International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

DWT-PCA Based Image Fusion Using GPU

Vengayil Nayana Murali[1], Rahul C[2]

[1]M. Tech Scholar, L.B.S College of Engineering, Kasargod, India

[2]Assistant Professor, Department of Computer Science, L.B.S College of Engineering, Kasargod, India

Abstract: ***Image fusion techniques have been widely used in areas like satellite imaging, bio-medical imaging, military purpose etc. Image fusion is classified into two group namely spatial domain fusion and transform domain fusion. When spatial domain fusion technique like Principal Component Analysis (PCA) alone is used, it might create some spatial distortion. Such distortions can be handled with the help of frequency domain approach. In the proposed method, in order to combine a multispectral image and panchromatic image of satellite, the PCA can be combined with Discrete Wavelet Transform (DWT). Daubechies wavelets (db4) are used. This is a new approach for image fusion. For parallelizing and increasing the execution speed of the proposed work, a general purpose Graphics Processor Unit (GPU) is used. Driven by the insatiable market demand for real time, high definition 3D graphics, the programmable Graphic Processor Unit or GPU has evolved into a highly parallel, multi-threaded, many core processor with tremendous computational horsepower and very high memory bandwidth. CUDA is a general purpose parallel computing platform and programming model that leverages the parallel compute engine in GPUs to solve many complex computational problems in a more efficient way than on a CPU. When the image fusion is done using GPU, it will be more efficient and it helps to partition the problem into coarse sub problems that can be solved independently in parallel by blocks of threads and each sub problems into finer pieces that can be solved cooperatively in parallel by all threads within the block. With the help of GPU the image fusion can be done faster.***

Keywords: PCA, DWT, GPU, CUDA, Daubechies Wavelet

1. Introduction

Image fusion is the process that combines relevant information from two or more images into a single image. The resulting image will contain all the important information as compared to input images. The new image will extract all the information from source images. Image fusion is a useful technique for merging single sensor and multi-sensor images to enhance the information. The objective of image fusion is to combine information from multiple images in order to produce an image that deliver only the useful information. Image fusion takes place at three different levels i.e. pixel, feature and decision. Pixel level is a low level of fusion which is used to analyse and combine data from different sources before original information is estimated and recognised. Feature level is a middle level of fusion which extracts important features from an image like shape, length, edges, segments and direction. Decision level is a high level of fusion which points to actual target. One of commonly used satellite image fusion method is the Principle Component Analysis (PCA). Since the PCA [17] method can produce some distortions in the fused image, this problem can be solved by the transform domain approach. Also during replacement process of PCA fusion, some information may be lost. In this study knowledge based principal component analysis combined with Discrete Wavelet Transform can be developed to improve the fusing result. Fusion speed has also emerged as an important factor in the image fusion literature with the development of high performance sensors committed to high resolution and quality. GPU has evolved into a very powerful and flexible streaming processor, which includes fully programmable floating point pipelines giving good computational power and memory bandwidth. In most cases like image processing, linear algebra, data sorting and database queries, GPU based implementations are more faster than CPU based implementations. CUDA is the programming model used in GPU, which provides an extended version of ANSI-C for general purpose applications based on GPU. This paper also presents high speed fusion processing on GPU using CUDA.

a) Principal Component Analysis

PCA is a mathematical tool which transforms a number of correlated variables into a number of uncorrelated variables. These uncorrelated variables are called principal components. The first principal component is taken to be along the direction with maximum variance. The second principal component lies in the subspace perpendicular to that of the first principal component. The third principal component is taken in the maximum variance direction in the subspace perpendicular to the first two and so on. The PCA was initially applied for multivariate data reduction and was first developed for a linear transformation based on linear algebra. Due to the linear property, PCA algorithm requires only the computation of eigenvectors.

b) Graphics Processing Unit (GPU)

Recent trends indicate that GPU is widely being used for gaming applications as well as 3D image rendering in an accelerated speed. Being simple with its parallel architecture it is found that the computational power of GPU is at least 3-4 times faster than CPU. The massive computing power and the speed of SIMD architecture can be harnessed and at the same time the CPU can be used for other tasks. The design of graphics processor follows a common structure called graphics pipeline. This design helps hardware implementations maintain high computation rates through parallel execution. Nvidia and AMD are the main vendors of GPU.

c) Compute Unified Device Architecture (CUDA)

CUDA has become a standard platform for general purpose GPU computing in NVIDIA graphics cards. CUDA has many SIMD (Single Instruction Multiple Data) stream multiprocessors(SM), and each SM consists of 8 stream processors (SP). A multithreaded program is partitioned into blocks of thread that execute independently from each other, so that a GPU with more multiprocessor will automatically execute the program in less time than a GPU with fewer multiprocessors. Shared memory enables parallel data cache from global memory for accelerating memory access.

