

ISRO RESPOND PROJECT

Development of a Real-time Image Super-resolution System using Parallel Processing Hardware for Remote Sensing Application
(Sanction order no. ISRO/RES/4/642/17-18 dated 07.07.2017)

Final Report [August 2017-May 2021]



Prof. Bhabesh Deka
Principal Investigator
Dept. of ECE, Tezpur University, Assam -784028



Sri. AVV Prasad
Co-Principal Investigator
Dy. Director, BG&DDA
NRSC, ISRO, Hyderabad -500037

November 2021

FINAL REPORT ON RESPOND PROJECT

A. GENERAL INFORMATION	
1. Title of the Project	Development of a Real-time Image Super-resolution System using Parallel Processing Hardware for Remote Sensing Application
2. Principal investigator(s) with address(es)	Prof. Bhabesh Deka, Dept. of ECE School of Engineering, Tezpur University Tezpur University, Napaam, Sonitpur, Assam 784028
3. Other Investigator(s) with address(es)	Sri. AVV Prasad (Co-PI) Dy. Director, BG&DDA Phone: 040-23884457 Mail: prasad_avv@nrsc.gov.in
4. Period of the Project (with Starting date and Completion date)	August 1, 2017-May 31,2021
5. Total grant approved/released by ISRO	Rs. 24,88,000.00
6. Total amount spent during the period of the project	Rs. 24,06,496.00
7. Name and address of the University/Institution	Tezpur University Napaam, Tezpur Sonitpur, Assam (India) Pin-784 028
8. Name of the ISRO/DOS Centre involved	National Remote Sensing Centre (NRSC) Hyderabad
9. Names and present addresses of Research Fellows who worked in the Project	Trishna Barman Department of ECE Tezpur University Napaam, Sonitpur, Assam 784028
10. Names and present addresses of Research Associates/other research Staff recruited for the project	Not applicable
11. Names of Research staff (in 8 & 9 above) who have completed(ing) Ph.D. under the project	Trishna Barman (pursuing Ph.D)
12. Abstract of the Report (in about 200 words)	

Single Image Super-Resolution (SISR) has been gaining much attention in the field of remote sensing. Sparse representation approach is one of the most successfully learning-based method for SISR that shows effective result in MS remote sensing image. The aim of the proposed work is to develop a fast SISR reconstruction framework for multispectral remote sensing images based on dictionary learning and sparse representation using Computed Unified Device Architecture (CUDA) enabled General-Purpose computing on Graphics Processing Units (GP-GPU) hardware. The proposed work is divided into two major parts: Firstly, a novel super-resolution approach is developed for multispectral remote sensing images using sparse coding based on global and adaptive dictionary learning methods for different upscaling factors. Secondly, massively parallel algorithms are designed for dictionary learning and reconstruction phases of the proposed method using Open Multiprocessing (OpenMP)-based multicore parallel processing and CUDA-enabled GP-GPU programming model. In the reconstruction phase, high resolution (HR) image is estimated from low resolution (LR) image by utilizing overcomplete learned dictionaries. The dictionaries are adaptively learned from the given LR images using K-Singular Value Decomposition (K-SVD). Some feature extraction schemes such as Morphological Component Analysis (MCA), first and second order gradient filters are adopted for improving the performance. The proposed method is implemented first on a multicore parallel paradigm based on OpenMP, and then on Intel® Xeon® CPU with NVIDIA Tesla P100 GP-GPU hardware. It not only gives better results in terms of visual quality and objective criteria, but also significantly reduces the computation time compared to the CPU-based sequential counterparts to achieve the near-real time operating speed.

B. SCIENTIFIC/TECHNICAL INFORMATION

1. Summary of the work carried out in about 500 words

The proposed work is divided into three major parts: 1) Design and implementation of MS image SR (MSISR) based on sparse representations over trained overcomplete dictionaries 2) Development of MCA and sparse representations-based MSISR and its multicore implementation 3) CUDA-based GP-GPU implementation of MSISR based on adaptive dictionary learning and sparse representations. In the first work, a multispectral

image super-resolution method was developed based on sparse representations and dictionary learning. It was divided into two phases. In the first phase, global coupled dictionary was trained from an external database of multispectral satellite images using the K-SVD algorithm. A HR dictionary was prepared from high resolution MS patches, while a LR dictionary was prepared from feature patches, which were extracted from LR images after applying the high pass filters. In the second phase, the HR MS image was reconstructed based on sparse representations using the pre-trained dictionary. The results of the proposed MSISR work were compared with bicubic interpolation and the Sparse Fusion of Images (SparseFI). Objective evaluation parameters using Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measurement (MSSIM), Erreur Relative Globale Adimensionnelle de synthese (ERGAS), Spectral angle mapper (SAM), Universal Image Quality Index (Q-index) were computed and compared to validate the performance of the proposed work. Since the reconstruction involves considerable time due to feature extraction and sparse representation, a fast single-image super-resolution technique using OpenMP-based multicore parallel computing technique was implemented. The proposed method can reduce the computation time around 10 times.

The effective feature selection and extraction from the LR image for good dictionary construction is one of the important criteria of sparse representation-based model. In the second work, an improved feature extraction algorithm was developed for constructing efficient dictionary learning and reconstruction. MCA driven feature extraction was used for better feature extraction from LR image. Then, a single image SR approach based on sparse representation was proposed for MS and PAN images. During SR reconstruction, textural features were extracted using MCA for effective sparse representation of patches. Next, a coupled over-complete dictionary learning approach was proposed for both LR-MS and PAN images. During dictionary learning, principal component analysis (PCA) was considered for band reduction. Real remote sensing images from a few Indian satellites, RESOURCESAT-2 and CARTOSAT-2, as well as other satellites, such as QuickBird, were used in simulations to test the proposed method. The proposed method was also compared to those of other current SR methods to determine its superiority in terms of both quantitative measurements and visual analysis. To achieve near real-time performance, the proposed algorithm was implemented using an OpenMP-based multicore parallel processing technique. The proposed algorithm could achieve an acceleration up to 12 times.

Finally, we developed a fast SISR method for multispectral remote sensing images using sparse representations and CUDA-enabled GPU hardware. It was divided into two sections: A) Developed a novel framework for multispectral remote sensing image processing using the sparse coding and self-example-based dictionary learning method. B) Implemented dictionary learning technique (i.e., K-SVD) and the proposed sparse optimization on GP-GPU using CUDA programming. Instead of training from external HR images, an effective technique can be used to learn the coupled dictionary from the given LR image itself. To obtain the improved results and speed, an efficient convex optimization technique and improved dictionary learning technique, i.e., K-SVD approach was used for SR of multispectral images. Experiments were conducted using a few real remote-sensing image databases from the NRSC, ISRO, and a publicly available standard remote-sensing database. For quantitative measurements of the proposed method, performance evaluation parameters such as PSNR, MSSIM, Q-index, SAM, ERGAS, and Spatial Correlation Coefficient (sCC) were performed. For standardization and benchmarking, various test images on multispectral image SR were tested for different upscaling factors and compared the proposed method with state-of-the-art methods. The proposed method can perform better both quantitatively as well as visually. Parallel implementation of the proposed sparse-based multispectral image SR was done in GP-GPU for hardware acceleration. NVIDIA P100 GP-GPU embedded hardware with 3584 cores was used with the CUDA programming model to implement dictionary learning and sparse reconstruction. The speed up achieved using CUDA-GPU based proposed method was upto 10-13x for dictionary learning and 3-185x for reconstruction as compared to the sequential CPU counterparts. Furthermore, an application of the proposed method was demonstrated with image classification on real remote sensing images; features present both in the input LR and reconstructed HR image are detected and used for classification and results are compared both for the LR and HR images.

2. Introduction – Background and Objectives

- **Introduction**

Image super-resolution (SR) is a demanding research area in image processing and computer vision due to abundance of high-resolution display systems in many vision-based applications. It is basically used to achieve a HR image from single or multiple LR as inputs. SR algorithms can be classified into two categories based on their

inputs: single image super-resolution (SISR) and multiple image super-resolution (MISR). MISR techniques need a complex registration process in which alignment of multiple LR images is required with sub-pixel precision. On the contrary, SISR technique is more practical for many real-world applications, such as satellite imaging, medical imaging, video surveillance, television display, etc.



Figure 1: An example of remote sensing image super-resolution

Due to the physical limitation of imaging sensors and complex atmospheric disturbances, it is quite challenging to capture a HR image for remote sensing applications. Since multispectral (MS) images are obtained at low spatial resolution, there are some limitations to deploy these images in various remote sensing applications such as environment monitoring, military target identification, etc. To overcome these adversities, SR offers a cost-effective technique to produce HR images from the LR MS imagery.

In general, there are many types of SISR approaches, including interpolation-based methods [1], reconstruction-based methods [2], and learning-based methods [3]. Interpolation-based method usually miss high frequency details supposed to be present in target HR images, which leads to jagged artifacts and blurring. Although reconstruction methods can recover better edges information, they cannot reconstruct effectively when zooming factor is larger. In addition, post processing and sophisticated registration process are required for these methods. In recent years, SR based on learning methods have become a growing field in image processing [3]-[9]. In these methods, prior information is extracted between LR and HR image pairs. Those methods exhibit good performance because information synthesization is done from the image data itself. Since, SISR is an ill-posed problem due to

information loss and unique solution is impossible to obtain. Therefore, it needs strong prior information about the expected HR images to produce a robust solution in SR methods [10] [11].

Learning-based methods can be classified into two sub-categories: external and internal example-based learning methods. External example learning-based method learns the relationships between HR and LR image patches from similar external databases. They are obtained from many known LR and HR image pairs. In the internal example learning-based method, a single image is used instead of the database. Among external example learning-based algorithms, sparse representation based on dictionary learning is one of the most popular methods. Sparse representation-based SR method using global dictionary very much depends upon the HR images in the training database. Global dictionary might not be able to represent all the image patches of the LR image accurately. More image samples can be employed to learn the database, but the correctness of information yield by the training database for any LR input image cannot be guaranteed. The reconstructed HR may seem reasonable, but the information acquired from the training database may be irrelevant because the image patches of global dictionary are not available during reconstruction. To overcome the above limitations of global dictionary, adaptive dictionary has been broadly used to solve the SR problem. In the adaptive dictionary learning, only the LR image is used instead of external database for dictionary training. It is assumed that many similar image patches may exist both within the same and across different scales of the LR image.

Sparse representation-based methods are computationally very intensive due to their inverse problem. Additionally, another reason of having slow performance in sparse representation-based SR method is that many image patches are being processed sequentially. To solve those problems, parallel computing approach can be exploited to design highly parallelized algorithms by using graphics processing units (GPU) with computed unified device architecture (CUDA) environment for real-time SR reconstruction of remote sensing images. Since computation process becomes highly exhaustive in MS remote sensing image because of its large dimension, CUDA enabled GPU is the best choice to handle this computational problem

effectively.

- **Background**

Learning-based super-resolution approaches have drawn much attention in recent years [3], [4], [8], [9]. In these methods, prior knowledge is exploited from the training model to establish an end-to-end mapping between LR and HR images patches. The estimated HR image can be restored by adopting the training model through the reconstruction phase. Various external example learning-based methods have been reported in the literature such as the nearest neighbor embedding approach [4], manifold learning [13], sparse coding [14] [16] [17] and deep learning [18] [19], etc. Freeman et al. presented a nearest neighbor embedding approach that learns a relationship between LR-HR patches and between neighboring HR patches using Markov network [4]. But the reconstructed image not only has poor visual quality, but also does not have fine structure at the edge. To overcome this, SR based on nearest neighbor embedding technique has been proposed by Chang et al. [13]. This method has fewer artifacts, only blurring is occurred during LR patch matching. On the other hand, sparse coding can give improved reconstruction results and outperforms existing previous SR methods. Therefore, sparse coding becomes a promising and effective research topic in the field of super-resolution. Sparse coding-based SR method can be divided into two stages: first, training of sparse overcomplete dictionaries and second, reconstruction of the SR image by using the sparse representations on a suitable dictionary. According to the database used for dictionary training, it can be categorized into two types: global and adaptive dictionary based sparse coding methods. Yang et al. proposed a SR method using sparse priors; a global dictionary was learned from both LR and HR image patches from an external dataset [14]. Zeyde et al. simplified the algorithm of Yang et al. that makes the proposed method computationally less heavy and used a different training approach for a dictionary pair, i.e., the K-SVD technique [20]. Huihui presented a remote sensing SR method, which can reconstruct LR Landsat image based on global dictionary and sparse coding algorithm [21]. SPOT5 and simulated Landsat TM/ETM+ are used as HR and LR images, respectively to learn the dictionary pair using K-SVD. Since global dictionary learning-based SR methods are not so favorable for remote sensing application, therefore adaptive dictionary-

based methods are used where dictionary is learned using LR-HR image patches from a single image. Zhu et al. proposed a fast novel SR algorithm based on sparse representation and adaptive dictionary learning [22]. This method gives similar results with the state-of-the-art methods, and it is computationally less heavy. Pan et al. presented a SR technique for remote sensing images based on compressive sensing (CS), structural self-similarity (SSSIM), and dictionary learning [23].

Recently, parallel implementations of SR methods have been proposed for real-time applications [24], [25], [26], [27]. Kulkarni et al. proposed an efficient parallel orthogonal matching pursuit algorithm for SR reconstruction. This algorithm is platform independent, which was tested on different platforms such as CPUs, GPUs, multi-cores, and a Virtex-7 FPGA [24]. They achieved relatively better speed up around 2x to 13x, when compared with the previously reported methods. Hanlin et al. proposed a fast SISR method with CUDA acceleration [27]. A fast least absolute shrinkage and selection operator (LASSO) approximation has been used to estimate the sparse code. Results show that LASSO outperforms state-of-the-art methods and gives an acceleration up to 6.2x. Remote sensing images consist of several spectral bands, thereby their data volume is considerably large. Therefore, it is highly computationally exhaustive to restore HR images of high volumetric images. CUDA-enabled GPU is the best choice to handle this computational problem effectively for remote sensing images [28] [29]. Moustafa et al. presented a fast CUDA enabled GPU accelerated multispectral image SR using the morphological component analysis and adaptive dictionary. This proposed method achieves speedup of around 2 to 40 for different image sizes.

- **Objectives (as per proposal approved by ISRO):**

1. To design single image SR (SISR) algorithms using the concept of parallel computing.
2. To design and develop novel approaches for creating GP-GPU modules for real-time super-resolution reconstruction of remote sensing images.
3. To evaluate and compare performances of different algorithms for setting benchmarks for super-resolution reconstruction of remote sensing images

3. Instrumentation – Development of equipment/hardware

Not applicable

4. Techniques and methods employed for the investigation/study including details of development of the software

4.1 Sparse Representation model and its formulation for Super-resolution

In recent years, sparse representation has gained popularity in many fields especially in the field of image processing, computer vision, signal processing, and pattern recognition. It is found that sparse representation is definitely a beneficial tool for image super-resolution, image denoising, and object recognition. A signal is known to be sparse, if there are only a few non-zero elements present in the signal. In sparse representation model, a signal $X \in R^{N \times 1}$ can be represented as a linear combination of k “atoms” from the over-complete dictionary $D \in R^{N \times k}$ with $N \ll k$ as

$$X = D\alpha \quad (1)$$

where $\alpha \in R^{k \times 1}$ is the column vector with the weighting sparse co-efficients. The signal can be called as “ k -sparse” if only k ($k \ll N$) nonzero entities are present in the column vector α . The recovery of α from X is an ill-posed problem that cannot provide a unique solution. By imposing a prior information or an appropriate regularizing constraint, this problem is a sparse representation problem with the l_0 -norm minimization, i.e.:

$$\min_{\alpha} \|\alpha\|_0 \quad s.t. \quad X = D\alpha, \quad (2)$$

where $\|\cdot\|_0$ refers to the number of non-zero elements present in the vector. The optimization problem of Eq. 2 is a non-deterministic polynomial-time hard (NP-hard) problem and approximation of its solution is very difficult [30]. In the optimization problem, l_1 -norm provides the sufficient sparse solution which is equivalent to a solution obtained by l_0 -norm minimization. Therefore, l_1 -norm is used instead of l_0 -norm that converts the non-convex problem into a convex optimization problem, as follows:

$$\min_{\alpha} \|\alpha\|_1 \quad s.t. \quad \|D\alpha - X\|_2^2 \leq \epsilon \quad (3)$$

This is the basis pursuit denoising (BPDN) problem, which can be efficiently solved by using the recently developed fast l_1 -minimization algorithms such as the Fast Iterative Shrinkage Algorithm (FISTA) [31], LASSO [32]. In super-resolution model, the observed LR image X is obtained from the HR image Y based on following reconstruction constraint, as follows:

$$X = SHY \quad (4)$$

Here, H and S represent the blurring and down-sampling operators, respectively. Recovering the HR image Y from the LR image is an ill-posed problem because many HR images Y may obey the above equation with the same reconstruction constraint. In sparse coding approach for super-resolution, image patches can be represented by sparse linear combination of elements from an appropriately chosen over-complete dictionary. By considering this observation, patch-wise sparsity prior regularization is used to solve the ill-posed problem of Eq. 4. To reconstruct the HR image Y , sparse co-efficients can be obtained by using a LR dictionary (D_l) containing corresponding LR image patches taken from the training data. The block diagram of sparse representation-based super-resolution for multispectral image is shown in Fig. 2. To compute the sparse co-efficients of $X = D_l\alpha$, the minimization problem can be used:

$$\min_{\alpha} \|\alpha\|_1 \quad \text{subject to} \quad \|D_l\alpha - X\|_2^2 \leq \epsilon, \quad (5)$$

By using Lagrange multipliers, the above optimization problem can be reformulated to:

$$\min_{\alpha} \|D_l\alpha - X\|_2^2 + \lambda \|\alpha\|_1, \quad (6)$$

where the regularization parameter is used to trade-off between sparsity of the solution and accuracy of the output. Since LR and HR patches share the same sparse coefficient vector, the desired HR image patches can be obtained by:

$$y = D_h\alpha, \quad (7)$$

Due to noise, the reconstructed HR image Y_0 may not exactly project onto the assumed image acquisition model i.e., $X = SHY$. In order to fit into the imaging model accurately, a global reconstruction constraint is applied on Y_0 by solving the following optimization problem using gradient descent method:

$$Y = \arg \min_Y \|SHY - X\|_2^2 + C \|Y - Y_0\|_2^2 \quad (8)$$

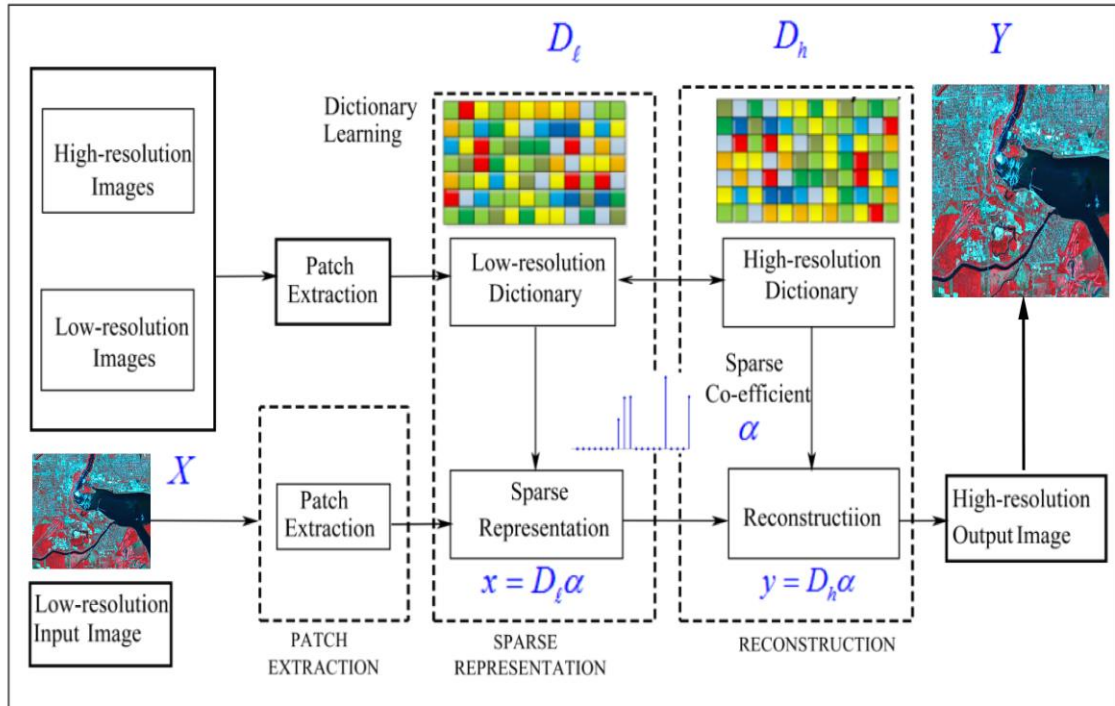


Figure 2: Block diagram of Sparse representation based Super-resolution for Multispectral Image

4.2 Dictionary Learning for Super-resolution:

Suppose $X_l = \{x_1, x_2, x_3, \dots, x_n\}$ and $Y_h = \{y_1, y_2, y_3, \dots, y_n\}$ are sets of related LR and HR multispectral images patches in the training dataset. Due to the ill-posed nature of super-resolution, this is a challenging task. Assuming that LR and HR MS images share the same sparse coefficients, individual HR and LR dictionaries can be learned by following minimizations:

$$D_h = \arg \min_{\{D_h, Z\}} \|Y_h - D_h Z\|_2^2 + l \|Z\|_1 \quad (9)$$

and

$$D_l = \arg \min_{\{D_l, Z\}} \|X_l - D_l Z\|_2^2 + l \|Z\|_1 \quad (10)$$

Instead of learning the dictionaries from LR and HR patches separately, a joint sparse coding problem for both the types of patches may be defined by combining the above equations as follows [9]:

$$\min_{\{D_h, D_l, Z\}} \frac{1}{P} \|Y_h - D_h Z\|_2^2 + \frac{1}{Q} \|X_l - D_l Z\|_2^2 + l \left(\frac{1}{P} + \frac{1}{Q} \right) \|Z\|_1, \quad (11)$$

where P and Q are the sizes of HR and LR image patch vectors, respectively. The l_1 norm term $\|Z\|_1$ enforces the uniform sparsity for both the dictionaries. The problem in

Eq. 11 is solved iteratively to obtain the dictionaries D_h and D_l . The example of HR and LR dictionaries are shown in Fig. 3.

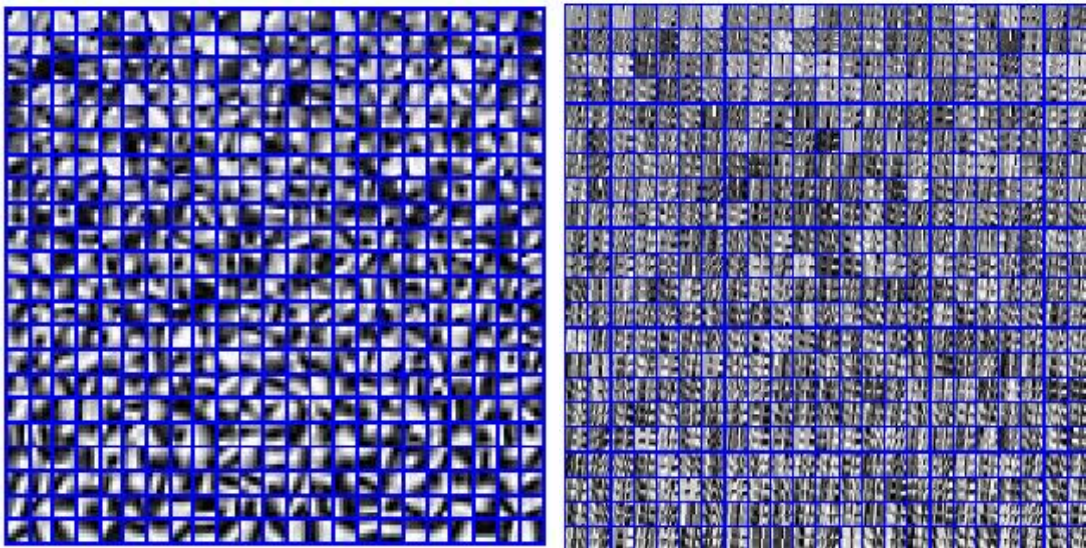


Figure 3: Example of HR (left side) and LR (right side) dictionaries

4.3 OpenMP-based Multicore Processing for Super-resolution

In recent years, the demand for high-computing systems have increased by manifold due to regular improvements, upgradations, and development of more computationally intensive applications. Increased clock speeds and, more recently, the addition of several processor cores on the same chip have been used to improve performance. Multicore processors are the latest way in which semiconductor firms are concentrating their efforts to improve processor performance. The data-intensive parts of sequential programme can be executed in parallel on multi-core processors. Multi-core processors have the advantage of consuming less resources and providing greater processing power as compared to single-core processors. Each core of the homogeneous multicore system has the same configuration and location. The Master/Slave model [40], the data flow model [41], and the OpenMP model [42] are the three most popular parallel programming models. OpenMP is now commonly used in general-purpose processors because of its scalability and versatility. The OpenMP implementations use a fork-join model. The master thread oversees executing the sequential pieces of a programme. When the master thread enters a parallel domain, it forks a group of worker threads that run in parallel with the master thread [33]. When the parallel section is over, the programme waits for all threads to finish before returning to single-threaded execution for another sequential section.

Figure 4 shows OpenMP model based on fork-join paradigm.

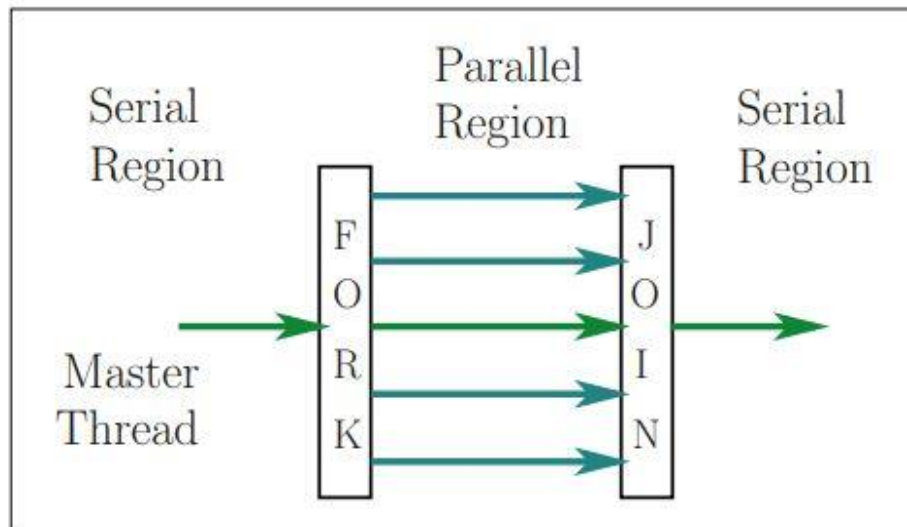


Figure 4. The fork-join paradigm of parallel computing

Since, single image SR (SISR) methods based on sparse representations are highly computationally intensive as they need to solve several regularization problems for dictionary learning as well as sparse reconstructions. Therefore, the computational overhead of SR algorithm is considerably very high in sequential manner. In order to reduce the time complexity of the SR algorithm, OpenMP based multicore techniques are one of the best parallel paradigms for real-time applications. Based on the fork-join model of parallelism, the basic idea of implementing parallel sparse based SR algorithm using multicore technique with OpenMP tools is shown in Fig. 5.

