

Frequency Domain Pseudo-color to Enhance Ultrasound Images

Jakia Afruz (Corresponding author)

Electrical and Computer Engineering

Southern Illinois University Edwardsville, IL-62026, USA

Tel: 618-709-2079 E-mail: jafruz@siue.edu

Va'Juanna Wilson

Electrical and Computer Engineering

Southern Illinois University Edwardsville, IL-62025, USA

Tel: 314-397-6982 E-mail: vawilso@siue.edu

Professor Dr. Scott E Umbaugh

ECE 439

Abstract

In digital image processing, image enhancement is employed to give a better look to an image. Color is one of the best ways to visually enhance an image. Pseudo-color refers to coloring an image by mapping gray scale values to a three dimensional color space. In this paper we used a pseudo-color technique in frequency domain to enhance ultrasound images. We used three different types of transforms to do this. These are the Fourier transform, Discrete Cosine transform and Walsh- Hadamard transform. After obtaining these pseudo-color images, we applied a high frequency emphasis filter or histogram stretch as a post process. In this paper we used a subjective study to compare images. First we compared pseudo-color images to their original monochrome images. Secondly, we compared all the three different types of transforms. Lastly, we compared the post processing techniques.

Keywords: Image enhancement, Ultrasound images, Pseudo-Color, Transforms, Filters

1. Introduction

This paper was based on the idea that the human visual system is more responsive to color than binary or monochrome images. We wanted to experiment with the importance of pseudo-color. To create our images we use the frequency domain pseudo color method that was discussed in Dr. Scott Umbaugh's book, Computer Imaging: Digital Image Analysis and Processing. We already had access to a program with the ability to do it spatially.

The purpose of this paper is to be able to understand image enhancement with color. We used ultrasound images as gray scale images and used pseudo color technique to enhance them.

1.1 Image Enhancement

Image Enhancement techniques are employed to improve the quality of images for human viewing. Color is important for image enhancement because the human visual system has the ability perceive thousands of colors in a small spatial area compared to only about 100 gray levels. (Umbaugh, 2005) In addition, color contrast can be more dramatic than gray level contrast and various colors have different degrees of psychological impact on the observer. Taking these advantages of human visual perception to enhance an image a technique is applied in a technique called pseudo color. (Owens, 1997)

1.2 Pseudo-color

Pseudo-color (false color) image processing consists of assigning colors to gray values based on a specified criterion. The term "pseudo-color" emphasizes that the colors are assigned artificially opposing to the true (real) colors. Pseudo-color is used to map the gray level values of a monochrome image to red, green and blue values. (Tcheslavski, 2008 and Umbaugh, 2005)

The pseudo-color techniques can be applied in both the spatial and frequency domains. The principal use of pseudo-color is for human visualization and interpretation of gray scale details on an image or their sequence and

it is often applied to images where relative values are important but not the specific representation like the ultrasound images we used.

In this paper, we performed pseudo-color in frequency domain. This is typically accomplished by performing a Fourier transform on the image and then applying a highpass, bandpass and lowpass filter to the transformed image. These filtered outputs are then applied to an inverse Fourier transform and the individual outputs are used as RGB components of the color image. But in this project we applied also the Discrete Cosine, Walsh-Hadamard transform and their respective inverse transform. Finally we compared the output pseudo-color images from these different processes.

2. Materials

Two types of software were used: Microsoft Visual Studio and CVIPtools. CVIPtools is a user interface that allows for quick image processing and analysis. It was developed at Southern Illinois University Edwardsville under the direction of Dr. Scott E Umbaugh, PhD. CVIPtools was used during the experimental stage to test different possible outcomes of changing certain variables. We experimented with band assignments and low and high frequency cutoff values. Based on those outcomes a pseudo color code was developed using C language in Visual Studio. Using this C code allowed us to automate the process and experiment with different transforms discussed in the next section. CVIP tools is only equipped to perform frequency domain pseudo coloring using the Fourier Transform.

3. Methods

3.1 Transforms

Figure -1 depicts the method we used. First a transform is applied to an input image.

