iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** III **Month of publication:** March 2024

DOI: https://doi.org/10.22214/ijraset.2024.58850

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue III Mar 2024- Available at www.ijraset.com

FPGA-Based Satellite Vision Systems using Verilog HDL

Palak Jain[1], Atif Ali[2], Ahtisham Ahmad[3], Ashish Yadav[4], Ayush Kumar[5]
Department of Electronics and Communication Engineering, ABES Engineering College, AKTU, Ghaziabad, India

Abstract: ***It is believed that the most effective way to collect information about the Earth's surface is through high- quality satellite images. Extracting a feature from an image is really difficult because you have to choose the best image segmentation methods and combine many strategies to find the Region in the most effective manner. This study makes recommendations for the classification techniques for objects in the satellite. On high-resolution satellite images, applying image processing methods. The methods used to define region mostly focus on urban, agricultural, and forest regions. There are several methods for extracting these traits. Using a Grey Level Cooccurrence Matrix is the most used method. It is employed to unveil specific characteristics regarding the spatial arrangement of gray levels in the texture image.***
The Grey Level Co-occurrence Matrix (GLCM) captures statistical details of neighboring pixels in an image, enabling the computation of textural features that enhance the comprehension of visual content. This research presents a VLSI implementation aimed at extracting four texture characteristics from the grey level co-occurrence matrix. Verilog was employed to model the hardware, with MATLAB used for software simulation. The simulation utilized the Verilog HDL language from the XILINX tool, and the implementation was executed on the SPARTAN FPGA board.
Keywords: Satellite image, features, field programmable gate arrays (FPGAs), Image segmentation

I. INTRODUCTION

A frequent phase in Verilog-based satellite image processing is the creation and execution of digital signal processing algorithms to enhance or extract information from satellite images. Verilog is a widely used language for describing hardware that is used in the design of digital circuits and systems. Below is a broad overview of the actions you may take to utilize Verilog to handle satellite images:

1) *Recognize the Image Format:* Digital formats like JPEG or BMP are typically used for satellite images. Recognize the picture formats that you will be using.
2) *Describe the Conditions:* Clearly state what processing operations you wish to carry out on the satellite photos. This might involve operations like color enhancement, edge detection, and picture filtering, among others.
3) *Construct the Algorithm:* Select or create the image processing algorithm that Verilog will use. Fourier transformations, convolution, and filtering are examples of common algorithms.
4) *Algorithm to Verilog Code Conversion:* Convert the algorithm to Verilog programming. This entails utilizing Verilog syntax to describe the hardware elements and their relationships.
5) *Play with the Design:* To simulate your Verilog code, use tools such as Model Sim. This stage assists in locating and resolving possible problems prior to implementing hardware.
6) *Create a circuit for FPGA or ASIC:* Utilize synthesis software (such as Intel Quartus or Xilinx Vivado) to transform your Verilog code into a netlist suitable for FPGA or ASIC implementation.
7) *Hardware Setup:* Utilize the synthesized design to program the FPGA or ASIC. Attach the source of the satellite picture to your hardware system's input.
8) *Examine and Fix:* Check to see that the hardware implementation yields the desired outcomes. To ensure that your solution is working as intended, use real satellite data and test photos.
9) *Enhancement:* Make sure your design is optimized for power, performance, and resource utilization. This might entail tweaking the synthesis parameters and improving the Verilog code.
10) *System Integration:* Integrate your image processing component with other components and interfaces as necessary if it is a part of a bigger system.
11) *Record-keeping:* For future use or cooperation, record your Verilog code, the decisions you made regarding design, and any other pertinent data.