4.4 CUDA-GPU Parallel Processing for Super-resolution

Graphics processing unit (GPU) has become a very popular parallel computing device for engineering and scientific computing process since parallel processes are concurrently executed on hundreds of processor cores and thousands of threads. Since computing power of GPUs is highly intensive, they are much faster than the CPU. It is specially designed to effectively handle parallel processing required by graphics algorithms and video rendering. The computational overhead of the central processing unit (CPU) can be reduced by exploiting parallel computational power of GPU. The architecture comparison is shown in Fig. 6.

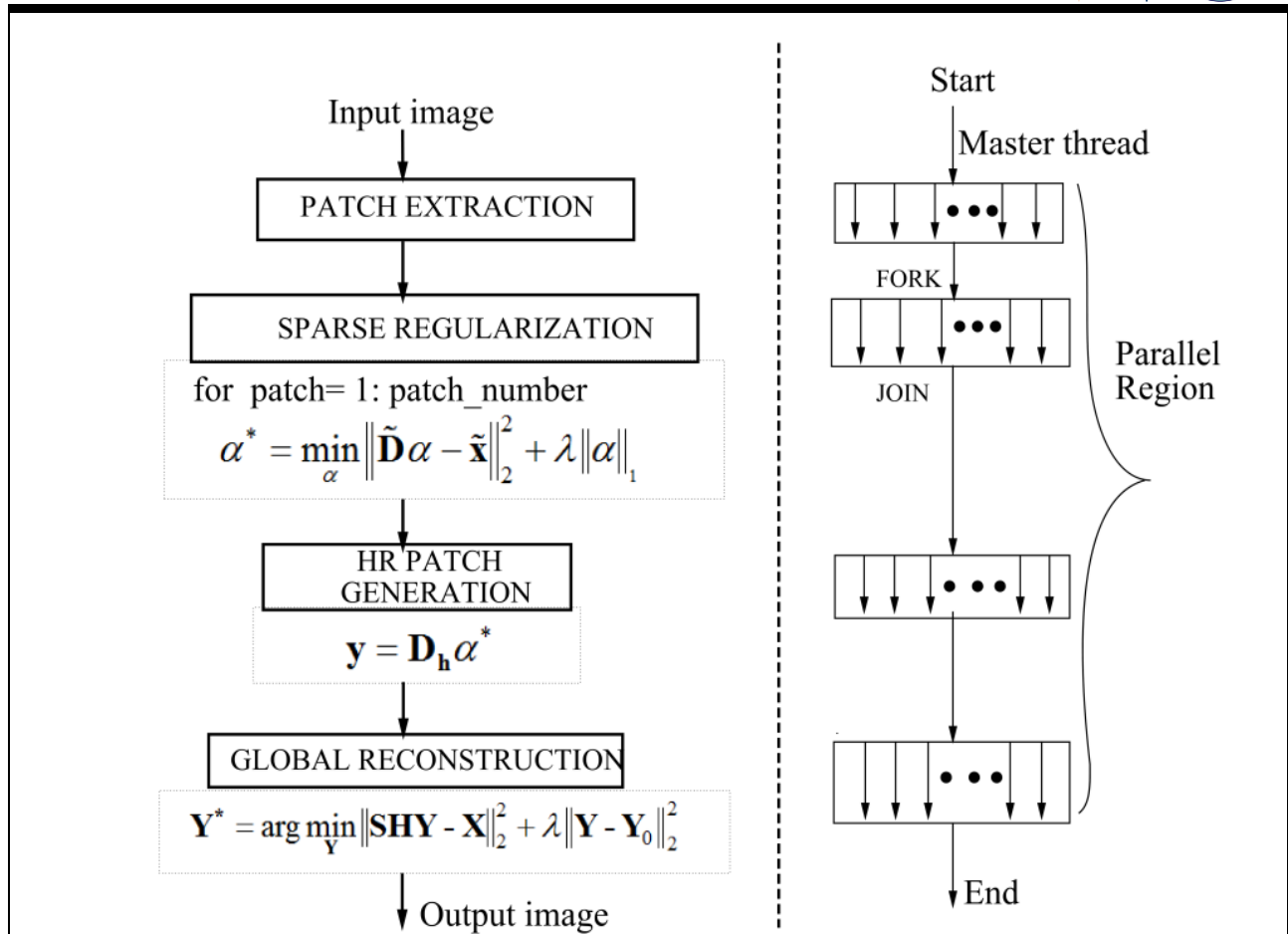


Figure 5: Multicore parallel SR algorithm

The computational power of GPU can be harnessed with the help of many applications program interfaces (APIs) without knowledge of graphics programming which allow users to boost the performance of any time consuming and computationally heavy algorithms. General-purpose programming on GPU (or GPGPU) has become very popular since the introduction of advanced programming environments, like CUDA [34], OpenCL [35], and DirectX compute shader [36]. It can also increase the computing efficiency of many applications by using existing hardware on end-user devices.

CUDA is a parallel programming platform introduced by NVIDIA in 2007. It is used to create software for graphics processors and to create a wide range of general-purpose applications for GPUs that are extremely parallel in design and run on hundreds of GPU processor cores. The CUDA API enables users to build a huge number of threads to run code on the GPU. A block is composed of several threads, which are indexed in the block using threadIdx. A grid is arranged in the same way,

and each block in a grid is indexed using “*blockIdx*”. “*ThreadIdx*” and “*blockIdx*” are both CUDA pre-defined variables.

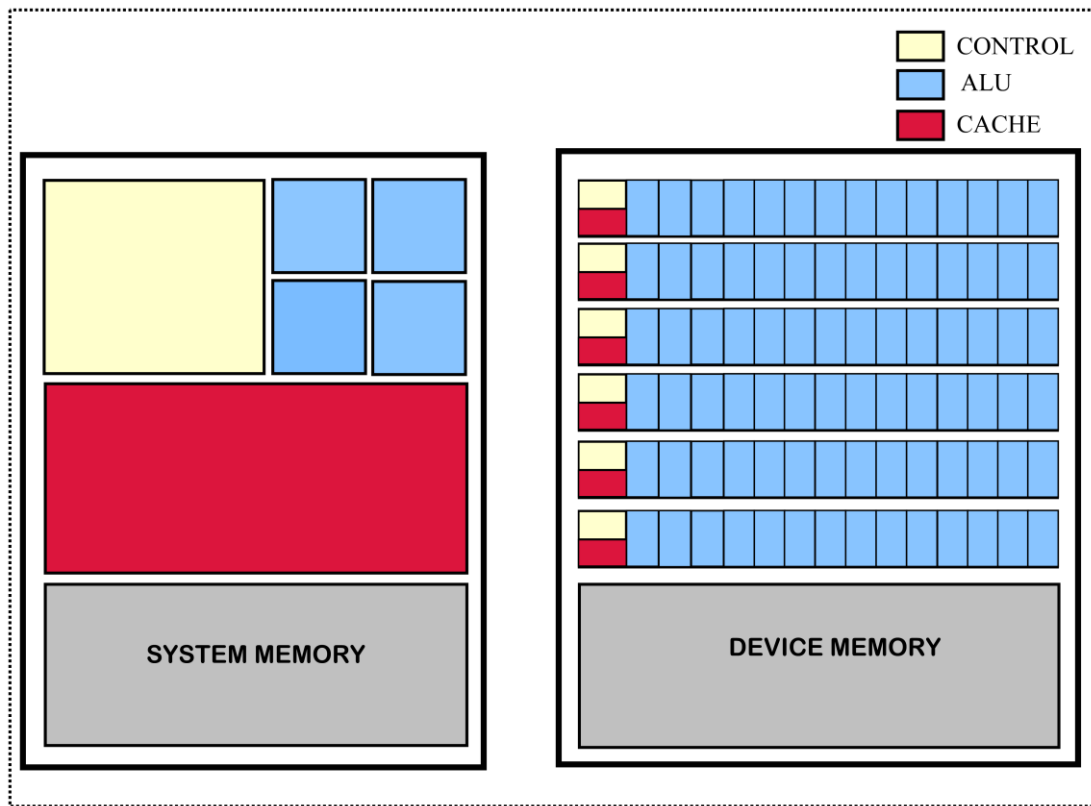


Figure 6: Comparison between CPU and GPU architecture

In addition, there are also two pre-defined variables “*blockDim*” and “*gridDim*”, which are used to specify the size of a block or grid determined by the total number of threads per block or the total number of blocks per grid. As seen in Fig. 6 all threads in CUDA are arranged [34] into a hierarchical way: block and grid. Kernels are specific function used in CUDA programming. A kernel is a function or a complete programme that is called by the CPU. It is run N times in parallel on the GPU using N threads. CUDA also supports shared memory and thread synchronization. The CUDA programming model combines serial and concurrent processing. Fig. 7 depicts an example of this type of heterogeneous programming. An ordinary CUDA programme consists of three steps: copying data from the CPU/host memory to the device's global memory, execution of CUDA codes in kernels, and restoring data from the device memory to the host memory. CUDA uses the bottom-up approach of parallelism, with a thread serving as an atomic unit of parallelism.

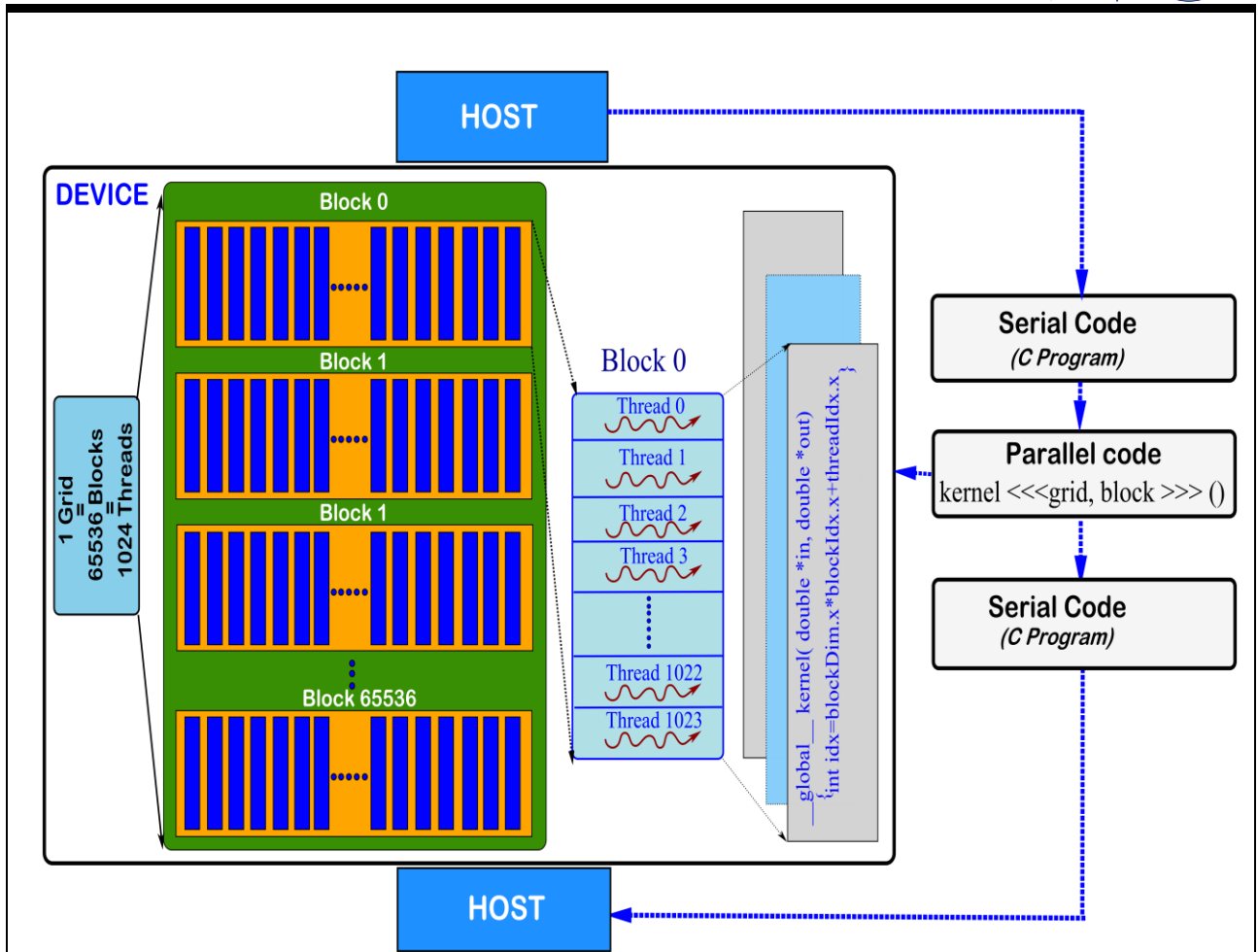


Figure 7: Heterogeneous architecture of CUDA and its three-level thread hierarchy

Due to inverse problem of sparse-based SR, it is computationally intensive to compute sparse co-efficients. Since, input image is divided into multiple feature patches by overlapping of one-pixel, super-resolution operation is applied on every patch independently. The processing of image patches is computationally very heavy and independent. Those two features make the algorithm appropriate to be implemented on GPU. Since CUDA can run the same code for different data in parallel, it is an excellent choice to process those image patches concurrently. The basic procedure for sparse representation-based SR using CUDA-enabled GPU is shown in Fig. 8.

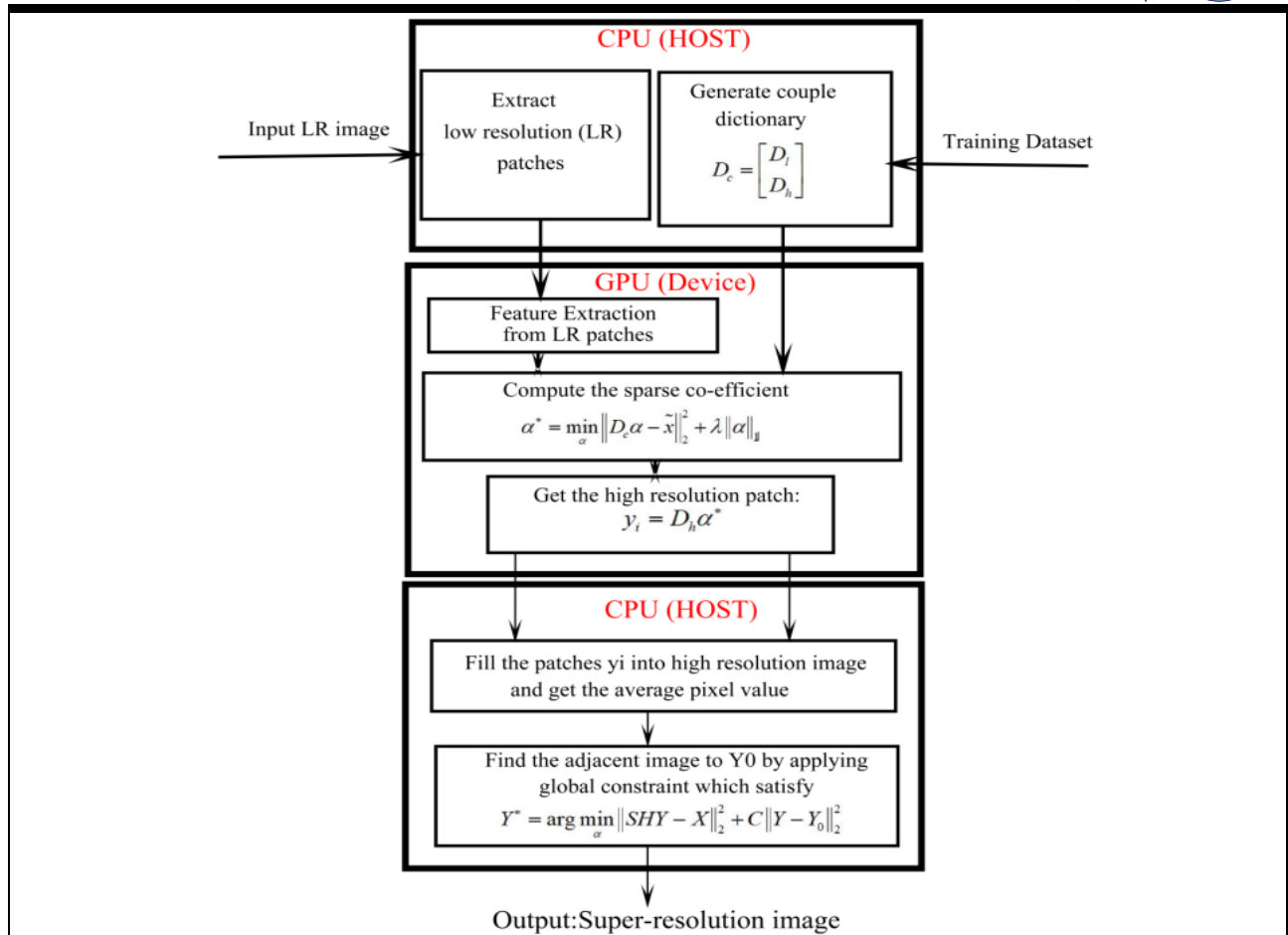


Figure 8: Procedure of Single Image Super-resolution based on GPU acceleration

5. Method of Analysis/Methodology

5.1 Design and implementation of MS image SR (MSISR) based on sparse representations and overcomplete dictionaries

5.1.1 Dictionary Training

A schematic of the dictionary learning procedure is detailed in Fig. 9. In this work, we learned two sparse overcomplete dictionaries by using the K-SVD dictionary learning approach on the training dataset that consists of 25 HR panchromatic and LR MS remote sensing images. HR dataset consisting of 10x10 size patches are extracted from panchromatic images, while feature-patches of size 10x10 are extracted from LR multispectral training images. Then on these two sets of patch vectors X^h and X^l , we apply the K-SVD algorithm [3]. Overcomplete Discrete cosine transform (DCT) is used to initialize LR and HR dictionaries D_l and D_h with 256 atoms. Dictionary learning is done iteratively until convergence, which solves a sparse regularization problem by applying the orthogonal matching pursuit

(OMP) algorithm. To train D_l and D_h from 50,000 patches, on an average, five K-SVD iterations are carried out. It takes approximately 3 hrs. for dictionary training.

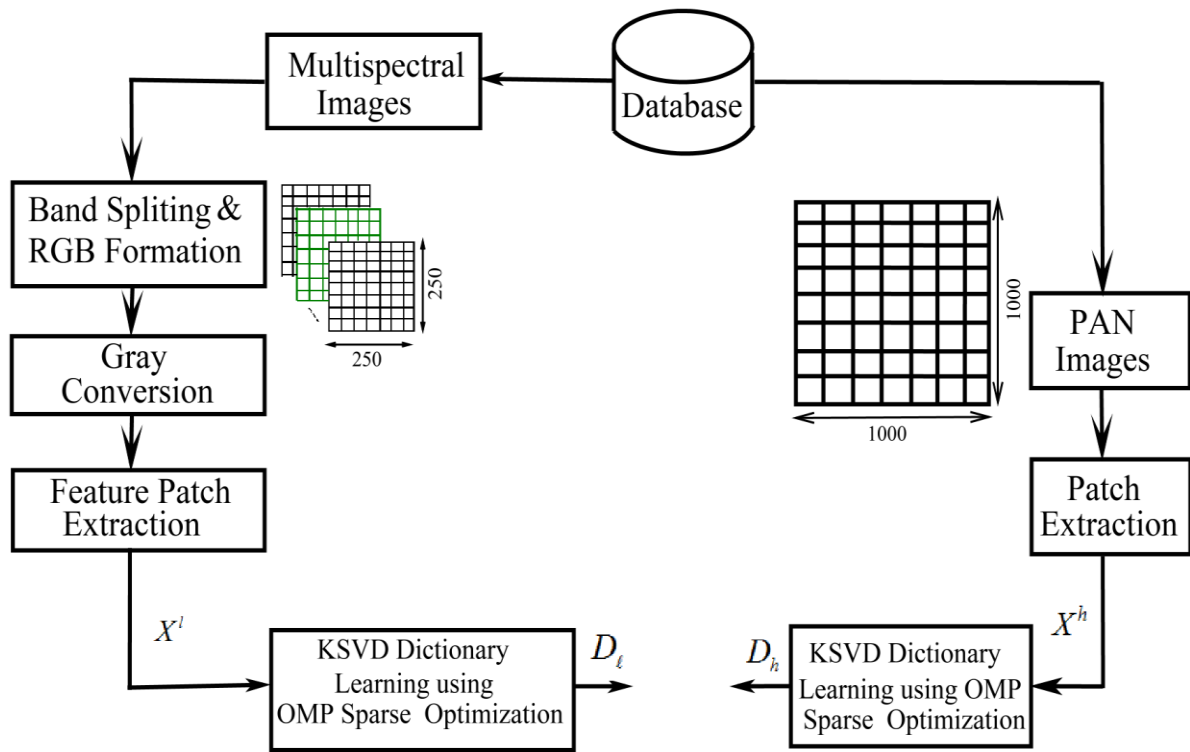


Figure 9: Schematic of the proposed dictionary learning procedure

5.1.2 Super-resolution Reconstruction

We developed a SISR algorithm-based on sparse representations for multispectral image super-resolution by considering each band of input LR MS image separately as shown in Fig. 10. For this, we split the four bands of QuickBird MS input image into RED, GREEN, BLUE and Near Infrared (NIR). Then each band is reconstructed by using pre-trained LR and HR dictionaries. In order fit into the imaging model, a global reconstruction constraint is applied on each reconstructed band by solving Eq. 4. Finally, the super-resolved MS is obtained by merging all the reconstructed MS bands. In this work, the sparse coding problem with the input MS band and LR dictionary D_l is solved by using an l_1 minimization based sparse optimization algorithm. The sequential time required for two times up-scaling of a 256 x 256 image is around 640 sec, whereas the same for a 128 x 128 image is around 154 sec.

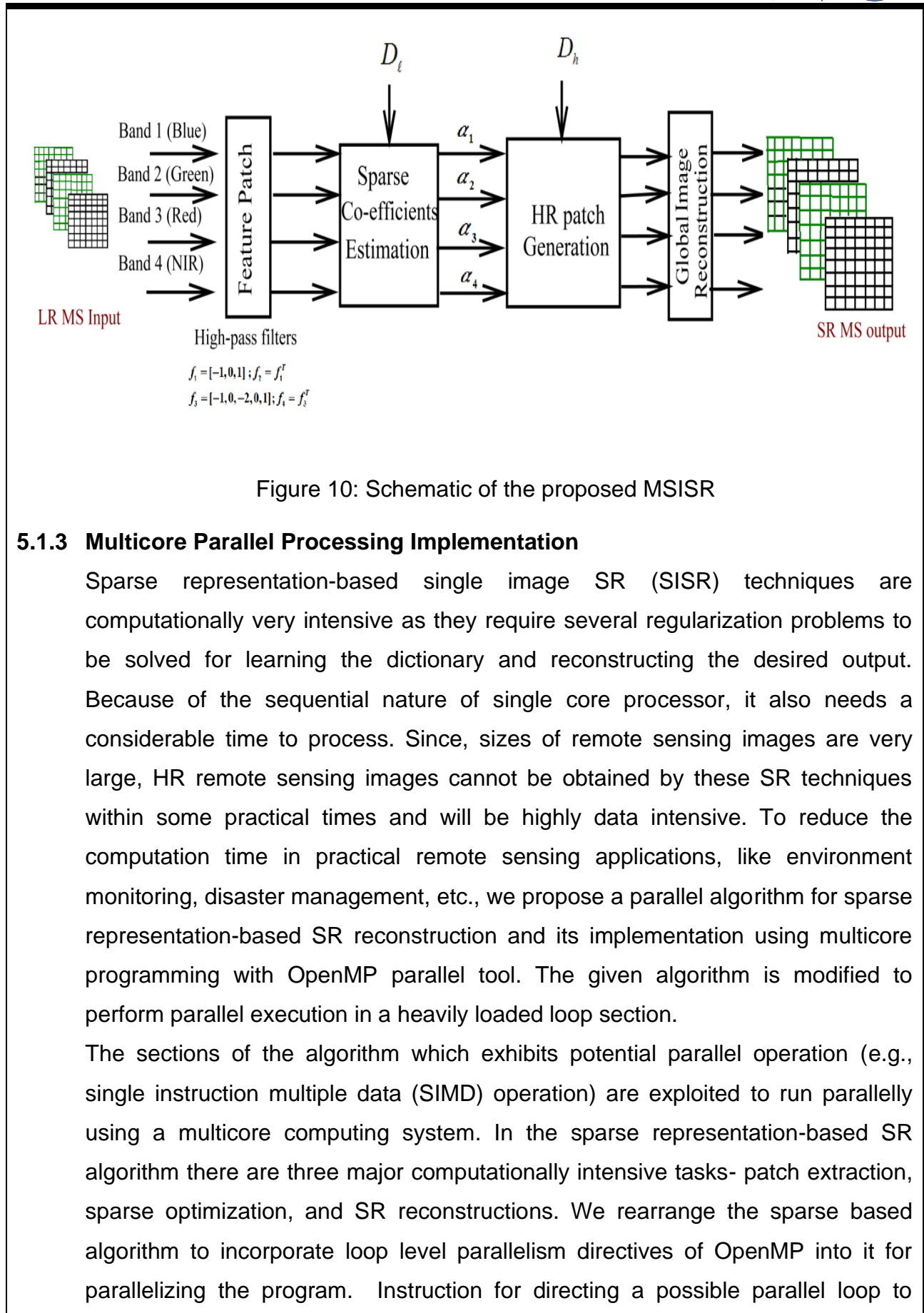


Figure 10: Schematic of the proposed MSISR

5.1.3 Multicore Parallel Processing Implementation

Sparse representation-based single image SR (SISR) techniques are computationally very intensive as they require several regularization problems to be solved for learning the dictionary and reconstructing the desired output. Because of the sequential nature of single core processor, it also needs a considerable time to process. Since, sizes of remote sensing images are very large, HR remote sensing images cannot be obtained by these SR techniques within some practical times and will be highly data intensive. To reduce the computation time in practical remote sensing applications, like environment monitoring, disaster management, etc., we propose a parallel algorithm for sparse representation-based SR reconstruction and its implementation using multicore programming with OpenMP parallel tool. The given algorithm is modified to perform parallel execution in a heavily loaded loop section.

The sections of the algorithm which exhibits potential parallel operation (e.g., single instruction multiple data (SIMD) operation) are exploited to run parallelly using a multicore computing system. In the sparse representation-based SR algorithm there are three major computationally intensive tasks- patch extraction, sparse optimization, and SR reconstructions. We rearrange the sparse based algorithm to incorporate loop level parallelism directives of OpenMP into it for parallelizing the program. Instruction for directing a possible parallel loop to

execute in multicore style is written as follows: #pragma omp <directive> [clauses]. Among different directives, the ‘parallel for’ directive provides loop level parallelism in a signal or image processing algorithm in an easy manner. The number of threads or cores among which the task is to be divided can be set during runtime by using omp_set_num_threads (integer). Algorithm 1 describes the implementation of a parallel sparse representation-based image SR (Parallel ScSR) algorithm.