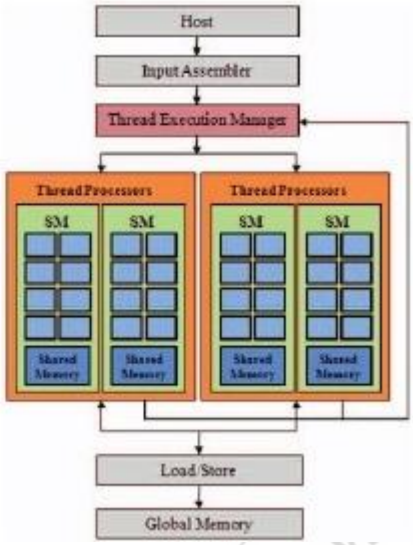

Figure 1: Simplified cuda architecture

The advantages of CUDA architecture are:

- Hardware abstraction: NVIDIA has hidden the architectures of its GPUs beneath an application programming interface (API). Programmers need not know the underlying details of GPU hardware.
- Comfortable development Environment: It provides a relatively simple path for users familiar with C programming language so that they can easily write programs for execution by the device.
- General DRAM memory addressing: Programming flexibility increases when a general DRAM memory addressing is provided. It can retrieve and store data to any location in DRAM, as how it is in CPU.
- Parallel data cache: The on-chip shared memory of CUDA is used by threads to share data with each other. It also improves the general read and write access."
- Thread Synchronization: When a kernel is allowed to complete and a new kernel is started, synchronization is achieved.

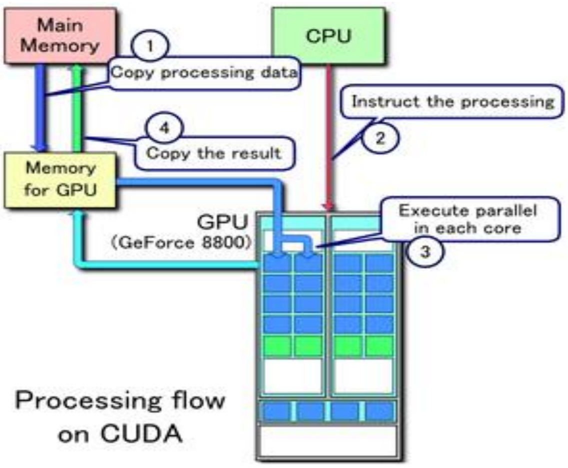

Figure 2: Processing flow on cuda

The source code supplied contains both host (CPU) and kernel (GPU) code. The host code transfers data to and from the GPU"s global memory and initiates the kernel code by calling a function. The kernel runs several blocks of threads and each thread performs a single computation. Threads are organized into a hierarchy of grids of thread block and a grid consists of a number of blocks that execute the same kernel. The block consists of threads that access data from the shared memory and executes instructions in parallel. Each thread has a private local memory. During the execution, threads may access data from multiple memory spaces.

d) Wavelet Transform and Wavelet Based Fusion

Wavelet transform is widely used in signal processing. It can divide a given signal into different scale components [15] wherein each can be studied with a resolution that it matches. The basis functions in wavelet transforms are called wavelets. The wavelets that are used in image fusion are of different categories namely orthogonal, bi-orthogonal and since they have unique image decompression and reconstruction characteristics they can yield different fusion result. The DWT of image signal produces a non redundant image representation, which provides better spatial and spectral localization of image information. The signal is passed through two complementary filters and emerges as two signal, approximation and details, which is known as decomposition or analysis. This can be reconstructed without loss of information by assembling the components back into original signal. The mathematical manipulation, which implies analysis and reconstruction, is called discrete wavelet transform and inverse discrete wavelet transform. Due to the multi-resolution representation capability, the wavelet transform has been used effectively in transient signal analysis, numerical analysis, computer vision etc. The DWT decomposes signal x[n] into different frequency sub-bands with different resolution using the scaling function(φ_j k[n]) and the wavelet function(ψ_j ,k[n]) where j and k are integers..These functions are the dilated and shifted version of φ[n] and ψ[n] defined by:

$$\varphi_{j,k}[n]=2^{\frac{-j}{2}}.\varphi[2^{-jn}-k]$$
$$\psi_{j,k}[n]=2^{\frac{-j}{2}}.\psi[2^{-jn}-k]$$

In the remaining, section II lists out the related works. Section III describes the existing system and section IV has the proposed system. Section V has the results and discussion.

2. Related Works

There are wide varieties of image fusion techniques. Some of them yielded poor performance while some others gave good results. Each of them has its own advantages and disadvantages. Also it improves the quality by removing the noise and blurriness of the image. We briefly review the techniques.