3.2 Fourier Transform

The Fourier transform is the most well-known and widely used transform. This transform decomposes an image into complex sinusoidal terms and these terms include a zero frequency term, also called the DC term which related to average value. highpass, lowpass and bandpass filters. (Umbaugh, 2005)

3.3 Discrete Cosine Transform

The cosine transform is similar to Fourier transform uses sinusoidal basis function but only difference is that the cosine transform basis function is not complex because they use only cosine function not the sine function.

3.4 Walsh-Hadamard Transform

The basis function of this transform is based on the square or rectangular waves with peak ± 1 . These types of basis functions are computationally very simple

3.5 Filters

After the transform is performed, the resulting spectrum is separated into 3 sections with the use of low pass, band pass and high pass filters. The high pass filter emphasizes the high frequency information mainly edges. The low pass filter will keep the low frequency information while attenuating any frequency beyond the cutoff. The bandpass filter retains a certain band of frequencies given high and low cut offs. All of the filters in our program are Butterworth filters. Using a non-ideal filter insured that there would be no rippling in the output image caused by the sharp edges of an ideal filter. Butterworth filters are rounded like most objects in nature are, most certainly babies. Figures 3 through 5 are examples of the intermediate filtered images before they were assigned to a band.

3.6 Inverse Transforms

At this point there are three separate transforms. The respective inverse transforms are applied to each spectrum to produce the desired lowpass, bandpass or highpass version of the image. Each image is assigned to either the red, green or blue band. For example, if we want the high frequency information to be green, we would assign the highpass image pixel values to the second band of our output image. Note that before the bands can be assigned, the lowpass, bandpass and highpass images must be log remapped back to byte images. This is essential to the display of the resulting image. Now a pseudo-color image has been created.

3.7 Post-Processing

The first type of post processing we explored was histogram equalization on the value. This was done by transforming the pseudo-color RBG image into an HSL (Hue, Saturation, Lightness) image, applying a histogram stretch on the third, lightness, band and finally applying the inverse transform to get a post processed

RGB image. This did not show any difference in images so it was only experimented with and not used. Ultimately, we used high frequency emphasis filters or a histogram stretch. Some of the images responded differently to each process. From that observation we decided to compare post processes, discussed in the next section.

4. Results and Discussion

2D and 3D ultrasound images were used. We really only wanted to use the images of fetuses after about 10 weeks in the womb, ideally at least 20 weeks. Below are some of the results. All other results are placed in the appendix. Figure 6 is a sample of one of the 2D ultrasounds. Figure 7 is 3D.

The subjective study was separated into three categories: pseudo-color image vs. monochrome image, histogram stretch vs. high frequency emphasis filters (as a post processing step) and comparisons of the transforms. The graphs below are from the study. We found first that 60% of the people tested preferred pseudo-color ultrasounds over the original monochrome (Figure 8). Then we compared transforms we used. At first FFT vs. Walsh and we got 50%-50% results so we could not say which one is better (Figure 9). Secondly, we tried with FFT vs. DCT and we got different result. DCT was chosen more than FFT and the result is DCT-67% and FFT 33% (Figure 10). And the last test, DCT vs. Walsh since FFT and Walsh were at same result. And here again we got DCT better than Walsh and the test result is DCT- 64% and Walsh- 36% (Figure 11)

The final comparison was done between using a histogram stretch or high frequency emphasis filter as a post processing step. To create these images we used the same image and performed both processes to them and compared them. The figure 12 shows the results. The histogram stretch images were chosen 65% of the time over the high frequency emphasis.

Some of the ultrasounds produced better results than others. It was a bit surprising, at first, to see that the images created using the Discrete Cosine Transform was chosen over those using the Fourier Transform, generally used in books. After more thought, we realized that the basis function of the DCT is a cosine wave, even function, and the FFT basis function is a sine wave, odd. So, it must be acceptable that DCT will provide better result than FFT cause the even function is easier to map.

