A

DISSERTATION

ON

HYPERSPECTRAL ANOMALY DETECTION

In Partial Fulfillment Of The Requirement For The Award Of The Degree Of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

R Sivaram	-	[4511-19-733-008]
M Shiva	-	[4511-19-733-041]
B Srivani	-	[4511-19-733-004]

Under the Esteemed Guidance of

Mrs. M. JYOTHI RANI

Assistant Professor (C)

Dept. of CSE, UCE&T, MGU.

UNIVERSITY COLLEGE OF ENGINEERING & TECHNOLOGY

MAHATMA GANDHI UNIVERSITY

COMPUTER SCIENCE & ENGINEERING

NALGONDA – 508254

APRIL - 2023

A

Major Project Report

On

HYPERSPECTRAL ANOMALY DETECTION

Submitted to

UNIVERSITY COLLEGE OF ENGINEERING & TECHNOLOGY

MAHATMA GANDHI UNIVERSITY

In Partial Fulfillment Of The Requirement For The Award Of The Degree Of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

R Sivaram	-	[4511-19-733-008]
M Shiva	-	[4511-19-733-041]
B Srivani	-	[4511-19-733-004]

Under the Esteemed Guidance of

Mrs. M. JYOTHI RANI

Assistant Professor (C)

Dept. of CSE, UCE&T, MGU.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

UNIVERSITY COLLEGE OF ENGINEERING & TECHNOLOGY

MAHATMA GANDHI UNIVERSITY

NALGONDA – 508254

APRIL - 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

UNIVERSITY COLLEGE OF ENGINEERING & TECHNOLOGY

MAHATMA GANDHI UNIVERSITY

NALGONDA – 508254

CERTIFICATE

This is to certify that this project entitled "**Hyperspectral Anomaly Detection**" is a bonafide work carried out by R Sivaram [4511-19-733-008], M Shiva [451119-733-041], B Srivani [4511-19-733-004] in partial fulfillment of the requirements for the award of Major Project in Bachelor of Technology in Computer science and Engineering from Mahatma Gandhi University, Nalgonda during the year 2022-2023. This project report satisfies the academic requirements in respect of the project work prescribed for the said Degree.

Project Guide	Head of the Department
Mrs. M. JYOTHI RANI	**Mrs. CH. SWARNALATHA**
Assistant Professor (C)	Assistant Professor (C)
Dept. of CSE	Dept. of CSE
UCE&T, MGU	UCE&T, MGU
Nalgonda	Nalgonda

Viva-voice held on…………………   SIGNATURE OF EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our project guide **Mrs. M. JYOTHI RANI,** Assistant Professor (C), Department of Computer Science and Engineering, UCE&T, MGU, for her guidance with unsurpassed knowledge and immense encouragement.

We are grateful to **Mrs. CH. SWARNALATHA**, Head of the Department, Computer Science and Engineering, for the valuable guidance and for permitting us to carry out this project.

We would like to thank Principal **Prof. R. REKHA** for her expert guidance and encouragement at various levels of our project.

We show our gratitude to our honourable Registrar **Prof. KRISHNA RAO THUMMA** for having provided all the facilities and support.

We avail this opportunity to express our deep sense of gratitude to our honourable Vice Chancellor **Prof. CH. GOPAL REDDY,** congenial atmosphere to complete this project successfully.

We thank all **teaching faculty** of Department of CSE, whose suggestions during reviews helped us in accomplishment of our project.

We express our thanks to all those who contributed directly or indirectly for the successful completion of our project work.

With Gratitude,
Our Team,

R Sivaram [4511-19-733-008]

M Shiva [4511-19-733-041]

B Srivani [4511-19-733-004]

DECLARATION

We, hereby declare that the Major Project entitled, **“Hyperspectral Anomaly Detection”** has been carried out by us under the guidance of **Mrs. M. Jyothi Rani, Asst. Professor**, UCE&T, MGU, Nalgonda, in partial fulfillment of the requirements for the award of degree of Bachelor of Technology in **Computer Science and Engineering**, of the University College of Engineering & Technology, Mahatma Gandhi University, Nalgonda during the academic year 2022-23. The work done in this dissertation report is original and it has not been submitted for any other degree in any university.

(Signature of the Student)

R Sivaram [4511-19-733-008]

M Shiva [4511-19-733-041]

B Srivani [4511-19-733-004]

ABSTRACT

Hyperspectral images (HSIs) are of wide spectral range and high spectral resolution so it contains rich spectral information to discriminate physical properties of different materials. Therefore, HSI finds many applications in different areas. In recent years, many anomaly detection methods based on the HSI have emerged. Especially, it is more appealing to detect interesting materials, e.g., targets or anomalies which are very different form background pixels in an image scene . Anomaly detection based on the HSI has been widely studied in the fields of agriculture, mineral exploration, maritime rescue, and military defense. Different from the supervised target detection, anomaly detection is achieved without knowing the prior information of targets. The difference of statistical distributions between background and anomalies can be utilized for detection. However, due to the absence of spectral information of anomalies, anomaly detection brings more challenges to traditional detection methods. we propose an anomaly detection method for hyperspectral images based on two well-designed dictionaries: background dictionary and potential anomaly dictionary. In order to effectively detect an anomaly and eliminate the influence of noise, the original image is decomposed into three components: background, anomalies, and noise. In this way, the anomaly detection task is regarded as a problem of matrix decomposition. Considering the homogeneity of background and the sparsity of anomalies, the low-rank and sparse constraints are imposed in our model. Then, the background and potential anomaly dictionaries are constructed using the background and anomaly priors. For the background dictionary, a joint sparse representation (JSR)-based dictionary selection strategy is proposed, assuming that the frequently used atoms in the overcomplete dictionary tend to be the background. In order to make full use of the prior information of anomalies hidden in the scene, the potential anomaly dictionary is constructed. We define a criterion, i.e., the anomalous level of a pixel, by using the residual calculated in the JSR model within its local region. Then, it is combined with a weighted term to alleviate the influence of noise and background.

CONTENTS

LIST OF FIGURES

CHAPTER 1: PREAMBLE

1.1 MOTIVATION FOR WORK

In a multidisciplinary education context, experimental-based learning appears one of the most interesting instructional strategies which tries to engage students in authentic real-world tasks to enhance learning. In experimental -based learning, students typically engage individually or in groups with an instructor or coach or mentor. Each of the project designs and implements an approach to understand practical professional environment in the field of computer science engineering. In this project, we have applied experimental -based learning to develop an anomaly detection model for hyperspectral images; in addition to technical knowledge, we also learned to manage resources and time execution and work in team.

1.2 INTRODUCTION

Imaging spectrometers have been successfully developed, marking that hyperspectral remote sensing has become one of the frontier remote sensing earth observation technologies. They can record electromagnetic wave response value reflected or emitted by ground material within the imaging region, forming an image at each wavelength. Hyperspectral image (HSI) will be obtained by stacking these images, which can cover almost the entire electromagnetic spectra, providing up to ten, hundreds, or even thousands of spectral bands. The spectral information from all bands can form an almost continuous spectral curve for each pixel, which reflects the spectral characteristic of ground material and can be utilized to distinguish different ground materials. Therefore, based on the abundant spectral information, HSI provides great advantages for image classification, spectral unmixing, and target detection . Target detection concentrates on separating specific targets from various backgrounds. Anomaly detection is one type of target detection techniques, which separates anomaly targets from typical backgrounds without any prior knowledge. Anomaly targets are usually distinguished by their spectra or quantity proximity to backgrounds, which can reveal important information in many circumstances including medical diagnosis, military reconnaissance, disaster rescuing, and so on. Particularly, detecting anomaly targets is usually more rewarding when the prior knowledge cannot obtain easily or quickly. For example, detecting ship wreckage in a sea background without prior knowledge can improve search and rescue efficiency as compared to supervised target detection. Therefore, the past decades became an eruptive period for anomaly detection research. And the research papers mainly involve two models [3]: statistical and geometrical models.

Statistical modelling-based methods are the earlier stand widely used for hyperspectral anomaly detection, which assume that anomalies are distant with respect to backgrounds obeying certain statistical distribution. For example, the benchmark Reed-Xiaoli (RX) algorithm models backgrounds with a multivariate Gaussian model and divides anomalies based on their Mahalanobis distances to background center. The statistics of background center are estimated with pixels in a local or global image. Different from this statistics estimation method, the blocked adaptive computationally efficient outlier nominator (BACON) algorithm employs reliable background pixels based on iteratively detection to obtain more robust statistics approximation. In addition to these methods in original spectral feature space, a nonlinear version of RX is proposed for linearly inseparable problem with kernel technique. Another successful variant is the fractional Fourier entropy-based RX algorithm. It utilizes an intermediate domain which includes both original spectral feature and Fourier domain feature to enhance the discrimination between anomalies and backgrounds. Although these statistical-based models can obtain good anomaly detection performance, the statistical distribution assumption does not always hold.

Geometrical modelling-based methods assume that anomalies are far away from backgrounds obeying certain geometrical representation. Take the support vector data description as an example, it achieves an optimal hypersphere to approximately enclose backgrounds and distinguishes anomalies by comparing their distances to hypersphere center with hypersphere radius. Another kind of geometrical modelling-based methods is subspace-based approach, which deems that anomalies are far away from backgrounds' subspace. The classic collaborative representation based detector (CRD) algorithm and its variants construct backgrounds' subspace with spatial neighbourhoods based on collaborative representation.

1.3 HSI's CLASSIFICATION & DETECTION – CHALLENGES, CONSTRAINTS

Since their emergence, several difficulties have caused issues in analysing and performing operations on hyperspectral images. Initially, it suffered from spectroscopy technology due to the bad quality of hyperspectral sensors and poor quality with insufficient data. However, along with the advancement in applied science, things have come to ease, but there are still some well-known non-dispersible hitches [2] that need to be overcome. Some of them are stated as follows:

- Lack of high-resolution Earth observation (EO) noiseless images.
- Hindrances in the extraction of features.
- The large spatial variability and interclass similarity.

- Lack of balance among interclass samples.
- The higher dimensionality.

Hyperspectral techniques are used extensively in a variety of research areas and environmental applications due to their capability to detect and discriminate between the specific features of objects with several contiguous spectral channels. Hyperspectral images provide better capability to see the previously unseen through their higher spectral detail. Imaging a large number of spectral bands and the advantages that this brings over multi-spectral imaging comes with several limitations and drawbacks. The prime disadvantages of hyperspectral data are cost and data complexity. To simplify hyperspectral data analysis, fast computers and large data storage capacities are needed and can be rented on-demand via cloud computing services.

A drawback affecting Hyperion images is linked to the spectral and spatial artifacts related to array malfunctions in some sensors such as flag pixels .4095, which represent pixels affected by echo and smear, dark current and background removal of the residual charge in the detector, sensor bias effects, and bad pixels.