Integrated DCT and PCA technique [1] proposed by Shaveta Mahajan combines information from multiple images of the same scene in order to deliver only the useful information. The discrete cosine transforms (DCT) based methods of image fusion are more suitable and time-saving in real-time systems using DCT based standards of still image or video.

Since DCT based image fusion results produced results with lesser quality, it was integrated with PCA and non linear enhancements. The image fusion methods using discrete cosine transform (DCT) are considered to be more appropriate and time-saving in real-time systems using still image or video standards based on DCT. The image fusion method using DCT alone produced low clarity image. After cross validation with image performance parameters it was seen that the proposed algorithm provided better results than the existing ones.

Seung, J et al., (2009) [2] has shown the approaches to accelerate multiscale image fusion algorithm using CUDA software platform. They have experimented in wavelet algorithm by using Daubechies-4 wavelet for discrete wavelet transform and a shift invariant discrete wavelet transform using Haar wavelets. Averaging fusion scheme was used after wavelet decomposition. Using CUDA platform they have utilized the power of GPU.

Himanshi et al., [3] in their paper presents a combination of Principal Component Analysis and Dual Tree Complex Wavelet as an improved fusion approach for MR and CT-scan images. Unlike real valued discrete wavelet transforms, DTCWT provides shift invariance and improved directionality along with preservation of spectral content. The decomposed images are then processed using PCA a based fusion rule to improve upon the resolution and reduce the redundancy. Simulation results demonstrate an improvement in visual quality of the fused image supported by higher values of fusion metrics.

Sonali, S et al(2014) [4] have fused the images of two modalities CT and MRI by integrating the DWT and PCA. DSP processors such as ADSP-BF533/32/31 provide higher performance and low power utilization. The fusion image will have the qualities of both the methods. It has provided less Mean Square Error and high signal to noise ratio.

Vani, S et al(2015) [5] has discussed the multi focus and multi modal image fusion using wavelet transform. In this the multimodal and multifocus images to be fused are decomposed by dual tree discrete wavelet transform (DTDWT). The fusion takes place by electing the average of the approximation coefficients and maximum of the detailed coefficients. The inverse of DTDWT will provide the fused image.

Wencheng Wan and Faliang Chang [6] presented a simple and efficient algorithm for multi-focus image fusion, which used a multiresolution signal decomposition scheme called Laplacian pyramid method. Firstly, the Laplacian pyramids of each source image are deconstructed separately, and then each level of new Laplacian pyramid is fused by adopting different fusion rules. To the top level, it adopts the maximum region information rule; and to the rest levels, it adopts the maximum region energy rule. Finally, the fused image is obtained by inverse Laplacian pyramid transform.

Senthil Kumar Sadhasivam et al [7] proposed a new scheme that overcomes the limitations of PCA weighted fusion. The source images of size are passed through Gaussian low pass filter to give the smoothened image from the source image. The high frequency difference image is obtained by calculating the deviation of the smoothened images from the source image. The high frequency difference images D_{ir} and D_{vis} are fused using the weighted average rule, the weights being determined by applying the PCA to the images. The fusion performance was evaluated using parameters like entropy, mutual information and the SSIM based index. From the values obtained it was easy to conclude that PCA-Max Fusion algorithm was an efficient technique for fusing IR and visible spectrum images.

Namratha H.N and Raghu M.T [8] presented a novel fusion rule via high pass modulation using Local Magnitude Ratio in Fast Discrete Curvelet Transforms (FDCT) domain. The source images are transformed into curvelet coefficients which contains more edge information of the source images. The coefficient values are small for details like texture or angle points. So further the coefficients are fused using wavelet transform-based method so that it contain small details information.

3. Existing System

PCA is one of the commonly used image fusion method in satellite image fusing. PCA fusion can transform a dataset into a new coordinate system with orthogonal basis. In the existing system [11] 3-band multispectral images (XS1, XS2 and XS3) and a high resolution panchromatic image of a SPOT satellite is considered. In order to obtain the useful landcover information of the geological location, a preprocessing step is used, wherein it will group the image pixels into K groups. A maxima likelihood classifier is used in the pre-processing step.

The SPOT images XS1, XS2 and XS3 are considered as 3 random variables. Then the eigenvalues (λ_1, λ_2, λ_3) and eigenvectors (e1, e2, e3) are determined, where the relationship between them can be shown by the following equation.

$$(A - \lambda I)\, X = 0$$

Where A is an n x n matrix, λ is an eigenvalue of matrix A and X is the eigenvector.