Algorithm 1. Parallel ScSR Algorithm

Input: input image \mathbf{X} , dictionaries \mathbf{D}_l and \mathbf{D}_h

```

1: use #pragma omp parallel for
2: for each  $3 \times 3$  patch  $\mathbf{x}_i$  in  $\mathbf{X}$  do
3:    $z_i \leftarrow$  mean pixel value of  $\mathbf{x}_i$ ;
4:    $\mathbf{x}_i \leftarrow \mathbf{x}_i - z_i$ 
5: end for
6: use #pragma omp parallel for
7:  $xx \leftarrow$  patch indexes in  $x$ -dimension of SR image
8:  $yy \leftarrow$  patch indexes in  $y$ -dimension of SR image
9: for (int  $xx = 0$ ;  $xx < rows$ ;  $xx^{++}$ ) do
10:  for (int  $yy = 0$ ;  $yy < cols$ ;  $yy^{++}$ ) do
11:    $\alpha^* = \min_{\alpha} \|\mathbf{E}\mathbf{D}_l\alpha - \mathbf{E}\mathbf{x}\|_2^2 + \lambda \|\alpha\|_1$ 
12:  end for
13: end for
14: for each HR patch do
15:   $\mathbf{y}_i \leftarrow \mathbf{D}_h\alpha^*$ ;  $\mathbf{y}_i \leftarrow \mathbf{y}_i + z_i$ 
16: end for
17: use #pragma omp parallel for
18: for (int  $i = 0$ ;  $i < (maxIter)$ ;  $i^{++}$ ) do
19:   $\mathbf{Y}^* = \arg \min_{\mathbf{Y}} \|\mathbf{S}\mathbf{H}\mathbf{Y} - \mathbf{X}\|_2^2 + c \|\mathbf{Y} - \mathbf{Y}_0\|_2^2$ 
20: end for

```

Output: super-resolution image \mathbf{Y}^*

Sequential Loop	Parellelized Loop
<pre> for(int x = 0; x< input.rows; x++){ for(int y = 0; y< input.rows; y++){ output.at(x,y) = 255 - input.at(x,y); } } </pre>	<pre> # pragma omp parallel for for(int x = 0; x< input.rows; x++){ for(int y = 0; y< input.rows; y++){ output.at(x,y)= 255 - input.at(x,y); } } </pre>

5.2 Multicore Implementation of MS image SR based on MCA and Sparse Representation

5.2.1 MCA-based image decomposition

Morphological component analysis (MCA) is a technique for extracting textural features from images, which reveals the image's high-frequency information [1].

In an $\sqrt{N} \times \sqrt{N}$ image X , a texture component X_t is supposed to exist alongside a structure X_s . Two overcomplete dictionaries: $\{D_t, D_s\} \in R^{N \times L}$ establish sparse representations corresponding to X_t and X_s , separately [30], i.e.

$$X_t = D_t \alpha_t, \quad (12)$$

$$X_s = D_s \alpha_s, \quad (13)$$

where α_t and α_s are the texture and structure components' sparse coefficient vectors, respectively. Different transformations, such as the wavelet and the local DCT, are used to initialize dictionaries for the texture portion, while the wavelet, the curvelets, the contourlets, etc. are used for the structure part. We can write

$$X = X_t + X_s, \quad (14)$$

or,

$$X = D_t \alpha_t + D_s \alpha_s, \quad (15)$$

Over a joint dictionary containing both D_t and D_s , MCA finds an optimal sparse representation of X as follows:

$$\{\alpha_t^{opt}, \alpha_s^{opt}\} = \arg \min_{\{\alpha_t, \alpha_s\}} \|\alpha_t\| + \|\alpha_s\| \quad (16)$$

$$s.t. \|X - D_t \alpha_t - D_s \alpha_s\|_2^2 \leq \varepsilon$$

A total variation (TV) regularisation concept is applied to the unconstrained version of (16) for improved recovery of piecewise-smooth artefacts and pronounced edges, i.e.

$$\{\alpha_t^{opt}, \alpha_s^{opt}\} = \arg \min_{\{\alpha_t, \alpha_s\}} \|\alpha_t\| + \|\alpha_s\| + \lambda_1 \|X - D_t \alpha_t - D_s \alpha_s\|_2^2 + \gamma TV(D_s \alpha_s) \quad (17)$$

Equation (17) is a ℓ_1 -minimisation problem that is solved iteratively using the block-coordinate-relaxation algorithm [31] and a simultaneous sparse approximation and dictionary update method [37]. Here, ε is the error value,

which represents the consistency of the decomposed layers as they are approximated. An example of MCA decomposition of an input MS image is shown in Fig. 11.

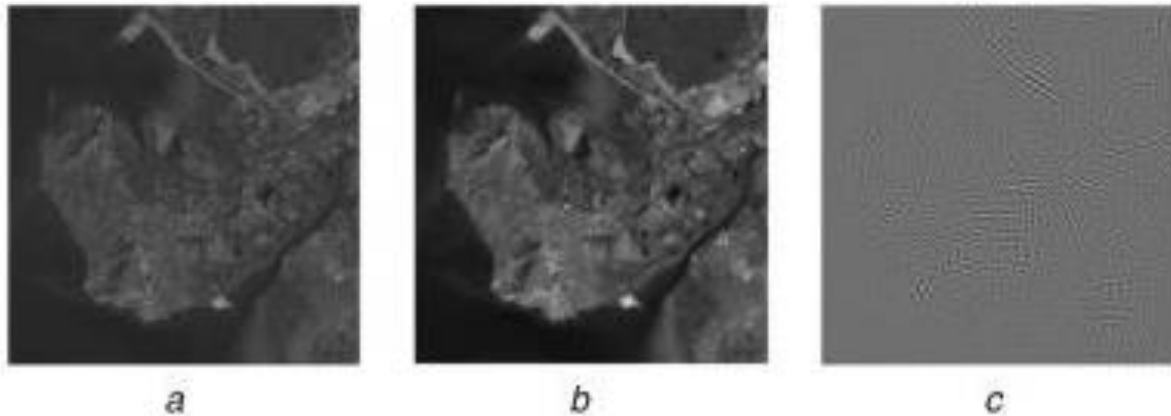


Figure 11: MCA decomposition results (a) Original image, (b) Structure part, (c) Texture part

5.2.2. Multi-spectral image dictionary learning

A schematic of the proposed MS image dictionary learning is presented in Fig. 10. By selecting regions of interests (Rois) with useful land cover features from ortho-rectified GeoTIFF photos, an image database consisting of cropped MS images of sizes 512×512 and 1024×1024 is developed. Each MS image is subjected to 2D PCA, after which the bands corresponding to the three highest principal components are chosen and converted to an RGB image. The three chosen channels are guaranteed to contain approximately 99% of the information in the actual MS image.

Now, to prepare two sets of training patch vectors: x^h and x^l , first, the grey-scale version of the RGB image is down-sampled and blurred to produce an LR image. The optimal upscale ratio for the final SR output of the scheme is equal to the down-sampling factor d . The grey-scale image is used to extract HR patch vectors x^h of size $n^2 \times 1$. LR patch vectors, on the other hand, are made in the following manner: using bi-cubic interpolation, the LR image is first up-scaled by 2 to produce an intermediate HR image. To extract high-frequency features from the bi-cubic interpolated image, first perform MCA on the bi-cubic image, which provides the image's structure and texture components. We use first- and second-order gradients in the horizontal and vertical directions on the texture part. As a result, four gradient maps are created, from which four LR feature

patches are extracted. Finally, they are concatenated to obtain the LR patch vector for training. Thus, the size of a LR feature patch x^l becomes $\left(\left(\frac{2n}{d} \times \frac{2n}{d}\right) \times 4 \times 1\right)$. We follow the same strategy for different zooms in factors. In [4], the authors learnt HR and LR dictionaries, namely, D_h and D_l from x^h and x^l , respectively. Both x^h and x^l are presumed to share a similar sparse representation vector z with their respective dictionaries in the sparse representation framework, as seen in (18).

$$\min_{\{D_h, D_l, Z\}} \frac{1}{H} \|x^h - D_h Z\|_2^2 + \frac{1}{L} \|x^l - D_l Z\|_2^2 + \lambda_2 \left(\frac{1}{H} + \frac{1}{L} \right) \|Z\|_1, \quad (18)$$

where H and L are the patch vector sizes for HR and LR, respectively, and λ_2 is the regularization parameter. By rearranging D_h and D_l into a joint dictionary, the above equation can be simplified. Similarly, x^h and x^l can be seen together. As a result, (18) can be rewritten as:

$$\min_{\{D_c, Z\}} \|x^c - D_c Z\|_2^2 + \lambda_2 \|Z\|_1, \quad (19)$$

where,

$$x^c = \begin{bmatrix} \frac{1}{\sqrt{H}} x^h \\ \frac{1}{\sqrt{L}} x^l \end{bmatrix} \quad \text{and} \quad D_c = \begin{bmatrix} \frac{1}{\sqrt{H}} D_h \\ \frac{1}{\sqrt{L}} D_l \end{bmatrix}$$

The K-SVD dictionary learning technique [32] is used to solve equation (19) since it solves the following two steps alternately.

- **Sparse coding step:** The dictionary is randomly initialized, and then it finds the sparse representation vector z_i for each patch vector x_i using this dictionary. This can be solved using a variety of sparse representation algorithms, such as greedy algorithms (e.g. matching pursuit (MP), OMP) or convex relaxation algorithms (e.g. BP).
- **Dictionary update step:** The next step is to update the dictionary using the sparse representations estimated above. There are many methods to choose

from, but the K-SVD-based approach is the most popular. Here, each column of dictionary d_i is updated for above-obtained sparse vector z_i by minimizing the error term $\|x_i - d_i z_i\|_2^2$ iteratively.

Since, it is simpler and can be trained with a wide number of sample patches, the K-SVD approach has benefits. A coupled K-SVD dictionary preparation for the PAN image database could be done in the same way as the training from MS images. For the PAN image, we can start using the proposed learning method at the patch extraction point, as seen in Fig. 12. PCA-based band reduction and RGB image conversion from MS images are not needed in this situation. Fig. 13 shows a visual representation of trained HR and LR dictionaries.

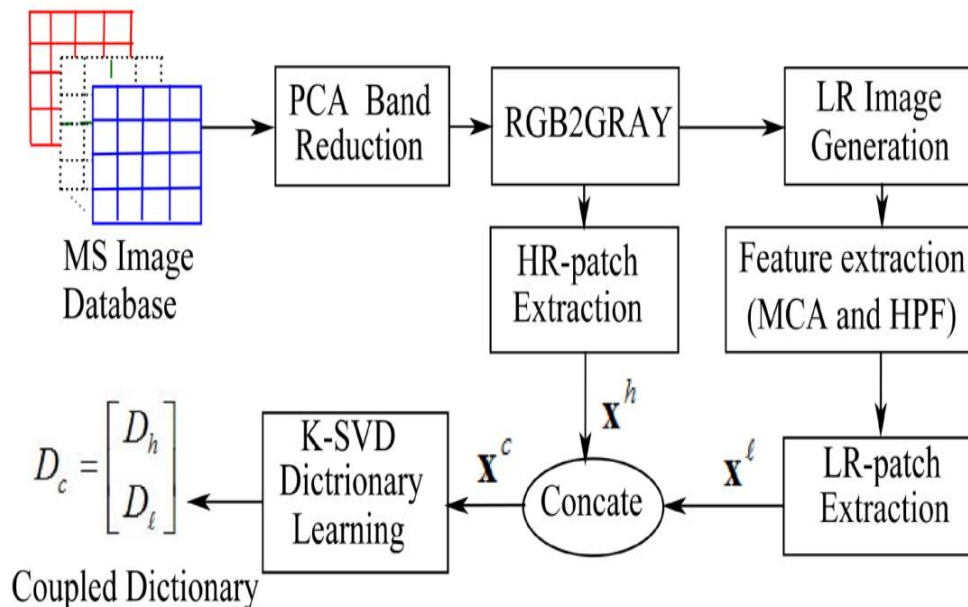


Figure 12: Multispectral image overcomplete dictionary learning

5.2.3 SR image reconstruction

The proposed approach solves two regularisation problems using the trained dictionary: one is to find a patch-wise sparse solution and the other is to approximate the super-resolved output using the image formation model as a global prior. Sparse reconstruction is done separately for each MS band. Each LR-MS band is applied to MCA separately to extract texture and cartoon components. The texture image ' X_t ' has a variety of high-frequency features, it favors sparse representation. The structure or cartoon image ' X_s ', on the other hand, includes low-frequency structural information that is upsampled to a size

equal to the size of the proposed super-resolved output using bicubic interpolation, which is comparatively simpler and reduces the method's overall computational cost.

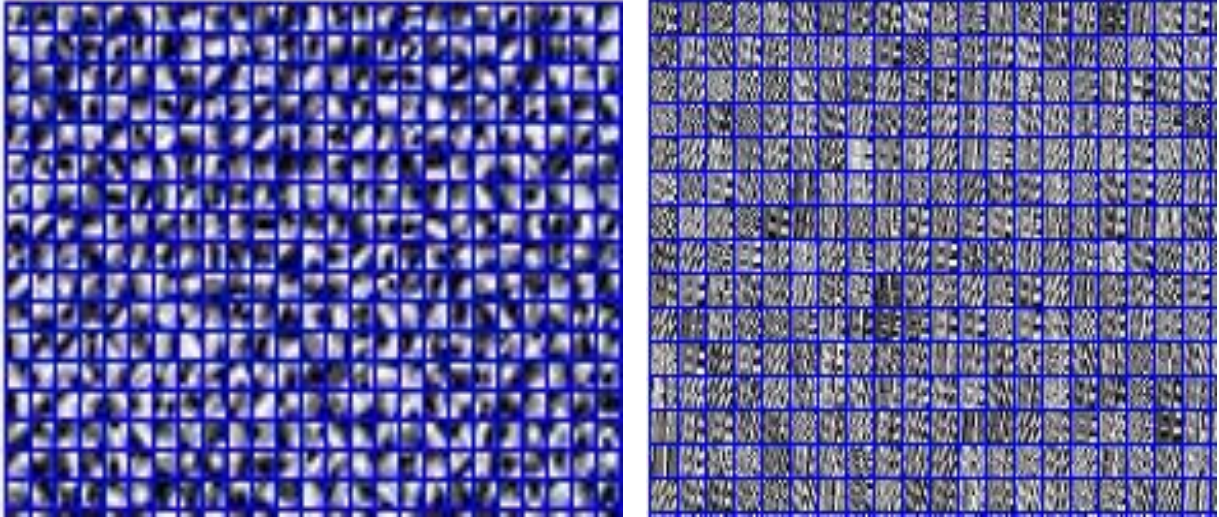


Figure 13: Representation patch images of the trained HR (left) and LR (right) dictionary

For the texture, ' X_t ' is first up scaled by 2 using bicubic interpolation after MCA decomposition. Upsampling before feature extraction is preferable because it is easier to create a correspondence between HR and up sampled LR image patches [14].

Now, an up sampled LR image is passed through 1D feature extraction filters of first and second orders, respectively to extract the high frequency features, as follows:

$$f = [-1, 0, 1] \quad (20)$$

$$s = [-2, -1, 0, 1, 2] \quad (21)$$

Linear convolution on X_t with the four filters f, f^T, s and s^T produces four filtered outputs; each of size equal to the size of X_t . Next, feature patches with single-pixel overlapping are extracted from each of the above outputs. To create a single feature vector x_t^ℓ , four feature patches corresponding to each pixel position of the filtered images are concatenated. As a result, each feature vector will be four times the size of the LR patch vector after upsampling. Finally, for sparse representation, the feature vector representing high-frequency information will be used.

The pre-trained over-complete dictionary is then used to represent a sparse representation problem, which can be written as follows:

$$\hat{\alpha} = \arg \min_{\alpha} \|\mathbf{D}_c \alpha - \mathbf{x}_t\|_2^2 + \lambda \|\alpha\|_1, \quad (22)$$

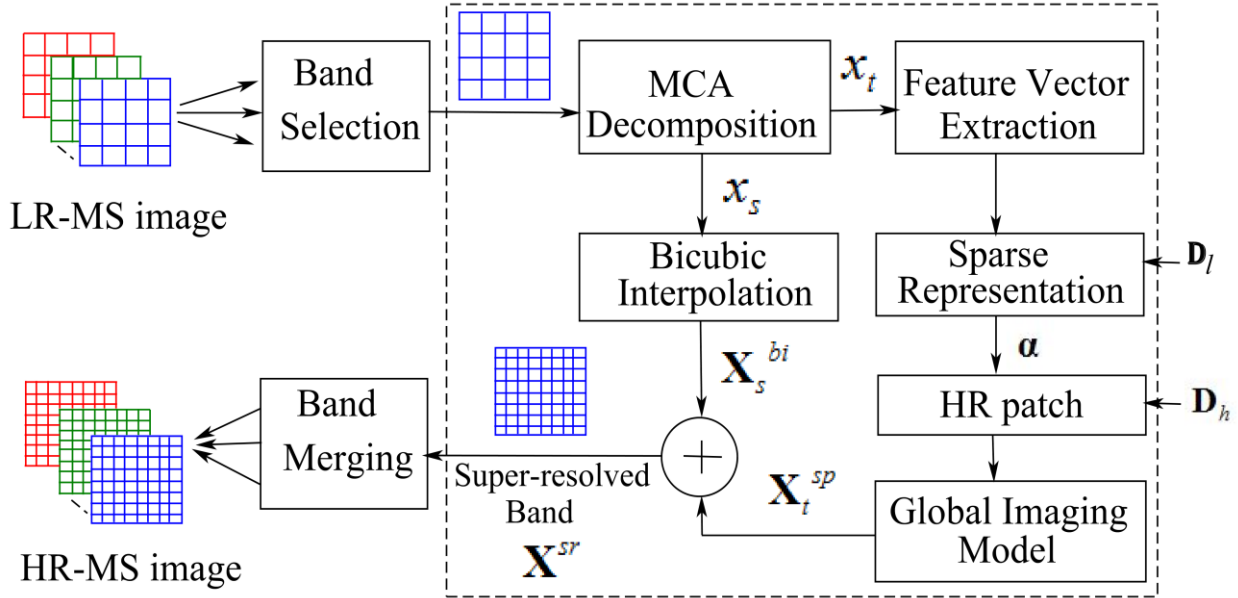


Figure 14: Proposed SR image reconstruction scheme

By multiplying with the HR dictionary D_h , the solution to (22) $\hat{\alpha}$ will be used to produce the corresponding HR patch vector x_t^ℓ :

$$\mathbf{x}_t^h = \mathbf{D}_h \hat{\alpha}, \quad (23)$$

Finally, to create an intermediate reconstructed image X_0 , reconstructed HR patches x_t^ℓ are positioned onto the target HR image map. Since, X_0 does not satisfy the imaging model perfectly due to noise or other reconstruction errors, the target HR image for the texture part X_t^{sp} is obtained by applying the global image constraint-based regularization as follows:

$$\mathbf{X}_t^{sp} = \arg \min_{\mathbf{X}_t^{sp}} \left\| SH\mathbf{X}_t^{sp} - \mathbf{X}_t \right\|_2^2 + c \left\| \mathbf{X}_t^{sp} - \mathbf{X}_0 \right\|_2^2, \quad (24)$$

where c is the regularization parameter. The target super-resolved image for an MS band X^{sr} is obtained by combining the solution of (24) with the super-resolved structural part obtained by MCA, as shown in Fig. 14. The target HR MS image can be created by combining individual super-resolved bands obtained as mentioned

above.

5.2.4 Parallel implementation of proposed algorithm

Computation is quite intensive in the K-SVD training algorithm, especially when the dimension of the dictionary increases, or the number of training image become large. In a sequential manner, it takes many hours to train the coupled dictionary using K-SVD. To reduce the computation time, implementation of the KSVD algorithm in a highly parallel environment is very much needed. Multicore parallel computing is one of the best choices for parallelizing the K-SVD algorithm. A parallel K-SVD based dictionary learning algorithm can be implemented based on the fork-join model of parallelism which utilizes loop level parallelism directives and library function from of OpenMP toolbox. In the OpenMP based multicore K-SVD algorithm, “for” loop operations are divided into multiple threads and execute each and all consequently. Thus, the computation time is reduced significantly.

5.3 CUDA-based GP-GPU implementation of MSISR based on adaptive dictionary learning and sparse representations

5.3.1 Parallel Adaptive dictionary learning for MS image SR in GPGPU:

In the sequential implementation, the two dictionaries were trained globally from a large database of MS images. Sparse representation-based SR method using global dictionary very much depends upon the HR image in the training database. Global dictionary might not be able to represent all the image patches accurately. More external images can be employed to learn the database, but the correctness of information yield by the training database for any LR input image cannot be guaranteed. The reconstructed HR may seem reasonable, but the information acquired from the training database may be irrelevant because the image patches of global dictionary are not currently available. To overcome the aforementioned limitations of global dictionary, adaptive dictionary has been broadly used to solve SR problem. In the adaptive dictionary, only the input LR image is used instead of external database, for dictionary training. It is assumed that many similar image patches may exist in the same LR image, both for the same and across different scales. Therefore, to make the dictionary learning method more effective and accurate for remote sensing applications, adaptive

dictionary technique is used where the LR/HR dictionaries have been learned and updated directly from all the overlapping patches of the given LR MS image. Once the dictionaries are trained, the SR MS image is reconstructed from its LR MS image by solving the sparse coding problem.

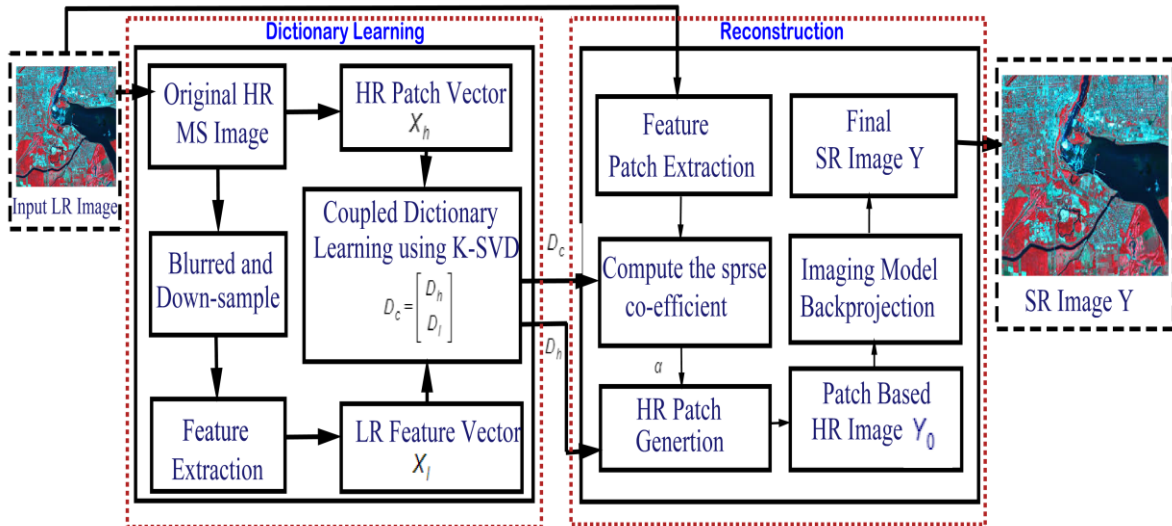


Figure 15: Block Diagram of proposed super-resolution method

The proposed adaptive dictionary learning-based MS image super-resolution method is shown in Fig. 15. A coupled overcomplete dictionary D_c is trained from the input test image itself. The adaptive dictionary learning is performed through the following steps-

- **Feature Extraction Stage:**

The original LR MS input image is blurred and down-sampled by zooming factor and then by using bicubic interpolation, the blurred and down-sampled LR image is resized again to its original size. Next, the feature extraction step is being applied on the degraded LR image to extract the high frequency components to improve the prediction accuracy. Four 1-D high pass filters from Eqs. 20 and 21 are used for feature extractions.

- **K-SVD dictionary training:**

LR and HR dictionaries are jointly learned by the coupled K-SVD [24] using the LR feature patches obtained from the feature extraction step and HR patches directly extracted from the original input image. The K-SVD dictionary learning has two iterative steps: firstly, a sparse coding of the signal is computed using the OMP by fixing the dictionary. Secondly, atom wise dictionary updating is done iteratively to simplify the updating step.

- **Parallel Implementation:**

K-SVD is a highly iterative process, its computation is very time consuming. Another time-expensive iterative process in K-SVD method is sparse coding computation using OMP. As OMP consists of many matrix/vector operations, such as matrix inverse and matrix-vector multiplication, it is very much suitable for parallel implementation on GPU platform. The computationally intensive operations of K-SVD are implemented on GPU environment by designing different kernels to reduce computational time.