Another problem that exists with hyperspectral data is spectral smile. Spectral smile, also known as the "smile" or "frown" curve, is a spectral distortion[1] referring the spectral distortion to an across-track wavelength shift from the central wavelength, which is due to a change of the dispersion angle with the field position. It is typically found in pushbroom sensors. Therefore, it is a shift in wavelength in the spectral domain, which is a function of an across-track pixel (column) in the swath. For VIS-NIR-SWIR wavelength regions, it shifts slightly to some extent. In visible-near infrared bands the shift range is between 2.6 and 3.5 nm, with the maximum shift occurring at column 256 in band 10. In SWIR bands, the spectral shift is less than 1 nm.

1.4 RELEVANCE OF THE PROJECT

In remote sensing, the anomaly detection aims to detect anomalous pixels in multispectral and hyperspectral images. Anomalies are pixels that deviate from the expected behaviour and can hold interesting information. For instance, anomaly can be rare vegetation species, anomaly in vegetation growth, illegal plants associated to drug commerce, polluted area in coastal waters, adventurers lost in the dessert, buried archaeological structures, illegal border crossing or military vehicle under vegetative cover. The magnitude of anomalies motivated researches in anomaly detection and interpretation for hyperspectral images. Since the beginning, researchers had to cope with the problem of the absence of any prior knowledge about the treated data.

1.5 SCOPE OF THE PROJECT

Interest on anomaly detection for hyperspectral images is increasingly growing the last decades due to the diversity of applications that aims for detecting small distinctive objects dispersed in a large geographic zone, without any prior knowledge about the scene. Hyperspectral imaging measures the amount of electromagnetic energy across the instantaneous field of view at a very high resolution in hundreds or thousands of spectral channels. This enables objects to be detected and the identification of materials that have subtle differences between them. However, the increase in spectral resolution often means that there is a decrease in the number of photons received in each channel, which means that the noise linked to the image formation process is greater. This degradation limits the quality of the extracted information and its potential applications.

The project aims to detect the anomalies of hyperspectral images with an acceptable degree of accuracy. By having an idea about the anomalies in data, we will know beforehand how to utilize the data for analysis & studies and thus it makes us to improve its application areas.

1.6 PROBLEM DEFINITION

By considering the effect of anomalies in hyperspectral images, the problem of the project is to detect the anomalies and it is stated as, Implementation of Dictionary Learning concept practically to detect the anomalies in Hyperspectral Images.

1.7 OBJECTIVE OF THE PROJECT

The objectives of the “Hyperspectral Anomaly Detection” can be stated as follows:

1. Make use of machine learning algorithms to increase the accuracy of Anomaly Detection in Hyperspectral Images.

2. Allow the researchers & other areas to invest wisely in study and development of applications corresponding to HIS’s that makes the people to advance in their work.

CHAPTER 2: LITERATURE SURVEY

2.1 INTRODUCTION

What other people think" has always been an important piece of information for most of us during the decision-making process. The Internet and the Web have now (among other things) made it possible to find out about the opinions and experiences of those in the vast pool of people that are neither our personal acquaintances nor well-known professional critics — that is, people we have never heard of. And conversely, more and more people are making their opinions available to strangers via the Internet. The interest that individual users show in online opinions about products and services, and the potential influence such opinions wield, is something that is driving force for this area of interest. And there are many challenges involved in this process which needs to be walked all over in order to attain proper outcomes out of them. In this survey we analysed basic methodology that usually happens in this process and measures that are to be taken to overcome the challenges being faced.

2.2 EXISTING SYSTEM

2.2.1 A Novel Hyperspectral Image Anomaly Detection Method Based On Low Rank Representation [13]

This research presents a novel method for anomaly detection in hyperspectral image(HSI) based on low Rank representation. In the observed HSI, the anomalies can be separated from the background. Since each pixel in the background can be approximately represented by a background dictionary, and the representation coefficients of the background pixels are correlative, a low-rank representation model is used to model the background part. Besides, to gain robust representation coefficient, the sum-to one constraint is added. The advantage of the proposed low-rank representation sum-to-one (LRRSTO)[5] method is that it makes use of the global correlation of the background and strength the robustness of the representation. Experiments results have been conducted using both simulated and real data sets. Taking into consideration that, in a real hyperspectral scene, anomalous signals are different from the background signals, and there usually exists strong correlation among the background signals. The low rank representation can be used to model the separation of anomalous part and non-anomalous part in the scene. Moreover, a sum-to-one constraint is added to the representation vectors to enhance the stability of the solution.

2.2.2 Anomaly Detection In Hyperspectral Imagery: An Overview [12]

Interest on anomaly detection for hyperspectral images is increasingly growing the last decades due to the diversity of applications that aims for detecting small distinctive objects dispersed in a large geographic zone, without any prior knowledge about the scene. In addition to the absence of prior knowledge, many problems are particularly challenging for the anomaly detection such as the differentiation between real targets, false alarms and noise, the detection of anomaly of different shapes and sizes, and the high computational cost of the proposed approaches.

To achieve anomaly detection, researchers adopt several techniques relevant to diverse disciplines such as statistic, information theory, graph theory, and so on. These approaches can be grouped into four families based on the underling techniques.

A. Statistic Based Anomaly Detectors

1) First and second order statistics Anomaly detectors:

The real genesis of the anomaly detection for multi/hyperspectral images is marked by the proposition of the RX detector (RXD)[7], which is considered as the benchmark anomaly detector for hyperspectral imagery. Reed and Ye were presented a generalization of the algorithm of the constant false alarm rate (CFAR) proposed by Chen and Reed. It can be considered as an adaptation of the binary hypothesis of Neyman-Pearson approach to the generalized maximum likelihood ratio test (GLRT) where under the first hypothesis the background is modelled as a zero mean normal distribution and under the second hypothesis, the signal of the target is linearly combined with the distribution of the residual background plus noise. The resolution of this detection rules is derived by the maximization of the detection probability with constraint of maintaining a constant false alarm probability at a desired value.

2) High order statistics anomaly detectors:

It is shown by experiments that using the second order statistics to detect target occurring with low probability and small population is less effective than using high order statistics. In fact, the contribution of anomalies to high order statistic is more effective than its contribution to the second order statistics. The approach (HOS-ATR) is based on a whitening process, to subtract the background signals, preceded by an Orthogonal Subspace Projection (OSP). For each projection, the optimal projection vector is searched by resolution of the kth order central moment. An evolution of this approach is presented in where the virtual dimensionality is used for detecting automatically the number of the spectrally distinct signatures present in the data. For this approach the projection is performed by a fast independent component analysis, with the negentropy criterion to approximate the mutual information, to speed up the progression of detection.

B. Kernel Based Anomaly Detectors

The major problem with statistics based anomaly detectors is the assumption of the linearity of the background. In fact, in reality the distribution of the probability density function of hyperspectral images is far away from the normal distribution. Therefore, nonlinear anomaly detectors were proposed using kernel strategies. Heesung and Nasrabadi proposed (KRX) a nonlinear version of the RXD based on its kernelization in the feature space in terms of kernels that implicitly compute dot products in the feature space. To estimate the kernel matrix, which is a Gram matrix; proprieties of the kernel principal component analysis are used. For the experimentation, the Gaussian RBF kernel is used to represent the RX kernel. Kwon and Nasrabadi proposed (KASD) a nonlinear version of the Adaptive Subspace Detector (ASD)[4]. This approach represents data with a nonlinear function using the Gaussian RBF kernel, which ensures the translation invariance propriety. Results overcome the conventional ASD.

C. Projectionl Based Anomaly Detectors

Projection based anomaly detectors try to minimize the amount of treated data and focus on the interesting features for anomaly detection. Johnson et al. presented an approach (autoGAD) based on the Independent Component Analysis (ICA). This approach is based on the statistical approach proposed by Robila and Varshney for target detection. To avoid the calculation of the kurtosis proposed, the zero detection histogram is used for feature selection and potential anomalies determination, and then the noise is smoothed by the use of the adaptive Winer filter that recurrently reduces the power of the background while maintaining the power of anomalies. This approach proposes to determinate the number of principal component automatically. Results show that this approach overcomes the RXD and the approach described mainly in term of computational cost. The projection pursuit is also used for anomaly detection. Chiang et al. proposed the use of the projection pursuit for target detection (PPTD). This approach looks for the optimal projection index by an evolutionary algorithm. It proposes the use of a zero detection thresholding method. Non detected pixel will be considered for the next projection process till convergence. Huck and Guillaume in proposed to establish a link between the projection pursuit and the binary testing hypothesis that leads to an asymptotically constant false-alarm rate resulting approach (BHPP)[6] that benefit of the spectral discrimination rate of the projection pursuit.

D. Segmentation Based Anomaly Detectors

To cope with the effect of the hypothesis of homogeneity of the background on the increasing of the false alarm rate, segmentation based anomaly detectors were proposed. The segmentation can be performed either with respect to spectral characteristic or to spectral and spatial characteristics.

1) Spectral segmentation based anomaly detectors:

Carlotto proposed the Cluster Based Anomaly Detector (CBAD) which is considered later as the benchmark anomaly detector for the segmentation based approaches. This approach tends to group in clusters image pixels according to the histogram quantization of image's principal components. Inside each cluster, a Gaussian mixture model (GMM) is supposed. Then, the Mahalanobis distance is calculated between the PUT and the center of each cluster. Pixels that exceed the threshold are considered as anomalies.

2) Spatial spectral segmentation based anomaly detectors:

Spatial spectral segmentation based anomaly detectors adds the contextual information to the spectral one in the segmentation process. For this purpose, Goovaerts et al. proposed a geostatistical filtering of noise and regional background using the factorial kriging followed by a local indicator of spatial autocorrelation (GLCAAD). Anomalies are guided by the sign and the magnitude of the local indicator of spatial autocorrelation for all images of principal components. This approach presents a lower FAR than the RXD mainly in presence of noise. Kim and Finkel proposed to start by grouping pixels of the image by the k-means clustering method, and then representative pixels of each cluster are chosen for the Locally Linear Embedding (LLE) algorithm (LLEAD). This method decreases the dimensionality while preserving the spatial spectral relationship between pixels and succeeds to detect anomalies.

2.2.3 Anomaly Detection And Classification For Hyperspectral Imagery [11]

In this research, these two anomaly detectors are investigated and explored. In particular, several variants of the RXD and LPD are derived, and a real-time processing version of the RXD is also introduced where the sample correlation matrix instead of the sample covariance matrix is used. It is referred to as causal RXD (CRXD)[9]. The term "causal" borrows from signal processing terminology which means that the information used for data processing is up to the pixel being processed and updated solely based on the pixels that were already processed. "Real-time process" refers to a process that processes data samples when they come with no time delay. It can be implemented in two ways: line-by-line and pixel-by-pixel. A line-by-line real-time process can be implemented by considering a line of pixels as an input array. It processes each line while the line is being scanned. A pixel-by-pixel real-time process considers each pixel as an input and processes a pixel while it being scanned. An experiment of a pixel-by-pixel CRXD will be demonstrated for illustration. Two advantages can be benefited from the CRXD. Since the computation of the inverse of a sample correlation matrix can be carried out in parallel via a QR-decomposition, the CRXD has an ability of processing data in a real-time fashion. In other words, unlike the RXD, which requires the knowledge of all the data samples to form the sample covariance matrix prior to processing, the CRXD processes and updates data either line-by-line or sample-by-sample. A second advantage results from the fact that the sample correlation matrix accounts for both the first-order and second-order statistics. For most remotely sensed images which are generally considered to be nonstationary, the CRXD can capture spectral variability more effectively than the RXD, which only takes care of second-order statistics.