After the transformation of multispectral images in to the PC space, the three principal components PC1, PC2 and PC3 are determined.

$$\text{If XS} \in \text{Class1,} \quad \begin{bmatrix} PC_{1,c1} \\ PC_{2,c1} \\ PC_{3,c1} \end{bmatrix} = \begin{bmatrix} e'_{1,c1}\ XS \\ e'_{2,c1}\ XS \\ e'_{3,c1}\ XS \end{bmatrix}$$

$$\text{If XS}_i \in \text{Class2,} \quad \begin{bmatrix} PC_{1,c2} \\ PC_{2,c2} \\ PC_{3,c2} \end{bmatrix} = \begin{bmatrix} e'_{1,c2}\ XS \\ e'_{2,c2}\ XS \\ e'_{3,c2}\ XS \end{bmatrix}.$$

$$\text{If XS}_i \in \text{Class n,} \quad \begin{bmatrix} PC_{1,cn} \\ PC_{2,cn} \\ PC_{3,cn} \end{bmatrix} = \begin{bmatrix} e'_{1,cn}\ XS \\ e'_{2,cn}\ XS \\ e'_{3,cn}\ XS \end{bmatrix}$$

The PCA is computed for each class. Then the first principal component is replaced with the high resolution panchromatic image, a method generally known as pan-sharpening. The new PC1 along with the old PC2 and PC3 are retransformed back into the original space. Then pixels of each class were fused by the PCA method. By combining all the classes, final fused result is obtained.

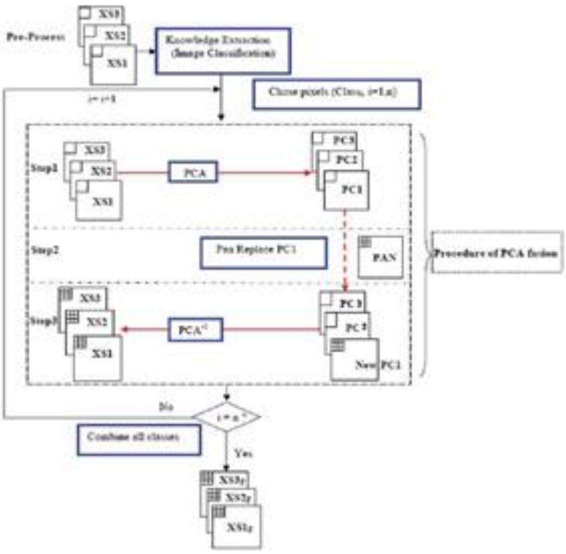

Figure 3: Flowchart Of KBPCA Fusion

4. Proposed System

In the existing system the only method used is PCA which is one of the spatial domain techniques. The spatial domain techniques may sometimes introduce spatial distortions and will not provide any spectral information. Moreover the first principal component that has maximum variance need not contain the actual data. Such spatial distortions are handled by frequency domain methods. So in the proposed system both PCA and DWT is combined to benefit from both. Here the wavelet class used is Daubechies wavelet. It is a family of orthogonal wavelets defining a discrete wavelet transform and characterized by a maximal number of vanishing moments for some given support. For each wavelet type in this class, there is a scaling function which generates an orthogonal multi resolution analysis. They are chosen to have highest number of vanishing moments i.e., compact support. These wavelets are continuous. The wavelets with fewer vanishing moments give less smoothing and remove less details, but the wavelets with more vanishing moments produces distortions. The wavelet used here is db4.

a) Implementation Using CPU

1. By applying DWT find LL, HL, LH and HH sub bands of PAN and RGB image.
2. Apply PCA on LL, HL, LH and HH sub bands of RGB image so that their corresponding PC1, PC2 and PC3 components are obtained.
3. Now replace PC1 components with the LL, LH, HL and HH of PAN image.
4. Perform inverse PCA upon the principal components so that it produces LL, LH, HL and HH components of RGB image.
5. By applying inverse DWT the fused image is obtained.

First a multispectral and panchromatic image is selected. Then the multispectral image is decomposed into R, G and B components. Discrete wavelet transform is then performed on the R, G and B as well as the grey components of the panchromatic image. Since db4 is the wavelet used, it should be specified. Then the LL, LH, HL and HH components of the R, G and B channels are separated. Now Principal Component Analysis is performed on the dwt components, so that the corresponding principal components are obtained.

PCA algorithm:

1. Find the mean of the values
2. Find the covariance matrix
3. Now compute the eigenvalues and its corresponding eigenvectors.
4. Sort the eigenvalues so that its corresponding eigenvectors also gets sorted.
5. Multiply the values with the eigenvectors.
6. Resultant matrix contains the Principal Components (PC1, PC2 and PC3).