The CUDA-OMP differs from the sequential implementation only by using interleaved arrays. The interleaved access pattern speeds up memory reads and writes in a massive-parallel environment by accessing sequential elements in threads or workitems at the same time. From the CUDA perspective, that means working with strided vectors, e.g., α_i is implemented as **alpha[j+i*m]** where m is the number of active threads and **j = blockIdx.x* blockDim.x + threadIdx.x** is the index of the current CUDA workitem. Each threads solves a separate OMP problem. The CUDA-GPU based OMP is summarised in Algorithm 1

The KSVD algorithm manipulates large training dataset; therefore, it is implemented on the host computer, while calling CUDA subroutines. In the algorithm below, the steps that are run on the device, are marked as “on device”, and the data that are stored on the device are marked with the “device” superscript. Some on-device routines are borrowed from the standard cuBLAS library in which case we denote the operation as “using cuBLAS”. The NVIDIA cuBLAS is a dedicated GPU implemented library for the standard basic linear algebra subroutines (BLAS). This library gives the user to access the computational resource of GPU. Algorithm 2 summarize the KSVD algorithm implemented in CUDA-GPU. The dictionary clearing subroutine is shown in Algorithm 3. In the algorithms discussed below, the notation $\|A\|_2^2 = \text{Diag}(A^T A)$ means the column-wise l_2 -norm, and $\|x\|_2 = \sqrt{\sum x_i^2}$ for a vector x. The $\partial(x)$ is defined as following:

$$\delta(x) = \begin{cases} 0, & \text{if } x = 0 \\ 1, & \text{if } x \neq 0 \end{cases}$$

Algorithm 1: OMP algorithm implemented in CUDA-GPU

Data:

n : the size of the input signal vectors
 m : the number of input data (=CUDA workitems)
 j : the current vector index (CUDA workitem), $0 \leq j \leq m - 1$
 S : the $n \times m$ matrix. Each column is a separate vector we want to sparse-encode
 w : the number of codewords
 T : the number of non-zero elements in a sparse representations
 D : the $n \times w$ dictionary matrix
 dtd : precomputed $D^T D$
 dtx : precomputed $D \leftarrow D^T S$

Result: $result_j$: the sparse vector of size w

$A \leftarrow \{\}$: the set of the selected codeword indexes, a bitmap;

$\alpha \leftarrow dtx_{,j}$;

$i \leftarrow 0$;

while $i < T$ **do**

$pos \leftarrow \operatorname{argmax}_k |\alpha_k|$;

$val \leftarrow \max_k |\alpha_k|$;

if $pos \in A$ **or** $val < \epsilon$ **then**

\perp break;

$A \leftarrow A \cup \{pos\}$;

 Cholesky update step:

if $i = 0$ **then**

$L_{0,0} \leftarrow 1$: the cholesky decomposition of $dtd_{\{k|k \in A\}, \{k|k \in A\}}$;

else

$t \leftarrow L \setminus dtd_{pos, \{k|k \in A\}}$: solve the triangle matrix by backsubstitution;

$L \leftarrow \begin{bmatrix} L \\ t \end{bmatrix}$: add the row t to the L ;

$s \leftarrow \sum_{k=0}^{i-1} L_{i,k}^2$;

if $1 - s < \epsilon$ **then**

\perp break;

$L_{i,i} \leftarrow \sqrt{1 - s}$;

$i \leftarrow i + 1$

 Perform orthogonal projection and compute sparse coefficients:

$t \leftarrow dtx_{\{k|k \in A\}, j}$;

$c \leftarrow LL^T \setminus t$: Cholesky solve;

$\alpha \leftarrow D^T S_{,j} - dtd_{-, \{k|k \in A\}} c$;

$result_{j, \{k|k \in A\}} \leftarrow c$;

Algorithm 2: CUDA-GPU based KSVD algorithm

Data:

n : the size of the training vectors
 m : the number of training vectors
 X : the $n \times m$ matrix of training data
 $iternum$: the number of iterations to run
 w : the number of codewords
 T : the number of non-zero elements in a sparse representations
 $useThreshold$: minimum number of usage of each codeword by the training set

Result: D : the $n \times w$ dictionary matrix

$D^{device} \leftarrow$ some initial normalized dictionary (e.g. random);

$iter \leftarrow 0$;

while $iter < iternum$ **do**

$dt^{device} \leftarrow D^T D$: using cuBLAS;

$\Gamma^{device} \leftarrow OMP(X, D, T)$: on device;

$p \leftarrow randperm(w)$;

$replacedAtoms \leftarrow \{\}$;

$j \leftarrow 0$;

while $j < w$ **do**

$ind^{device} \leftarrow \{k \mid \Gamma_{j,k} \neq 0\}$: on device;

$\gamma^{device} \leftarrow \Gamma_{j,ind}$: on device;

if $ind = \{\}$ **then**

 this atom is dead; replace with a signal with the worst MSE

$e \leftarrow \|D\Gamma - X\|_2^2$: on device;

$i \leftarrow \operatorname{argmax}_k e_k$;

$atom^{device} \leftarrow X_{:,i} / \|X_{:,i}\|_2$: on device;

$replacedAtoms \leftarrow replacedAtoms \cup \{p_j\}$;

else

$\beta^{device} \leftarrow \gamma\gamma^T$: on device;

$atom^{device} \leftarrow X_{:,ind}\gamma^T - D\Gamma_{:,ind}\gamma^T$: on device;

$atom^{device} \leftarrow atom + D_{:,j}\beta$: on device;

$atom^{device} \leftarrow atom / \|atom\|_2$;

$t^{device} \leftarrow atom^T D_j$: using cuBLAS;

$\gamma^{device} \leftarrow atom^T X_{:,ind} - t\Gamma_{:,ind} + t\gamma$: on device;

$\Gamma_{p_j,ind}^{device} \leftarrow \gamma$: on device;

$D_{:,p_j}^{device} \leftarrow atom$: on device;

$j \leftarrow j + 1$;

$errors^{device} \leftarrow \|D\Gamma - X\|_2^2$: on device;

 Clear the dictionary (see below);

$iter \leftarrow iter + 1$;

Algorithm 3: The dictionary cleansing subroutine

```

usesj ← ∑k=0m-1 δ(Γj,k) for 0 ≤ j ≤ w - 1 : on device;
j ← 0;
while j < w do
    if j ∈ replacedAtoms then
        if usesj ≥ useThreshold then
            Gdevice DT Dj : on device;
            Gjdevice ← 0; maxCorr ← maxk G;
            anotherJ ← argmaxk G;

            replace if there are highly correlated atoms
            the condition anotherJ ≥ j here is to prevent replacing both
            correlating atoms
            if |maxCorr| ≤ 0.99 or anotherJ ≥ j then
                return;
            end
        end
        end

        choose a signal with the most error value as a new atom
        replaceWith ← argmaxk errorsk;
        D-jdevice ← X-,replaceWith / ||X-,replaceWith||2 : on device;

        zero the error value to prevent using this signal as the new atom
        again
        errorsreplaceWith ← 0;
        replacedAtoms ← replacedAtoms ∪ {j};
    end
    j ← j + 1;
end

```

6 GPGPU parallel reconstruction of super-resolution MS image

In the SR reconstruction, input MS LR image is divided into multiple feature patches by overlapping of one pixel and each patch is processed independently. The processing of image patches for a whole image is computationally very heavy and independent. As the image size increases, the processing time of image patches is also increased. In such case, the parallel data structure characteristics of image patches is the key feature to exploit in the GPU platform for accelerating the computation time. Here, for harnessing the GPU acceleration, a parallel programming platform named CUDA, released by NVIDIA is adopted to achieve a fast SR reconstruction method for real-time applications. The proposed parallel SR image reconstruction method is schematically shown in Fig. 16.

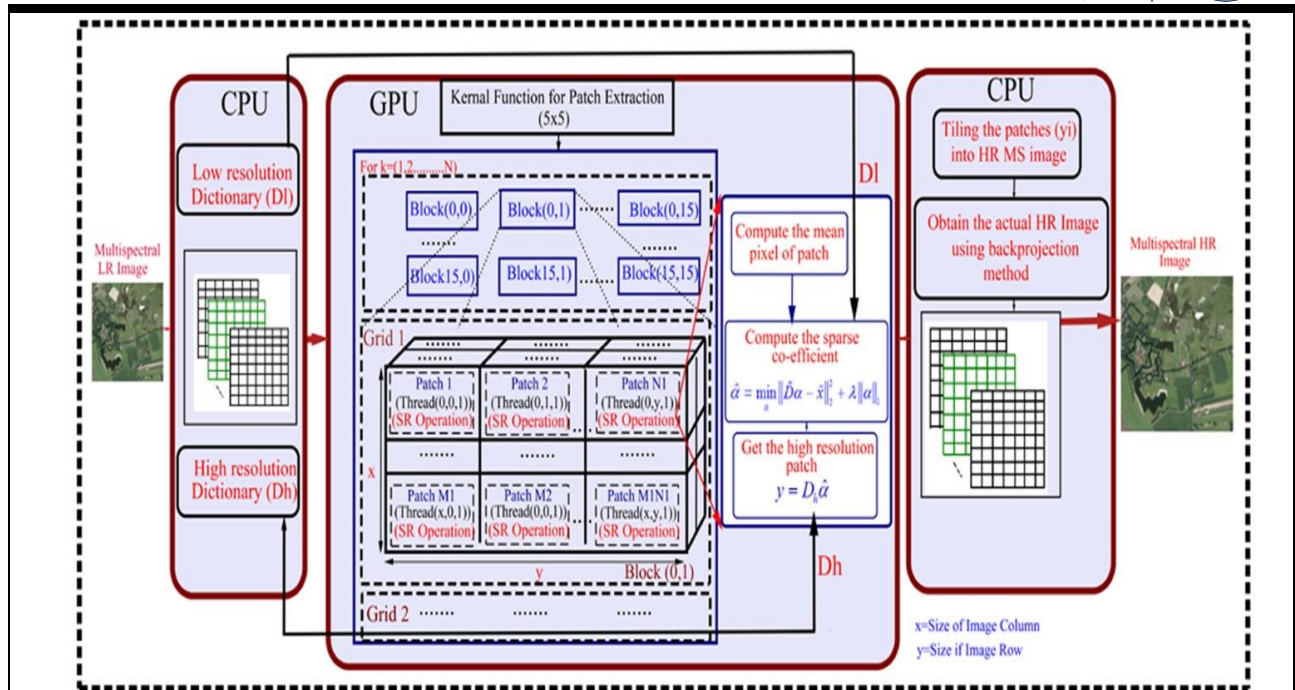


Figure 16: Schematic of proposed GPGPU SR image reconstruction method

Since, SR operation is applied on every image patch independently, so for each LR MS patch a sparse coefficient vector is calculated using the coupled dictionaries D_h and D_l learned using K-SVD technique as described in the previous dictionary learning section. Then a HR version of that patch is obtained by multiplying the sparse coefficients vector with HR dictionary D_h .

For CUDA-GPU based SR reconstruction, a kernel function is defined on the global memory to perform the SR operation on every patch in parallel. Since SR works patch-wise that offers fine-grained thread parallelism, each thread is qualified for processing a patch. Input MS image patches and the HR/LR dictionaries (D_h and D_l) are transferred from host (CPU) memory to global memory (GPU). Next, sparse coefficients for each LR patch are solved by LASSO that runs on an individual thread. Since, each HR and LR MS patches share the same sparse representation coefficients in the joint coupled dictionary pair, the desired HR multispectral image patches can be obtained by multiplying high resolution dictionary with sparse representation coefficient. Once all the HR patches are reconstructed, all these patches are transferred from global memory (GPU) to host memory (CPU). Finally, the desired HR image is obtained by tiling of all the HR image patches in the CPU. By applying back projection as given in Eq. 8, the final high resolution MS image is obtained.

Algorithm 4 GPU based Super-resolution via Sparse Representation for Multispectral Satellite Images

Input: Low resolution MS input image (X_k^{ms}) and training input pairs for dictionary training

Output: High resolution image Y_k^{ms*}

- 1: Prepare low resolution input patches and dictionaries (D_h and D_l)
Transfer the input patches and dictionaries from CPU to GPU
- 2: Define kernel function by using `_global_` specifier for processing the 5×5 patches parallelly (patch_size = 5)

– Initialization of index in GPU

int $t_{id1} = blockDim.x * blockDim.x + threadIdx.x$

int $t_{id2} = blockDim.y * blockDim.y + threadIdx.y$

– Check the boundary condition

if $t_{id1} + patch_size + 2$ and $t_{id2} + patch_size + 2$ are less than H (image height) and W (image width) then

1. Extract the feature and calculate the mean pixel value of the LR patches (\tilde{x}_i^{ms})

2. Solve the sparse co-efficient:

$$\alpha_k^* = \min_{\alpha_k} \|\tilde{D}\alpha_k - \tilde{x}_k^{ms}\|_2^2 + \lambda \|\alpha_k\|_1$$

3. Get the high resolution patch (y_k^{ms}):

$$y_k^{ms} = D_h \alpha_k^*$$

Transfer the estimated HR patches from GPU to CPU

- 3: Place the HR multispectral image patches y_k^{ms} into HR MS image
 - 4: Find the adjacent image to Y_0 which satisfies the reconstruction constraint
 $Y_k^{ms*} = \underset{Y}{\operatorname{argmin}} \|SHY_k^{ms} - X_k^{ms}\|_2^2 + C \|Y_k^{ms} - Y_{ok}^{ms}\|_2^2$
-

6. Results

E. Database preparation

F. QuickBird Multispectral (MS) Data collected from GLCF:

Multispectral satellite image data collected by QuickBird are taken via FTP from the Global Land Cover Facility (GLCF*). The images of various land cover types have been captured over the areas of Indonesia, Sri Lanka, Bangladesh and India during the period from August 2 to December 3, 2002. The QuickBird satellite data has a high-resolution panchromatic (PAN) image with spatial resolution of 0.7 meters and low resolution multispectral (MS) images in four bands (i.e., red, green, blue, and near infrared) with a spatial resolution of 2.8 meters.

*<http://glcf.umiacs.umd.edu/> (Last accessed on 20 March 2018)

a. MS Data Purchased from NRSC Data Center, ISRO:

MS datasets containing Resourcesat 2A LISS-III and LISS-IV images have been purchased from NRSC data center (<https://uops.nrsc.gov.in/>). The satellite Resourcesat 2 LISS-III provides four MS bands- three in visible near infrared (VNIR) and one in Short Wave Infrared (SWIR) with 23.5meters resolution. LISS-

IV sensor also provides same four bands with a spatial resolution of 5.8 meters. These images are downloaded via FTP. These Land cover type images captured during January 6, 2013, to April 10, 2018, over many places in India are available in the datasets.

b. Freely available MS Data collected from Bhuvan portal:

This dataset consisting of Resourcesat 2 (LISS-III) images have been freely downloaded from Bhuvan portal (<http://bhuvan.nrsc.gov.in/data/download/index.php>). These MS images were collected from various parts of India during the period November 24, 2009, to February 02, 2016.

G. Experimental Set-up

Experiments were performed on a server equipped with the Intel(R) Xeon(R) E5-2620V4 processor running at 2.1 GHz, and having 20 M cache, and 128 GB RAM. Further, it embeds an NVIDIA tesla P100 GPU card with pascal architecture and comprises of 3584 cores.

Table 1: Hardware specification of GPU

GPU Architecture	NVIDIA Pascal NVIDIA CUDA®
NVIDIA CUDA®	Cores 3584
Double-Precision Performance	4.7 TeraFLOPS
GPU Memory	16GB CoWoS HBM2 at 732 GB/s or 12GB CoWoS HBM2 at 549 GB/s
System Interface	PCIe Gen3
Max Power Consumption	250 W

H. Evaluations Parameters:

For quantitative and qualitative measurements, PSNR, MSSIM, SAM, ERGAS, Q-Index and sCC are used as objective evaluation criteria to measure the reconstruction quality of an image.

I. Peak signal-to-noise ratio (PSNR): PSNR can be calculated by using the following formula:

$$PSNR(dB) = 10 \log_{10} \frac{255^2}{MSE}$$

$$\text{where, } MSE = \frac{1}{M_w \times M_h} \sum_{i=1}^{M_w-1} \sum_{j=1}^{M_h-1} (r(i, j) - t(i, j))^2$$

where r is the ground truth HR image and t is the reconstructed HR image.

J. Structural similarity index measurement (MSSIM): SSIM is described by

$$SSIM = \frac{4\mu_r\mu_t\sigma_{rt}}{(\mu_r^2 + \mu_t^2)(\sigma_r^2 + \sigma_t^2)}$$

Where μ_r , μ_t and σ_r , σ_t represent the means and variances of ground truth and recovered HR images, respectively. σ_{rt} denotes the covariance of HR ground truth and recovered images.

Besides PSNR and MSSIM, we evaluated five other objective parameters for validation of the proposed MS image SR.

K. Spectral angle mapper (SAM): It is commonly used for spectral analysis of MS images and it calculates the average angle between the pixels of input (Y) and output image (X) using each band as a coordinate axis.

$$SAM(Y, X) = \frac{1}{N} \sum_i^N \cos^{-1} \left(\frac{Y_i X_i}{\|Y_i\| \|X_i\|} \right) \in [0, p]$$

where N is the total number of pixels in the image. The ideal value for SAM is 0.

L. ERGAS: For quality evaluation, it takes into account scaling factor and root mean square error (RMSE) values and is expressed as follows:

$$ERGAS(Y, X) = \frac{100}{S} \sqrt{\frac{1}{K} \sum_j^K \left(\frac{RMSE(Y^j, X^j)}{\bar{Y}^j} \right)^2} \in [0, \infty]$$

where S is the SR scaling factor ERGAS values of 0 indicate the best quality, while higher values indicate distortions in the reconstructed output.

M. Universal image quality index (Q-index): It gathers three properties in the image evaluation, which are correlation, luminance and contrast, respectively. In case of MS image it is computed as the average value of spectral band. Its value also lies in the range $[-1, 1]$. It is defined by:

$$Q(Y, X) = \sum_j^K \left(\frac{\overset{a}{s_{YX}} \overset{b}{2\bar{Y}\bar{X}} \overset{c}{2s_Y s_X}}{s_Y s_X (\bar{Y})^2 (\bar{X})^2 (s_Y)^2 (s_X)^2} \right) \in [-1, 1]$$

N. Spatial Correlation Coefficient (sCC): It measures linear relationship between edges of reference image to that of the reconstructed image. It is expressed as

$$sCC = \frac{\sigma_{XY}}{\sigma_X \sigma_Y}$$

where σ_{xy} denotes covariance between the ground truth (X) and the super-resolved image (Y). Similarly, σ_X and σ_Y represent standard deviations of X and Y, respectively. Its value lies in the range [-1,1].

O. Speed-up calculation:

For quantitative measurements, the speed-up S_t is calculated to compare the time taken by CPU (sequential) and GPU to reconstruct an image:

$$S_t = \frac{t_s}{t_p}$$

D. Results of CPU implementation of MSISR models based on dictionary learning and sparse representations

D.1 Results of MSISR based on sparse representations and overcomplete dictionaries

D.1.1 Performance Evaluations

Results of visual study and objective performance criteria are explained below. Output images are analyzed in terms of visual study as well as PSNR, MSSIM, SAM, Q-index and ERGAS for the multispectral image SR. We also analyzed the time complexity of both the experiments.

P. Visual Study:

a. SISR from RGB MS images

Super-resolution outputs for three different inputs Test1, Test2 and Test3 having resolutions of 128×128 , 256×256 and 512×512 are shown in Fig.17 for two times zooming. A small portion of the zoomed image is also shown superimposed on the output image in the left-hand side to highlight the visual quality of reconstruction¹. The dataset for this work is collected from [11].

b. SISR from four band MS images

The multispectral image SR based on sparse representation is tested on two MS low resolution images: a 256×256 MS land cover image collected from the area of Indonesia and Ujong and a 128×128 size MS image from

Sundarban, India. Visual quality of the output images of bicubic interpolation and proposed work are displayed in Figs. 18 and 19 below.

B. Objective Evaluations:

c. For SISR of RGB images

PSNR and MSSIM are shown for the reconstructed images in Table 2. The proposed method is able to give better results in terms of PSNR and MSSIM for all the test images compared to bicubic interpolation¹.

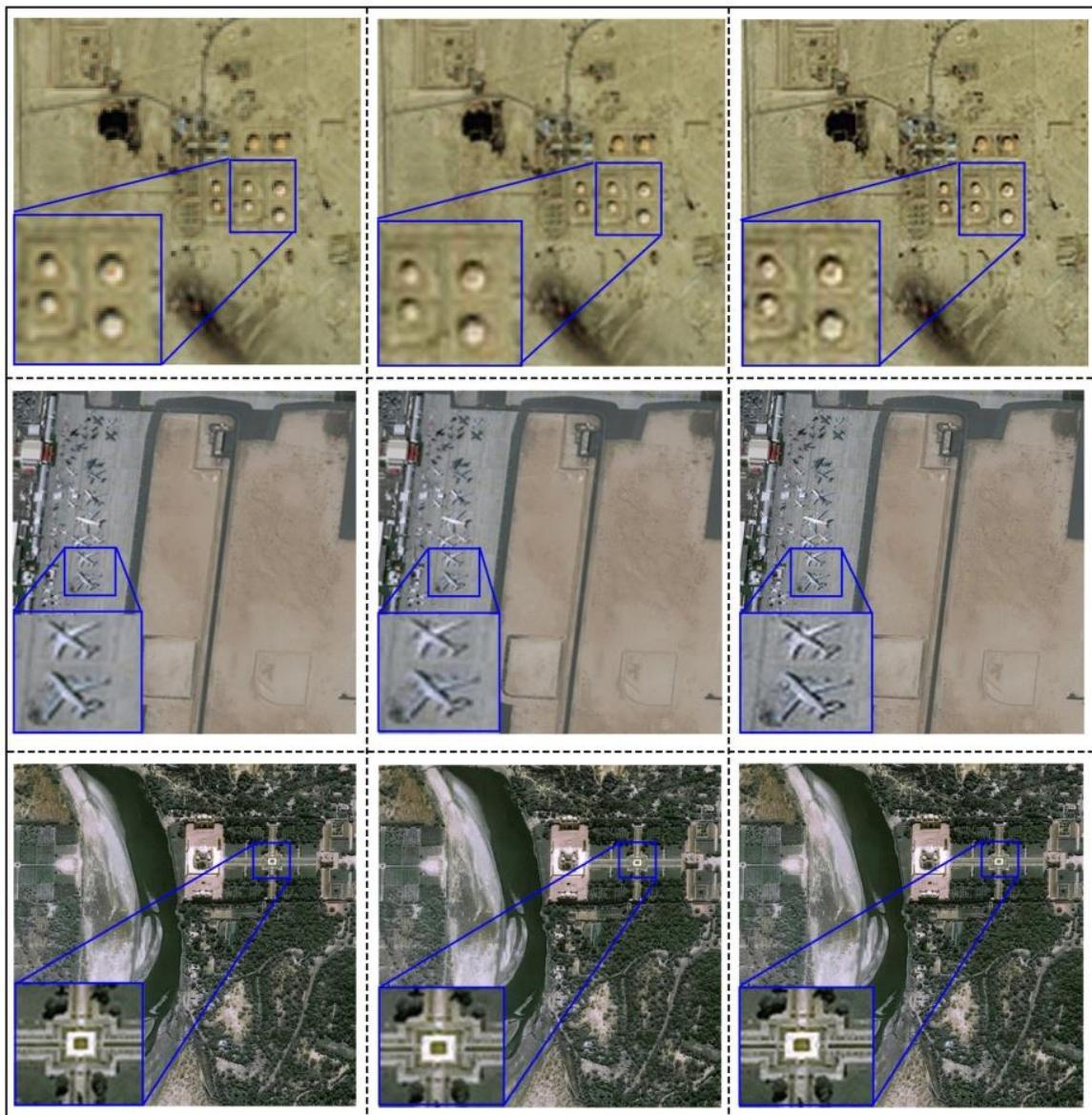


Figure 17: Results for a zooming factor of 2. First row left to right: Test1 and results of bicubic and the proposed method, respectively. Second row left to right: Test2 and results of bicubic and the proposed method. Third row left to right: Test3 and results

of bicubic and the proposed method, respectively.

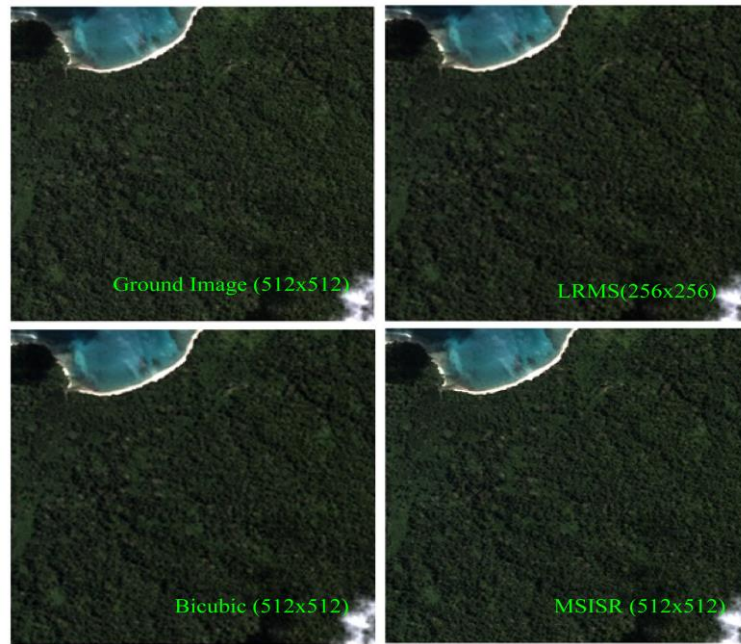


Figure 18: Results for a zooming factor of 2 of QuickBird MS land cover image of the area Indonesia and Ujong. First row left to right: ground image and input LR MS image of half size. Second row left to right: Results of bicubic and the proposed method.

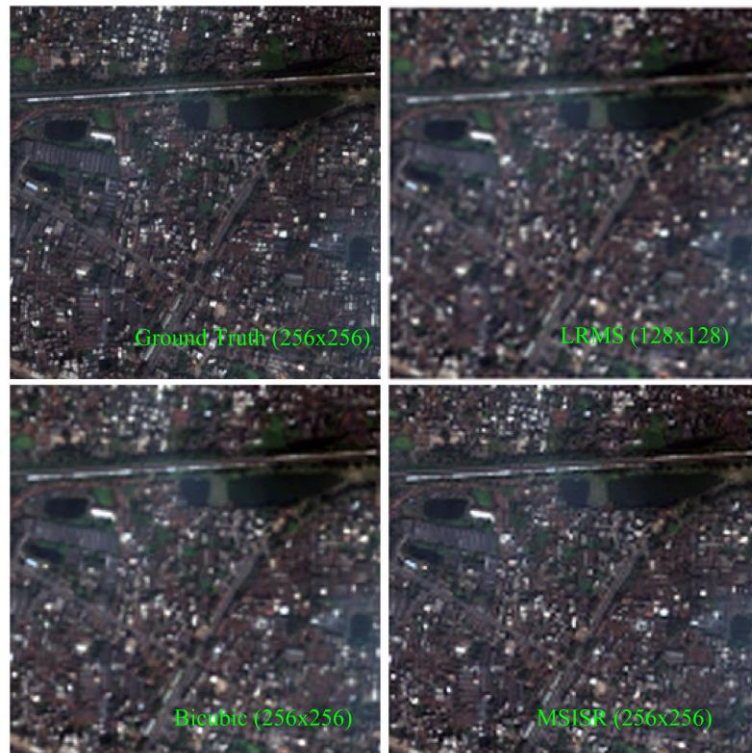


Figure 19: Results for a zooming factor of 2 of QuickBird MS land cover image of the area Sunderland, India. First row left to right: ground image and input LR MS image of

half size. Second row left to right: Results of bicubic and the proposed method¹.

Table 2: PSNR and MSSIM values of test images for different upscale factor

Input	PSNR (in dB)				MSSIM			
	ZOOM=2		ZOOM=4		ZOOM=2		ZOOM=4	
	Bicubic	Proposed	Bicubic	Proposed	Bicubic	Proposed	Bicubic	Proposed
Test-1	29.30	30.16	26.57	26.83	0.756	0.805	0.740	0.778
Test-2	29.31	30.33	27.03	27.36	0.956	0.968	0.945	0.953
Test-3	30.40	31.76	28.69	29.21	0.969	0.979	0.946	0.955

d. For SISR of MS images:

Here, besides PSNR and MSSIM, we evaluated three other objective parameters, i.e. Q-index, SAM, ERGAS parameters for validation of MSSISR.

Table 3: Performance comparison in terms of objective parameters

Indexes		Experiment 1 Input= 256x256; Output=512x512			Experiment 2 Input= 128x128; Output=256x256		
		Bicubic	MSISR	SPARSFI	Bicubic	MSISR	SPARSFI
PSNR	R	30.01	30.92	30.83	25.60	26.67	26.66
	G	25.83	26.77	26.77	26.62	27.54	27.54
	B	34.01	34.72	34.65	33.44	34.14	34.15
	N	10.07	10.99	10.99	18.82	19.66	19.65
	RGB	29.40	30.42	30.39	28.44	29.38	29.32
SSIM	R	0.973	0.994	0.994	0.807	0.853	0.853
	G	0.968	0.993	0.993	0.796	0.849	0.849
	B	0.971	0.992	0.992	0.840	0.872	0.872
	N	0.961	0.993	0.993	0.798	0.850	0.849
	RGB	0.971	0.993	0.994	0.810	0.859	0.859
Q-index		0.823	0.845	0.885	0.960	0.988	0.992
SAM		1.333	1.254	1.250	0.779	0.767	0.767
ERGAS		1.543	1.390	1.400	1.23	1.11	1.124

e. Related Publication:

¹H. U. Mullah and B. Deka, "A Fast Satellite Image Super-Resolution Technique Using Multicore Processing", *Advances in Intelligent Systems and Computing*, Springer, vol. 734, 2018.

D.2 Results of MSISR based on MCA and sparse representations

Experiments are performed to obtain the SR of multispectral images for different zooming value with a trained coupled dictionary of size 256. Quantitative evaluations are done for comparisons of results. Moreover, the spectral properties of the reconstructed images are also examined.

D.2.1 Datasets :

Three multispectral image databases are considered for experiments which are QuickBird, LISS-III and LISS-IV respectively, which band information is as mentioned in Table I. Results are shown for three test images shown in Fig. 20 below².