©IJRASET: All Rights are Reserved

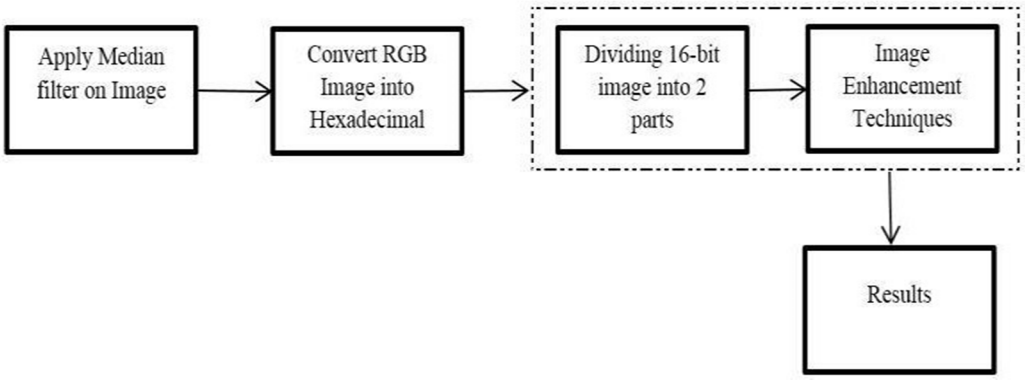
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue III Mar 2024- Available at www.ijraset.com

II. LITERATURE REVIEW

The most crucial instrument for gathering data about the Earth's surface is the satellite picture. Satellite photos may be used to look at a variety of interesting human activities, such as suburban regions, residential developments, and the proportion of forest land in a certain area. Over the past 20 years, remote sensing imagery has found application in a various sector, encompassing areas such as land-use and land-cover. Furthermore, because of its extensive coverage, satellite data may be used to calculate estimations of agricultural acreage [1].

Over half of the world's population resides in Asia, which is also the main rice-producing region. Monitoring, mapping, and forecasting paddy rice cultivation can lead to effective management of the production of food and water [2].

Large-scale data processing methods may now be implemented with great promise thanks to Field Programmable Gate Array (FPGA) technology. The dimensions of images, the depth of bits, computational intricacy, and processing time have all seen an escalation in applications like remote sensing and satellite image processing. The ideal solution is to implement hardware, however doing so necessitates considering how much longer it takes to construct a hardware design [3]. FPGA offers multispectral image compression that is effective in reducing bandwidth and storage requirements. [4][5]. Without sacrificing dependability, it lowers latency and mathematical complexity [6]. multispectral data on board satellite imaging systems to lessen the need for a downlink connection and to more effectively utilize multispectral data sets across a range of applications [7][8]. Reducing the amount of data is the primary difficulty in satellite image processing. FPGA offers multispectral image compression that is effective in reducing bandwidth and storage requirements [9]. It decreases latency and mathematical complexity without sacrificing dependability [10]. Feature extraction seeks to diminish the resources needed for precisely representing extensive data. Managing the multitude of variables inherent in intricate data analysis stands out as a key challenge [11].



MATLAB and XILINX are the two operational tools used in the system design. A software program called MATLAB is used to compute manual calculations. The Verilog HDL language was employed to develop the filtering algorithm, utilizing the quarts tool.

The language used for hardware representation is Verilog. Finally, contrast XILINX's and MATLAB's PSNR calculations. The following is a full explanation of this technique-

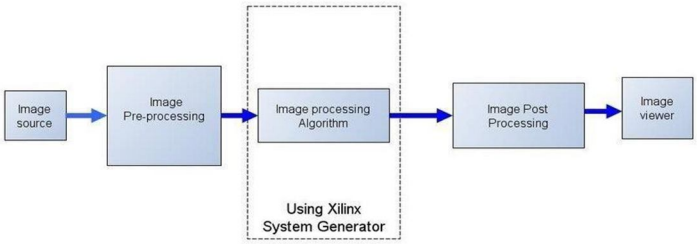
1) *Preprocessing:* Enhancing some visual properties that are crucial for additional processing or suppressing undesired distortions is the primary goal of preprocessing. The input image is a colored, three-dimensional picture that has to be converted to grayscale before being downsized to a 25*25 resolution. The XILINX simulator receives the resized picture as input for additional processing.
2) *Grayscale Conversion:* An RGB picture is converted to grayscale using MATLAB. Consequently, an 8-bit picture with potential values ranging from 0 to 255 is obtained. However, the picture captured by the camera in real time is RGB, but it lacks a function, thus it is converted to greyscale using the basic formula below. Grayscale: 0.2*R + 0.59*G + 0.11*B Where R-Red value, G-Green Value, B-Blue Value
3) *Converting Image to Hexadecimal File:* The bmp picture is transformed from bitmap to hexadecimal format in order to be processed on FPGA. Since the hardware equipment cannot interpret direct images, the bitmap picture must be transformed to hex file using MATLAB. The bitmap image's R, G, and B data are included in the hex file, which has dimensions of 25 by 25.
4) *Impulse Noise:* Impulse noise is caused by timing errors during the digitizing process, damaged pixels in the camera sensors, and incorrect memory locations. Other names for the impulse noise include shot noise, spike noise, and salt and pepper noise. In an 8-bit picture, the noise is represented by 255 pixels for salt and 0 pixels for pepper.

