters, all of which have a circular region of support. For the examples the original square image is sampled onto a 5-layer (16807 hexagonal pixels or hexels) HIP image. The resulting hexagonal image and its frequency domain analog is illustrated in figure 3.

(a)



(b)

Figure 3. The hexagonally sampled test image : a) spatial domain b) frequency domain

Four distinct filters were implemented two were low pass and two were high pass. They all had the same cutoff frequencies. The transfer functions for the two low pass filters are :

$$H_{IL}(\boldsymbol{g}) = \begin{cases} 1, & |\boldsymbol{g}| \leq R \\ 0, & |\boldsymbol{g}| > R \end{cases} \quad (15)$$

$$H_{BL}(\boldsymbol{g}) = \frac{1}{1 + C(\frac{|\boldsymbol{g}|}{R})^{2n}} \quad (16)$$

Where $|\boldsymbol{g}|$ is the radius from the image centre of a HIP index, R is the cutoff frequency, C is the magnitude at the point where $|\boldsymbol{g}| = R$, and n is the filter order. Equation (15) gives an ideal filter with a sharp cutoff and equation (15) gives a Butterworth filter with a smooth cutoff. Both have a circular region of support. For the results shown here $R = 22$ hexels, $C = (\sqrt{2} - 1)$, and $n = 2$. The cutoff frequency radius, R, was chosen to make sure that 80% of the power spectra would be contained in the ideal low pass filter. C was chosen to give 3 dB attenuation at the point where $|\boldsymbol{g}| = R$. The results after filtering are illustrated in figure 4 (a-f).

The ideal filters exhibit significantly more ringing than the Butterworth filters. This is due to the Gibbs phenomena. In the case of high pass filtering of the original image there is significant aliasing at the edges of the image. An equivalent square image (of size 128×128 pixels) would not display images with such circular rings but would exhibit blockiness due to the low resolution. Furthermore, the spectrum would exhibit more energy in the horizontal and vertical directions due to the quantisation along these axes. The consequence of this is that the filtering process illustrated here would perform relatively poorly.

6. Conclusion

This paper presented a radix-7 decimation in space algorithm for the HIP framework. The resulting algorithm was of computational complexity $N \log_7 N$ and exploited the single index addressing scheme inherent to the HIP framework. The linear filtering example showed that HIP is well suited to filtering with circular regions of support. This meant that the discriminating power of the filter was better due to its better fit.

References

[1] L. Gibson and D. Lucas. Vectorization of raster images using hierarchical methods. *Computer Graphics and Image Processing*, 20:82–89, 1982.

[2] N. P. Hartman and S. L. Tanimoto. A Hexagonal Pyramid data structure for Image Processing. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-14(2):247–256, Mar/Apr 1984.

[3] I. Her. Geometric Transforms on the Hexagonal Grid. *IEEE Transactions on Image Processing*, 4(9):1213–1222, September 1995.

[4] D. E. Knuth. *The Art of Computer Programming : Seminumerical Algorithms*, volume 2. Addison Wesley, 1969.

[5] R. Mersereau and T. Speake. A Unified Treatment of Cooley-Tukey Algorithms for the Evaluation of the Multidimensional DFT. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-29(5):1011–1018, 1981.

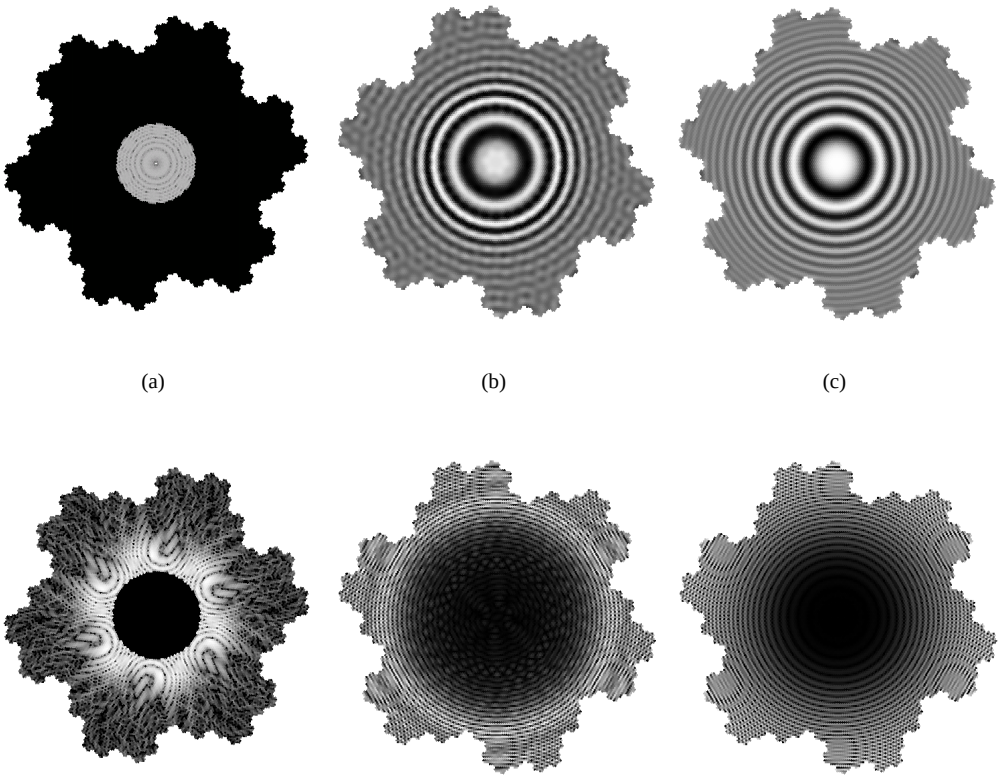
[6] L. Middleton and J. Sivaswamy. A framework for practical hexagonal-image processing. *Journal of Electronic Imaging (to appear)*, 2001.

[7] L. Middleton and J. Sivaswamy. Edge Detection in a Hexagonal-image Processing Framework. *Image and Vision Computing*, 19/14:1071–1081, Nov. 2001.

[8] D. P. Petersen and D. Middleton. Sampling and Reconstruction of Wave-Number-Limited Functions in N-Dimensional Euclidean Spaces. *Information and Control*, 5:279–323, 1962.

[9] A. Rosenfeld and A. Kak. *Digital Picture Processing*. Academic Press, 2nd edition, 1982.

[10] J. Serra. Introduction to Mathematical Morphology. *Computer Vision, Graphics, and Image Processing*, 35:283–305, 1986.



(a) (b) (c)

(d) (e) (f)

Figure 4. The filtered results (a) frequency domain ideal low pass filter (b) spatial domain ideal low pass filter (c) spatial domain Butterworth filter (d) frequency domain ideal high pass filter (e) spatial domain ideal high pass filter (f) spatial domain Butterworth filter.