Table 4: Details of MS image datasets

Satellite	LISS-III	LISS-IV	QuickBird
Band Information	B2: 0.52 - 0.59 B3: 0.62 - 0.68 B4: 0.77 - 0.86 B5: 1.55 - 1.70	B2: 0.52 - 0.59 B3: 0.62 - 0.68 B4: 0.77 - 0.86	Pan: 0.45-0.90 Blue: 0.45-0.52 Green: 0.52-0.60 Red: 0.63-0.69 NIR: 0.76-0.90
Swath	140 Km	23 Km	16.8 Km
Spatial-resolution	23.5 m	5.8 m	Pan: 0.65 m MS: 2.62 m



Figure 20: Test images: from left to right: QuickBird, LISS-IV, LISS-III

D.2.2 Simulation Results:

A. Super-resolution of QuickBird MS image:

QuickBird takes high-resolution MS images, which are extremely useful for assessing land cover. This experiment's dataset is multispectral as seen in Table 4.

It has three visible bands, blue, green, and red, as well as one near-infrared (NIR) band with a resolution of 2.8 m. The training dataset contains 25 distinctive MS images from areas such as Chilika Lake, the Sundarbans, Ujong Kulon, Yala, and others. Fig. 20 shows a few examples of images from this dataset. LR and HR patches of size 2×2 and 8×8 , respectively, are extracted from the dataset and train HR and LR dictionaries of size 64×256 . The ground truth for reconstruction is a 256×256 MS image with a resolution of 2.8 m (capturing a scene over the India-Sundarban region passed by the satellite on November 2, 2002). We use the given LR-MS image as the ground-truth since we do not have an HR-MS image. The LR MS test image is obtained by applying a Gaussian low-pass filter (LPF) of size 5 and standard deviation = 0.5, followed by a factor of 4 down-sampling, as seen in Fig. 21.

B. Super-resolution of LISS-IV image:

This experiment's dataset is multi-spectral, as seen in Table 4. LISS-IV MS data has a resolution of 5.8 m and consists of two visible bands (B2 and B4) and one NIR band (B3). The coverage swath is 70 km, and bands are quantized using 10 bits. The training dataset consists of 30 images of various sizes: 256×256 , 512×512 , and 1024×1024 . Region of Interest (RoI) can be chosen so that it includes homogeneous regions in each band when selecting training images. This means that each of the training images has high feature content and therefore facilitates improved dictionary learning. To learn HR and LR dictionaries, 50,000 sample patches are chosen from the training dataset for this experiment. Fig. 20 depicts a visual representation of the above-mentioned dataset's trained HR and LR dictionaries. We consider extracting patches of size 5×5 from both HR and upsampled LR MS images, which results in LR and HR dictionaries of sizes 25×256 and 100×256 , respectively. As seen in Fig. 21, a 512×512 size ground-truth image is obtained from a 5.8 m resolution LISS-IV earth observation image passing over an area of India on March 18, 2017. We generate test LR MS images for reconstruction by blurring and downsampling each band of the input LISS-IV MS image. The results of SR reconstruction of test LR MS images are compared to the ground-truth.

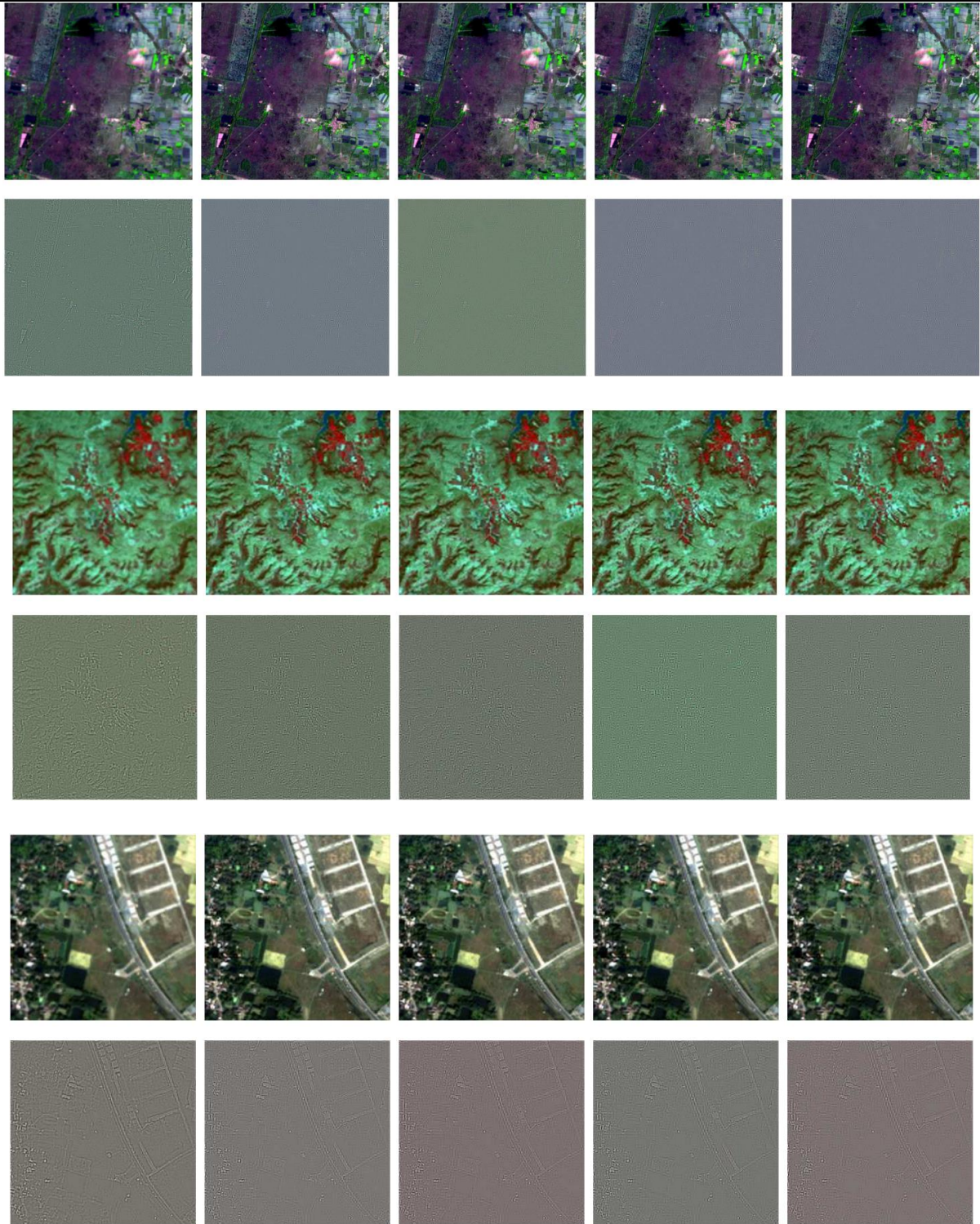


Figure 21: Visual outputs of different methods and the error images with ground truth: from top to bottom: first and second row are LISS-IV results for 2 times zooming, third and fourth row are LISS-III results for 3 times zooming, fifth and sixth rows are QuickBird results for 4 times zooming. Again, from left to right: Bicubic method, SparseFI, Yang et al. method, Moustafa et al method and the proposed method.

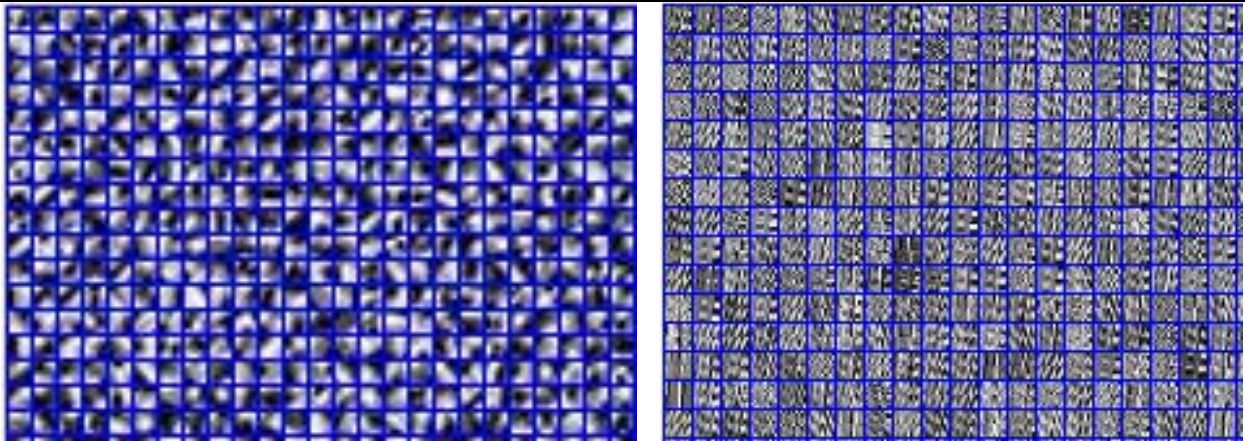


Figure 22: Representation patch images of the trained HR (left) and LR (right) dictionary

Table 5: Evaluation of parameters for the three test images

Image	Method	PSNR	MSSIM	ERGAS	SAM	UIQI	sCC	NIGE
QuickBird	Bi-cubic	25.32	0.8015	7.663	4.438	0.7766	0.9690	17.92
	SparseFl	26.46	0.8402	6.714	4.296	0.8274	0.9761	13.10
	Yang et al.	26.47	0.8499	6.762	4.280	0.8305	0.9767	14.0
	Moustafa et al.	26.52	0.8496	6.694	4.284	0.8298	0.9764	13.56
	Chen et al.	26.39	0.8290	7.149	4.328	0.8080	0.9740	13.7
	Lucas et al.	26.18	0.8070	7.477	4.423	0.7830	0.9700	13.41
	Proposed	26.60	0.8505	6.611	4.275	0.8312	0.9772	12.96
LISS-IV	Bi-cubic	30.11	0.9694	5.492	3.316	0.8454	0.9818	15.79
	SparseFl	32.54	0.9968	4.205	2.726	0.9135	0.9892	14.11
	Yang et al.	31.37	0.9835	4.748	3.310	0.8783	0.9862	14.80
	Moustafa et al.	32.48	0.9937	4.233	2.769	0.9073	0.9892	14.25
	Chen et al.	32.35	0.9910	4.649	2.925	0.8900	0.9870	12.90
	Lucas et al.	32.45	0.9920	4.827	3.027	0.8870	0.9860	12.44
	Proposed	32.76	0.9974	4.106	2.663	0.9181	0.9897	12.91
LISS-III	Bi-cubic	25.95	0.8108	7.724	4.023	0.7915	0.9454	18.06
	SparseFl	26.54	0.8414	7.082	3.917	0.8273	0.9545	15.23
	Yang et al.	26.88	0.8607	6.787	3.859	0.8402	0.9570	14.96
	Moustafa et al.	27.10	0.8636	6.764	3.614	0.8491	0.9586	13.56
	Chen et al.	27.07	0.8510	7.014	3.683	0.8360	0.9550	13.89
	Lucas et al.	26.96	0.8500	7.091	3.757	0.8350	0.9540	12.01
	Proposed	27.24	0.8719	6.663	3.567	0.8601	0.9590	12.41
	Ideal value	High	1	0	0	High	High	Low

C. Super-resolution of LISS-III MS image:

The dataset of this experiment is multi-spectral as referred in Table 5. LISS-III images have a medium resolution (i.e., 23.5 m) and four spectral bands: three visible (B2, B3, and B4) and one near-infrared (NIR) (B5). The swath coverage is

140 km, and each band is quantized with 10 bits. We create a training dataset of 25 MS images, which are chosen in the same manner as described for LISS-IV.

These images are regarded as HR data, from which LR images for learning HR and LR dictionaries for our experiments are produced. We considered extracting HR-patches of size 9×9 with 3 pixels overlap and LR-patches of size 3×3 with 1 pixel overlap to obtain HR and LR dictionaries of sizes 81×256 and 144×256 , respectively. For reconstruction, an image of size 510×510 with a resolution of 23.5 m is obtained from an original LISS-III image containing a scene over the area of Giri Forest, India, and with a pass date of 8 January 2017.

D. Super-resolution of panchromatic image:

This experiment's dataset is Panchromatic, as seen in Table 5. Consider a monochrome CARTOSAT-2 PAN image with a swath coverage of 9.6 km and a spatial resolution of about 0.65 m. We choose 20 HR-PAN ROI images with homogeneous information for learning the dictionary, as we do in MS image SR.

LR versions of each of these images are created using the same method as the LR-MS test data. PAN image SR essentially applies the SISR algorithm to each band of the MS image during SR. A ground-truth image of size 512×512 is obtained from the CARTOSAT-2B satellite of ISRO passing over the area, Mumbai, India on 10 December 2015. Blurring and downsampling are used to generate the LR test image of scale 256×256 . Fig. 21 depicts the SR findings. On an average, the proposed method for PAN image SR improves PSNR by around 1.4 dB over bi-cubic performance, while Yang's method and Chen's method improve PSNR by 0.1–2 dB.

Table 6: Qualitative measures for CARTOSAT-2 PAN 1 image SR

Image	Methods	PSNR	MSSIM	UIQI	NIQE
CARTOSAT-2 PAN	Bi-cubic	40.13	0.9921	0.8586	24.29
	Yang et al.	41.43	0.9965	0.8825	17.39
	Chen et al.	41.32	0.9950	0.8720	18.16
	Lucas et al.	40.97	0.9950	0.8730	20.94
	Proposed	41.52	0.9966	0.8862	17.07

f. Related Publication:

²H. U. Mullah, B. Deka, and AVV Prasad, “Fast Multispectral Image Super-resolution via Sparse Representation”, *IET Image Processing*, 2020

D.3 Results of MSISR based on adaptive dictionary learning and sparse representations

D.3.1 Performance evaluation

Simulations of the proposed method are conducted on the generated test data sets of LISS-IV imagery. LISS-IV MS images, as we know, are made up of three spectral bands. In this work, rather than processing a false colour RGB image by combining the spectral bands, an individual band image of the test MS image is processed. The reason for not processing a false RGB image is that it loses individual band characteristics. Also, if the dictionary is learnt from one band image, it will not properly work for processing of any other bands. This is because the range of numerical values of pixels in each band is very different. Although the content in a MS image remains same but band wise, they highlight different objects. The coupled dictionary pair is learned from 10000 training patch-pairs from a single LR MS image band-wise.

Table 7: Performance Evaluation of Test Image size of 512 using different methods for upscaling factor 2

Indexs		Bicubic	ScSR [14]	ANR [7]	CCR [38]	JRSR [39]	Proposed
PSNR (dB)	Band 1	38.36	40.23	40.02	40.02	41.78	43.15
	Band 2	34.64	37.02	35.92	36.03	37.98	39.18
	Band 3	31.72	33.47	33.13	33.34	35.62	34.88
	Average	34.90	36.90	36.36	36.46	38.46	39.07
SSIM	Band 1	0.949	0.966	0.961	0.959	0.967	0.977
	Band 2	0.922	0.952	0.940	0.938	0.953	0.969
	Band 3	0.900	0.938	0.919	0.920	0.947	0.944
	Average	0.923	0.952	0.940	0.939	0.956	0.963
ERGAS	Band 1	5.91	4.76	4.86	4.88	3.98	3.40
	Band 2	12.00	9.12	10.36	10.21	8.16	7.11
	Band 3	5.54	4.54	4.72	4.60	3.54	3.85
	Average	7.82	6.14	6.65	6.56	5.22	4.79
SAM	Band 1	0.029	0.023	0.024	0.024	0.019	0.016
	Band 2	0.058	0.044	0.050	0.049	0.040	0.034
	Band 3	0.027	0.022	0.023	0.022	0.018	0.019
	Average	0.038	0.030	0.032	0.032	0.026	0.023
UIQI	Band 1	0.777	0.834	0.829	0.808	0.842	0.894
	Band 2	0.816	0.830	0.859	0.846	0.878	0.925
	Band 3	0.769	0.793	0.784	0.798	0.832	0.789
	Average	0.788	0.819	0.824	0.817	0.851	0.869
sCC	Band 1	0.794	0.835	0.853	0.842	0.835	0.882
	Band 2	0.843	0.890	0.885	0.851	0.876	0.919
	Band 3	0.851	0.894	0.877	0.883	0.898	0.863
	Average	0.830	0.873	0.871	0.859	0.870	0.888

For validation of the proposed algorithm, quantitative evaluation of reconstructed images for upscaling factor 2 is performed as shown in Table 7. SR results are compared with the Bicubic interpolation and four state-of-the-art learning-based SR methods: Yang's sparse coding super-resolution (ScSR [14]) methods, anchored neighborhood regression method (ANR) [7], clustering and collaborative representation (CCR) [38] and joint regularization based SR method (JRSR) [39]. Here, for quantitative analysis, the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) measures, ERGAS, SAM, Q-index and sCC are employed.

Table 8: Performance Evaluation of Test Image size of 512 using different methods for upscaling factor 4

Indexs		Bicubic	ScSR [14]	ANR [7]	CCR [38]	JRSR [39]	Proposed
PSNR(dB)	Band 1	32.10	32.65	33.18	33.30	33.66	33.48
	Band 2	27.56	28.23	28.47	28.60	28.91	28.78
	Band 3	27.86	28.48	29.02	29.37	29.56	29.37
	Average	29.17	29.79	30.22	30.42	30.71	30.54
SSIM	Band 1	0.819	0.838	0.842	0.845	0.866	0.851
	Band 2	0.725	0.757	0.760	0.765	0.800	0.775
	Band 3	0.724	0.749	0.753	0.762	0.786	0.767
	Average	0.756	0.781	0.785	0.790	0.817	0.798
ERGAS	Band 1	10.52	9.89	9.30	9.18	8.81	9.00
	Band 2	17.59	16.27	15.84	15.59	15.04	15.28
	Band 3	11.48	10.68	10.04	9.66	9.43	9.64
	Average	13.20	12.28	11.720	11.48	11.09	11.31
SAM	Band 1	0.052	0.049	0.046	0.045	0.044	0.044
	Band 2	0.084	0.078	0.076	0.074	0.071	0.073
	Band 3	0.054	0.046	0.047	0.045	0.043	0.044
	Average	0.0633	0.0577	0.0563	0.0547	0.0527	0.0537
UIQI	Band 1	0.450	0.507	0.518	0.519	0.596	0.547
	Band 2	0.457	0.520	0.526	0.530	0.607	0.557
	Band 3	0.412	0.467	0.472	0.480	0.549	0.501
	Average	0.440	0.498	0.505	0.509	0.584	0.535
sCC	Band 1	0.406	0.427	0.467	0.476	0.513	0.481
	Band 2	0.426	0.450	0.480	0.492	0.529	0.499
	Band 3	0.401	0.429	0.456	0.480	0.503	0.478
	Average	0.411	0.435	0.468	0.483	0.515	0.486

The visual quality of the reconstructed images using various methods is shown in Fig. 23 for 2 times zoom factor. For better visual perception, zoomed in area is highlighted in the original image; shown for different methods to distinguish the visual difference accurately.

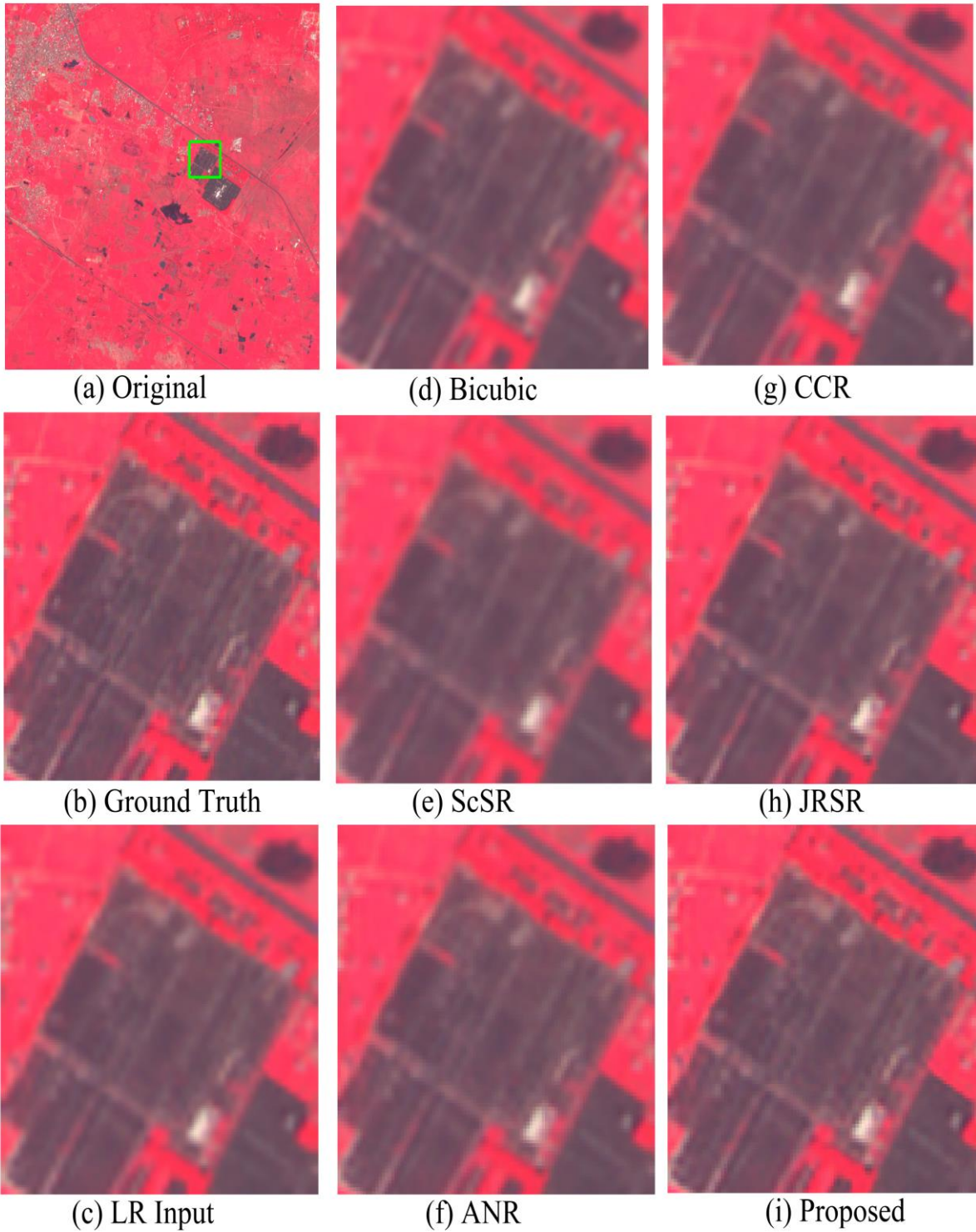


Figure 23: Visual outputs of different SR results on test image 512×512 by using different methods for upscaling factor 2. (Zoom in for better view)

The qualitative comparisons of the HR reconstructions MS images of size 512×512 using various algorithms are shown in Table 8 with a zoom factor of 4. Also, the reconstructed images (zoom in) are visually compared by using different methods in Fig. 24 for upscale factor 4

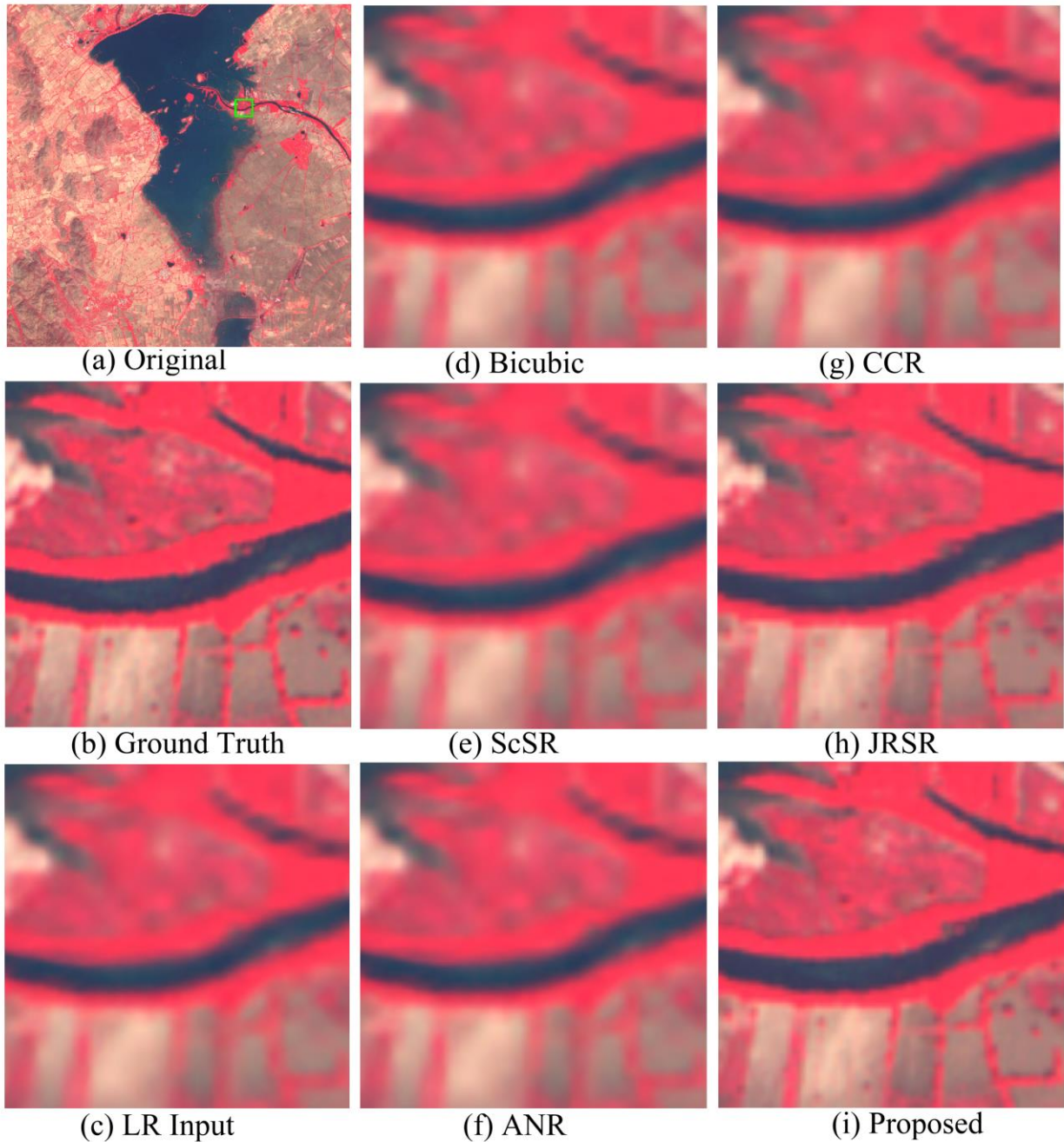


Figure 24: Visual outputs of different SR results on test image 512×512 by using different methods for upscaling factor 4. (Zoom in for better view)

Q. Results of Parallel implementations of MSISR models based on dictionary learning and sparse representations

E.1 Results of Multicore Parallel Processing Implementation of MSISR based on sparse representations and overcomplete dictionaries¹

The execution time is computed for reconstruction of the test images in both sequential and parallel approaches. Fig. 25 shows that the Yang's method [1] requires 91 seconds using the sequential approach for reconstructing a 256×256 image from a 128×128 image of Test1. On the contrary, with the application of multicore parallel processing it is reduced to 8.4 secs. when 24 cores are used. Thus, the overall speed up that is achieved using the proposed method is $= 91/8.4 = 10.83$

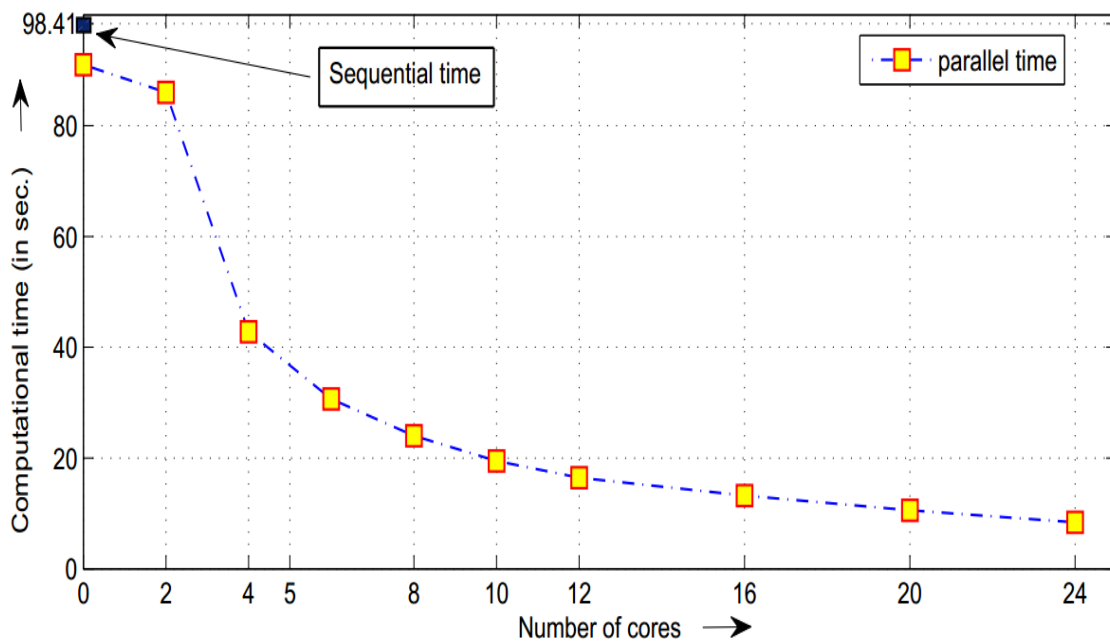


Figure 25: Plot of computation time versus number of cores used in multicore computing system

a. Related Publication:

¹H. U. Mullah and B. Deka, "A Fast Satellite Image Super-Resolution Technique Using Multicore Processing", *Advances in Intelligent Systems and Computing*, Springer, vol. 734, 2018

E.2 Results of Multicore Implementation of MSISR based on MCA and

Sparse Representation

The proposed method as discussed in section 5.1.2, when implemented using multicore parallel computing is able to provide an speed-up of about 28 times in dictionary training, as shown in Fig. 26 (above), while it reduces the reconstruction time by about 12 times as shown in the Fig. 26 (below)².