©IJRASET: All Rights are Reserved

5) *Using A Mean Filter to Remote Noise:* The Mean filter is a linear technique used to eliminate noise from the picture. It is employed to lessen the variance in intensity between successive pixels. It is frequently used to smooth out and eliminate noise from images. The mean filter operates by substituting the value of each pixel in an image, one pixel at a time, with the average value of all its neighboring pixels, including itself. Therefore, the values of the pixels that do not correspond to their surroundings are eliminated upper Noise.
6) *Ratio of Peak Signal to Noise:* The mean square error and peak signal-to-noise ratio serve as two metrics for evaluating the effectiveness of image compression. The peak signal-to-noise ratio measures the ratio between the maximum potential power of a signal and the detrimental noise that degrades the representation's quality. This ratio is expressed on a logarithmic decibel scale.
7) *Image Conversion from 2D TO 1D:* The XILINX program cannot directly read the image file; it necessitates the prior conversion of the file into the hex file format through MATLAB. In XILINX, picture values are expressed in an array format, contrasting with MATLAB's 2D format. Each memory address within XILINX stores the 16-bit data.
8) *File Reading using XILINX:* In XILINX tool, the hex file is read using the Verilog HDL language. The image has dimensions of 25 by 25 and is represented as a matrix in hex file with values of 25 by 25 * 3 = 1875.The values are not arranged in the right format in relation to the image. The main difficulty is arranging the hexadecimal values such that they match the image's unsigned decimal value. The hex file is organized in relation to the picture using the Verilog HDL programming language. The mean filtering technique effectively eliminates salt and pepper noise from an image, often termed as average filtering. This method involves systematically traversing the image pixel by pixel, replacing each pixel value with the average of its neighboring values. Verilog HDL is employed to articulate the filtering algorithm, and the hexadecimal values of the filtered image are showcased in the waveform generated by simulating the filtering process using the XILINX tool.

III. PROPOSED SYSTEM ALGORITHM

The proposed system architecture describes the analysis of the texture feature parameter obtained with HDL and a simulator. The input for this method is the multispectral image obtained by the IRS satellite. The input image is split into an urban image and a vegetation image using the preprocessing method.



In order to detect changes, this work extracts texture characteristics from multispectral satellite pictures using GLCM. The image was preprocessed and converted to a hex file since the hardware module is unable to comprehend it. Using the GLCM computing unit, the grey level co-occurrence matrix was recovered from the image. Four second order properties (correlation, energy, contrast, and angular second moment) are computed using the GLCM matrix. Finally, compare the features of the software and hardware.

IV. PREPROCESSING



Figure 1. satellite image

The multispectral satellite picture input is displayed in Fig. 1. The IRS satellite sensor to take the picture.

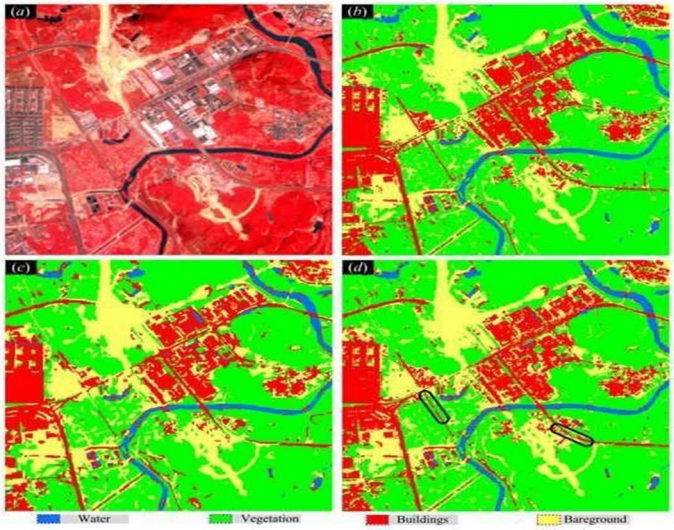


Figure.2 Processed image

In fig. 2, the preprocessing step involves cropping and dividing the input multispectral satellite image into pictures of vegetation and cities. There is a reduction in size to 14 by 14. The XILINX application doesn't accept the image directly as input; it requires the conversion of the image to a hexadecimal file format. Figure 2 illustrates the arrangement of pixel values from the image in an array format within that hexadecimal file.