5. Summary and Conclusion

For this paper we wanted to explore pseudo-color in the frequency domain and apply the process to ultrasound images. We got great resulting images. An interesting result from the subjective study is that the use the Fourier Transform may not result in the best images. The Discrete cosine transform was chosen over the FFT. Also, the Walsh-Hadamard transforms are least desirable for use of pseudo-color imaging. This could be attributed to the simplicity of the basis functions. Ultimately, we found that most of our subjects preferred the look of the pseudo-color ultrasounds over their respective monochrome images. This agrees with the theory stated before that the human visual system is more responsive to color contrast as opposed to monochrome contrast. This gives hope for future study to know that the process is worth more experimentation. In the future, we would like to explore the wavelet theory and be able to compare it to the other transforms.

Special Thanks:

Tawannia Wilson (BSN RN) Clinical Administrator
Peoples Health Center-Headquarters
5701 Delmar Blvd
St. Louis, MO 63112

References

- Gonzalez, R.C. Woods, R.E. (1993). *Digital Image Processing*. USA: Addison-Wesley Publishing Company, Inc.
- Horadam, K.J. (2007). *Hadamard Matrices and Their Applications*. Princeton University, New Jersey: Princeton University Press.
- Li, J., Li J., Wei, P. (2007). Pseudocolor Coding of Medical Images Based on Gradient. *International Conference on Bioinformatics and Biomedical Engineering Proceeding (ICBBE), 2007*. ISBN: 1-4244-1120-3. p932 – 935
- Owens, R. (1997). Image Enhancement. *Computer Vision IT412*. Retrived October 29, 1997.
- Tcheslavski, G.V. (2008). Color Image Processing: Pseudo color Processing. *Tcheslavski's Homepage*. Retrieved April 28, 2008.

Umbaugh, S. E. (2005). *Computer Imaging: Digital Image Analysis and Processing*. Boca Raton, Florida: CRC Press.

Umbaugh, S. E. (2010). *Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIPtools*. Boca Raton, Florida: CRC Press.

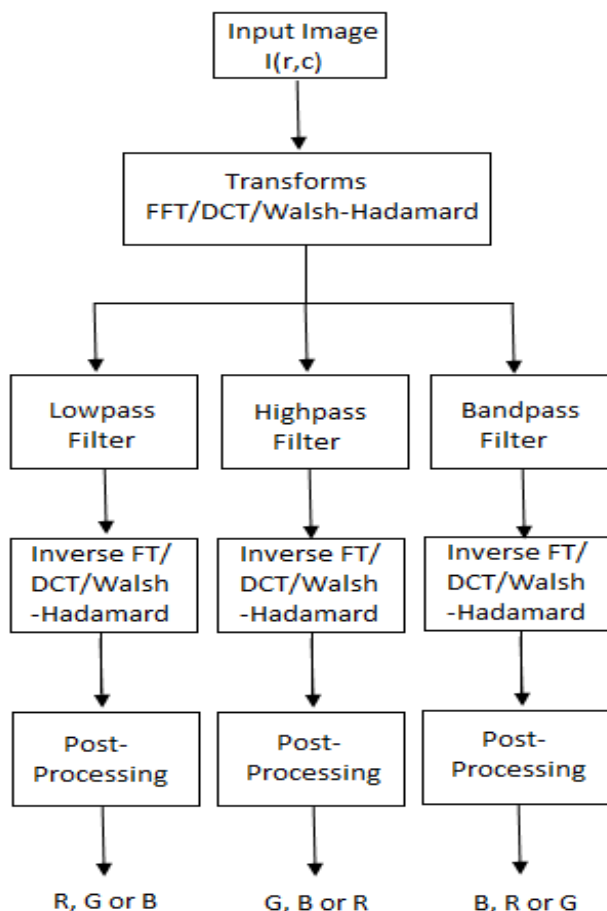


Figure 1. Block diagram of process



Figure 2. Original image



Figure 3. Lowpass filtered image

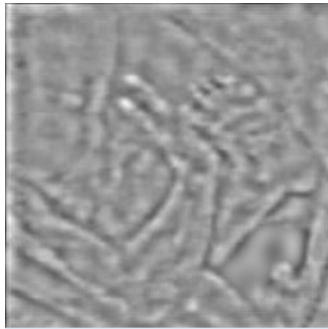


Figure 4. Band pass filtered image

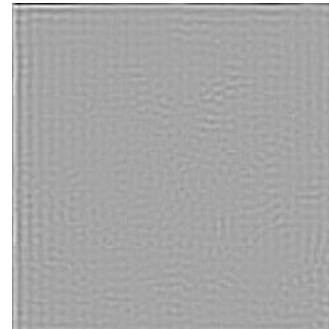


Figure 5. Highpass filtered image



Figure 6. (left to right) original ultrasound image, FFT and DCT pseudo-color enhanced image