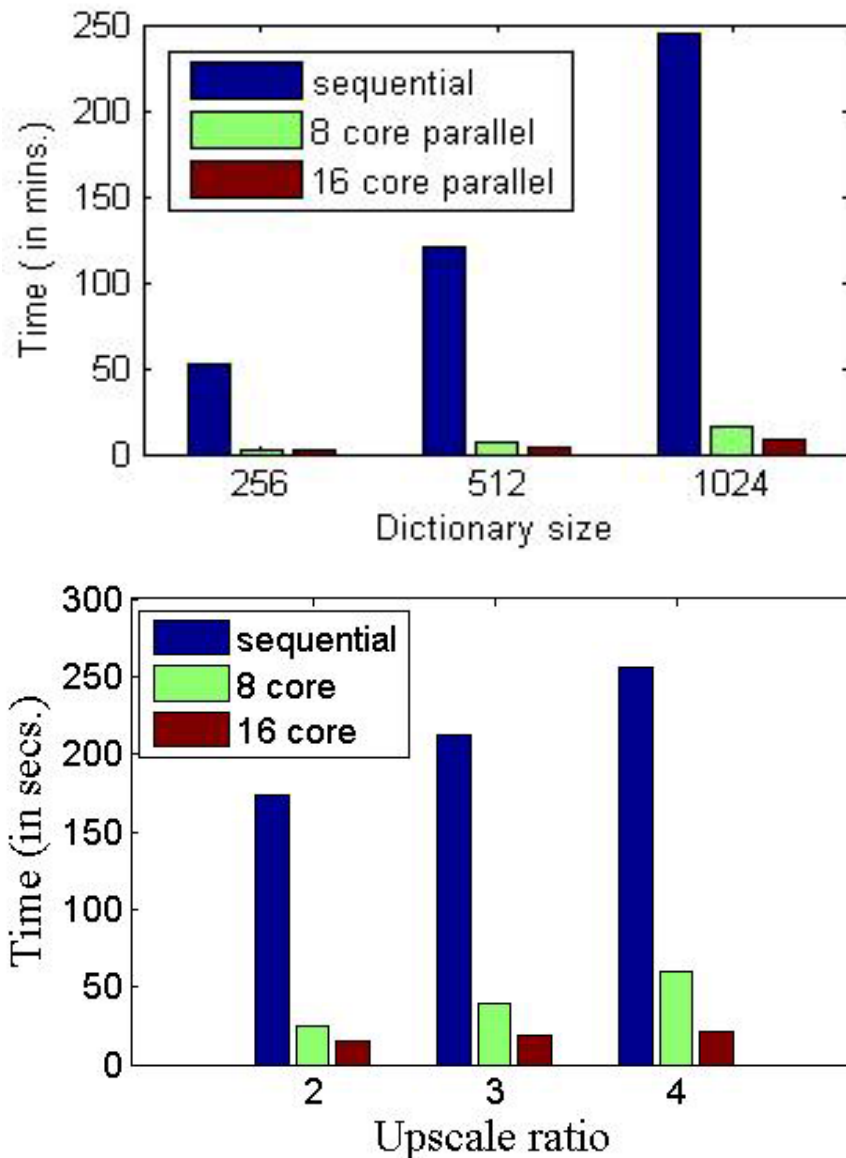


Figure 26: Speed-up comparisons: dictionary training (above) and SR reconstruction (below)

b. Related Publication:

²H. U. Mullah, B. Deka, and AVV Prasad, "Fast Multispectral Image Super-resolution via Sparse Representation", *IET Image Processing*, 2020

E.3 Results of CUDA-based GPGPU Implementation of MSISR based on

adaptive dictionary learning and sparse representations

The experiment is carried out to analyze the scalability of our implementation in terms of dictionary size and input image size. Table 9 shows the execution time taken for both sequential CPU and GPU implementation of dictionary learning for different dictionary sizes. Dictionaries of different size are trained using internal dataset i.e., LR image itself size of 512×512 . The D_h and D_l are trained using fixed sample patches around 100000. As the sample patches are kept in fixed numbers, it takes same execution time for different upscaling factors on CPU.

Table 9: CPU vs GPU speed-up for different dictionary size

Dictionary Size	CPU (secs)	GPU (secs)	Speed-up
256	105	10	10.5
512	193	17	11.3
1024	363	27	13.44

As the dictionary size increases, the speed up of GPU implementation increases from 4 to 5 times. Fig. 27 shows the running time comparison of CPU and GPU implementation of dictionary learning for various dictionary sizes.

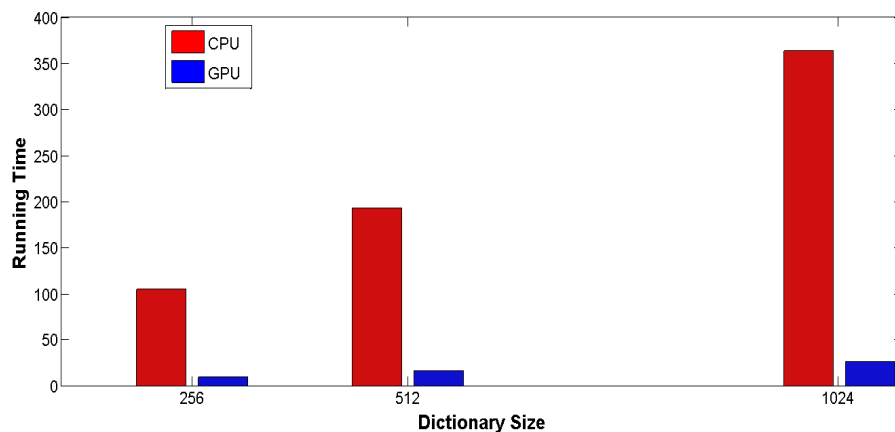


Figure 27: Execution time of CPU and GPU implementations of different dictionary sizes

In the reconstruction, the image size (number of pixels) was varied from 128×128 , 256×256 , 512×512 , and 1024×1024 . Table 10 compares the execution time for sequential CPU and GPU parallel implementations for upscale factor 2. We note that as the image size used in the experiment increases, so does the speedup. As the image sizes increases, the GPU implementation's speed improved steadily from 11 to 36 times as compared to CPU equivalents. In Fig. 28, the

reconstruction time comparison between CPU and our proposed GPU SR method is shown for different image sizes with zooming factor 2

Table 10: CPU vs GPU Reconstruction time comparison for zoom factor 2

Image Size		Proposed Method		Average CPU Time (secs) (t_s)	Average GPU Time (secs) (t_p)	Speed up (t_s/t_p)
		CPU	GPU			
128x128	Band 1	40.33	3.77	42.65	3.78	11.28
	Band 2	43.99	3.80			
	Band 3	43.68	3.79			
256x256	Band 1	188.10	4.25	188.10	4.28	29.93
	Band 2	188.10	4.27			
	Band 3	187.11	4.32			
512x512	Band 1	318.20	10.02	317.04	9.36	33.87
	Band 2	315.59	10.02			
	Band 3	317.34	8.05			
1024x1024	Band 1	856.20	24.01	865.53	23.67	36.56
	Band 2	907.20	24.01			
	Band 3	834.20	23.00			

Table 10 shows the comparison of reconstruction time of sequential CPU and GPU parallel implementations for different image sizes using upscale factor 4. Our proposed method takes about 4-83 secs and reduces the reconstruction time by 60 to 186 times as compared to CPU counterparts.

Table 11: CPU vs GPU Reconstruction time comparison for zoom factor 4

Image Size		Proposed Method		Average CPU Time (secs) (t_s)	Average GPU Time (secs) (t_p)	Speed up (t_s/t_p)
		CPU	GPU			
128x128	Band 1	249.01	4.11	250.60	4.11	60.97
	Band 2	252.35	4.12			
	Band 3	250.44	4.11			
256x256	Band 1	1028.22	9.54	1028.59	9.49	108.38
	Band 2	1029.03	9.42			
	Band 3	1028.54	9.53			
512x512	Band 1	4295.23	23.01	4295.04	23.18	186.73
	Band 2	4294.02	23.42			
	Band 3	4295.44	23.13			
1024x1024	Band 1	9322.17	83.02	9322.78	83.93	111.07
	Band 2	9324.17	82.55			
	Band 3	9322.01	83.22			

Overall, the results show that executing our algorithm on GPU is 11 to 186 times

faster than performing it on CPU for different zooming factor, respectively The execution time comparison between CPU and our proposed GPU SR reconstruction method is shown in Fig. 29 for various image sizes with zooming factor 4. As the execution time difference between CPU and GPU is quite large, the running time in Fig. 29 is shown in log scale.

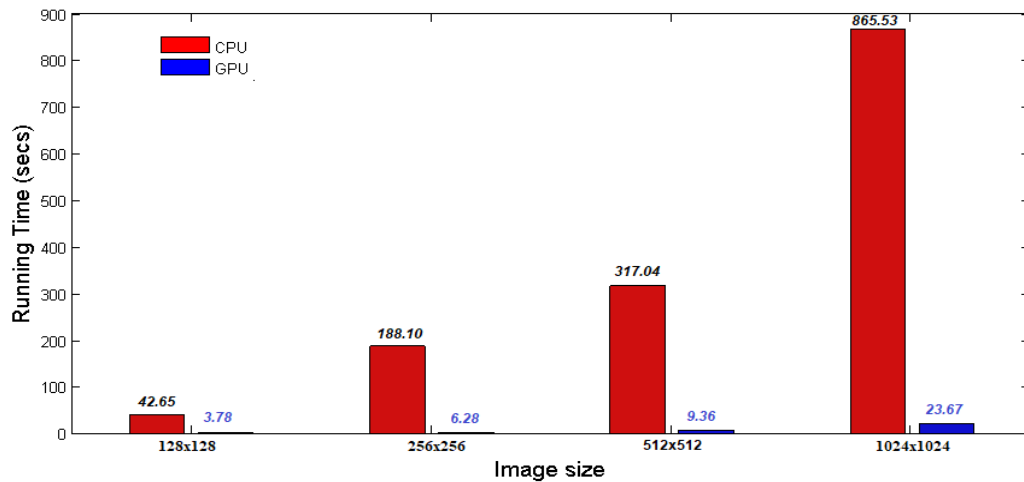


Figure 28 : Reconstruction time of CPU and GPU implementation for different Image sizes with zooming factor 2

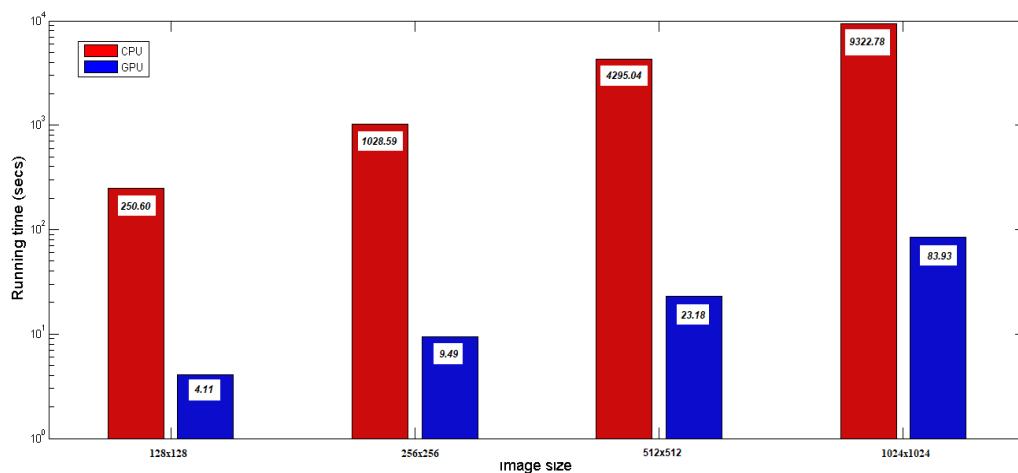


Figure 29: Reconstruction time of CPU and GPU implementation for different Image sizes with zooming factor 4

c. Related Publication:

Trishna Barman, Bhabesh Deka, and AVV Prasad, “GPU-Accelerated Adaptive Dictionary Learning and Sparse Representations for Multispectral Image Super-resolution,” *IEEE 18th India Council International Conference (INDICON), 2021*

R. Real-time Implementations and Applications

F.1 Real-time Implementations of MSISR models

It is one of the major conflicts that there is no ground truth image in case of real satellite data. We usually synthesize a LR image from original images, then produce a SR image from the LR image, finally compare the original and SR images. But, in real case, we can only produce a HR version of the original LR MS image itself. In such case, we can compare the resulted image with other HR MS images acquired from a HR sensor. This can be done by first finding the same area images for the HR sensor, then allotting similar geospatial coordinates to the reconstructed images, and finally re-projecting the new sensor image into similar resolution to the SR reconstructed image. Thus, we obtain spatially and spectrally equivalent images which can be compared both visually as well as by comparing performance by NIQE measure or by any classification algorithm

To implement the MCA based MSISR method as discussed in section 5. 2 for real-time applications, experiments are carried out in this work to test the output of the proposed method when input LR images are fed directly to the proposed algorithm for SR reconstructions for various zooming factors, without any preprocessing, such as blurring and down-sampling. In addition to visual inspection, we assess the quality of super-resolved outputs using the no-reference assessment metric, i.e. NIQE. When we use the QuickBird MS image as the input LR image directly and do SR reconstruction on it for 2x zooming, the NIQE values for the input and output images are 14.86 and 11.51, respectively. Similarly, we compute the NIQE values of reconstructed images using LISS-IV and LISS-III sensor inputs. The input and reconstructed LISS-IV images have NIQE values of 21.70 and 19.37, respectively, while the LISS-III image has NIQE values of 24.25 and 21.49, respectively².

Since the spatial size of remote sensing images is significantly large, it is very important to process the larger remote sensing images within a real-time. Our efficient CUDA-GPU based SR algorithm can easily process the larger remote sensing images (up to 3000×1500 image size) within a few minutes for different upscaling factors. The proposed method has also the ability to enhance visual quality of SR results for upscaling factor 2 and 4. To validate the effectiveness of the proposed method for real remote sensing images, quantitative SR results of different Rols are shown for upscale factor 2 and 4 in Tables 11 and 12, respectively.

Execution times taken for dictionary training and reconstruction for different Rols using this CUDA-GPU-based parallel SR algorithm are also shown in Table 12 and Table 13 for upscale factors 2 and 4. The proposed method is only compared with the bicubic method because other state-of-the-art methods cannot be processed for the larger remote sensing images due to the memory limitation imposed by their sequential implementations.

Table 12: Band wise results for two real MS images and GPU dictionary training/ reconstruction time with upscale factor 2

Input	Parameters	Bicubic (dB)	Proposed method (dB)	Average PSNR (dB) & SSIM		Dictionary training Time (secs)	Reconstruction Time (secs)	
				Bicubic	Proposed			
2048x2048	Band 2	PSNR	41.49	44.79	PSNR: 38.28 SSIM: 0.923	PSNR:42.10 SSIM: 0.967	49.61	83
		SSIM	0.940	0.976				
	Band 3	PSNR	38.60	42.66			49.41	83
		SSIM	0.937	0.972				
	Band 4	PSNR	34.75	38.87			50.42	82
		SSIM	0.894	0.954				
3000x1500	Band 2	PSNR	41.43	44.29	PSNR: 38.28 SSIM:0.903	PSNR: 41.57 SSIM:0.965	49.01	147
		SSIM	0.956	0.974				
	Band 3	PSNR	38.73	42.25			53.12	146
		SSIM	0.859	0.970				
	Band 4	PSNR	34.68	38.17			54.02	147
		SSIM	0.895	0.952				

Table 13: Band wise results for two real MS images and GPU dictionary training/ reconstruction time with upscale factor 4

Input	Parameters	Bicubic (dB)	Proposed method (dB)	Average PSNR (dB) & SSIM		Dictionary training Time (secs)	Reconstruction Time (secs)	
				Bicubic	Proposed			
2048x2048	Band 2	PSNR	37.98	38.29	PSNR: 34.34 SSIM: 0.841	PSNR: 34.66 SSIM: 0.850	53.21	403
		SSIM	0.912	0.916				
	Band 3	PSNR	34.78	35.07			53.25	402
		SSIM	0.859	0.866				
	Band 4	PSNR	30.28	30.64			49.68	403
		SSIM	0.754	0.769				
3000x1500	Band 2	PSNR	37.58	37.86	PSNR: 34.05 SSIM: 0.790	PSNR: 34.68 SSIM: 0.798	54.00	415
		SSIM	0.761	0.765				
	Band 3	PSNR	34.68	35.97			56.20	415
		SSIM	0.859	0.866				
	Band 4	PSNR	29.91	30.23			55.02	415
		SSIM	0.751	0.765				

The visual results of an input LR image of size 2048×2048 and its corresponding SR results for upscale factor 2 is shown in Fig. 30. We can see that the visual quality of output SR is much improved than that of LR input image in the enlarged rectangular area. For upscale factor 4, the visual result of input LR image size of 2048×2048 is shown in Fig. 31.

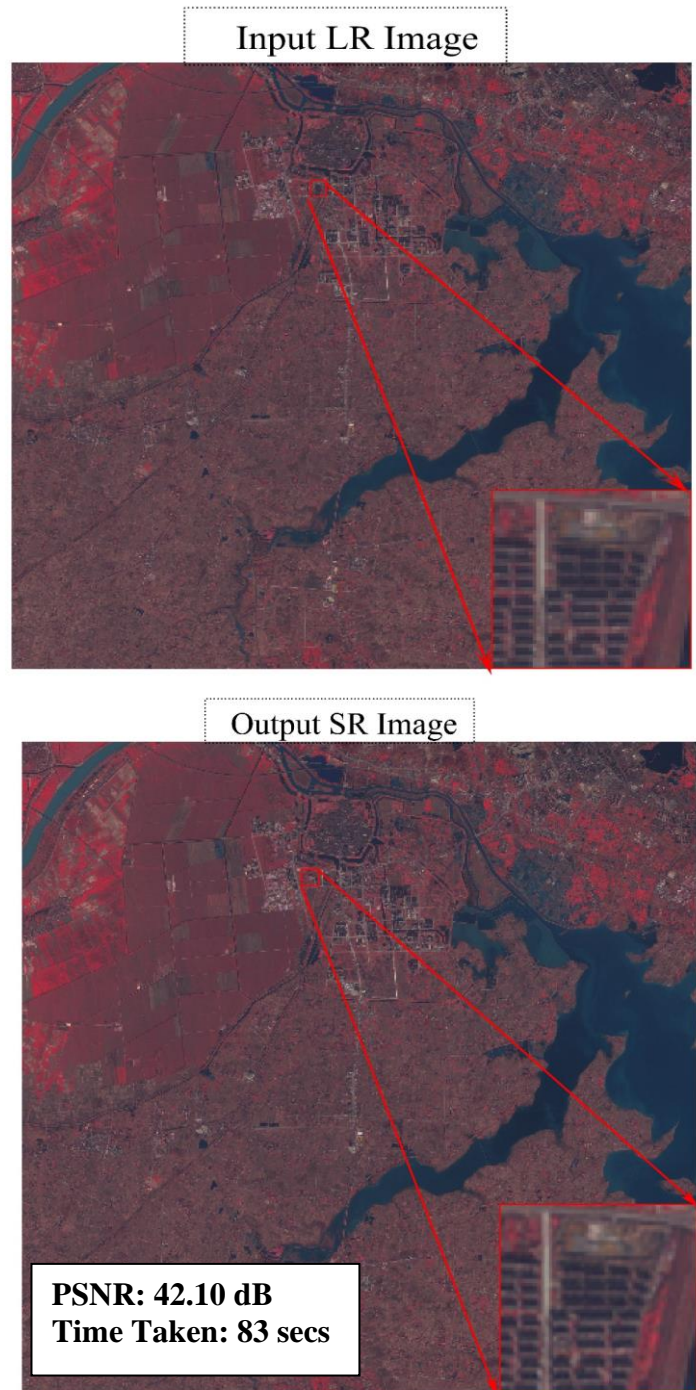
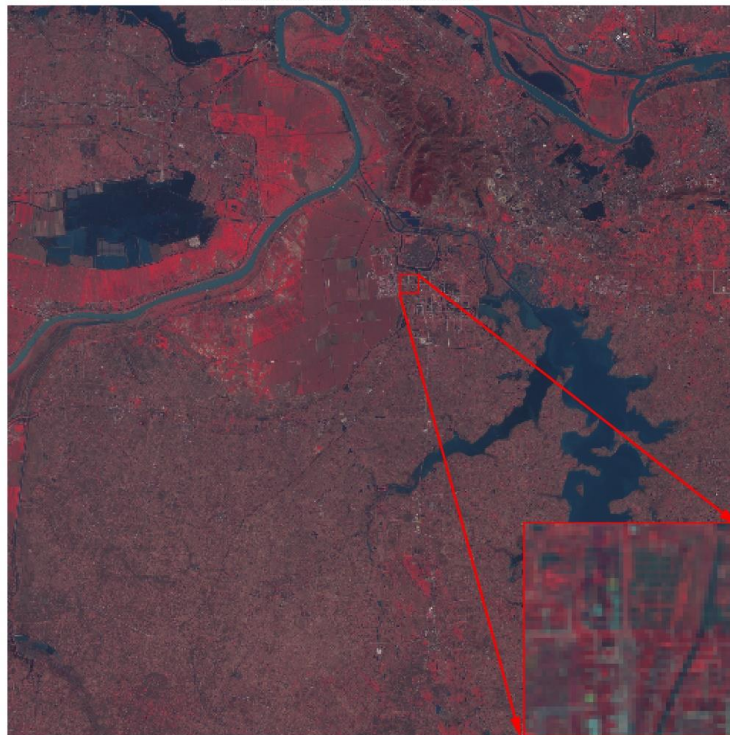


Figure 30: The visual results of LR and SR image of LISS-IV sensor for upscale factor 2; An area in the red rectangle is enlarged in the lower right corner of the image.

Input LR Image



Output SR Image

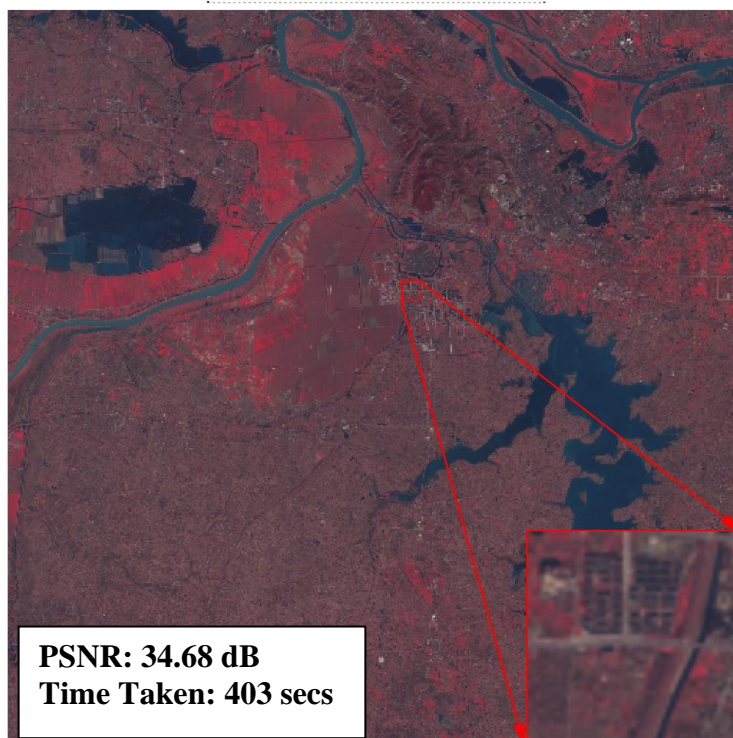


Figure 31: The visual results of LR and SR image of LISS-IV sensor for upscale factor 4; An area in the red rectangle is enlarged in the lower right corner of the image

F.2 Application and Evaluations

SR based multispectral remote sensing images play an important role in various applications such as various surveys, monitoring and management applications. To evaluate the MSISR image, spectral property evaluation and classification are being applied to the proposed method as post-processing steps.

F.2.1 Spectral properties evaluation: Spectral Signature

Spectral signature plots the spectral reflectance of objects as a function of wavelength. A ROI of homogeneous regions corresponding to identical location in each band is selected. Now, average reflectance of pixels in the ROI corresponding to a band are then plotted against the band number. Spectral profile plots of the QuickBird test MS image and HR MS image produced from it are shown in Fig. 32, where it can be observed that spectral orientations of ground-truth and test images are not disturbed after SR reconstruction².