If the image is 14 by 14, the 196 values are arranged in an array structure in the hex file.

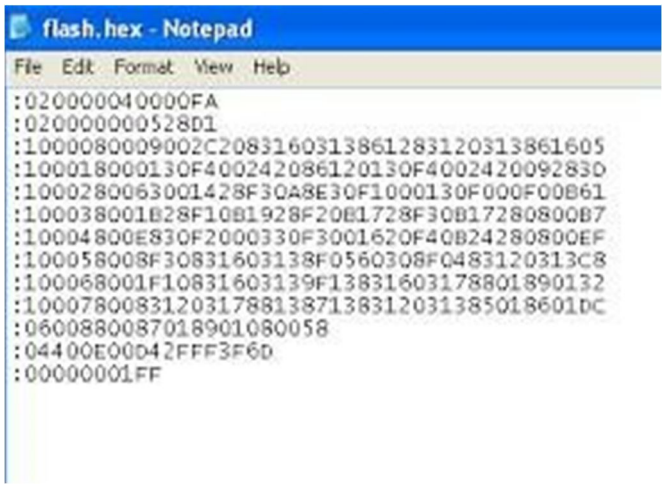


Figure.3 hexadecimal file

V. GLCM CALCULATION UNIT

In the realm of statistical texture analysis, texture features are determined by examining the statistical distribution of observed intensity combinations at specific locations within an image. Statistical properties, including those of first, second, and higher orders, are categorized based on the number of intensity points (pixels) within each set. Utilizing the Grey Level Co-occurrence Matrix (GLCM) method allows for the extraction of second-order statistical texture properties. The GLCM is derived from a Grayscale Image and computes the frequency of occurrences—horizontally, vertically, or diagonally— where a pixel with a gray-level value of i is adjacent to neighboring pixels with a value of j. Both the Grey Level Co-occurrence Matrix (GLCM) and the associated texture feature calculations are employed as techniques in image analysis. These calculations utilize the GLCM contents to gauge the intensity fluctuations at the respective pixels. The technique of creating co-occurrence matrices for a grayscale image is summarized in Figure 4 below.

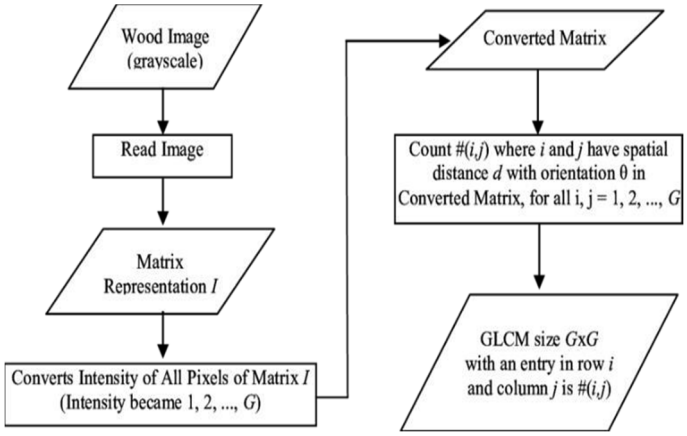


Figure. 4 block diagrams of GLCM

The intensity of each pixel in the picture divides the objects into distinct groups. The clusters matching to ROI can be found using a variety of methods. In the original design, the threshold is determined by selecting pixels at random for every item and manually identifying the correct group. The maximum, lowest, and average values are among the statistical aspects that are examined in this sampling data. Table 1 displays the test data that was recorded, and the threshold for the example pixels is determined using these statistical characteristics of each group.

VI. EXPERIMENTAL RESULT

The outcomes of employing the GLCM approach for texture feature extraction from satellite images through FPGA implementation were simulated using MATLAB and XILINX software.