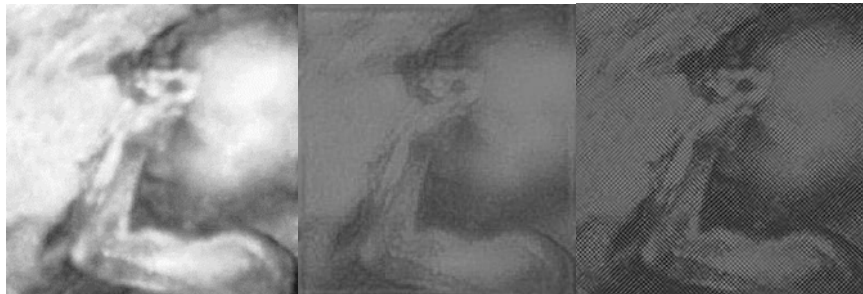


Figure 7. (left to right) original ultrasound image, FFT and DCT pseudo-color enhanced image

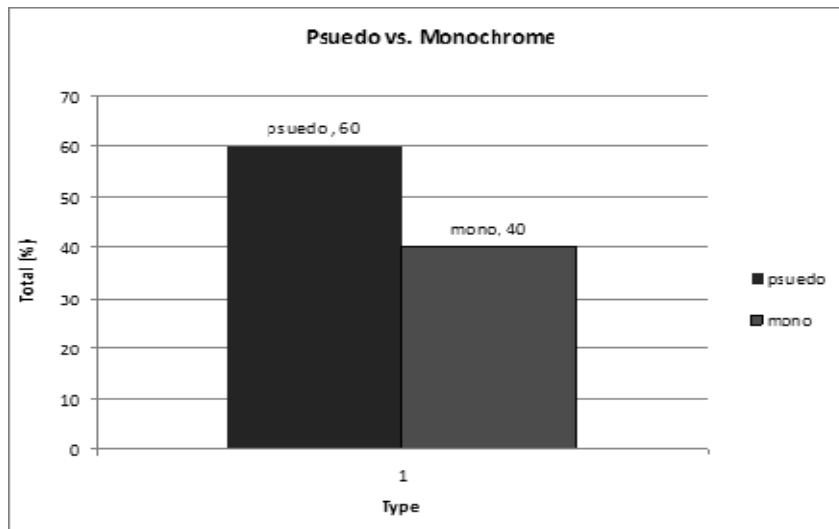


Figure 8. Graph showing 60% of people tested chose pseudo-color images over monochrome