Now replace the PC1 with corresponding LL, LH, HL and HH components of the panchromatic image. Next the inverse PCA is performed by multiplying the eigenvector with the transpose of new components. The resultant matrices will be LL, LH, HL and HH for R, G and B. Inorder to obtain the fused result inverse DWT is performed. The resultant matrix contains the pixel values of the fused image.

b) Implementation Using GPU

GPUs have in recent years evolved into highly parallel, multithreaded, many-core processors with tremendous computational speed and very high memory bandwidth [4]. Therefore, they are well suited for massively data parallel processing with high arithmetic floating point intensity. With the dramatic increase of the processing power of GPUs, it is possible to use GPUs for efficient general purpose processing nowadays, namely in the field of general purpose GPU (GPGPU) computing.

CUDA architecture enables NVIDIA GPUs to execute parallel programs. A CUDA program executes kernels in parallel across a set of parallel threads organized in thread blocks and grids consisting of those thread blocks as shown in Fig. 5. Correspondingly, Fig. 5 also presents different levels of memory, i.e., registers and local memory for a thread, shared memory for the block and global as well as constant memory and texture memory for the grid on the GPU. The GPU instantiates a kernel program on a grid of thread blocks, whereas each thread within a thread block executes an instance of the kernel. The CUDA kernel and the main part of the GPU implementation are given in pseudo-code in Fig. 4.

```
__global__ void transformKernel(float *outputData, int width, int height, float theta)
{
 unsigned int x = blockIdx.x*blockDim.x + threadIdx.x;
 unsigned int y = blockIdx.y*blockDim.y + threadIdx.y;
 float u = (float)x - (float)width/2;
 float v = (float)y - (float)height/2;
 float tu = u*cosf(theta) - v*sinf(theta);
 float tv = v*cosf(theta) + u*sinf(theta);
 tu /= (float)width;
 tv /= (float)height;
 //variable declaration for R,G, B and grey
 for( i = 0 ; i < img->height ; i++ )
 {
   for( j = 0 ; j < img->width ; j++ )
   {
     //Reads and Stores the value for R,G, B and grey
   }
 }
}
_global_void _dwt(float *img, int width, int height, int b)
{
 unsigned int x = blockIdx.x*blockDim.x + threadIdx.x;
 unsigned int y = blockIdx.y*blockDim.y + threadIdx.y;
 height = img->height;
 width  = img->width;
 string nm = "db4";
 vector<double> l1,h1,l2,h2;
 filtcoef(nm,l1,h1,l2,h2);
//Finds the coefficients for the wavelet db4
 if(b==1)
 {
  //Find the dwt_2d of R and save it in dwt_output
 }
 Elseif(b==2)
 {
   //Find the dwt_2d of G and append it in dwt_output
 }
 Elseif(b==3)
 {
   //Find the dwt_2d of B and append it in dwt_output
 }
  Else
  {
  // Find the dwt_2d of grey and append it in dwt_output
  //combine the LL, LH, HL and HH bands of R, G and B
  //Find the PCA of all subbands
    sdkloadPPM4(); //replaces the first principal component with grey's subbands
  //Find the inversePCA
  idwt_2d( dwt_coef2,flag, nm, idwt_output2,length);
//Finds the inverse dwt and stores the new pixel values in idwt_output2
}
```

```
int main(int argc, char **argv)
{
  // Enter both the input images
  //Find the cuda device
  //Find the path to the respective images
  dim3 dimBlock(8, 8, 1);
  dim3 dimGrid(width / dimBlock.x, height / dimBlock.y, 1);
   transformKernel<<<dimGrid, dimBlock, 0>>>(dData, width, height, angle);
  checkCudaErrors(cudaDeviceSynchronize());
  StopWatchInterface *timer = NULL;
  sdkCreateTimer(&timer);
  sdkStartTimer(&timer);
  for(int  i=1;i<5;i++)
  {
   _dwt<<<dimGrid, dimBlock, 0>>>(dData, width, height, i);
  getLastCudaError("Kernel execution failed");
  checkCudaErrors(cudaDeviceSynchronize());
  }
   sdkStopTimer(&timer);
// print the time taken to execute
sdkSaveImg(outputFilename, hOutputData, width, height);
//fused image is saved in the file _out.pgm
//Free both the paths to input images.
}
```

Figure 4: Pseudocode for GPU Implementation Including the Cuda Kernel

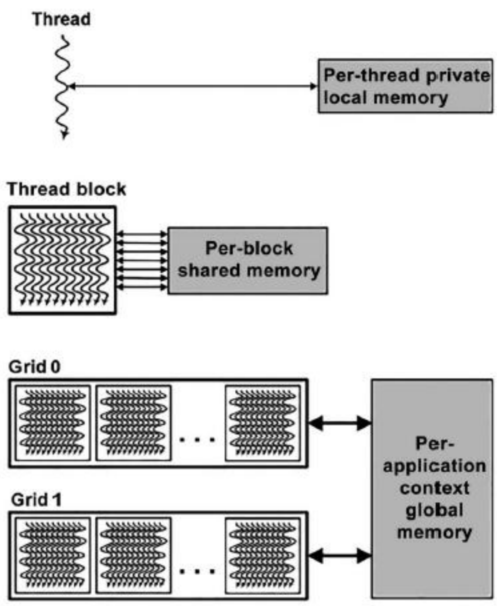

Figure 5: Cuda Heirarchy of Threads, Blocks and Grids with Corresponding per Thread Private, Per-Block Shared and Per Application Global Memory Spaces

The main implementation techniques and strategies are described as follows.