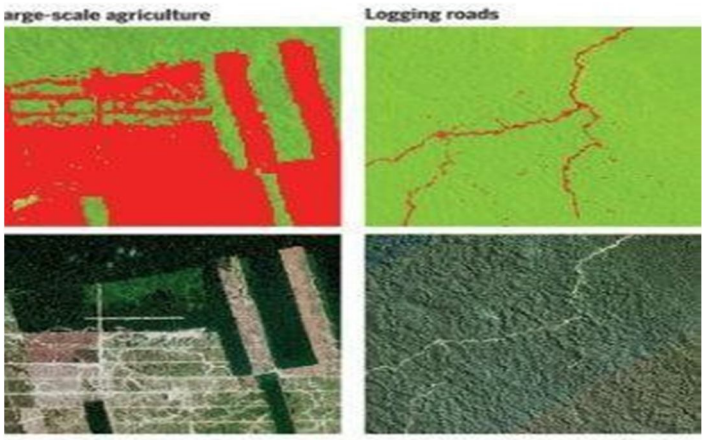


Figure.5 Result

Eight bits of memory are used for each pixel value in the register. The values of the image's pixels are contained in the hexadecimal file that is seen in Figure 3. File reading is used to read the pixel values from the hex file and store them in temporary memory. The simulation result for the pixel value of the improved satellite image is depicted in Figure 5 above.

VII. CONCLUSION

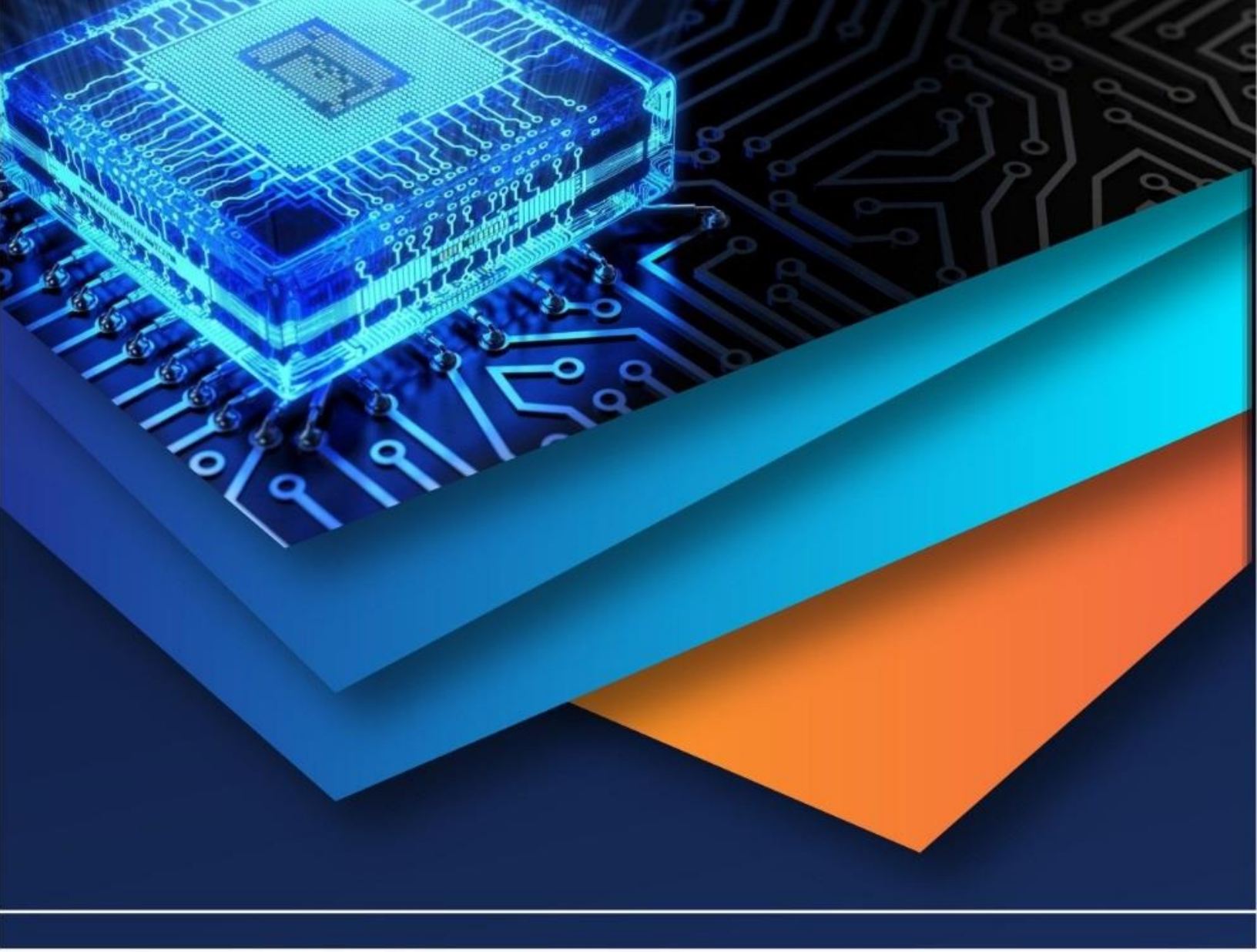
Field Programmable Gate Array (FPGA) technology has surfaced as a promising avenue for implementing algorithms designed to handle extensive datasets. The research aimed to implement the GLCM method in hardware for extracting texture features from multispectral images and assess the performance of the feature extraction parameters based on both simulation and implementation results. Through the use of XILINX for hardware implementation and MATLAB for simulation, the goal has been accomplished. It was successfully possible to implement the novel proposal for the extraction of texture characteristics from multispectral satellite images using the Grey Level Co- occurrence Matrix in the Verilog HDL language.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Future work will thus focus on implementing automated histogram categorization using the image's histogram information as well as on identifying smaller, more focused processing in images, such as a rice field or a particular patch of land.