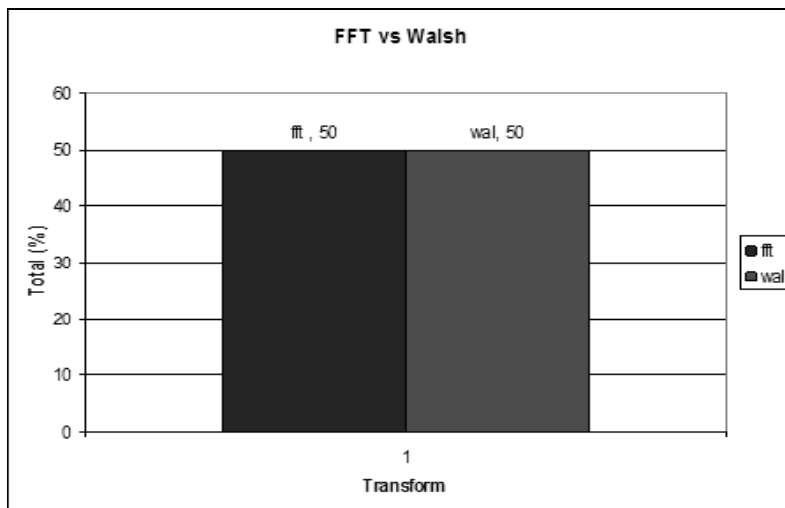


Figure 9. Graph showing a tie between FFT pseudo-color and Walsh pseudo-color

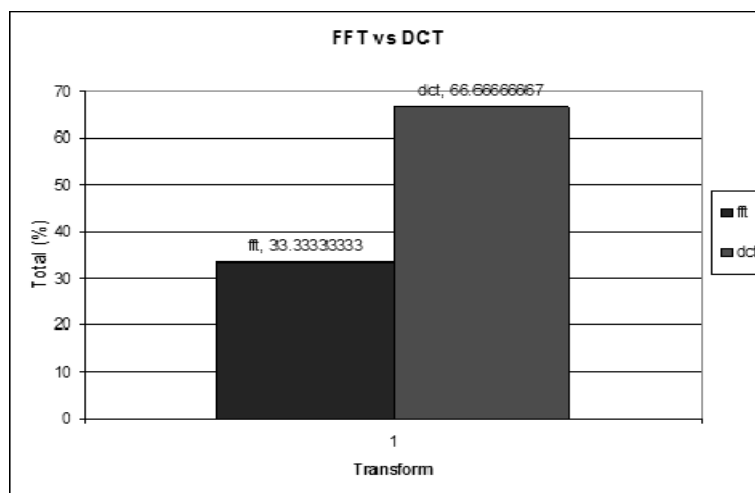


Figure 10. Graph showing 67% of people tested chose DCT pseudo-color over FFT pseudo-color

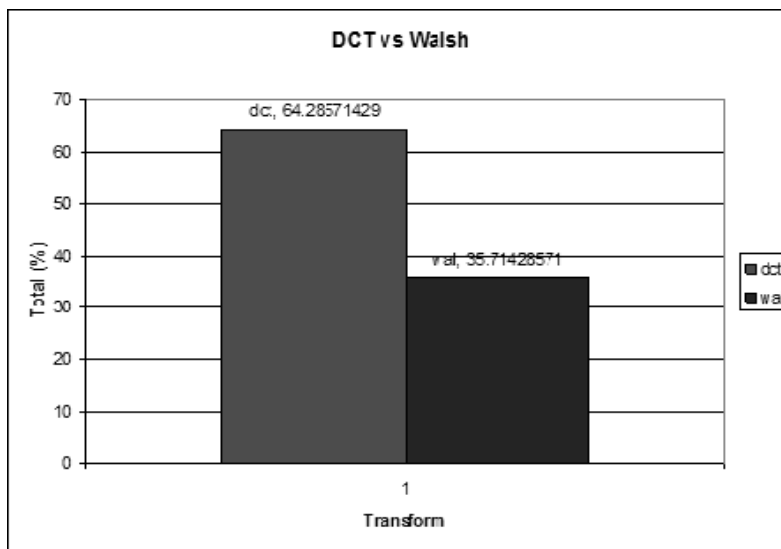


Figure 11. Graph showing 64% of people tested chose DCT pseudo-color over Walsh-Hadamard pseudo-color

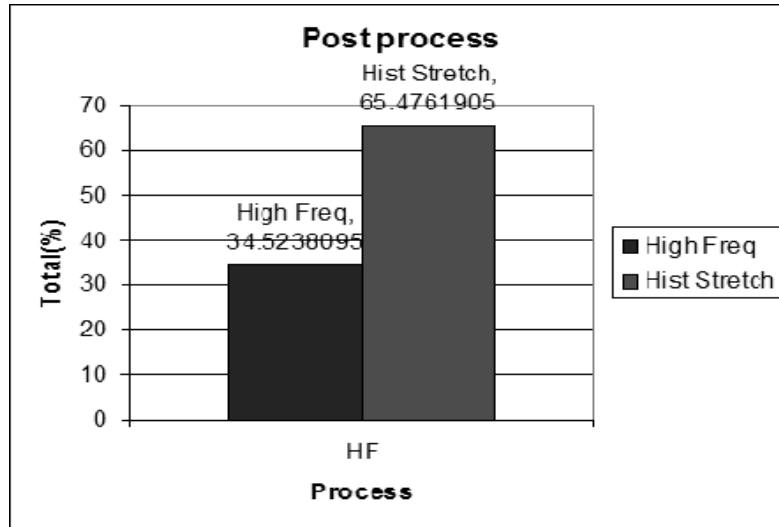


Figure 12. Graph showing 65% of people tested preferred histogram stretch over a high frequency emphasis filter