It is often the case that an anomaly detector can detect different types of anomalies but cannot discriminate the detected targets from one another. So, in this research, four target discrimination measures are further developed to resolve this dilemma. A target discrimination measure discriminates the detected anomalies and clusters them into separate target classes. Due to the fact that the number of the detected target pixels is generally small, the automatic thresholding methods commonly used in traditional spatial-based image processing may not be applicable. In this case, an automatic thresholding method is proposed and modified from the zero-detection thresholding technique proposed to cluster the detected targets into separate classes. The means of these target classes are then calculated and used as the target information for follow-up target classification. Because the resulting classification is supervised, the classification results may classify additional target pixels whose signatures are similar to the anomalies, but were not detected by the RXD. In analogy with the CRXD, a real-time implementation of anomaly classification is also possible. However, in this case, a certain time delay is inevitable between anomaly detection and target discrimination, but is negligible if the image size is manageable, such as one used in the experiments conducted in this paper. In order to make distinction of this subtle difference, a process is referred to as an online process if it can be implemented in a timely manner with negligible time lag. So, with this definition, anomaly classification can be implemented as an online process by the CRXD in conjunction with a target discrimination measure and followed by the real-time linearly constrained minimum-variance (LCMV)[10] classifier developed. More specifically, anomaly classification can be accomplished by three-stage processes. The anomaly detection is initiated in the first-stage process, then followed by a clustering process using a target discrimination measure in the second stage and, finally, concluded by the LCMV classification in the third stage. All these three stages can be processed online with no appreciable time delay.

2.2.4 Auto-AD: Autonomous Hyperspectral Anomaly Detection Based on Fully Convolutional Autoencoder [14]

A novel hyperspectral anomaly detection framework based on a fully convolutional AE is proposed. Specifically, the background is reconstructed with the same dimensionality as the original image, to avoid loss of spectral information, while the anomalies are detected based on reconstruction errors, to avoid the construction of additional detectors. The proposed Auto-AD method achieves autonomous hyperspectral anomaly detection. Furthermore, the Auto-AD method does not require any parameters to be manually set, and involves no pre-processing or postprocessing procedures. Owing to the ability of the proposed network to accurately reconstruct the background, the anomalies are automatically separated. An adaptive-weighted loss function is proposed to further suppress the anomaly reconstruction, in which the weights of potential anomalous pixels are reduced during training. Thus, anomalies have a higher contrast with the background in the map of reconstruction errors. The weights are adaptively updated since they are derived from the reconstruction errors.

CHAPTER 3: THEORITICAL BACKGROUND

Theoretical background highlighting some topics related to the project work is given below. The description contains several topics which are worth to discuss and also highlight some of their limitations that encourage going on finding solutions as well as highlights some of their advantages for which reason these topics and their features are used in this project.

3.1 PREDICTION MODEL

3.1.1 Hyperspectral Anomaly Detection Via Background And Potential Anomaly Dictionaries Construction

In this model, we propose a new anomaly detection method for hyperspectral images based on two well-designed dictionaries: background dictionary and potential anomaly dictionary. In order to effectively detect an anomaly and eliminate the influence of noise, the original image is decomposed into three components: background, anomalies, and noise. In this way, the anomaly detection task is regarded as a problem of matrix decomposition. Considering the homogeneity of background and the sparsity of anomalies, the low-rank and sparse constraints[8] are imposed in our model. Then, the background and potential anomaly dictionaries are constructed using the background and anomaly priors. For the background dictionary, a joint sparse representation (JSR)-based dictionary selection strategy is proposed, assuming that the frequently used atoms in the overcomplete dictionary tend to be the background. In order to make full use of the prior information of anomalies hidden in the scene, the potential anomaly dictionary is constructed. We define a criterion, i.e., the anomalous level of a pixel, by using the residual calculated in the JSR model within its local region. Then, it is combined with a weighted term to alleviate the influence of noise and background.

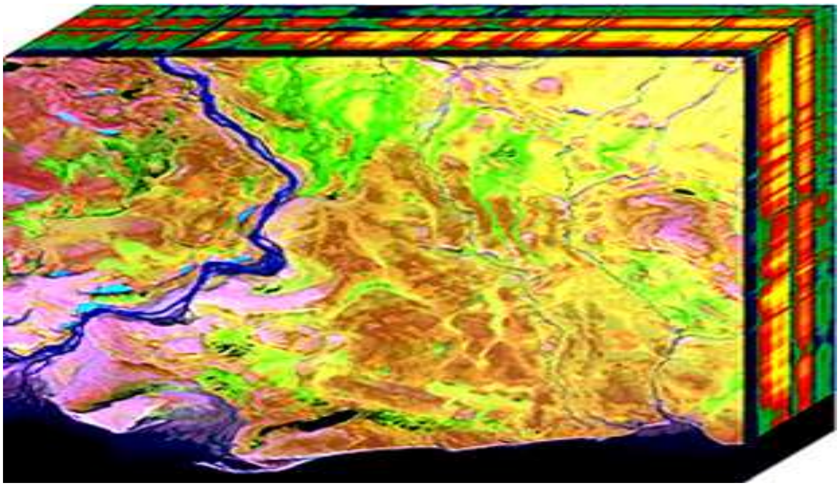

Fig. 1. Hyperspectral Image

A. Background, Anomaly, and Noise Decomposition Model

Let an HSI data be denoted as X ∈ Rh×w×d , where d is the number of the spectral bands, and h and w are the spatial size of the data. For convenience, we transform the 3-D cube X into a 2-D matrix X = {xi} n i=1 ∈ Rd×n, where each column of X is a spectral pixel vector in the HSI and n = h × w is the number of the pixels. In this paper, we formulate the anomaly detection task as a matrix decomposition problem. The HSI data matrix is decomposed into three components: background, anomaly, and noise. Considering that there usually exists a strong correlation between background pixels, which can be represented by the combination of other background pixels, and we want to distinguish the anomalies and noise simultaneously, our decomposition model is formulated as

$$X = BZ + A + E \qquad (1)$$

where BZ is the background component, B = [b1, b2,..., bnB] is the background dictionary, nB is the number of the atoms in the dictionary, Z = [z1,z2,...,zn] are the corresponding representation coefficients, and A = [a1, a2,..., an] and E = [e1, e2,..., en] are the anomaly and noise components, respectively. Intuitively, the whole spectral space can be divided into several underlying subspaces. For an HSI, pixels in a local region are most likely homogeneous, so we assume that the background holds a low-rank property. For the noise component, it has been investigated that there are mainly two kinds of noise existing in HSIs, including sparse noise (strip and deadline) and Gaussian random noise. Compared with l2 and l1 norms, l2,1 norm is more robust to describe both sparse noise and Gaussian random noise. Thus, the objective function can be further formulated as

$$\min_{\mathbf{Z},\mathbf{A},\mathbf{E}} \ \mathrm{rank}(\mathbf{Z}) + \beta\|\mathbf{A}\|_l + \lambda\|\mathbf{E}\|_{2,1}$$
$$\text{s.t } \mathbf{X} = \mathbf{BZ} + \mathbf{A} + \mathbf{E} \qquad (2)$$

where $\beta > 0$ and $\lambda > 0$ are the coefficients used to balance all the components. rank*(***Z***)* represents the rank of matrix **Z** which is the coefficient matrix of the low-rank representation. The $l_{2,1}$ norm is defined as the sum of the l_2 norm of the columns in a matrix, i.e.,

$$\|\mathbf{E}\|_{2,1} = \sum_{i=1}^{n} \sqrt{\sum_{j=1}^{d} (\mathbf{e}_{i,j})^2} \qquad (3)$$

which attempts to enforce each element of the matrix to approach zero except for some outliers. It is difficult to estimate anomalies using a particular distribution, because anomalies may be different in the same HSIs. Thus, pixels with significant differences from the background are extracted and used as potential prior of the other anomalous pixels. Intuitively, anomalies chosen as the atoms of the potential anomaly dictionary are related to the other ones hidden in the data set. Therefore, a hidden anomaly can be represented by the linear combination of the predetected strong anomaly atoms in the dictionary, namely, $\mathbf{A} = \mathbf{TS}$, where $\mathbf{T} = [t_1, t_2, ..., t_{nT}]$ is the potential anomaly dictionary, n_T is the number of atoms in potential anomaly dictionary, and $\mathbf{S} = [s_1, s_2, ..., s_n]$ is the corresponding coefficient matrix. However, when generating the potential anomaly dictionary, we may mistakenly include some pixels that are not anomalies. To avoid this situation, we assume that only the potential anomaly atoms are active when reconstructing an anomaly pixel. The atoms in the potential anomaly dictionary are expected to be the supportive bases for the reconstruction of an anomaly. Since these pixels are randomly distributed in the scene, anomalous pixels retain sparse characteristics. In this way, we constrain the coefficients matrix to be sparse. As a result, anomalous pixels are reconstructed using the atoms as few as possible from the potential anomaly dictionary. The above-mentioned formulation is nonconvex and NP-hard. Fortunately, under certain conditions, the problem of finding a low-rank approximation for a given matrix can be solved by minimizing its nuclear norm. Now, the model for our proposed anomaly detection becomes as

$$\min_{\mathbf{Z},\mathbf{S},\mathbf{E}} \|\mathbf{Z}\|_* + \beta\|\mathbf{S}\|_1 + \lambda\|\mathbf{E}\|_{2,1}$$
$$\text{s.t } \mathbf{X} = \mathbf{BZ} + \mathbf{TS} + \mathbf{E} \qquad (4)$$

where $\|*\|_*$ denotes the matrix's nuclear norm. The response value of each pixel belonging to anomalies can be calculated by the l_2 norm of each column of the anomaly component $\mathbf{A} = \mathbf{TS}$, i.e., $r_i = a_i 2$ $i = 1,2,...,n$. Finally, anomalies can be determined by a predefined threshold.

B. Background Dictionary Construction

For the pixels within a local region of an HSI, they may share common structures. Therefore, these regions can be jointly approximated by a sparse linear combination of a few common atoms. The objective of the JSR is

$$\min \|\psi\|_{\text{row},0}$$
$$\text{s.t. } \mathbf{U} = \mathbf{V}\psi + \mathbf{R} \qquad (5)$$

where $\mathbf{U} = [u_1, u_1, ..., u_L]$ is a 2-D matrix with L spectral pixels flattened from the 3-D local region cube and ψ is the corresponding representation coefficients where only a few rows are nonzero, $\mathbf{R} = [r_1, r_2, ..., r_L]$ are the residuals after construction based on $\mathbf{V}$ and ψ, and $\|*\|_{\text{rows},0}$ denotes the nonzero rows of ψ.