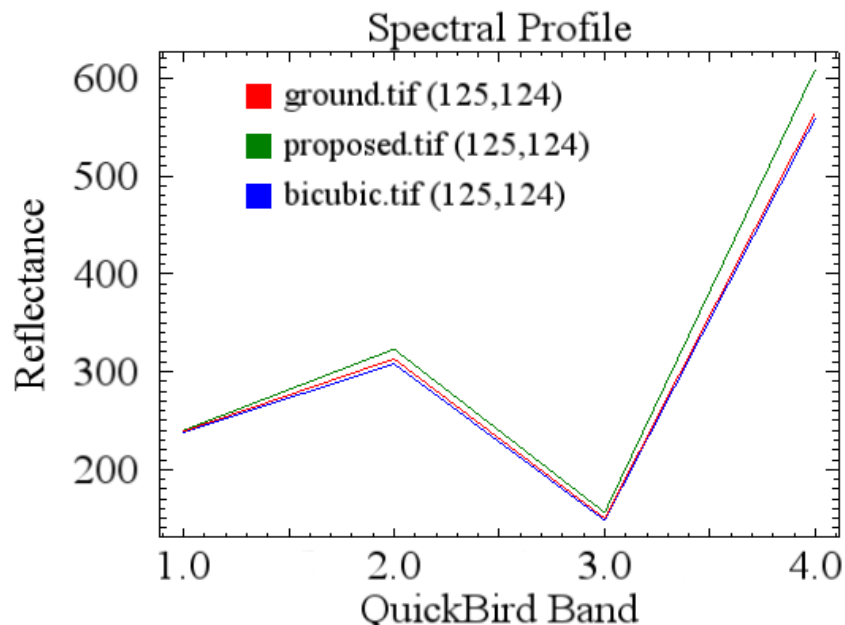


Figure 32: Spectral profile of QuickBird image

S. End-members identifications:

Spectral unmixing is the decomposition of a mixed pixel's spectral signature into a group of endmembers and their corresponding abundance. A linear mixture model considers a mixed image; each pixel is a linear mixture of endmembers with positive fractional abundances that always add up to one. The effects of spectral unmixing are highly dependent on the input endmembers; modifying the endmembers affects the results. There are various methods for doing linear

spectral unmixing, such as maximum likelihood-based unmixing, spectral angle mapper, etc. In this work, we compare the effects of SR in end-member identifications using ENVI software to perform spectral unmixing of LR and SR MS images using the maximum likelihood method².

In the example shown in Fig. 33, three different types of end members are selected from the given LISS-IV MS image that includes water body, vegetation, and barren soil. Here, the ROIs are selected for three end-member types, namely, “water body” consisting of total 457 numbers of pixels, “vegetation” consisting of 722 pixels and “barren soil” consisting of 606 pixels, respectively. The unmixed end-member profile distributions are displayed for the ground truth MS image, LR test image and SR MS image. It can be observed that, the SR reconstructed image by proposed algorithm is better able to identify a end-member type of the true MS image compared to the LR test image.

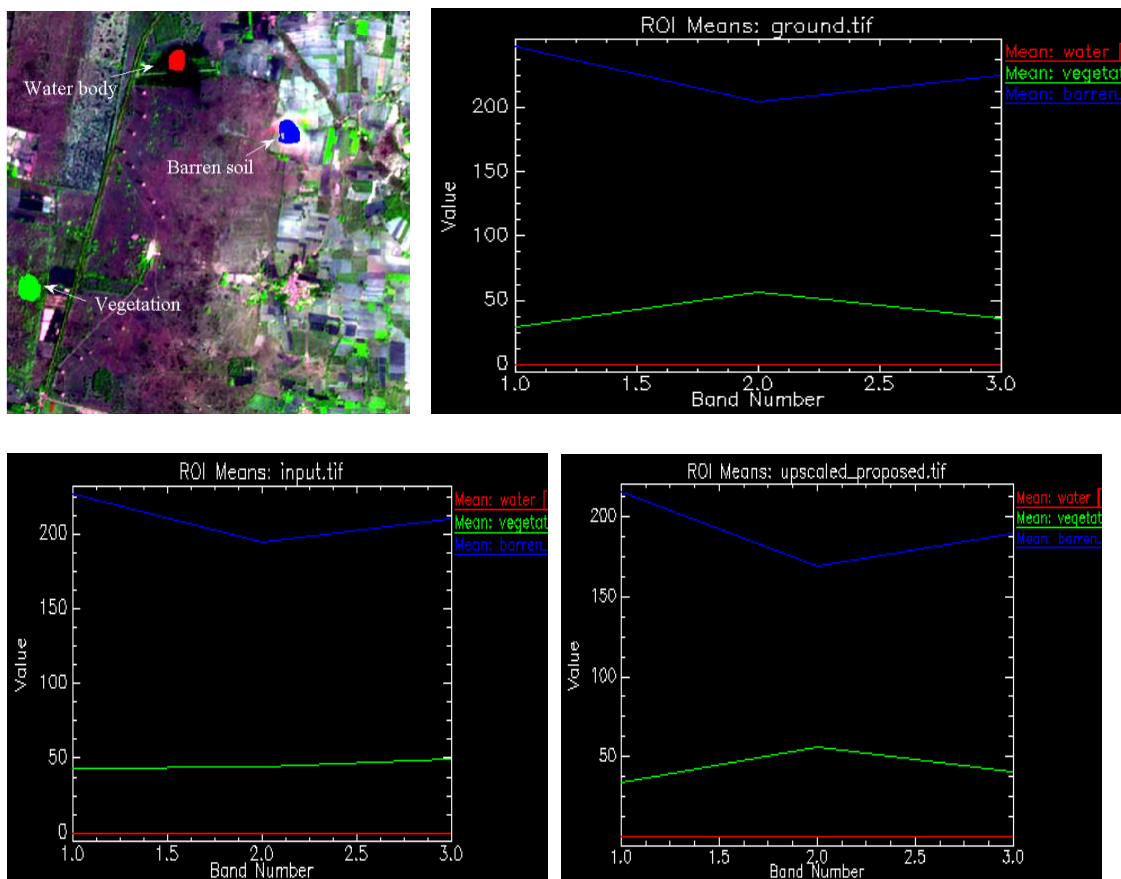


Figure 33: End-members comparisons

a. Related Publication:

²H. U. Mullah, B. Deka, and AVV Prasad, “Fast Multispectral Image Super-resolution via Sparse Representation”, *IET Image Processing*, 2020

F.2.2 Classification on SR images

As a post processing step, classification is performed with the proposed method and the-state-of-the-art methods to separate various regions present in the MS image. Envi classic 5.1 is used for classification and analysing the classified results. The unsupervised classification has been performed using K-means method on the test image for different methods. Without training classes, unsupervised classification can cluster pixels from the test images. Five classes are defined: vegetation, barren soil, water, road, and building. It is observed that the proposed method efficiently separates the classes that are quite similar to that of original ground truth image. But, in LR image and other SR methods, some of the classes are not clearly identified properly and misclassify some regions. Results of unsupervised classification on test images for different methods are shown in Fig. 34. The percentage of classification accuracy on each class of the test image for different methods is given in Table 14. The classification accuracies on each class of the proposed method are closed to that of the ground truth. But class wise accuracy rates of the LR, ScSR, ANR and CCR methods are diverse from the original image. Since the image quality of the proposed method is comparatively higher than other methods, the difference in percentage of accuracy between the proposed method and the original ground truth image is low, when compared to other methods. Therefore, it can be concluded that the reconstruction accuracy of the proposed method is clearly superior to its counterparts.

Table 14: Classification accuracy on each class of Test Image using unsupervised classification for different methods

Class	Ground Truth (%)	LR Input (%)	Bicubic (%)	ScSR (%)	ANR (%)	CCR (%)	JRSR (%)	Proposed (%)
I	3.60	3.16	2.98	3.16	3.02	3.04	3.45	3.49
II	17.63	16.67	17.94	17.92	17.81	17.75	17.23	17.33
III	51.75	53.01	50.13	50.51	50.45	50.54	51.68	51.69
IV	20.09	19.69	21.25	20.94	21.17	21.14	20.65	20.58
V	6.90	7.45	7.45	7.45	7.53	7.52	6.98	6.90

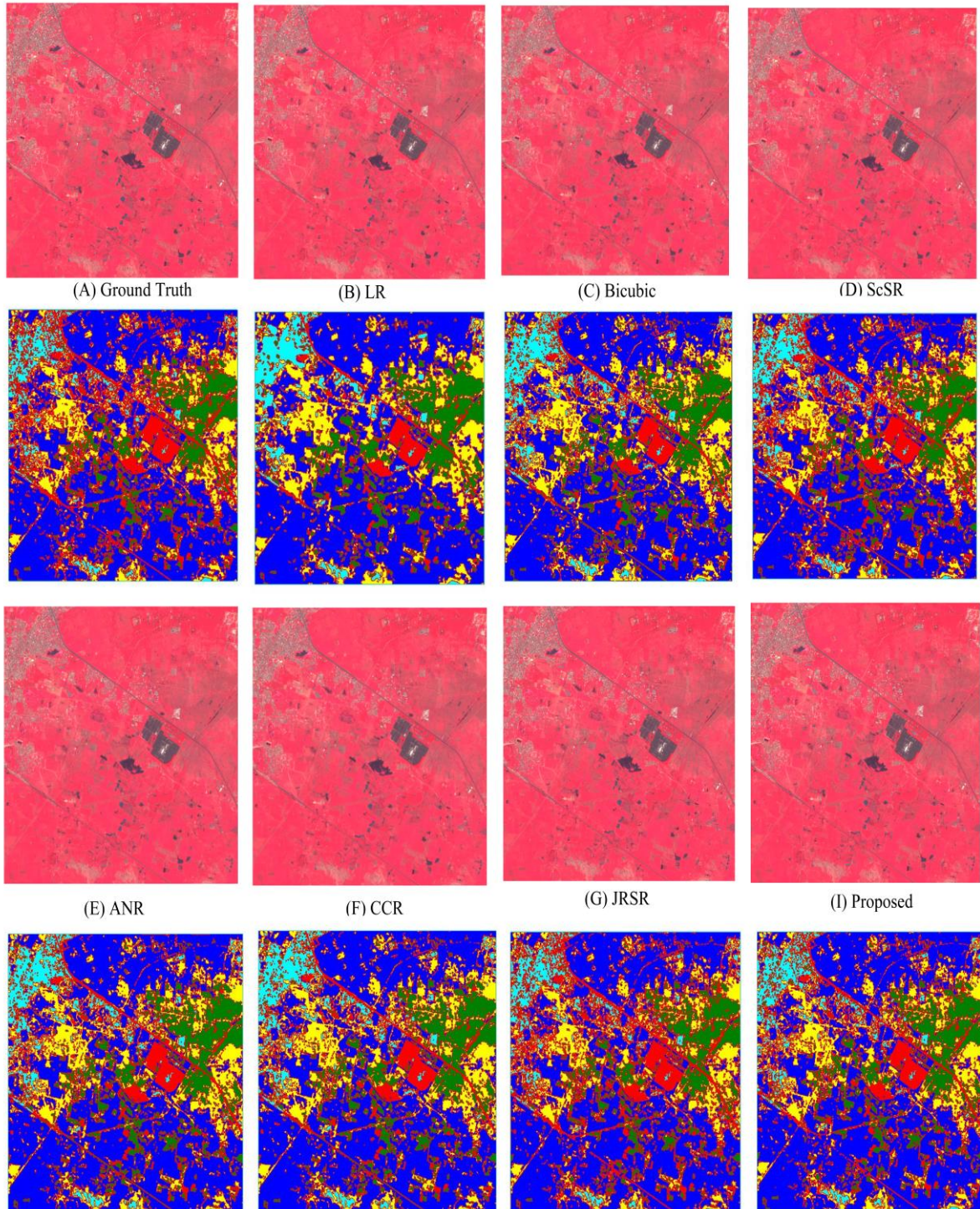


Figure 34: Results of unsupervised classification on the Test Image for different methods

7. Discussions & Interpretation of the results

7.1 Design and implementation of MS image SR (MSISR) based on sparse representations and overcomplete dictionaries

The existing sparse representation-based method has been employed on the remote sensing images. The quantitative assessments such as PSNR, MSSIM, Q-Index, SAM and ERGAS have been performed on various images size with upscaling factors 2 and 4 as shown in Table 2. Since both MSISR and SPARSFI are sparse representation-based methods, they give better results than the bicubic. However, these sparse representation-based SR methods are computationally heavy, OpenMP based multicore parallel processing technique was also applied to the algorithm to obtain the speed-up. From Fig. 25, we can see that, multicore implementation increases the speed upto 10 times when utilizing the maximum cores (16) than sequential CPU counterpart. In the first work, we have shown some preliminary results of how sparse representations-based SR works on remote sensing images and compared with some basic state-of-the-art methods. Also, we have conducted a preliminary analysis to speed-up the sparse representations-based SR method using OpenMP-based parallel processing paradigm during the first year of implementation.

7.2 Multicore Based Multispectral Image SR

7.2.1 Super-resolution of QuickBird MS image:

Results of SR are compared with those of other approaches. Fig. 23 depicts a visual representation of SR outputs. The results show that as compared to other methods, the proposed approach causes the least visible error. We also conduct a quantitative assessment in terms of PSNR, MSSIM, ERGAS, SAM, UIQI, sCC, and NIQE to support this point, and the results are shown in Table 5. It is discovered that with a zoom factor of 4, the proposed algorithm improves PSNR by 1.28 dB over bi-cubic and 0.2–0.4 dB over other MS SR approaches. Moustafa et al. obtain results that are similar to that of the proposed method.

We have experimented with different zooming factors. PSNRs obtained for 4x zooming for the proposed method, bicubic, and Moustafa et al. are 28.86, 26.43, and 28.35 dB, respectively. Again, for 3x zooming, the proposed, bi-cubic, and Moustafa et al. values are 27.48, 25.89, and 27.21 dB, respectively. As a result, when compared to bi-cubic, our approach improves by 2.43 and 1.59 dB for 2 and 3 zooming factors, respectively. Similarly, as opposed to Moustafa et al., the proposed approach improves by 0.51 and 0.27 dB for 2x and 3x zooming factors, respectively. The proposed method achieves superior results for other

parameters as well. Closer results in the case of Moustafa et al. may be attributed to the fact that it is a self-learning method; the dictionary is learned from the given LR image. The proposed method, on the other hand, makes use of a trained dictionary from an external dataset of noise-free images. An external database is preferred because it can have better features for learning from specific LR-MS images.

7.2.2 Super-resolution of LISS-IV MS image:

For various zooming factors, SR reconstruction is performed. Fig. 23 depicts a comparison of output images with a zooming factor of 2 using various methods. There are also error images associated with each method. In comparison to other methods, the proposed method produces the least apparent defect. Table 5 shows the results of various objective criteria for LISS-IV test images. It is obvious that the proposed approach outperforms other approaches in terms of efficiency. On an average, the proposed method for 2x zooming increases PSNR by around 2.7 dB over bi-cubic output, although it is 0.3–0.4 dB better than Moustafa et al. and Lucas et al. Table 5 shows the values for MSSIM and UIQI have improved as well. The NIQE value has also been reduced dramatically, indicating a better reconstructed image. We also run SR on the same image for 3x and 4x zooming, and the proposed approach outperformed the others.

7.2.3 Super-resolution of LISS-III MS image:

SR is carried out with a zooming factor of 3. The results are analysed and compared with other MS image SR algorithms both visually and quantitatively, as seen in Fig. 23 and Table 5, respectively. The proposed method outperforms others even when the test LR image is zoomed in 3x. Visual analysis reveals that the proposed method causes less apparent errors than others. On an average, the proposed method for 3x zooming increases PSNR by around 1.29 dB over bi-cubic output, which is 0.15–0.3 dB more than Moustafa et al., Chen et al., and Lucas et al.. Similarly, improvements in values have been observed for MSSIM, UIQI, and NIQE. We also performed SR on the same image for 2x and 4x zooming, and the proposed method outperformed the others.

7.2.4 Super-resolution of panchromatic image:

Fig. 23 depicts the SR results. On average, the proposed method for PAN image SR improves PSNR by around 1.4 dB over bi-cubic, while it improves PSNR by

0.1–2 dB more than Yang's and Chen's methods. Similarly, improvements in MSSIM and UIQI values have been observed. In the case of the proposed method, the NIQE value is also greatly decreased, indicating a better reconstruction.

7.2.5 Speed-up calculation

The plot of computing time versus core number for the proposed dictionary learning algorithm in the Intel Xeon processor-based server method is shown in Fig. 26 (above). The execution time of the multi-core implementation of 256 size dictionary training using 16 cores is reduced to 1 min 49 s from its sequential equivalent value of 52 min 33 s. In this case, the speed-up is about 28 times. Similarly, as seen in Fig. 26 (below), the highest speed-up of the reconstruction method with respect to a 256-size dictionary is obtained corresponding to four times up-scaling, which is given as $255.23 \text{ s}/21.26 \text{ s} \approx 12$.

7.3 MS image SR based on Dictionary Learning and Sparse Representation using GPGPU with CUDA acceleration

The proposed algorithm is compared to state-of-the-art approaches in terms of PSNR, SSIM, ERGAN, SAM, Q-Index and sCC. Our algorithm obtains the best results for most of the test images, achieving best results on an average. The PSNR increases for our proposed method by around 4.79 dB and 1.37dB over the bicubic interpolation for zooming factors 2 and 4, respectively. The PSNR gain of our proposed method over the second-best method i.e. JRSR [39] are 0.61dB for zoom factor 2. But for zooming factor 4, the proposed method underperforms than JRSR. It also can be found in Table 10 that the SSIM, ERGAS, SAM, Q-Index and sCC gains of our proposed method shows better results than that of others methods for upscale factor 2. These quantitative results show that our proposed method not only achieves lower reconstruction error but also retains more structural features than the other compared state-of-the-art methods. However, SSIM, ERGAS, SAM, Q-Index and sCC of the proposed method gives comparable results with JRSR for zoom factor 4. The most of compared state-of-the-art methods are based on global dictionary learning. It's possible that the global dictionary could not be able to represent all the image patches correctly. Although the restored HR may seem appropriate, the details obtained from the training database may be meaningless since global dictionary image patches are currently unavailable. Also, these methods have high

computational cost and memory consumption. In the remote sensing domain, global dictionary learning based SR methods are not convenient, because it is quite time consuming and expensive to produce the HR training data. On that account, our proposed method is based on adaptive dictionary learning where only the LR image is used for dictionary training instead of an external database. Reconstructed image may find many similar patches within the adaptive dictionary. Therefore, our proposed adaptive dictionary-based SR method gives state-of-the-art results in MS remote sensing imagery.

To reduce computational complexity of the proposed method, highly parallelized algorithms using CUDA enabled GPU has been designed for real time SR reconstruction of MS remote sensing images. The CUDA-GPU parallel algorithm gives around 10-13 times speed up for dictionary learning as shown in Table 9. From the Tables 10 and 11, we have seen that reconstruction time is reduced by upto 11-36 times and 60-186 times for upscaling factor 2 and 4, respectively than the CPU counterparts.

8. Deliverables & linkages to ISRO/DOS

The complete source codes, executable files, and tested data for execution of the project will be shared with the National Remote Sensing Centre (NRSC).

Developed parallel SR algorithm-based GPU-CUDA can be used in many remote sensing applications such as urban mapping, environment monitoring, military surveillance, intelligence gathering, and disaster management, *etc.* It can also be applied to ISRO's Bhuvan geoportal for navigation.

8. Conclusions

From the initial work of MSISR, we have shown some preliminary SR results and compared with a few basic state-of-the-art methods. We have also developed an efficient multicore parallel processing-based single image SR for multispectral satellite images. The developed system can reduce computational complexity by ten times compared to a highly efficient sparse representation based single image SR method. Besides improved visual quality, the results show fair PSNR and MSSIM improvements. In the MCA based SR method, we have demonstrated a sparse representation and morphological feature extraction-based MS SISR technique. Both the CARTOSAT-2 PAN image dataset and the LR-MS datasets from QuickBird and RESOURCESAT-2 are used to learn overcomplete dictionaries. For evaluating reconstructed HR MS

images, we use both objective and visual analysis. As the results are compared to those of other recent methods, it is found that it outperforms them in both qualitative and quantitative terms. The proposed algorithm is also accelerated by using OpenMP-based parallel processing.

In the GPU based SR work, we have found that adaptive dictionary learning SR method is preferable than global dictionary learning. Experiments have shown that our algorithm offers advantages in terms of better visual outputs and objective criteria as compared to other state-of-the-art methods considered. The CUDA-GPU implementation of the proposed method achieves significant speeds up in the computational time; about 11-36 times and 60-186 times for upscaling factors 2 and 4, respectively than the corresponding CPU implementations.

9. Future Plans

In the future work, the MCA-based proposed SR algorithm can also be implemented on GPU using CUDA to obtain speed-up that would be suitable for real remote sensing applications. In GPU-based proposed work, new feature extraction schemes and dictionary learning techniques can be explored. Various optimization techniques can also be used to improve the perceptual quality of SR results. The proposed GPU-CUDA based parallel dictionary learning and reconstruction algorithms for super-resolution might be improved for achieving more speed-ups for real time remote sensing applications.

In the recent years, deep learning approaches have achieved promising results in SISR system. As a future work, deep learning model will be incorporated to the proposed work to obtain further improvement in results.

10. References

- [1] Z. Wei and K. K. Ma, "Contrast-guided image interpolation", *IEEE Transactions on Image Processing*, vol. 22, no. 11, pp. 4271-4285, Nov 2013.
- [2] S. S. Panda, M. S. R. S. Prasad and G. Jena, "Pocs based super-resolution image reconstruction using an adaptive regularization parameter", *International Journal of Computer Science*, vol. 8, no. 5, 2011.
- [3] D. Glasner, S. Bagon and M. Irani, "Super-resolution from a single image", *In IEEE 12th International Conference on Computer Vision*, pp. 349–356, Sept 2009.
- [4] W. T. Freeman, T. R. Jones and E. C. Pasztor, "Example-based super-resolution",

- IEEE Computer Graphic and Application*, vol. 22, no. 2, pp. 56-65, Mar 2002.
- [5] S. Schulter, C. Leistner and H. Bischof, “Fast and accurate image upscaling with super-resolution forests”, *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3791–3799, June 2015.
- [6] M. Cao, Z. Gan and X. Zhu, X, “Super-resolution algorithm through neighbor embedding with new feature selection and example training”, *In IEEE International Conference on Signal Processing*, vol. 2, pp. 825–828 , Oct 2012.
- [7] R. Timofte, V. De and L. V. Gool, “Anchored neighborhood regression for fast example based super-resolution”, *In IEEE International Conference on Computer Vision*, pp. 1920–1927, Dec 2013.
- [8] W. T. Freeman and E. C. Pasztor, “Learning low-level vision”, *In: Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, pp. 1182–1189, Sept 1999.
- [9] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image super-resolution via sparse representation”, *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861-2873, Nov 2010.
- [10] R. Fattal, “Image upsampling via imposed edge statistics”, *ACM Trans. Graph.*, vol. 26, no. 3, Jul 2007.
- [12] M. F. Tappen, B. C. Russell and W. T. Freeman, “Exploiting the sparse derivative prior for super-resolution and image demosaicing”, *In IEEE Workshop on Statistical and Computational Theories of Vision*, 2003.
- [13] H. Chang, D. Y. Yeung and Y. Xiong, “Super-resolution through neighbor embedding”, *In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2004.
- [14] J. Yang, J. Wright, T. Huang and Y. Ma, “Image super-resolution as sparse representation of raw image patches”, *In: 2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008.
- [16] S. Gu, W. Zuo, Q. Xie, D. Meng, X. Feng and L. Zhang, “Convolutional sparse coding for image super-resolution”, *In: 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1823–1831, Dec 2015.
- [17] W. Dong, L. Zhang, G. Shi and X. Li, “Nonlocally centralized sparse representation for image restoration”, *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1620–1630, April 2013.

- [18] J. Kim, J. K. Lee and K.M. Lee, “Accurate image superresolution using very deep convolutional networks”, *In: IEEE Conf. Comp. Vis. Pattern Recog.* , 2016
- [19] J. Kim, J. K. Lee and K.M. Lee, “Deeply-recursive convolutional network for image super-resolution”, *In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [20] R. Zeyde, M. Elad and M. Protter, “On Single Image Scale-Up Using Sparse-Representations”, *International Conference on Curves and Surfaces*, pp. 711-730, 2010
- [21] H. Song, B. Huang, Q. Liu and K. Zhang, “Improving the spatial resolution of landsat TM/ETM+ through fusion with spot5 images via learning-based super-resolution”, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1195–1204, March 2015.
- [22] Z. Zhu, F. Guo, H. Yu and C. Chen, “Fast single image super-resolution via self-example learning and sparse representation”, *IEEE Transactions on Multimedia*, vol. 16, no. 8, pp. 2178–2190, Dec 2014.
- [23] Z. Pan, J. Yu, H. Huang, S. Hu, A. Zhang, H. Ma and W. Sun, “Super-resolution based on compressive sensing and structural self-similarity for remote sensing images”, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 9, pp. 4864–4876, Sep 2013.
- [24] A. Kulkarni and T. Mohsenin, “Accelerating compressive sensing reconstruction OMP algorithm with CPU, GPU, FPGA and domain specific many-core”, *In: IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 970-973.
- [25] Y. Yuan, X. Yang, W. Wu, H. Li, Y. Liu and K. Liu, “A fast single-image superresolution method implemented with CUDA”, *Journal of Real-Time Image Processing*, vol. 16, no. 1, pp. 81-97, 2018.
- [26] C. Jung, P. Ke, Z. Sun and A. Gu, “A fast deconvolution-based approach for single image super-resolution with GPU acceleration”, *Journal of Real-Time Image Process.*, vol. 14, no. 2, pp. 501–512, Feb 2018.
- [27] H. Tan, H. Xiao, Y. Liu, M. Zhang, and B. Wang, “Lasso approximation and application to image super-resolution with CUDA acceleration”, *In: 2nd International Conference on Image, Vision and Computing (ICIVC)*, 2017.
- [28] S. B. Buche, S. A. Dhondse and A. N. Khobragade, “Satellite image processing on parallel computing: A technical review”, *In: 2016 Online International Conference on Green Engineering and Technologies (IC-GET)*, pp. 1–9, Nov 2016.