Appendix A

Image Library

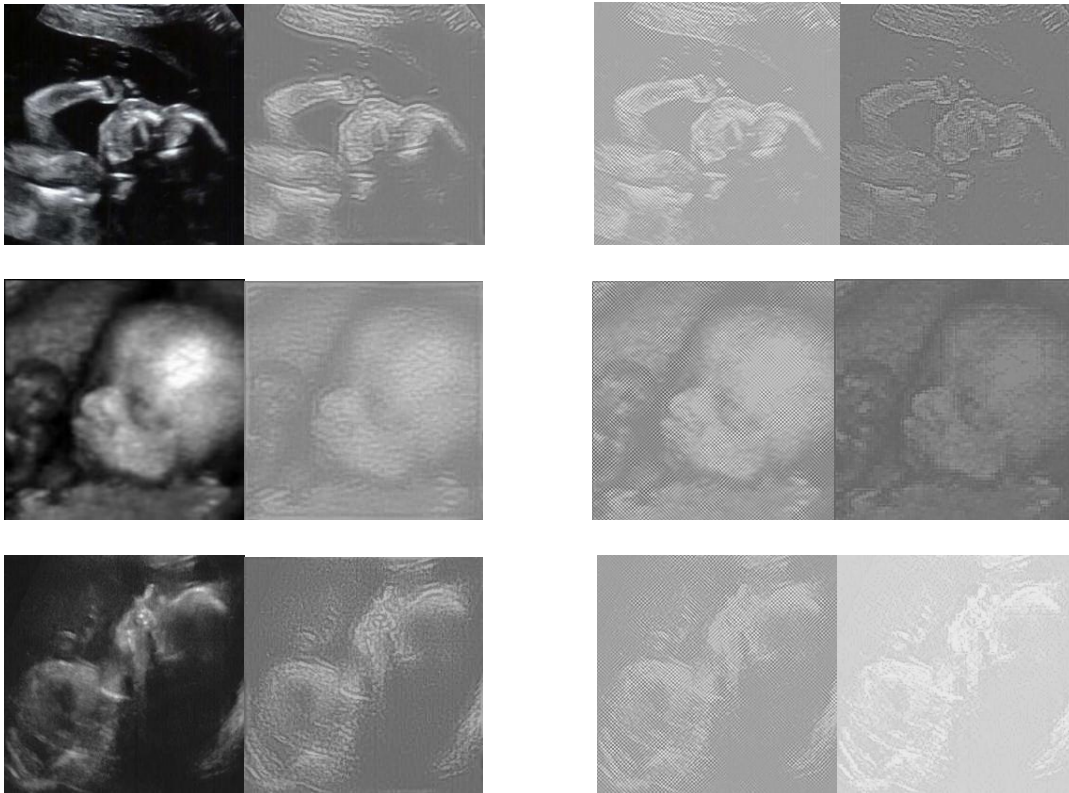
Some of the images created using this technique