The JSR model represents each spectral pixel within a local region using common atoms chosen from an overcomplete dictionary. Thus, it captures the common part from the original region by the linear combination of a few atoms, which reflects the consistent spectral information in this region. Obviously, for the anomaly detection task, the common part within a region tends to be regarded as the background, so we choose the atoms that are frequently used as the bases to reconstruct background. All the chosen atoms from different classes of background materials form the background dictionary. Now, the challenge is how to construct an overcomplete dictionary for the JSR model and how to design the metric to measure the frequently used atoms chosen from the dictionary.

For the first problem, a simple method to construct the dictionary in the JSR model is to utilize all the original spectral pixels in HSI. However, it is time-consuming to calculate each small region over such a large dictionary. Although a wide range of spectral information from different materials may boost the representation ability of the JSR model, it actually degrades the ability of describing a particular material. This is because too many pixels from different classes are involved when reconstructing a local region. Consequently, the global dictionary may be confused when intending to extract a common structure.

Considering the nonlocal similarity between the local regions, we use an extended *k*-means clustering algorithm to group all the pixels into several clusters so that each group contains a similar underlying structure for reducing complexity. Let the small region under investigation be denoted as a 3-D cube U ∈ $R^{win \times win \times d}$ of size win×win×*d*, where win is the window size. Its 2-D form is **U** ∈ $R^{\mathbf{d} \times \mathbf{L}}$, where L = win×win.

The image patch distance (IPD) is used to measure the distance between two regions, which is defined as

$$o_{IPD}(\mathbf{U_P},\mathbf{U_Q}) = o(P(u_i), Q(u_j)) = \sum_{h=1}^{L} \max \begin{pmatrix} \min\limits_{b \in Q(u_j)} o(a_h, b) \\ \min\limits_{a \in P(u_i)} o(b_h, a) \end{pmatrix} \quad (6)$$

where *o(a,b)* is a nonlocal spectral similarity function, and (here it is the Euclidean distance), and $\mathbf{U_P}$ and $\mathbf{U_Q}$ are two 2-D matrices representing two regions selected from the regions. The calculation procedure is shown in Fig. 1.

With the increase in the number of classes, the number of the regions in each class usually decreases. As a result, the number of the regions in the smaller clusters is insufficient to construct an overcomplete dictionary by the JSR model. To solve this problem, those non overcomplete classes will be merged into the nearest class based on the IPD distance.

In this way, we group all the regions into several clusters. For each cluster, the overcomplete dictionary is made up of the overall spectral items in the class which are used to reconstruct each local region. The frequently used atoms in the dictionary are treated as the final background dictionary atoms.

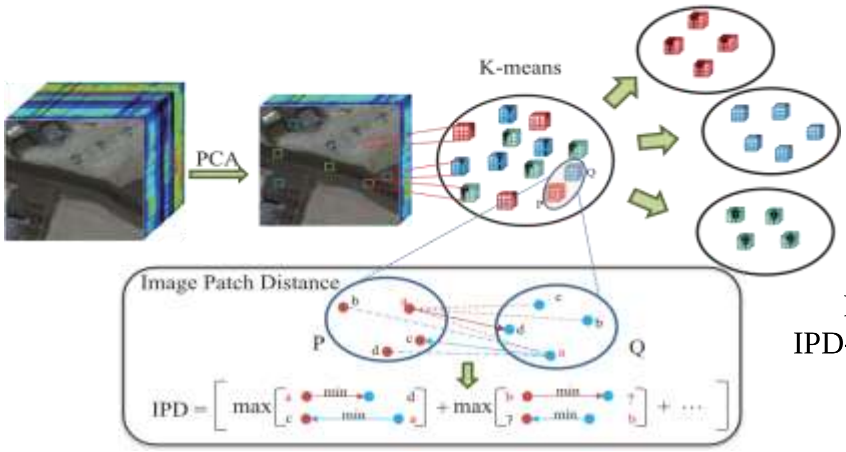

Fig. 2. Procedure of IPD-based *K*-means.

The ultimate formulation of the JSR model for each class is as

$$\begin{aligned} &\min \ \|\psi_i^c\|_{\text{row},0} \\ &\text{s.t. } \mathbf{U}_i^c = \mathbf{G}^c \psi_i^c + \mathbf{R}_i^c \end{aligned} \tag{7}$$

where $\mathbf{U}_i^c$ is the ith region of the cth class, $i = [1,2,\ldots,n^c]$, $n^c = [n^1,n^2,\ldots,n^K]$ and $c = [1,2,\ldots, K]$, n^c is the number of the regions in the cth class and K is the number of classes, $\mathbf{G}^c = [g_1^c, g_2^c,\ldots, g_{n^c}^c]$ is the overcomplete dictionary of the cth class with n^c atoms, and ψ_i^c and $\mathbf{R}_i^c$ are the corresponding coefficients and residual, respectively.

The second problem is to measure the representation frequency of each atom in each class. In this model, the frequency is defined as the sum of the JSR coefficients after normalization. It can be expressed as

$$\begin{aligned} P^c &= \frac{1}{\gamma} \sum_{i=1}^{n^c} \sum_{j=1}^{L} |\psi_{i,j}^c| \\ \gamma &= \text{sum}\left(\sum_{i=1}^{n^c} \sum_{j=1}^{L} |\psi_{i,j}^c| \right) \end{aligned} \tag{8}$$

where $\psi_{i,j}^{c}$ is the jth column of the ith region in the cth class, γ is a normalization term, P^c is a vector, in which the value of each element reflects the weighted frequency chosen as the background dictionary atom, and sum$(\cdot)$ denotes the elementwise sum of a vector. We sort it in descending order and the top atoms are chosen as background dictionary atoms for the cth class. Finally, all the chosen atoms from each class construct the background dictionary **B**.

C. Potential Anomaly Dictionary Construction

We believe that some anomalous pixels with strong responses to the background can be detected by the JSR model and they can be considered as the prior in order to detect the other anomalies. If a region is not completely homogenous, that is, there are outliers or different pixels, these pixels are more likely anomalies. Such heterogeneity in a local region is reflected by the residual computed by the JSR model. Therefore, the pixels in a region leading to a large reconstruction residual are claimed to be anomalous pixels. Meanwhile, in order to reduce the influence of isolated noise in the scene, we use the region-based residual to select the potential anomaly atoms. For the central pixel in a region, a larger average residual of its neighboring pixels means that the central pixel is most likely an anomaly. Here, the mean residual of the regions is regarded as the anomaly response of the central pixel.

For the ith region of the cth class, the response of the anomaly is calculated by

$$\bar{\mathbf{R}}_i^c = \frac{1}{L}\sum_{j=1}^{L} \|r_{i,j}^c\|_2 \tag{9}$$

where $r_{i,j}^{c}$ is the jth column of the ith region in cth class. Therefore, the error in the cth class is

$$\mathbf{R}_{\text{mean}}^c = \left[\bar{\mathbf{R}}_1^c, \bar{\mathbf{R}}_2^c, \ldots, \bar{\mathbf{R}}_{n^c}^c\right]. \tag{10}$$

The average residual of each region in K classes is concatenated, and then, we define the response value of a pixel being anomaly, namely, AL as

$$\begin{aligned} \text{AL} &= \frac{1}{\chi}\left[\mathbf{R}_{\text{mean}}^1, \mathbf{R}_{\text{mean}}^2, \ldots, \mathbf{R}_{\text{mean}}^K\right] \\ \chi &= \sum_{c=1}^{K}\sum_{i=1}^{n^c} \bar{\mathbf{R}}_i^c = \frac{1}{L}\sum_{c=1}^{K}\sum_{i=1}^{n^c}\sum_{j=1}^{L} \|r_{i,j}^c\|_2 \end{aligned} \tag{11}$$

where χ is the normalization part and AL is a vector in which each element is the anomaly response value of a region.

However, it is not enough to extract obvious anomalous pixels. On the one hand, noise is always involved. On the other hand, for a complicated scene, a region may contain different types of background materials that may be mistakenly regarded as potential anomaly pixels. In other words, not all the pixels with large reconstruction residuals are anomalies. For this concern, we take into account the importance of atoms participating in reconstructing other pixels. Compared with other atoms, the selected background and anomaly atoms are more significant. Therefore, we can eliminate the atoms that do not take part in reconstruction in order to alleviate the interference.

In addition, anomalies are selected as atoms only when the current regions contain an obvious anomaly structure. In order to identify these truly anomalous pixels, we utilize the difference between the JSR coefficients when reconstructing the background and anomaly. The way to estimating the atoms of the background dictionary actually includes two levels of information. The first level is the times of an atom being selected. It describes the participation quantity of each atom in a certain class. The second level is the absolute value of the corresponding coefficients, which describes the importance degree of an atom in region reconstruction. For the first level, the selection times of the background atoms are always high, while the selection times of anomaly atoms are low when reconstructing each of the local regions. For the second level, the chosen anomaly atoms have strong coefficients only when reconstructing a local region including many anomalies. So the frequently used atoms for each class tends to be background, while those atoms that only participate in reconstructing a region containing anomalies tends to be anomalies. The anomaly atoms in each class show the property of low selection frequency and large coefficient. We define the anomalous weight (AW) as follows to depict the abovementioned issue:

$$\mathrm{AW}^c = \frac{P^c}{F^c} \tag{12}$$

where F^c is a vector in which each element reflects the select times of each atom, which can be written as

$$F^c = \sum_{i=1}^{n^c} \sum_{j=1}^{L} \mathrm{sgn}\left(\left|\varphi_{i,j}^c\right|\right)$$

where sgn is the sign function. The weighted AL can be expressed as

$$\begin{aligned} \mathrm{AL} &= \frac{1}{\chi}\left[\mathbf{R}_{\mathrm{mean}}^1, \mathbf{R}_{\mathrm{mean}}^2, \ldots, \mathbf{R}_{\mathrm{mean}}^K\right] \odot \mathbf{AW} \\ \chi &= \sum_{c=1}^{K} \sum_{i=1}^{n^c} \bar{\mathbf{R}}_i^c = \frac{1}{L} \sum_{c=1}^{K} \sum_{i=1}^{n^c} \sum_{j=1}^{L} \left\| r_{i,j}^c \right\|_2 \end{aligned} \tag{13}$$

where denotes the elementwise multiplication and $\mathbf{AW} = \{AW^1, AW^2, ..., AW^K\}$. We sort them in a descending order and choose the top pixels as the atoms of the potential anomaly dictionary **T**.