1. CUDA kernel called „transformKernel" was designed to read the pixel values of the source image i.e, the multi-spectral image and the panchromatic image and store

them in separate 1-dimensional arrays and several necessary parameters,e.g., the picture width and height.

2. The kernel „_dwt‟ will help in finding the sub band after applying dwt. This kernel is kept in a for loop so that for the first time the sub bands for red is found, on the second iteration the sub bands for green is appended along with that of red. On the third iteration the same is applied for blue.
3. On the fourth iteration same is applied for grey of panchromatic image and is appended along with the other sub bands.
4. The PCA is applied to LL, LH, HL and HH sub bands of RGB and the first principal component is replaced with sub bands of grey.
5. Inverse PCA is done and on the obtained result inverse dwt is applied.
6. The result is stored in another variable.
7. For tuning the performance, the block size selected is 8 x 8 threads per block. The GPU used is Nvidia GeForce 710M. The runtime for all input sizes is measured on a two-dimensional block while the dimensions of the grid are set dynamically corresponding to the image size.
8. Both the input images are read.
9. Cuda device is found.
10. Path to both the image is found.
11. Block size is set to 8 x 8 and the grid size will be set dynamically.
12. The kernel „transformKernel‟ is called, and then a timer is created and started.
13. The kernel „_dwt‟ is called.
14. Timer is stopped and the output pixels are copied back to host.
15. The image will be displayed in the output file.

5. Results and Discussion

When the image fusion was done using the proposed algorithm, it was seen that it yielded better result. For the implementation of the fusion process, images were taken from the Global Land Cover Facility (GLCF). GLCF provides access to some free Quickbird imagery which is mainly used for land cover assessment. Fusion result produced a high resolution colour image by merging low resolution colour image and high resolution panchromatic (black and white) image. This process was done in both CPU and GPU. GPU was used inorder to increase the execution speed. This algorithm was found easier to be implemented in GPU because the parallelization process was implemented easily. As per the analysis, the GPU has achieved an average speed up of 32-35%. Moreover when images of different sizes were given, the execution time changed. The code for CPU was developed using Python 3.4.2. The processor used is 2.4 GHz Intel Core i5. The code for GPU was developed using CUDA with Nvidia GeForce 710 M as the graphics processor.

a) Sunderban Delta

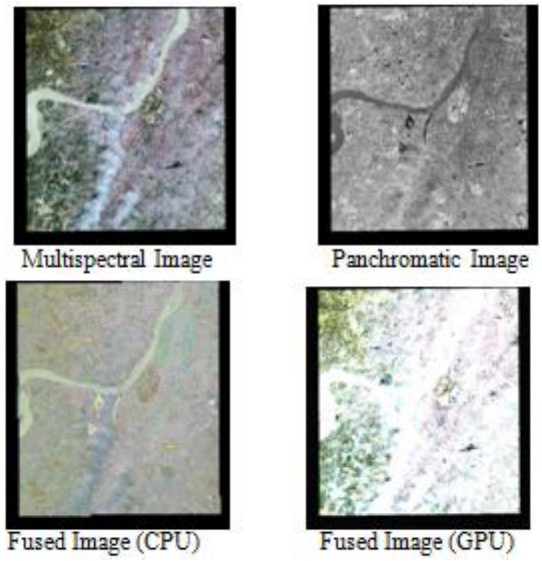

Table 1: Performance Comparison of sunderban ms-pan Fusion

Image Sizes	CPU (seconds)	GPU (seconds)
512 X 512	0.559388682	0.159825338
256 X 256	0.155208044	0.050067111
128 X 128	0.050182782	0.014337938
64 X 64	0.02736923	0.008049774
32 X 32	0.016498916	0.006599566