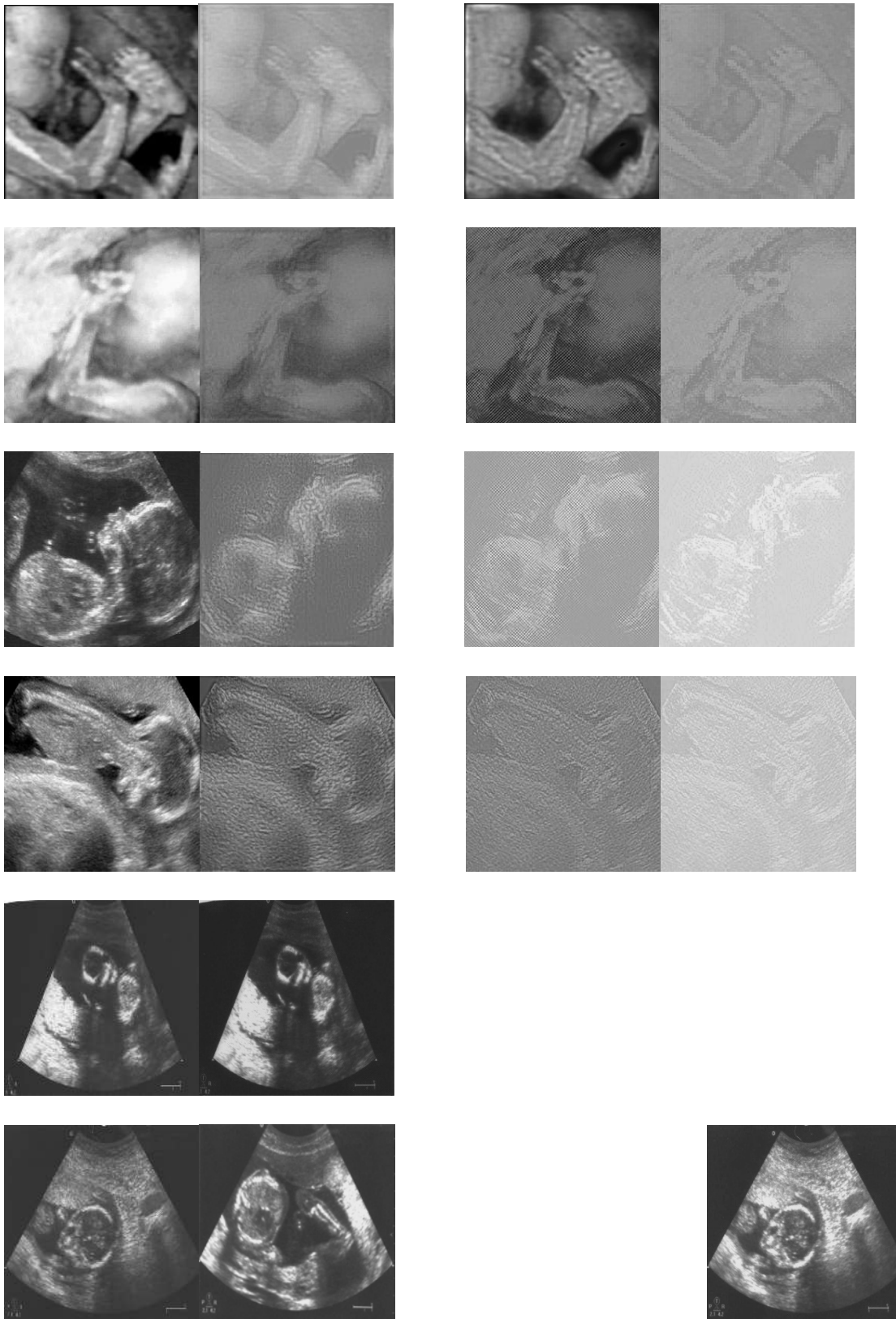
ORIGINALS:

FFT

DCT

WALSH





Appendix B

Pseudo color C function:

```

#include "CVIPtoolkit.h"
#include "CVIPconvert.h"
#include "CVIPdef.h"
#include "CVIPimage.h"
#include "CVIPfs.h"
#include <CVIPspfltr.h>
#include <CVIPmatrix.h>
#include "CVIPlab.h"

Image *PSUEDO( Image *inputImage)
{
IMAGE_FORMAT image_format;
COLOR_FORMAT color_space;
CVIP_TYPE    data_type;
FORMAT       data_format;

char          *outputfile;
Image        *cImage, /* New image*/
             *pImage,
             *bImage,
             *hImage,
             *lImage,
             *bpass,
             *hpass,
             *lpass,
             *cvipImage,
             *Image1,
             *Image2;

byte         **image, /* 2-d matrix data pointer */
             **psuedo, /* pointer*/
             **low,
             **high,
             **band;

Image *images[3];
float norm[3]={255.0,255.0,255.0};
float ref[3]={1.0,1.0,1.0};

unsigned int r, /* row index */
            c, /* column index */
            bands; /* band index */