D. Optimization and Computational Complexity

To solve the problem shown in (4), for convenience, two auxiliary variables **J** and **L** are introduced to make the objective function separable. Thus, the problem can be converted to the following form:

$$\min_{J,E,Z,S,L} \|\mathbf{J}\|_* + \beta\|\mathbf{L}\|_1 + \lambda\|\mathbf{E}\|_{2,1}$$
$$\text{s.t. } \mathbf{X} = \mathbf{BZ} + \mathbf{TS} + \mathbf{E}, \quad \mathbf{Z} = \mathbf{J}, \ \mathbf{S} = \mathbf{L}. \qquad (14)$$

We solve (14) by utilizing the augmented Lagrange multiplier method reported in [48], which is implemented by updating one variable with others being fixed

$$\begin{aligned}\ell = \ & \|\mathbf{J}\|_* + \lambda\|\mathbf{E}\|_{2,1} + \beta\|\mathbf{L}\|_1 \\ & + \langle \mathbf{Y_1}, \mathbf{X} - \mathbf{BZ} - \mathbf{E} - \mathbf{TS}\rangle + \langle \mathbf{Y_2}, \mathbf{Z} - \mathbf{J}\rangle \\ & + \langle \mathbf{Y_3}, \mathbf{S} - \mathbf{L}\rangle + \frac{\mu}{2}\big(\|\mathbf{X} - \mathbf{BZ} - \mathbf{E} - \mathbf{TS}\|_F^2 \\ & \qquad + \|\mathbf{Z} - \mathbf{J}\|_F^2 + \|\mathbf{S} - \mathbf{L}\|_F^2\big) \qquad (15)\end{aligned}$$

where $\mathbf{Y}_1$, $\mathbf{Y}_2$, and $\mathbf{Y}_3$ are the Lagrange multipliers and $\mu > 0$ is a penalty parameter. The problem can be resolved using the following steps.

1) Fix **E**,**L**,**S**,**Z** and update **J**. The objective can be derived as

$$\min_{J} \|\mathbf{J}\|_* + \frac{\mu}{2}\left\|\mathbf{J} - \left(\mathbf{Z} + \frac{\mathbf{Y_2}}{\mu}\right)\right\|_F^2. \qquad (16)$$

2) Fix **J**,**L**,**S**,**Z** and update **E**. The objective can be derived as

$$\min_{E} \lambda\|\mathbf{E}\|_{2,1} + \frac{\mu}{2}\left\|\mathbf{E} - \left(\mathbf{X} - \mathbf{BZ} - \mathbf{TS} + \frac{\mathbf{Y_1}}{\mu}\right)\right\|_F^2. \qquad (17)$$

3) Fix **J**,**E**,**S**,**Z** and update **L**. The objective can be derived as

$$\min_{L} \beta\|\mathbf{L}\|_1 + \frac{\mu}{2}\left\|\mathbf{L} - \left(\mathbf{S} + \frac{\mathbf{Y_3}}{\mu}\right)\right\|_F^2. \qquad (18)$$

The nuclear norm, l_1 and $l_{2,1}$ norms, can be solved by singular value thresholding, soft thresholding, and $l_{2,1}$ norm minimization operator. The complete procedure is summarized in Algorithm 1.

Algorithm 1 Algorithm for Solving Background, Anomaly, and Noise Decomposition

Input: Data matrix X, parameters $\lambda > 0$ and $\beta > 0$
Initialize: $\mathbf{Z} = \mathbf{J} = \mathbf{S} = \mathbf{L} = 0$, $\mathbf{E} = 0$, $\mathbf{Y_1} = \mathbf{Y_2} = \mathbf{Y_3} = 0$,
$\mu = 10^{-6}$, $\mu_{\max} = 10^{10}$, $\rho = 1.2$, $\varepsilon = 10^{-6}$
Output: Z, E, S

1. While not converged do
2. Fix others and update **J** by Eq. (16)
3. Fix others and update **L** by Eq. (17)
4. Fix others and update **E** by Eq. (18)
5. Fix others and update **Z** by

$$\mathbf{Z} := (\mathbf{B}^T\mathbf{B} + \mathbf{I})^{-1}[\mathbf{B}^T\mathbf{X} - \mathbf{B}^T\mathbf{TS} - \mathbf{B}^T\mathbf{E} + \mathbf{J} + (\mathbf{B}^\mathbf{T}\mathbf{Y_1} - \mathbf{Y_2})/\mu]$$

6. Fix others and update S by

$$\mathbf{S} := (\mathbf{T}^T\mathbf{T} + \mathbf{I})^{-1}(\mathbf{T}^T\mathbf{X} - \mathbf{T}^T\mathbf{BZ} - \mathbf{T}^T\mathbf{E} + \mathbf{L} + (\mathbf{T}^T\mathbf{Y}_1 - \mathbf{Y}_3)/\mu)$$

7. Update the three Lagrange multipliers

$$\mathbf{Y_1} := \mathbf{Y_1} + \mu(\mathbf{X} - \mathbf{BZ} - \mathbf{E} - \mathbf{DS})$$
$$\mathbf{Y_2} := \mathbf{Y_2} + \mu(\mathbf{Z} - \mathbf{J})$$
$$\mathbf{Y_3} := \mathbf{Y_3} + \mu(\mathbf{S} - \mathbf{L})$$

8. Update the parameter μ,

$$\mu = \min(\rho\mu, \mu_{\max})$$

9. Check the convergence conditions

$$\|\mathbf{X} - \mathbf{BZ} - \mathbf{DS}\|_F < \varepsilon$$
$$\|\mathbf{Z} - \mathbf{J}\|_F < \varepsilon$$
$$\|\mathbf{S} - \mathbf{L}\|_F < \varepsilon$$

10. End While

The computation of our method includes those of Algorithm 1 and dictionary construction. For the first aspects, the major computation is Step 2, which requires computing the singular value decomposition of an n × n matrix. Therefore, it is time-consuming if n is large. The optimal solution Z∗ (with respect to the variable Z) to (4) always lies within the subspace spanned by the rows of A.

This means that Z∗ can be factorized into Z∗ = P∗Z˜∗, where P∗ can be computed in advance by orthogonalizing the columns of BT. Therefore, our model shown in (4) can be rewritten as

$$\min_{\tilde{\mathbf{Z}},\mathbf{S},\mathbf{E}} \|\tilde{\mathbf{Z}}\|_* + \beta\|\mathbf{S}\|_1 + \lambda\|\mathbf{E}\|_{2,1}$$
$$\text{s.t } \mathbf{X} = \mathbf{B}^*\tilde{\mathbf{Z}} + \mathbf{TS} + \mathbf{E} \tag{19}$$

where $\mathbf{B}^* = \mathbf{BP}*$. Since the number of rows of $Z˜$ is at most r_b (the rank of matrix **B** r_b n), therefore the computation complexity of Step 2 is $O(r_B^3)$. Noted that, $\mathbf{X} = \{x_i\}_i^n{=}_1 O\in(dnr\mathrm{R}^d\times_B{}^n)$., so the computation complexity of Steps 3 and 4 is $O(dnr_B)$. For the dictionary **B** and potential anomaly dictionary **T**, their column numbers are n_B and n_T, respectively. In Steps 5 and 6, the term $(\mathbf{B}^T\mathbf{B} + \mathbf{I})^{-1}$ and $(\mathbf{T}^T\mathbf{T} + \mathbf{I})^{-1}$ can be calculated in advance, so the computation complexity of these two steps are $O(nr_B^2)$ and $O(nn^2_T)$, respectively. Therefore, the complexity of Algorithm 1 is $O(n_s(r_B^3 + dnr_B + nr_B^2 + nn^2_T))$, where n_s is the number of iterations. The major computation of the dictionary construction addresses on solving the JSR model. In this paper, the Orthogonal Matching Pursuit-Cholesky based method [51] is used to solve (14). However, the dictionaries in the JSR model vary across different classes. For convenience, we consider that there is only one dominant class so that we can reconstruct the regions using one dictionary. In this way, the upper bound of the computational complexity can be calculated as $O(nk(2ndL+nL^2+2n+k^2))$, where k is the sparsity level. The computational complexity of the IPD k means is $O(n_k(d_r LKn))$, where d_r is the number of the bands after dimensionality reduction, K is the number of clusters, and n_k is the number of the iterations. Therefore, the computation complexity of the entire algorithm is $(n_k(d_r LKn) + nk(2ndL + nL^2 + 2n + k^2) + n_s(r_B^3 + dnr_B + nr_B^2 + nn^2_T))$.

CHAPTER 4: SYSTEM REQUIREMENT SPECIFICATION

A System Requirement Specification (SRS) is basically an organization's understanding of a customer or potential client's system requirements and dependencies at a particular point prior to any actual design or development work. The information gathered during the analysis is translated into a document that defines a set of requirements. It gives a brief description of the services that the system should provide and also the constraints under which the system should operate. Generally, SRS is a document that completely describes what the proposed software should do without describing how the software will do it. It's a two-way insurance policy that assures that both theclient and the organization understand the other's requirements from that perspective at a given pointin time.

SRS document itself states in precise and explicit language those functions and capabilities a software system (i.e., a software application, an ecommerce website and so on) must provide, as well as states any required constraints by which the system must abide. SRS also functions as a blueprint for completing a project with as little cost growth as possible. SRS is often referred to as the "parent" document because all subsequent project management documents, such as design specifications, statements of work, software architecture specifications, testing and validation plans, and documentation plans, are related to it.

Requirement is a condition or capability to which the system must conform. Requirement Management is a systematic approach towards eliciting, organizing and documenting therequirements of the system clearly along with the applicable attributes. The elusive difficulties of requirements are not always obvious and can come from any number of sources.

4.1 SYSTEM REQUIREMENTS

4.1.1 HARDWARE REQUIREMENTS

Processor – Intel I3 or Higher
Speed – 1.1 GHZ
RAM – Minimum 4GB
Hard Disk – Minimum 128GB
Key Board – Standard Windows Keyboard
Mouse – Two or Three Button Mouse

4.1.2 SOFTWARE REQUIREMENTS

- Python 3.9 or latest version, is used for data pre-processing, model training andprediction.
- Python IDE: Spyder
- Operating System: windows 7 and above or Linux based OS or MAC OS

4.2 FUNCTIONAL REQUIREMENTS

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulationand processing and other specific functionality.

Following are the functional requirements on the system:

1. The entire control model set must be translated to C output Code.
2. Inputs must be models designed using CLAW design components along with standarddesign components, Multiple design models must be processed and the result must be combined to obtain asingle output file.

4.3 NON-FUNCTIONAL REQUIREMENTS

Nonfunctional requirements are the requirements which are not directly concerned with the specific function delivered by the system. They specify the criteria that can be used to judge the operation ofa system rather than specific behaviors. They may relate to emergent system properties such as reliability, response time and store occupancy. Nonfunctional requirements arise through the user needs, because of budget constraints, organizational policies, the need for interoperability with other software and hardware systems or because of external factors such as: -

- Product Requirements
- Organizational Requirements

4.3.1 PRODUCT REQUIREMENTS

Platform Independency: Standalone executables for embedded systems can be created so the algorithm developed using available products could be downloaded on the actual hardware and executed without any dependency to the development and modeling platform.

Correctness: It followed a well-defined set of procedures and rules to compute and also rigorous testing is performed to confirm the correctness of the data.

Ease of Use: Model Coder provides an interface which allows the user to interact in an easy manner.

Modularity: The complete product is broken up into many modules and well- defined interfaces are developed to explore the benefit of flexibility of the product.

Robustness: This software is being developed in such a way that the overall performance is optimized and the user can expect the results within a limited time with utmost relevance and correctness Nonfunctional requirements are also called the qualities of a system.

These qualities can be divided into execution quality & evolution quality. Execution qualities are security & usability of the system which are observed during run time, whereas evolution quality involves testability, maintainability, extensibility or scalability.

4.3.2 ORGANIZATIONAL REQUIREMENTS

Process Standards: The standards defined by DRDO are used to develop the application which is the standard used by the developers inside the defense organization.