b) hilka Lake

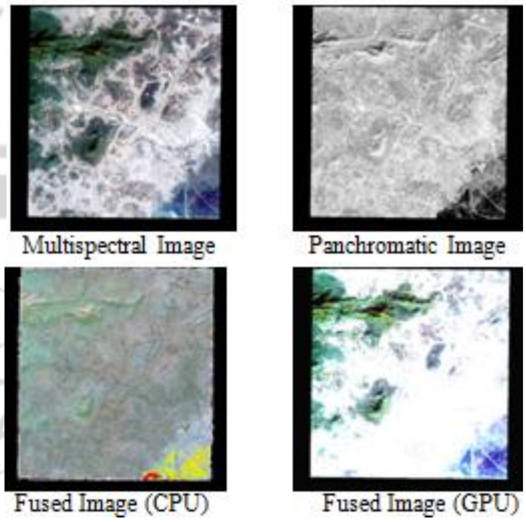

Table 2: Performance Comparison of Chilka ms-pan Fusion

Image Sizes	*CPU (seconds)*	*GPU (seconds)*
512 X 512	0.606260566	0.186541713
256 X 256	0.147181483	0.04672428
128 X 128	0.049086847	0.01533964
64 X 64	0.027976961	0.009325654
32 X 32	0.019002051	0.009048596

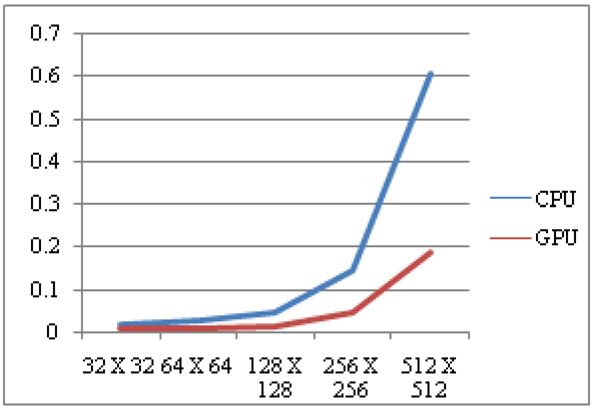

Figure 6: Chart for General Comparison Between the CPU and GPU Speeds

Table 1, 2 summarizes the performance comparisons in terms of speed of different sizes of different images. It is clearly evident that when the image size decreases the speed of execution also decreases. From the fig 6 the fusion speed of the algorithm in CPU and GPU can be seen. GPU takes lesser amount of time to fuse the images.

Table 3: Quality Metrics

Quality Assessment Parameters	*Existing System*	*Proposed System (CPU)*	*Proposed System (GPU)*
RMSE	10.4282	10.4182	9.7166298
PSNR	1.704014	1.7123514	4.3320865
SSIM	-0.413888	-0.4114035	0.2906483
UQI	-0.677144	-0.674937	0.2771212
WSNR	1.496559	1.49083911	4.2037716

The quality criteria [16] used is defined as follows:

Root-Mean-Square Error (RMSE): The RMSE calculates the changes in pixel values to compare the difference between the original and pan-sharpened image.

$$\text{RMSE} = \sqrt{\frac{1}{M.N}\sum_{i=1}^{M}\sum_{j=1}^{N}(X_{i,j} - \hat{X}_{i,j})}$$

$X_{i,j}$ is the pixel value of the original image X and $\hat{X}_{i,j}$ is the pixel value of the pan-sharpened image $\hat{X}$. The value of RMSE is smaller.

Peak Signal to Noise Ratio (PSNR): Ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation.

$$\text{PSNR} = 20.\log_{10}(\text{MAX}_1) - 10.\log_{10}(\text{MSE})$$

Where MAX_1 is the maximum possible pixel value of the image and MSE is the mean-squared error.

Structural Similarity Index (SSIM): It is the method for predicting the perceived quality of images. It is used for measuring the similarity between two images. The values can range between -1 to +1.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c1)(2\sigma_{x,y} + c2)}{(\mu_x^2 + \mu_y^2 + c1)(\sigma_x^2 + \sigma_y^2 + c2)}$$

Universal Image Quality Index (UQI): It combines 3 different factors namely loss of correlation luminance distortion and contrast distortion. It has been widely used to assess the quality of image sharpening recently.

$$\text{UQI} = \frac{\sigma_{x\hat{x}}}{\sigma_x\sigma_{\hat{x}}}.\frac{2\bar{x}\bar{\hat{x}}}{\bar{x}^2+\bar{\hat{x}}^2}.\frac{2\sigma_x\sigma_{\hat{x}}}{\sigma_x^2+\sigma_{\hat{x}}^2}$$

where x,$\hat{x}$ σ_x and $\sigma_{\hat{x}}$ are the mean values and standard deviations of the original image X and the pan-sharpened image $\hat{X}$, respectively.

Weighted Signal-to-Noise Ratio (WSNR): It is calculated in the spatial frequency domain.

$$\text{WSNR} = 10\log 10 \frac{255^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}[X(m,n) - B(m,n)]^2.W(m,n)}$$

W(m,n) is the weighting function, B(m,n) is the DFT of bi-level halftone image.
From the analysis it is seen that the proposed algorithm is better than the existing one.