REFERENCES

[1] Soontranon, N., Srestasathiern, P., and Rakwatin, P., 2015b. Rice crop calendar based on phenology analysis from time- series images. In Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI- CON), 2015 12th International Conference on, pages 1–5. IEEE
[2] Y.H. Tseng., P.H. Hsu and I.H Chen., 1998. Asian conference on remote sensing "Automatic recognition of rice fields from multitemporal satellite images"
[3] Ernestina Martel, Raul Guerra, Sebastian, Lopez and Roberto Sarmiento, ―A GPU-Based Processing Chain for Linearly Unmixing Hyperspectral Images'', IEEE journal of selected topics in applied earth observations and remote sensing, vol. 10, no. 3, march 2017
[4] QianXu, SrenivasVaradarajan, ChaitaliChakrabarti,Lina J. Karam,"A Distributed Canny Edge Detector: Algorithm and FPGA Implementation'',IEEE Transactions on image processing volume: 23, Issue: 7, July 2014
[5] Sebastian Lopez, Tanya Vladimirova, Carlos Gonza ´lez, Javier Resano, Daniel Mozos and Antonio Plaza, "The Promise of ReconFigureurable Computing for Hyperspectral Imaging OnboardSystems: A Review and Trends'' IEEE | Vol. 101, No. 3, March 2013
[6] Sebastian Lopez, Tanya Vladimirova, Carlos Gonza ´lez, Javier Resano, Daniel Mozos and Antonio Plaza, "The Promise of ReconFigureurable Computing for Hyperspectral Imaging OnboardSystems:A Review and Trends''IEEE | Vol. 101, No. 3, March 2013
[7] C. Gonzalez, S. Sanchez, A. Paz, J. Resano, D. Mozos, and A. Plaza, "Use of FPGA or GPU-based architectures for remotely sensed hyperspectral image processing," Integr. VLSI J., vol. 46, no. 2, pp. 89–103, 2013.
[8] Carlos González, Javier Resano,Antonio Plaza and Daniel Mozos,"FPGA Implementation of Abundance Estimation for Spectral Unmixing of Hyperspectral Data Using the Image Space Reconstruction Algorithm''IEEE Journal Of Selected Topics In Applied Earth Observations And Remote Sensing, Vol. 5, No. 1, February 2012
[9] Qian Xu, Srenivas Varadarajan, Chaitali Chakrabarti, Lina J.Karam,―A Distributed Canny Edge Detector: Algorithm and FPGA Implementation'',IEEE Transactions on image processing volume: 23, Issue: 7, July 2014.
[10] V.Pavithra, R.Pounroja, Dr.B.SathyaBama, "Machine Vision Based Automatic Sorting of Cherry Tomatoes'', IEEE Sponsored 2nd International Conference On Eletronics And Communication System(icecs 2015).



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)