Design Methods: Design is one of the important stages in the software engineering process. This stage is the first step in moving from problem to the solution domain. In other words, starting with what is needed design takes us to work how to satisfy the needs.

CHAPTER 5: SYSTEM ANALYSIS

Analysis is the process of finding the best solution to the problem. System analysis is the process by which we learn about the existing problems, define objects and requirements and evaluate thesolutions. It is the way of thinking about the organization and the problem it involves, a set of technologies that helps in solving these problems. Feasibility study plays an important role in system analysis which gives the target for design and development.

5.1 FEASIBILITY STUDY

All systems are feasible when provided with unlimited resources and infinite time. But unfortunately,this condition does not prevail in the practical world. So it is both necessary and prudent to evaluate the feasibility of the system at the earliest possible time. Months or years of effort, thousands of rupees and untold professional embarrassment can be averted if an ill- conceived system is recognized early in the definition phase. Feasibility & risk analysis are related in many ways. If project risk is great, the feasibility of producing quality software is reduced. In this case three key considerations involved in the feasibility analysis are:

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

5.1.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified.

Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available.

5.1.2 *TECHNICAL FEASIBILITY*

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

5.1.1 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

5.2 ANALYSIS

5.2.1 PERFORMANCE ANALYSIS

For the complete functionality of the project work, the project is run with the help of a healthy networking environment. Performance analysis is done to find out whether the proposed system. It is essential that the process of performance analysis and definition must be conducted in parallel.

5.2.2 TECHNICAL ANALYSIS

System is only beneficial only if it can be turned into information systems that will meet the organization's technical requirement. Simply stated this test of feasibility asks whether the system will work or not when developed & installed, whether there are any major barriers to implementation. Regarding all these issues in technical analysis there are several points to focus on: -

Changes to bring in the system: All changes should be in a positive direction, there will be increased level of efficiency and better customer service.

Required skills: Platforms & tools used in this project are widely used. So the skilled manpower is readily available in the industry.

Acceptability: The structure of the system is kept feasible enough so that there should not be any problem from the user's point of view.

5.2.3 ECONOMICAL ANALYSIS

Economic analysis is performed to evaluate the development cost weighed against the ultimate income or benefits derived from the developed system. For running this system, we need not haveany routers which are highly economical. So, the system is economically feasible enough.

CHAPTER 6: DESIGN

6.1 SYSTEM ARCHITECHTURE

The system architecture is the model that conceptually defines the views, structure, and behavior of the system. System architecture in other words is the representation and description of how the system works and communicates with other system components in general.

The architecture of a system reflects how the system is used and how it interacts with other systems and the outside world. It describes the interconnection of all the system's components and the data link between them. The architecture of a system reflects the way it is thought about in terms of its structure, functions, and relationships. In architecture, the term "system" usually refers to the architecture of the software itself, rather than the physical structure of the buildings or machinery. The architecture of a system reflects the way it is used, and therefore changes as the system is used.

The whole system is composed of the components and the subsystems that overall work together to make the system it should be in the first place.

When we talk about the components of the system then they are nothing but the hardware and software put together, their relationship, and their transmission and production of the data.

The system architecture is made by keeping business logic and needs in mind. This architecture can be both formal and detailed depending on the situation. The system is designed by keeping the view things in mind that is.

- The quality attributes of the system.
- The enterprise's IT environment.
- The design of the system is what the customer wants.
- The system should fulfill the business strategies.
- Human dynamics, means if the system is delivered to the non-technical guys then the system should be self-maintainable till a point.

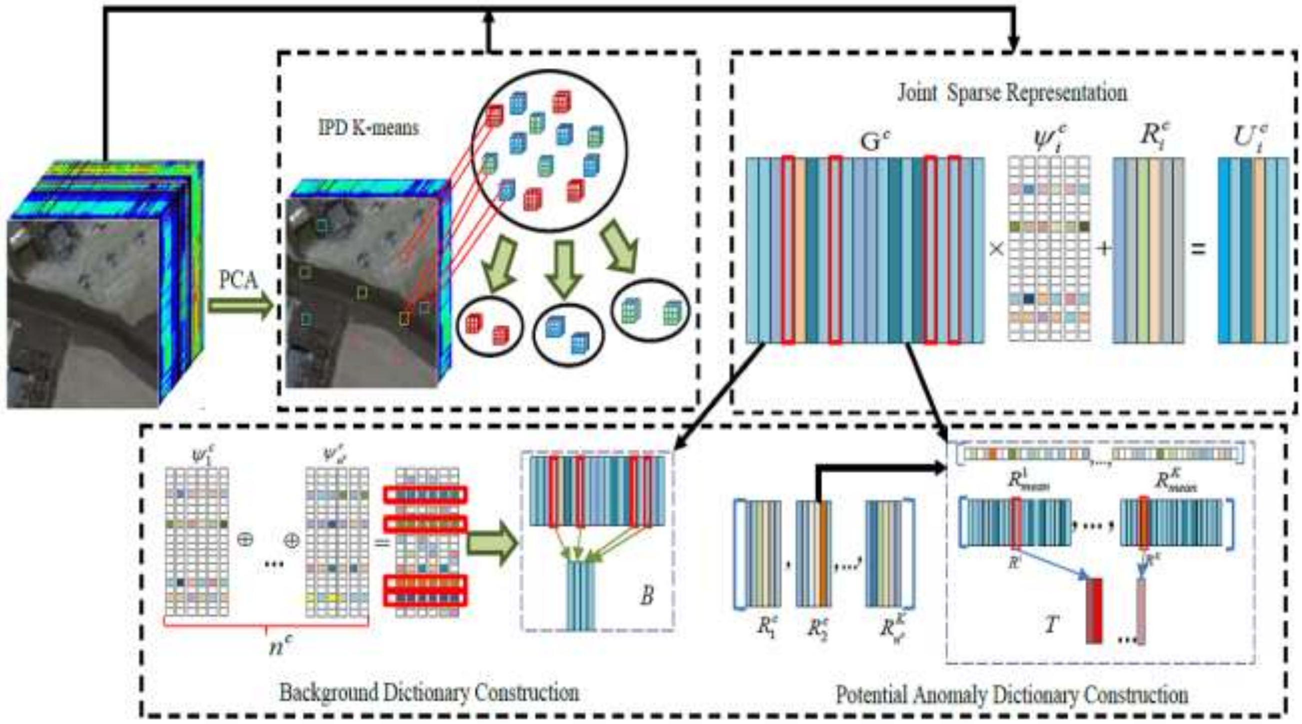

Fig. 3. System Architecture

6.2 UML DIAGRAMS

An UML diagram is a partial graphical representation (view) of a model of a system under design, implementation, or already in existence. UML diagram contains graphical elements (symbols) - UML nodes connected with edges (also known aspaths or flows) - that represent elements in the UML model of the designed system. The UML model of the system might also contain other documentation such as use cases writtenas templated texts.

UML diagrams have many benefits for both software developers and businesspeople, and the most key advantages are:

- *Problem-Solving* - Enterprises can improve their product quality and reduce cost especially for complex systems in large scale. Some other real-life problems including physical distribution or security can be solved.
- *Improve Productivity* - By using the UML diagram, everyone in the team is on the same page and lots of time are saved down the line.
- *Easy to Understand* - Since different roles are interested in different aspects of the system, the UML diagram offers non-professional developers, a clear and expressive presentation of requirements, functions and processes of their system.

The diagram can be used in many different fields including software engineering or business processes to strengthen efficiency.

- *Draft the System*- In this case, the UML diagram is used by the development team to discuss the outlines and structure the overall system. This may include the forward design and the backward design for different activities, roles, actors, and so on.
- *Visualize Programming Language* - Different types of UML diagrams in a certain system can be translated into code directly to save time for software or related application development.
- *Business Analysis* - In reality, the UML diagram can also be used to analyze the business sales pathway in order to improve customer service.

The kind of the diagram is defined by the primary graphical symbolsshown on the diagram. For example, a diagram where the primary symbols in the contentsarea are classes is class diagram. A diagram which shows use cases and actors is use case diagram. A sequence diagram shows sequence of message exchanges between lifelines.

UML specification does not preclude mixing of different kinds of diagrams, e.g. to combine structural and behavioral elements to show a state machine nestedinside a use case. Consequently, the boundaries between the various kinds of diagrams arenot strictly enforced. At the same time, some UML Tools do restrict set of available graphical elements which could be used when working on specific type of diagram. UML specification defines two major kinds of UML diagram: structure diagrams and behavior diagrams.

Structure diagrams show the static structure of the system and its parts on different abstraction and implementation levels and how they are related to each other. The elements in a structure diagram represent the meaningful concepts of a system, and may include abstract, real world and implementation concepts.

Behavior diagrams show the dynamic behavior of the objects in a system, which can be described as a series of changes to the system over time.

6.2.1 USE CASE DIAGRAM

In the Unified Modelling Language (UML), a use case diagram cansummarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

- Scenarios in which your system or application interacts with people, organizations,or external systems.
- Goals that your system or application helps those entities (known as actors)achieve.
- The scope of your system.

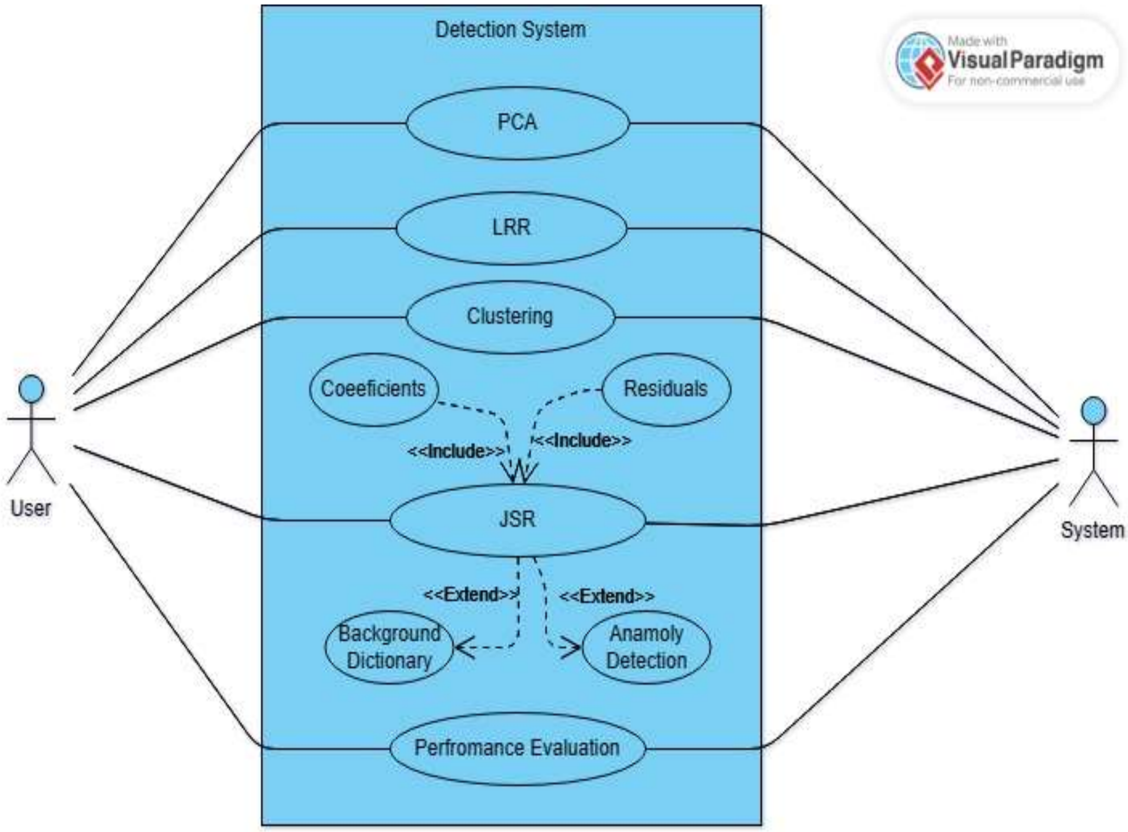

Fig. 4. Use Case Diagram

6.2.2 CLASS DIAGRAM

A Class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the structure of the application, and for detailed modelling, translating the models into programming code. Class diagrams can also be used for data modelling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