6. Conclusion

Image fusion has been used in several areas like military [9], medical imaging [18] etc. The research on the fusion of satellite images [14] has improved widely. Multispectral image will be of low resolution and panchromatic image will be of high resolution. The major information content of PC1 images is spatial information which can then be replaced by a high resolution image (PAN). First we analyze the result of PCA based technique. Then the results using the proposed method are also analyzed. From the quality metrics like PSNR, RMSE, SSIM etc, it is clear that the proposed algorithm performs well. The fusion speed was also increased by implementing the proposed algorithm using the GPU. Many image processing applications like fast image registration [12], fast 2D ultrasound image straining [13] etc has been developed with the help of GPU, which exhibited better performance. The wavelet used is „db4" and fewer vanishing moments results in lesser loss in details. The fused image will be having the spatial information of panchromatic image and colour of multispectral image.

References

[1] Shaveta Mahajan and Arpinder Singh, "Integrated PCA & DCT based fusion using consistency verification & Non-Linear Enhancement," International journal of Engineering and Computer Science ISSN:2319-7242 Volume 3 Issue 3 March 2014
[2] Seung-Hun Yoo, Jin-Hyung Park and Chang-Sung Jeong, "Accelerating Multi-scale Image Fusion Algorithms using CUDA", International conference of Soft Computing and Pattern Recognition 2009.
[3] Himanshi, Vikrant Bhateja, Abhinav Krishn and Akansha Sahu, "An Improved Medical Fusion Approach Using PCA and Complex Wavelets", International Conference on Medical Imaging, m-Health and Emerging Communication System 2014
[4] Sonali Mane and S D Sawant, " Image Fusion of CT/MRI using DWT, PCA Methods and Analog DSP Processor", International Journal of Engineering Research and Applications, ISSN : 2248-9622, Vol 4, Issue 2(version 1), February 2014.
[5] Vani M and Saravanakumar S, "Multi Focus and Multi Modal Image fusion using Wavelet Transform", 3rd International Conference on Signal Processing, Communication and Networking 2015

[6] Wencheng Wan,Faliang Chang, "A Multi-Focus Image Fusion Based on Laplacian Pyramid", Journal of Computers, Volume 6, No. 12, December 2011.
[7] Senthil Kumar et.al, "Implementation of Max Principle with PCA in Image Fusion for Surveillance and Navigation Application", Electronic Letters on Computer Vision and Image Analysis 10(1):1-10,2011
[8] Namratha H.N and Raghu M.T, "Remote Sensing Satellite Image Fusion Using Fast Curvelet Transform", International Journal of Science and Research 2013
[9] V P S Naidu and J R Raol," Pixel level Image Fusion using Wavelets and Principal Component Analysis", Defense Science Journal, Vol. 58, No.3, May 2008
[10] M.Sifuzzamman et.al, "Application of Wavelet Transform and its Advantages Compared to Fourier Transform", Journal of Physical Sciences, Volume 13, 2009, 121-134
[11] Jie-Lun Chiang, "Knowledge based Principal Component Analysis for Image Fusion" International Journal of Applied Mathematics and Information Sciences **8**, No. 1L, 223-230 (2014)
[12] Seung-Hun Yoo et.al, " Fast Image Registration for Multisensor Fusion using Graphics Hardware" Proceedings of IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems Seoul, August 20-22, 2008
[13] Tim Idzenga et.al, " Fast 2D Ultrasound Image Straining- Benefits of using a GPU" IEEE transactions on Ultrasonics, Ferroelectrics and Frequency Control, Volume 61, No.1,January 2014.
[14] Firouz Abdulla Al-Wassai and N.V Kalyankar, "Major Limitations of Satellite Images" Journal of Global Research in Computer Science, 4 (5), May 2013, 51-59.
[15] Gagandeep Kour and Sharad P Singh, "Image Decomposition Using Wavelet Transform", International Journal Of Engineering and Computer Science ISSN: 2319-7242 Volume 2 Issue 12 Dec 2013 Page No.3477-3480.
[16] Jagalingam P and Arkal Vittal Hegde, "A Review of Quality Metrices for Fused Image" , International Conference on Water Resources, Coastal and Ocean Engineering 2015.
[17] Pramod Kumar Pandey, Yaduvir Singh and Sweta Tripathi, "Image Processing Using Principal Component Analysis", International Journal of Computer Applications(0975-8887) Volume 15, No. 4, February 2011.
[18] Pavithra C and Dr. S Bhargavi, " Fusion of two images based on wavelet transform", International Journal of Innovative Research in Science, Engineering and Technology, Volume 2, Issue 5, May 2013.