- [29] M. Moustafa, H. M. Ebeid, A. Helmy, T. M. Nazmy, M. F. Tolba, “Rapid real-time generation of super-resolution hyperspectral images through compressive sensing and GPU”, *International Journal of Remote Sensing*, vol. 37, no. 18, pp. 4201-4224, 2016.
- [30] E. Amaldi and V. Kann, “On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems”, *Theor. Comput. Sci.*, vol. 209, no. 12, 237-260, Dec 1998.
- [31] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring”, *In: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 693–696, April 2009.
- [32] D. L. Donoho and Y. Tsaig, “Fast solution of l_1 -norm minimization problems when the solution may be sparse”, *IEEE Transactions on Information Theory*, vol. 54, no. 11, pp. 4789-4812, 2008.
- [33] Texas Instruments, *OpenMP Programming for TMS320C66x multicore DSPs*. (2011). [Online]. Available: https://www.ti.com/lit/ml/sprt620a/sprt620a.pdf?ts=1622009346834&ref_url=https%253A%252F%252Fwww.google.com%252F.
- [34] NVIDIA Corporation Technical Staff, *NVIDIA CUDA Programming Guide 2.2*, NVIDIA Corporation, 2009.
- [35] A. Munshi, *OpenCL Specification V1.0*. (2003). [Online]. Available: <https://www.khronos.org/registry/OpenCL/specs/opencvl-1.2.pdf>.
- [36] C. Boyd, The DirectX 11 compute shader. (2008). [Online]. Available: <http://s08.idav.ucdavis.edu/boyd-dx11-compute-shader.pdf>.
- [37] A. Bruce, S. Sardy and P. Tseng, “Block coordinate relaxation methods for nonparametric signal de-noising”, *Proc. SPIE*, vol. 3391, pp. 75–86, 1998.
- [38] Y. Zhang, Y. Zhang, J. Zhang, and Q. Dai, “CCR: Clustering and collaborative representation for fast single image super-resolution,” *IEEE Trans. Multimedia.*, vol. 18, no. 3, pp. 405-417, Mar. 2016.
- [39] K. Chang, P. L. K. Ding, B. Li, “Single Image Super Resolution Using Joint Regularization”, *IEEE Signal Processing Letters*, vol. 25, pp. 596-600, 2018.
- [40] S. Sahni, G. Vairaktarakis, “The master-slave paradigm in parallel computer and industrial settings”, *Journal of Global Optimization*, vol. 9, pp. 357–377, 1996.
- [41] N. Ito, H. Shimizu, M. Kishi, et al., “Data-flow based execution mechanisms of Parallel and Concurrent Prolog”, *New Generation Computing*, vol. 3, pp. 15–41, 1985.
- [42] R. Chandra, R. Menon, L. Dagum, et. al., “Parallel Programming in OpenMP”,

11. List of Publications

- [6] B. Deka, H. U. Mullah, T. Barman and A. V. V. Prasad, “**JAMiSR- Joint Adaptive Multispectral Image Super-resolution via Sparse Representations and Applications**”, *IEEE Trans. on Comp. Img.* (Under review: TCI-01425-2021).
- [5] T. Barman, B. Deka and A.V.V Prasad, “**GPU-Accelerated Adaptive Dictionary Learning and Sparse Representations for Multispectral Image Super-resolution**”, *INDICON*, 2021 (Accepted).
- [4] H. U. Mullah, B. Deka, and AVV Prasad, “**Fast Multispectral Image Super-resolution via Sparse Representation**”, *IET Image Processing*, 2020
- [3] H. U. Mullah and B. Deka, T. Barman and A. V. V. Prasad, “**Sparsity Regularization Based Spatial/Spectral Super-resolution of Multispectral Imagery**”, *Inter. Conf. on Pattern Recognition and Machine Intelligence (PReMI)*, Springer, December 2019
- [2] H. U. Mullah and B. Deka, “**Parallel Multispectral Image Super-resolution Based on Sparse Representation**”, *IEEE 2nd Inter. Conf. on Innovations in Electronics, Signal Processing and Communications (IESC)*, February, 2019
- [1] H. U. Mullah and B. Deka, “**A Fast Satellite Image Super-Resolution Technique Using Multicore Processing**”, *Advances in Intelligent Systems and Computing*, Springer, vol. 734, 2018

12. Acknowledgement

I extend my deep and sincere gratitude to ISRO for providing me financial grants to carry out research through the RESPOND project scheme. Special thanks to Dr. M. A. Paul, Deputy Director, RESPOND, ISRO, Bangalore for his co-operation throughout the project and timely release of the funds. I am extremely grateful to Co-PI Shri A. V. V Prasad, Dy. Director, BG & DDA, NRSC for providing us continuous support, necessary guidance and advice during our progress review meetings conducted both at NRSC and online besides our time-to-time discussions on regular basis. His timely advice helped us immensely to complete this project as per objectives laid down in the project. Furthermore, I would also like to thank our former Co-PIs Ms. Archana Pragada, Scientist-SE, and Mr. Raja Shekhar, Scientist-SF, NRSC for their valuable guidance. I would like to offer my sincere thanks to Tezpur University for providing laboratory infrastructure. At last, but not the least, I highly acknowledge NESAC, Shillong for providing necessary support and information for the completion of the project.



GFR 12 - A
(See Rule 238(1))
FORM OF UTILISATION CERTIFICATE
FOR GRANTEE ORGNISATION

UTILISATION FOR THE YEAR 2020-21 in respect of GRANTS-IN-AID Released by Department of Space

1. Project title: Development of a Real-time Image super-resolution system using Parallel Processing Hardware for Remote Sensing Application
2. Name of the PI & Designation: Prof. Bhabesh Deka, Department of ECE, Tezpur University
3. Name & Address of the Institution :Tezpur University, Napaam, Tezpur, Assam
4. Name of the Scheme **Space Sciences (2792)/Space Applications(3508)**
5. Whether recurring or no recurring grants Non-recurring
6. Grants position at the beginning of the Financial year
 - (i) Cash in Hand/Bank 6,775.00
 - (ii) Unadjusted advances NIL
 - (iii) Total 6,775.00

7. Details of grants received, expenditure incurred and closing balances:(Actuals)

Rs. Cr.

Unspent Balances of grants received years (Figure as at Sl. no 3(iii))	Interest earned thereon	Interest deposited back to the Government	Grant received during the year			Total available Fund (1+2-3+4)	Expenditure incurred	Closing Balances (5-6)
			Sanction No. (i)	PAO Ref No:	Date (ii)			
1	2	3	4			5	6	7
6,775.00	NIL	NIL	NA	NA	NA	6,775.00	NIL	6,775.00

Details of Grant position at the end of the year

- (i) Cash in Hand/Bank 6,775.00
- (ii) Unadjusted advances NIL
- (iii) Total 6,775.00

Certified that I have satisfied myself that the conditions on which grants were sanctioned have been duly fulfilled/are being fulfilled and that I have exercised following checks to see that the money has been actually utilized for the purpose for which it was sanctioned:

- (i) The main accounts and the other subsidiary accounts and registers (including assets registers) are maintained as prescribed in the relevant Act/Rules/Standing instructions (mentioned the Act/Rules) and have been duly audited by designated auditors. The figures depicted above tally with the audited figures mentioned in financial statements/accounts.
- (ii) There exist internal controls for safeguarding public funds/assets, watching outcomes and achievements of physical targets against the financial inputs, insuring quality in asset creation etc. & the periodic evaluation of internal controls is exercised to ensure their effectiveness.
- (iii) To the best of our knowledge and belief, no transactions have been entered that are in violation of relevant Act/Rules/Standing instructions and scheme guidelines.
- (iv) The responsibilities among the key functionaries for execution of the scheme have been assigned in clear terms and are not general in nature.
- (v) The expenditure on various components of the scheme was in proportions authorized as per the scheme guidelines and terms and conditions of the grants-in-aid.
- (vi) It has been ensured that the physical and financial performance under the scheme Space Science has been according to the requirements, as prescribed in the guidelines issued by Govt. of India and the performance /targets achieved statement for the year to which the utilization of the fund resulted in outcomes given at **Reports submitted to the Programme Office.**

(vii) The utilization of the fund resulted in outcomes given at Reports duly forwarded to respective programme Office (to be formulated by the Department concerned as per their requirements/specifications).

(viii) Details of various schemes executed by the agency through grants-in-aid received from the same Ministry or from other Ministries is enclosed at Annexure

(ix) It is certified that for the same purpose we have not received any grant from the other Ministry/Department or any other Private Body.

Date:

Place:

Signature

Name.....

Chief Finance Officer (Head of the Finance)

Seal

Finance Officer

Tezpur University

Signature

Name.....

Head of the Organisation

Seal

Registrar

Tezpur University

Note: Any interest earned on unutilized funds during previous year and not remitted back to Govt. should be included in the opening balance at Sl. No. 3 (i)

GFR 12 - A
(See Rule 238(1))
FORM OF UTILISATION CERTIFICATE
FOR GRANTEE ORGANISATION

UTILISATION FOR THE YEAR 2020-2021 in respect of GRANTS-IN-AID Released by Department of Space

1. Project title: Development of a Real-time Image super-resolution system using Parallel Processing Hardware for Remote Sensing Application
2. Name of the PI & Designation: Prof. Bhabesh Deka, Department of ECE, Tezpur University
3. Name & Address of the Institution :Tezpur University, Napaam, Tezpur, Assam
4. Name of the Scheme **Space Sciences (2792)/Space Applications(3508)**
5. Whether recurring or no recurring grants **Non-recurring** Recurring
6. Grants position at the beginning of the Financial year
 - (i) Cash in Hand/Bank 3,47,855.00
 - (ii) Unadjusted advances NIL
 - (iii) Total 3,47,855.00

7. Details of grants received, expenditure incurred and closing balances:(Actuals)

Rs. Cr.

Unspent Balances of grants received years (Figure as at Sl. no 3(iii))	Interest earned thereon	Interest deposited back to the Government	Grant received during the year			Total available Fund (1+2-3+4)	Expenditure incurred	Closing Balances (5-6)
			Sanction No. (i)	PAO Ref No:	Date (ii)			
1	2	3	4			5	6	7
3,47,855.00	NIL	NIL	NA	NA	NA	3,47,855.00	2,24,000.00	1,23,855.00

Details of Grant position at the end of the year

- (i) Cash in Hand/Bank 1,23,855.00
- (ii) Unadjusted advances NIL
- (iii) Total 1,23,855.00

Certified that I have satisfied myself that the conditions on which grants were sanctioned have been duly fulfilled/are being fulfilled and that I have exercised following checks to see that the money has been actually utilized for the purpose for which it was sanctioned:

- (i) The main accounts and the other subsidiary accounts and registers (including assets registers) are maintained as prescribed in the relevant Act/Rules/Standing instructions (mentioned the Act/Rules) and have been duly audited by designated auditors. The figures depicted above tally with the audited figures mentioned in financial statements/accounts.
- (ii) There exist internal controls for safeguarding public funds/assets, watching outcomes and achievements of physical targets against the financial inputs, insuring quality in asset creation etc. & the periodic evaluation of internal controls is exercised to ensure their effectiveness.
- (iii) To the best of our knowledge and belief, no transactions have been entered that are in violation of relevant Act/Rules/Standing instructions and scheme guidelines.
- (iv) The responsibilities among the key functionaries for execution of the scheme have been assigned in clear terms and are not general in nature.
- (v) The expenditure on various components of the scheme was in proportions authorized as per the scheme guidelines and terms and conditions of the grants-in-aid.
- (vi) It has been ensured that the physical and financial performance under the scheme Space Science has been according to the requirements, as prescribed in the guidelines issued by Govt. of India and the performance /targets achieved statement for the year to which the utilization of the fund resulted in outcomes given at **Reports submitted to the Programme Office.**

(vii) The utilization of the fund resulted in outcomes given at Reports duly forwarded to respective programme Office (to be formulated by the Department concerned as per their requirements/specifications).

(viii) Details of various schemes executed by the agency through grants-in-aid received from the same Ministry or from other Ministries is enclosed at **Annexure**

(ix) It is certified that for the same purpose we have not received any grant from the other Ministry/Department or any other Private Body.

Date:

Place:

Signature

Name.....

Chief Finance Officer (Head of the Finance)

Seal

Finance Officer
Tezpur University

Signature

Name.....

Head of the Organisation

Seal

Registrar
Tezpur University

Note: Any interest earned on unutilized funds during previous year and not remitted back to Govt. should be included in the opening balance at Sl. No. 3 (i)

GFR 12 - A
(See Rule 238(1))
FORM OF UTILISATION CERTIFICATE
FOR GRANTEE ORGNISATION

UTILISATION FOR THE YEAR April, 2021-May,2021 in respect of GRANTS-IN-AID Released by Department of Space

1. Project title: Development of a Real-time Image super-resolution system using Parallel Processing Hardware for Remote Sensing Application
2. Name of the PI & Designation: Prof. Bhabesh Deka, Department of ECE, Tezpur University
3. Name & Address of the Institution :Tezpur University, Napaam, Tezpur, Assam
4. Name of the Scheme **Space Sciences (2792)/Space Applications(3508)**
5. Whether recurring or no recurring grants Non-recurring
6. Grants position at the beginning of the Financial year
 - (i) Cash in Hand/Bank 6,775.00
 - (ii) Unadjusted advances NIL
 - (iii) Total 6,775.00

7. Details of grants received, expenditure incurred and closing balances:(Actuals) Rs. Cr.

Unspent Balances of grants received years (Figure as at Sl. no 3(iii))	Interest earned thereon	Interest deposited back to the Government	Grant received during the year			Total available Fund (1+2-3+4)	Expenditure incurred	Closing Balances (5-6)
			Sanction No. (i)	PAO Ref No:	Date (ii)			
1	2	3	4			5	6	7
			Sanction No. (i)	PAO Ref No:	Date (ii)	Amount (iii)		
6,775.00	NIL	NIL	NA	NA	NA	6,775.00	NIL	6,775.00

Details of Grant position at the end of the year

- (i) Cash in Hand/Bank 6,775.00
- (ii) Unadjusted advances NIL
- (iii) Total 6,775.00

Certified that I have satisfied myself that the conditions on which grants were sanctioned have been duly fulfilled/are being fulfilled and that I have exercised following checks to see that the money has been actually utilized for the purpose for which it was sanctioned:

- (i) The main accounts and the other subsidiary accounts and registers (including assets registers) are maintained as prescribed in the relevant Act/Rules/Standing instructions (mentioned the Act/Rules) and have been duly audited by designated auditors. The figures depicted above tally with the audited figures mentioned in financial statements/accounts.
- (ii) There exist internal controls for safeguarding public funds/assets, watching outcomes and achievements of physical targets against the financial inputs, insuring quality in asset creation etc. & the periodic evaluation of internal controls is exercised to ensure their effectiveness.
- (iii) To the best of our knowledge and belief, no transactions have been entered that are in violation of relevant Act/Rules/Standing instructions and scheme guidelines.
- (iv) The responsibilities among the key functionaries for execution of the scheme have been assigned in clear terms and are not general in nature.
- (v) The expenditure on various components of the scheme was in proportions authorized as per the scheme guidelines and terms and conditions of the grants-in-aid.
- (vi) It has been ensured that the physical and financial performance under the scheme Space Science has been according to the requirements, as prescribed in the guidelines issued by Govt. of India and the performance /targets achieved statement for the year to which the utilization of the fund resulted in outcomes given at **Reports submitted to the Programme Office.**

- (vii) The utilization of the fund resulted in outcomes given at Reports duly forwarded to respective programme Office (to be formulated by the Department concerned as per their requirements/specifications).
- (viii) Details of various schemes executed by the agency through grants-in-aid received from the same Ministry or from other Ministries is enclosed at **Annexure**
- (ix) It is certified that for the same purpose we have not received any grant from the other Ministry/Department or any other Private Body.

Date:

Place:

Signature

Name.....

Chief Finance Officer (Head of the Finance)

Seal

Finance Officer
Yezpur University

Signature

Name.....

Head of the Organisation

Seal

Registrar
Yezpur University

Note: Any interest earned on unutilized funds during previous year and not remitted back to Govt. should be included in the opening balance at Sl. No. 3 (i)

GFR 12 - A
(See Rule 238(1))
FORM OF UTILISATION CERTIFICATE
FOR GRANTEE ORGANISATION

UTILISATION FOR THE YEAR April, 2021-MAY,2021 in respect of GRANTS-IN-AID Released by Department of Space

1. Project title: Development of a Real-time Image super-resolution system using Parallel Processing Hardware for Remote Sensing Application
2. Name of the PI & Designation: Prof. Bhabesh Deka, Department of ECE, Tezpur University
3. Name & Address of the Institution :Tezpur University, Napaam, Tezpur, Assam
4. Name of the Scheme **Space Sciences (2792)/Space Applications(3508)**
5. Whether recurring or no recurring grants **Non-recurring Recurring**
6. Grants position at the beginning of the Financial year
 - (i) Cash in Hand/Bank 1,23,855.00
 - (ii) Unadjusted advances NIL
 - (iii) Total 1,23,855.00

7. Details of grants received, expenditure incurred and closing balances:(Actuals)

Rs. Cr.

Unspent Balances of grants received years (Figure as at Sl. no 3(iii))	Interest earned thereon	Interest deposited back to the Government	Grant received during the year			Total available Fund (1+2-3+4)	Expenditure incurred	Closing Balances (5-6)
			Sanction No. (i)	PAO Ref No:	Date (ii)			
1	2	3	4			5	6	7
1,23,855.00	NIL	NIL	NA	NA	NA	1,23,855.00	NIL	74,729.00

Details of Grant position at the end of the year

- (i) Cash in Hand/Bank 74,729.00
- (ii) Unadjusted advances NIL
- (iii) Total 74,729.00

Certified that I have satisfied myself that the conditions on which grants were sanctioned have been duly fulfilled/are being fulfilled and that I have exercised following checks to see that the money has been actually utilized for the purpose for which it was sanctioned:

- (i) The main accounts and the other subsidiary accounts and registers (including assets registers) are maintained as prescribed in the relevant Act/Rules/Standing instructions (mentioned the Act/Rules) and have been duly audited by designated auditors. The figures depicted above tally with the audited figures mentioned in financial statements/accounts.
- (ii) There exist internal controls for safeguarding public funds/assets, watching outcomes and achievements of physical targets against the financial inputs, insuring quality in asset creation etc. & the periodic evaluation of internal controls is exercised to ensure their effectiveness.
- (iii) To the best of our knowledge and belief, no transactions have been entered that are in violation of relevant Act/Rules/Standing instructions and scheme guidelines.
- (iv) The responsibilities among the key functionaries for execution of the scheme have been assigned in clear terms and are not general in nature.
- (v) The expenditure on various components of the scheme was in proportions authorized as per the scheme guidelines and terms and conditions of the grants-in-aid.
- (vi) It has been ensured that the physical and financial performance under the scheme Space Science has been according to the requirements, as prescribed in the guidelines issued by Govt. of India and the performance /targets achieved statement for the year to which the utilization of the fund resulted in outcomes given at **Reports submitted to the Programme Office.**

(vii) The utilization of the fund resulted in outcomes given at Reports duly forwarded to respective programme Office (to be formulated by the Department concerned as per their requirements/specifications).

(viii) Details of various schemes executed by the agency through grants-in-aid received from the same Ministry or from other Ministries is enclosed at **Annexure**

(ix) It is certified that for the same purpose we have not received any grant from the other Ministry/Department or any other Private Body.

Date:

Place:

Signature

Name.....

Chief Finance Officer (Head of the Finance)

Seal

Finance Officer
Tezpur University

Signature

Name.....

Head of the Organisation

Seal

Registrar
Tezpur University

Note: Any interest earned on unutilized funds during previous year and not remitted back to Govt. should be included in the opening balance at Sl. No. 3 (i)

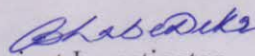
FORM OF FUND UTILISATION CERTIFICATE

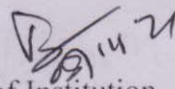
(PROJECTS/SCHEMES)

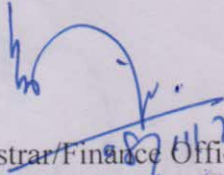
2020-21

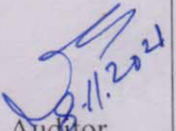
Name of the Nodal Institution/ Department of Organisation	Tezpur University/ Dept. of ECE
Name of the Project /Scheme	Development of a Real-time Image Super-resolution System using Parallel Processing Hardware for Remote Sensing Application/RESPOND- ISRO (DOS)

Certified that out of ₹ NIL only Grant-in-aid sanctioned and released during the financial year 2020-21 in favour of Registrar, Tezpur University on the project "Development of a Real-time Image Super-resolution System using Parallel Processing Hardware for Remote Sensing Application" for the Third year by Government of India, Department of Space, Bangalore, an amount of ₹3,54,630.00 (Three Lakh Fifty Four Thousand Six Hundred Thirty) only was unspent balance carried forwarded from the previous year 2019-20. Out of these, a sum of ₹2,24,000.00 (Two Lakh Twenty Four Thousand) only has been utilized during Apr. 2020-Mar. 2021 on the project /Scheme and leaving an unspent balance of ₹1,30,630.00 (One Lakh Thirty Thousand Six Hundred Thirty) only to be utilized on or before 31.05.2021. The unspent balance in any on completion of the project will be/was surrendered to Pay and Accounts officer, Department of space, Bangalore duly supported by consolidated Audited statement of account, reports, papers, compendium of data analysis etc.


Project Investigator
(with seal)


Head of Institution
(with seal)
Registrar
Tezpur University


Registrar/Finance Officer
(with seal)
Finance Officer
Tezpur University


Auditor
(with seal)
Internal Audit Officer
Tezpur University

FORM OF FUND UTILISATION CERTIFICATE

(PROJECTS/SCHEMES)

April, 2021-May, 2021

Name of the Nodal Institution/ Department of Organisation	Tezpur University/ Dept. of ECE.
Name of the Project /Scheme	Development of a Real-time Image Super-resolution System using Parallel Processing Hardware for Remote Sensing Application/RESPOND- ISRO (DOS)

Certified that out of ₹ NIL only Grant-in-aid sanctioned and released during the financial year 2021-22 in favour of Registrar, Tezpur University on the project “Development of a Real-time Image Super-resolution System using Parallel Processing Hardware for Remote Sensing Application” for the Third year by Government of India, Department of Space, Bangalore, an amount of ₹1,30,630.00 (One Lakh Thirty Thousand Six Hundred Thirty) only was unspent balance carried forwarded from the previous year 2020-21. Out of these, a sum of ₹49,126.00 (Forty Nine Thousand One Hundred Twenty Six) only has been utilized during April, 2021-May, 2021 on the project /Scheme and leaving an unspent balance of ₹81,504.00 (Eighty One Thousand Five Hundred Four) only. The unspent balance in any on completion of the project will be/was surrendered to Pay and Accounts officer, Department of space, Bangalore duly supported by consolidated Audited statement of account, reports, papers, compendium of data analysis etc. (Cheque/DD No..... 443499
Dated. 17/11/2021)

[Signature]
Project Investigator
(with seal)

[Signature]
Head of Institution
(with seal)
Registrar
Tezpur University

[Signature]
Registrar/Finance Officer
(with seal)
Finance Officer
Tezpur University

[Signature]
Auditor
(with seal)
Internal Audit Office
Tezpur University

**AUDITED STATEMENT SHOWING THE HEAD WISE EXPENDITURE FOR THE
FINANCIAL YEAR 2020-2021**

1. **Project Title:** Development of a Real-time Image super-resolution system using Parallel Processing Hardware for Remote Sensing Application
2. **Name of the PI & Designation:** Prof. Bhabesh Deka, Department of ECE, Tezpur University
3. **Name & Address of Institution:** Tezpur University, Napaam, Tezpur, Assam
4. **ISRO/DOS Letter/Sanction Order No & Date:** ISRO/RES/4/642/17-18 dated 07.07.2017
5. **Period of Statement:** FY 2020-21
6. **Total Grants Approved/ Grants for the Year:** Total grant approved: **24, 88,000.00**
Grant approved for FY 2020-21: Nil
7. **University/Institution Sanction No & Date:** DoRD/ECE/BD/20-336
8. **Expenditure Statement for the period:** FY 2020-21

SL. No	Item of Expenditure	In Rupees (₹)						Remarks
		Amount sanctioned by ISRO during FY 2020-2021	Opening Balances as on 1/04/2020	Amount released by ISRO in FY 2020-21	Fund Available	Expenditure incurred FY 2020-21	Unspent balance as on 1/04/2021	
1.	Equipment	Nil	6775.00	Nil	6,775.00	Nil	6775.00	
2.	SRF@28,000.00pm	Nil	2,17,000.00	Nil	2,17,000.00	2,24,000.00	(-)7,000.00	
3.	Satellite Data	Nil	25,000.00	Nil	25,000.00	Nil	25,000.00	
4.	Consumables and Supplies	Nil	24,616.00	Nil	24,616.00	Nil	24,616.00	Balance amount of ₹ 24,616.00 for consumables is committed expenditure for FY 2019-20 will be adjusted by May, 2021.
5.	Travel & Field Logistics	Nil	56247.00	Nil	56247.00	Nil	56247.00	Due to COVID-19 outbreak, travel grant could not be utilized.
6.	Contingency	Nil	24,992.00	Nil	24,992.00	Nil	24,992.00	Balance amount of ₹ 24,992.00 for contingency is committed expenditure for FY 2018-19 will be adjusted by May, 2021.
7.	Overhead	Nil	Nil	Nil	Nil	Nil	Nil	
	Grand Total	Nil	3,54,630.00	Nil	3,54,630.00	2,24,000.00	1,30,630.00	The amount of committed expenditure is ₹ 49,608.00***

***Committed expenditure of ₹ 49,608.00 (Forty Nine Thousand Six Hundred Eight) only will be spent at the end of FY 2020-21. Unspent amount of 81,022.00 (Eighty One Thousand Twenty Two) only will remain unutilized at the end of the FY 2020-21.

Bhabesh Deka
Project Investigator
(with seal)

Bhabesh Deka
Head of Institution
(with seal)

Bhabesh Deka
Registrar
Tezpur University

Bhabesh Deka
Registrar/Finance Officer
(with seal)

Bhabesh Deka
Finance Officer
Tezpur University

Bhabesh Deka
Auditor
(with seal)

Bhabesh Deka
Internal Audit Officer
Tezpur University

**AUDITED STATEMENT SHOWING THE HEAD WISE EXPENDITURE FOR THE
FINANCIAL YEAR April 2021-May, 2021**

1. **Project Title:** Development of a Real-time Image super-resolution system using Parallel Processing Hardware for Remote Sensing Application
2. **Name of the PI & Designation:** Prof. Bhabesh Deka, Department of ECE, Tezpur University
3. **Name & Address of Institution:** Tezpur University, Napaam, Tezpur, Assam
4. **ISRO/DOS Letter/Sanction Order No & Date:** ISRO/RES/4/642/17-18 dated 07.07.2017
5. **Period of Statement:** FY 2021-22
6. **Total Grants Approved/ Grants for the Year:** Total grant approved: **24, 88,000.00**
Grant approved for FY 2021-22: Nil
7. **University/Institution Sanction No:** DoRD/ECE/BD/20-336
8. **Expenditure Statement for the period:** April, 2021-May, 2021

SL. No	Item of Expenditure	In Rupees (₹)						Remarks
		Amount sanctioned by ISRO during FY 2020-2021	Opening Balances as on 1/04/2021	Amount released by ISRO in FY 2020-21	Fund Available	Expenditure incurred FY April, 2021-May, 2021	Unspent balance as on 31/05/2021	
1.	Equipment	Nil	6775.00	Nil	6,775.00	Nil	6775.00	
2.	SRF@28,000.00 pm	Nil	(-)7,000.00	Nil	Nil	Nil	(-)7,000.00	
3.	Satellite Data	Nil	25,000.00	Nil	25,000.00	Nil	25,000.00	
4.	Consumbles and Supplies	Nil	24,616.00	Nil	24,616.00	26,180.00	(-)1,564.00	The excess amount of 1,564.00 was adjusted from contingency head.
5.	Travel & Field Logistics	Nil	56247.00	Nil	56247.00	Nil	56247.00	Due to COVID-19 outbreak, travel grant could not be utilized.
6.	Contingency	Nil	24,992.00	Nil	24,992.00	22,946.00	2,046.00	
7.	Overhead	Nil	Nil	Nil	Nil	Nil	Nil	
Grand Total		Nil	1,30,630.00	Nil	1,37,630.00	49,126.00	81,504.00	

Unspent amount of 88,504.00 (Eighty One Thousand Five Hundred Four) only will remain unutilized at the end of the FY April 2021-May, 2021.

Bhabesh Deka
Project Investigator
(with seal)

Bhabesh Deka
Head of Institution
(with seal)
Registrar
Tezpur University

[Signature]
Registrar/Finance Officer
(with seal)
Finance Officer
Tezpur University

[Signature]
Auditor
(with seal)
Internal Audit Officer
Tezpur University