```

```

unsigned int          no_of_rows,      /* number of rows in image */
                    no_of_cols,      /* number of columns in image */
                    no_of_bands;     /* number of image bands */

image_format=getDataFormat_Image(inputImage);
color_space=getColorSpace_Image(inputImage);
data_format = getDataFormat_Image(inputImage);
data_type = getData_Type_Image(inputImage);

    /*** Gets the number of image bands (planes)*/
    no_of_bands = getNoOfBands_Image(inputImage);
    /*** Gets the number of rows in the input image*/
    no_of_rows =  getNoOfRows_Image(inputImage);
    /*** Gets the number of columns in the input image*/
    no_of_cols =  getNoOfCols_Image(inputImage);
/*To create a new images*/
cImage = new_Image(image_format, color_space, no_of_bands, no_of_rows, no_of_cols, data_type,
data_format);
//pImage = new_Image(image_format, color_space, 3, no_of_rows, no_of_cols, data_type, data_format);
Image1 = duplicate_Image(inputImage);
Image2 = duplicate_Image(inputImage);
images[0]= Image1;
images[1]= Image2;
images[2]= inputImage;
pImage = assemble_bands(images, 3);

/*Perform Transforms*/
//cImage = (Image *)fft_transform(inputImage,256);
//cImage = (Image *)dct_transform(inputImage,256);
cImage = (Image *)walhad_transform(inputImage, 1 , 512);      //Walsh
//cImage = (Image *)walhad_transform(inputImage, 3 , 256);      //Hadamard

hImage = duplicate_Image(cImage);
lImage = duplicate_Image(cImage);

/*Filter transforms [Highpass, lowpass, bandpass] */
                                bpass = (Image *)Butterworth_Band_Pass(cImage,512,1,10,100,2);
                                hpass = (Image *)Butterworth_High(hImage,512,1,100,2);
                                lpass = (Image *)Butterworth_Low(lImage,512,1,10,2);

/*Use inverse to get back images followed by a remap*/
                                /*******Inverse Fourier******/
/*
                                bpass = ifft_transform(bpass,256);
                                hpass = ifft_transform(hpass,256);

```

```

lpass = ifft_transform(lpass,256);
/*****Inverse Cosine*****/
/*
bpass = idct_transform(bpass,256);
hpass = idct_transform(hpass,256);
lpass = idct_transform(lpass,256);
/*****Inverse Walsh/Hadamard*****/
cImage = (Image *)walhad_transform(inputImage, 0 , 512);
//Walsh
/*
//Hadamard
cImage = (Image *)walhad_transform(inputImage, 2 , 256); */

lpass = remap_Image(lpass, data_type, 0 , 255);
hpass = remap_Image(hpass, data_type, 0 , 255);
bpass = remap_Image(bpass, data_type, 0 , 255);
/*view_Image(lpass, "lowpass");
view_Image(bpass, "highpass");
view_Image(hpass, "bandpass");*/
high = getData_Image(hpass, 0);
low = getData_Image(lpass, 0);
band = getData_Image(bpass, 0);
for(bands=0; bands < 3; bands++){
    psuedo= getData_Image(pImage, bands);
    for(r=0; r < no_of_rows; r++) {
        for(c=0; c < no_of_cols; c++) {
            if(bands == 0)
                psuedo[r][c] = low[r][c];
            else if(bands == 1)
                psuedo[r][c] = band[r][c];
            else
                psuedo[r][c] = high[r][c];
        }
    }
}
/*view_Image(pImage, "Before preprocess");
*****Post Processing*****
cvipImage = colorxform(pImage,HSL,norm,ref,1);
cvipImage = histeq(cvipImage, 0);
pImage = colorxform(cvipImage,HSL,norm,ref,0);*/
pImage = hist_stretch(pImage,100,200,10,230);
print_CVIP("\n\tEnter the Output File Name: ");
outputfile = getString_CVIP();
view_Image(pImage,outputfile);
write_Image(pImage,outputfile,CVIP_NO,CVIP_NO,image_format,
1);
free(outputfile);}

```