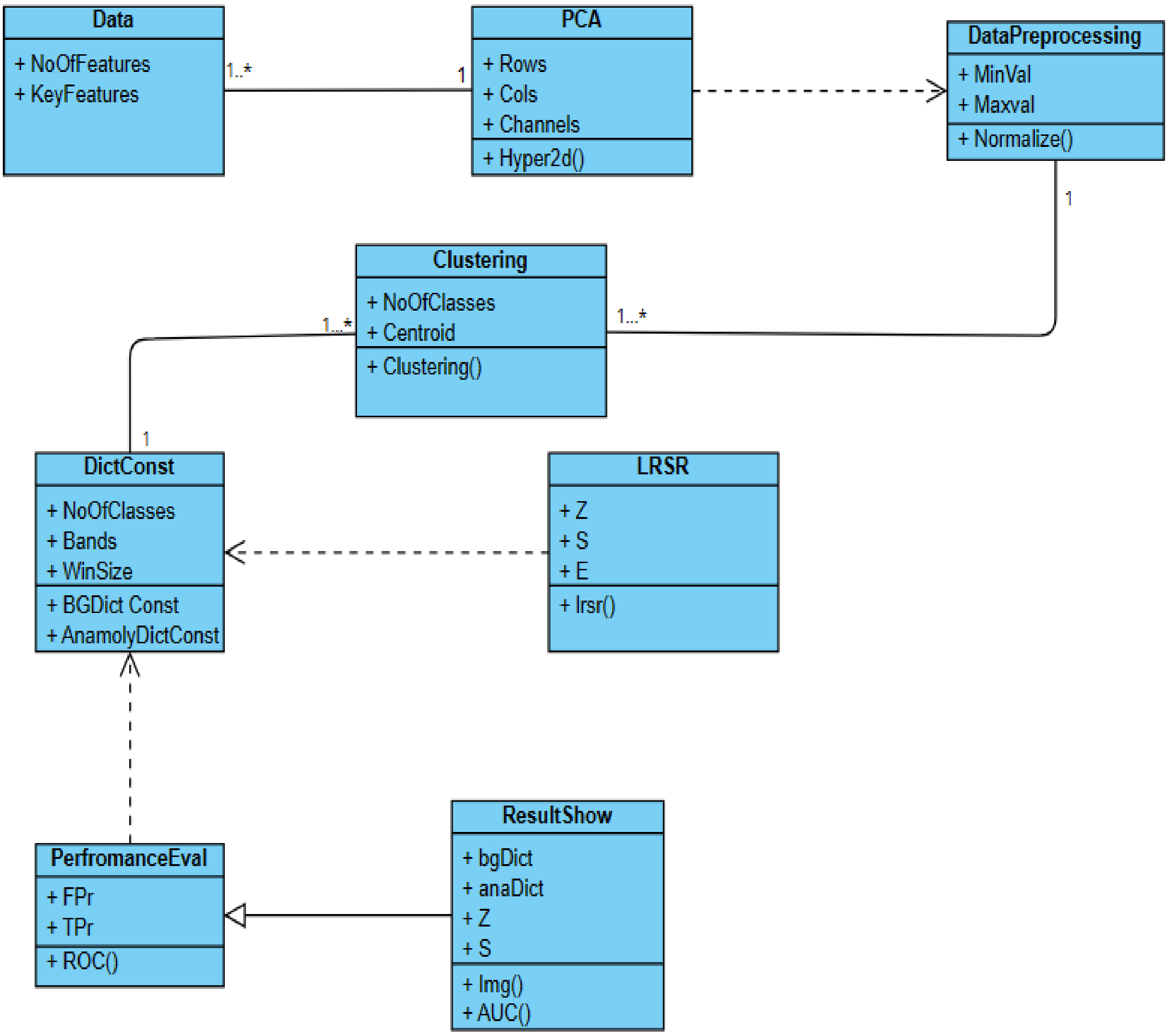

Fig. 5. Class Diagram

6.2.3 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes knownas event diagrams or event scenarios.

Sequence diagrams can be useful references for businesses and otherorganizations. Try drawing a sequence diagram to:

- Represent the details of a UML use case.
- Model the logic of a sophisticated procedure, function, or operation.
- See how objects and components interact with each other to complete a process.
- Plan and understand the detailed functionality of an existing or future scenario.

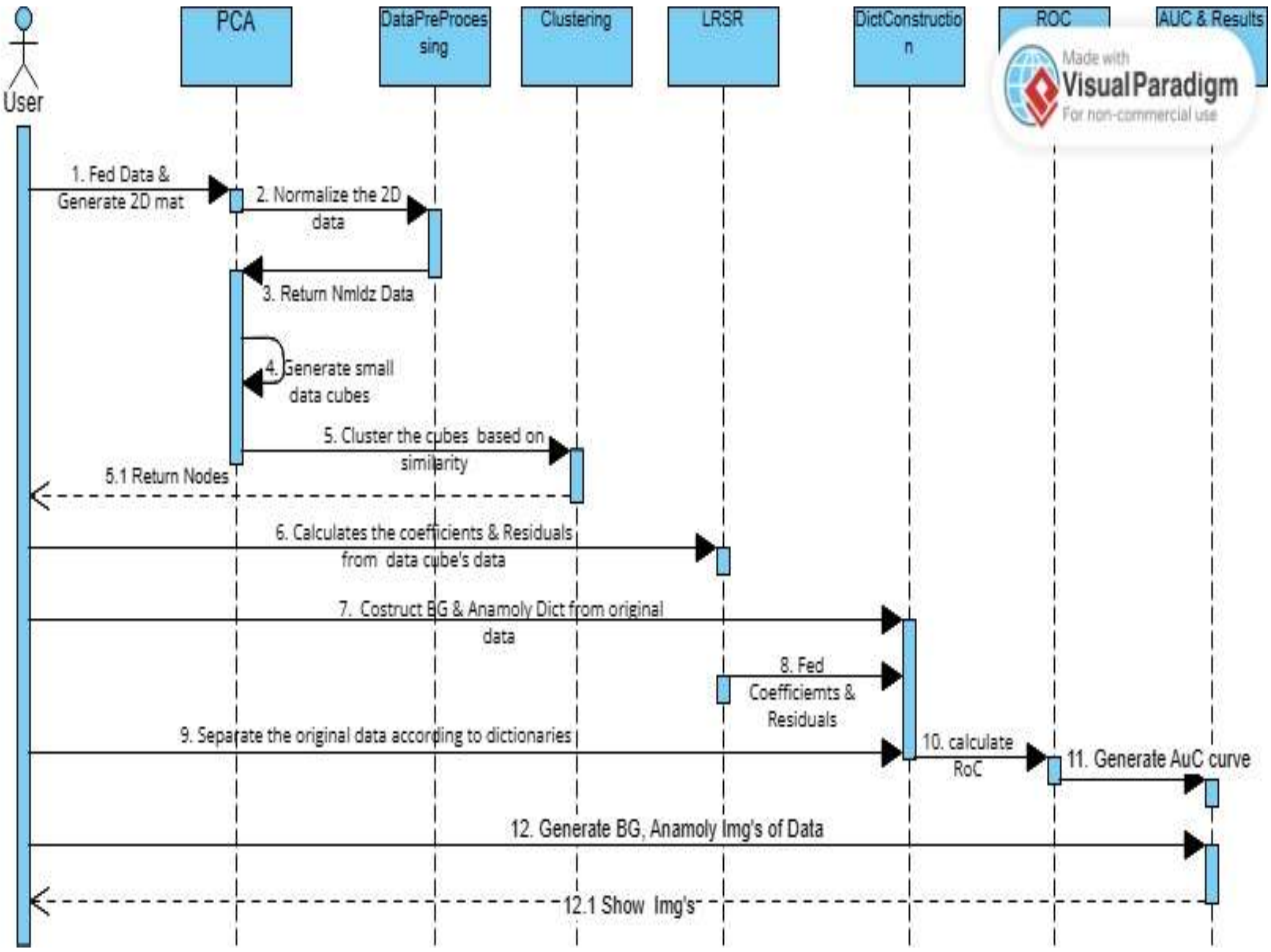

Fig. 6. Sequence Diagram

6.2.4 ACTIVITY DIAGRAM

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showingthe various decision paths that exist while the activity is being executed.

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modeling how a collection of use cases coordinate to represent business workflows

1. Identify candidate use cases, through the examination of business workflows
2. Identify pre- and post-conditions (the context) for use cases
3. Model workflows between/within use cases
4. Model complex workflows in operations on objects
5. Model in detail complex activities in a high level activity Diagram

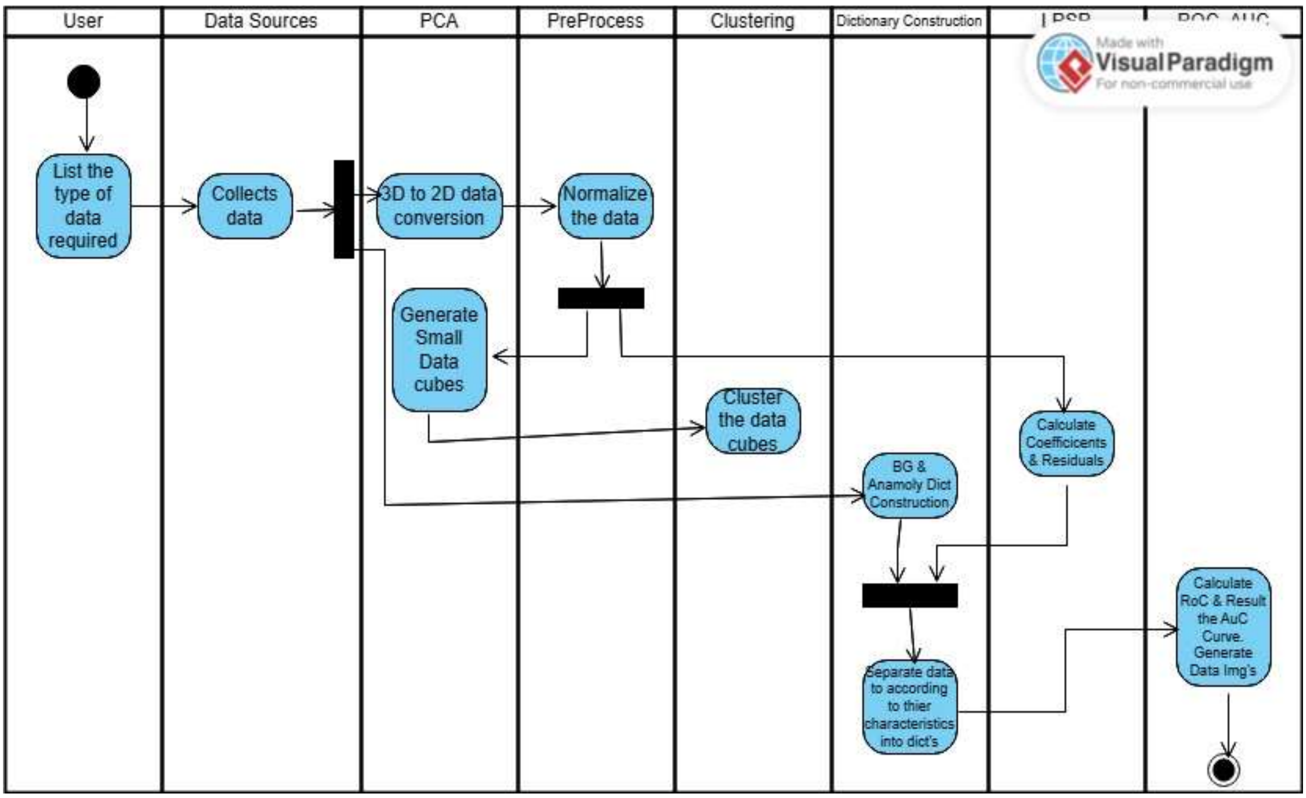

Fig. 7. Activity Diagram

CHAPTER 7: IMPLEMENTATION

7.1 SAMPLE CODE

7.1.1 HyperPro Tool. Py

```
import numpy as np
from scipy import linalg
from scipy import spatial
def load_dataset(file_name):
    dataset = []
    f = open(file_name)
    for line in f.readlines():
        curline = line.strip().split('\t')
        fltline = map(float, curline)
        dataset.append(fltline)
    return np.mat(dataset)
def hyperconvert2d(data3d):
    """ #Hyperconvert2d Convets an HSI cube to a 2D matrix#Converts a 3D HSI cube to a 2D matrix
of points #Usage* data3d = hyperconvert2d(data2d)  #Inputs* inputdata - 3D HSI cube  #Outputs*
outputdata - 2D data matrix"""
    rows, cols, channels = data3d.shape
    data2d = data3d.reshape(rows * cols, channels, order='F')
    return data2d.transpose()
def hyperconvert3d(data2d, rows, cols, channels):
"""#HyperConvert2D Converts an 2D matrix to a 3D data cube# Usage* data3d =
hyeprconvert3d(data2d) #Inputs* data2d - 2d data matrix #Outputs* data3d - 3d data cube"""
    channels, pixnum = data2d.shape
    data3d = data2d.transpose().reshape(rows, cols, channels, order='F')
    return data3d
def hypercorr(data2d):
```

```
"""   compute the sample autocorrelation matrix of a 2D matrix  # Usage* R = hyperCorr(data2d)
#Inputs* data2d - 2D matrix#Outputs* R - Sample autocorreltation matrix"""
    rows, cols = data2d.shape
    return np.dot(data2d.transpose(), data2d) / cols
```

def hypercov(data2d):

```
 """#hypercorr compute the sample covariance matrix of a 2D matrix#Usage* C  = hypercorr(data2d)
#Inputs* data2d - 2D matrix  # Outputs*C - Sample covariance matrix   """
    rows, cols = data2d.shape
    mu = np.mean(data2d, 1)
    for i in range(cols):
        data2d[:, i] = data2d[:, i] - mu
    return np.dot(data2d.transpose(), data2d) / (cols - 1)
```

def hypernorm(data2d, flag):

```
    """   hpernormalize Normalize data to be in range [0,1] # Usage *normdata = hypernorm(data2d)
#Inputs* data2d - 2D data matrix# Outputs*normdata - Normalize 2D data matrix   """
    normdata = np.zeros(data2d.shape)
    if flag == "minmax":
        minval = np.min(data2d)
        maxval = np.max(data2d)
        normdata = data2d - minval
        if maxval == minval:
            normdata = np.zeros(data2d.shape)
        else:
            normdata /= maxval - minval
    elif flag == "L2_norm":
        for i in range(data2d.shape[1]):
            col_norm = linalg.norm(data2d[:, i])
            normdata[:, i] = data2d[:, i] / col_norm
    return normdata
```

7.1.2 LRSR. Py

```
import numpy as np
from scipy import linalg

def LRSR(DictLRR, DictSRC, data, beta, lmda):
    """
    :param DictLRR: the background dictionary
    :param DictSRC: the anomaly dictionary
    :param data: the normalized data
    :param beta: parameters
    :param lmda: parameters
    :return: Z: the low rank coefficeinets
             S: the sparse coefficients
             E: the noise
    """
    dataRows, dataCols = data.shape
    DLRows, DLCols = DictLRR.shape
    DSRows, DSCols = DictSRC.shape
    ILRR = np.eye(DLCols)
    ISRC = np.eye(DSCols)
    Z = np.zeros((DLCols, dataCols))
    J = np.zeros((DLCols, dataCols))
    E = np.zeros((dataRows, dataCols))
    S = np.zeros((DSCols, dataCols))
    L = np.zeros((DSCols, dataCols))
    Y1 = np.zeros((dataRows, dataCols))
    Y2 = np.zeros((DLCols, dataCols))
    Y3 = np.zeros((DSCols, dataCols))
    mu = 0.0001
    mu_max = 10 ** 10
    p = 1.1
    err = 0.000001
    itera = 1
```

```
inv_Z = np.linalg.inv(np.dot(DictLRR.transpose(), DictLRR) + ILRR)
    inv_S = np.linalg.inv(np.dot(DictSRC.transpose(), DictSRC) + ISRC)

    while itera < 500:
        print("iteration:{0}".format(itera))
        # update J
        operator1 = 1 / mu
        tmpJ = Z + Y2 / mu
        Ju, Jsigma, Jvt = linalg.svd(tmpJ, full_matrices=False)
        # threshold1 =1/mu
        evp = Jsigma[Jsigma > operator1].shape[0]
        if evp >= 1:
            Jsigma[0:evp] -= operator1
            JsigmaM = np.diag(Jsigma[0:evp])
            print ("current evp is: {0}".format(evp))
        else:
            evp = 1
            JsigmaM = 0
        J = np.dot(np.dot(Ju[:, 0:evp], JsigmaM), Jvt[0:evp, :])

        # update E
        operator3 = lmda / mu
        tmpE = data - np.dot(DictLRR, Z) - np.dot(DictSRC, S) + Y1 / mu
        terows, tecols = tmpE.shape
        for i in range(tecols):
            tmpValue1 = linalg.norm(tmpE[:, i])
            if tmpValue1 > operator3:
                E[:, i] = ((tmpValue1 - operator3) / tmpValue1) * tmpE[:, i]
            else:
                E[:, i] = 0

        # update L
        tmpL = S + Y3 / mu
        operator2 = beta / mu
        tmpL[tmpL > operator2] -= operator2
```

```
tmpL[tmpL < -operator2] += operator2
     tmpL[(tmpL >= -operator2) & (tmpL <= operator2)] = 0
     L = tmpL.copy()

     # update Z
     tmpZ = np.dot(DictLRR.transpose(), data - np.dot(DictSRC, S) - E) + J + \
        (np.dot(DictLRR.transpose(), Y1) - Y2) / mu
     Z = np.dot(inv_Z, tmpZ)
     # update S
     tmpS = np.dot(DictSRC.transpose(), data - np.dot(DictLRR, Z) - E) + L + \
        (np.dot(DictSRC.transpose(), Y1) - Y3) / mu
     S = np.dot(inv_S, tmpS)

     # update Y1,Y2,Y3
     T1 = data - np.dot(DictLRR, Z) - E - np.dot(DictSRC, S)
     T2 = Z - J
     T3 = S - L
     Y1 += mu * T1
     Y2 += mu * T2
     Y3 += mu * T3
     # update mu
     err1 = linalg.norm(T1, np.inf)
     err2 = linalg.norm(T2, np.inf)
     err3 = linalg.norm(T3, np.inf)
     rlerr = max(err1, err2, err3)
     mu = min(p * mu, mu_max)

     itera += 1
     print("max err is:{0}".format(rlerr))
     print("current mu is:{0}".format(mu))
     if rlerr < err:
        break
 return Z, E, S
```

7.1.3 Dict_const. Py

```
import numpy as np
import HyperProTool as hyper
import scipy.io as sio
from sklearn import decomposition

def dic_constr(data3d, groundtruth, win_size, cluster_num, K, selected_dic_percent, target_dic_num):
    """
    :param data3d: the original 3D hyperpsectral image
    :param groundtruth:  a 2D matrix reflect the label of corresponding pixels
    :param win_size: the size of window, such as 3X3, 5X5, 7X7
    :param cluster_num: the number of classters such as 5, 10, 15, 20
    :param K: the level of sparsity
    :param selected_dic_percent: the selected percent of the atoms to build the background dictionary
    :param target_dic_num: the selected number to build the anomaly dictionary
    :return: data2d:  the normalized data
             bg_dic:  the background dictionary
             tg_dic:  the anomaly dictionary
             bg_dic_ac_label:  the index of background dictionary atoms
             tg_dic_label: the index of anomaly dictionary atoms
    """

    data2d = hyper.hyperconvert2d(data3d)
    rows, cols, bands = data3d.shape
    data2d = hyper.hypernorm(data2d, "L2_norm")
    sio.savemat("data2d.mat", {'data2d': data2d})
    data3d = hyper.hyperconvert3d(data2d, rows, cols, bands)
    pca = decomposition.PCA(n_components=20, copy=True, whiten=False)
    dim_data = pca.fit_transform(data2d.transpose())
    data3d_dim = hyper.hyperconvert3d(dim_data.transpose(), rows, cols, 10)
    win_dim = hyper.hyperwincreat(data3d_dim, win_size)
    cluster_assment = hyper.Kmeans_win(win_dim, cluster_num)
```

```
sio.savemat("cluster_assment.mat", {'cluster_assment': cluster_assment})
win_matrix = hyper.hyperwincreat(data3d, win_size)
sio.savemat("win_matrix.mat", {'win_matrix': win_matrix})
wm_rows, wm_cols, wm_n = win_matrix.shape
resdiual_stack = np.zeros((bands, win_size * win_size, wm_n))
save_num = 0
bg_dic_tuple = []
bg_dic_ac_tuple = []
bg_dic_fc_tuple = []
class_order_data_index_tuple = []
anomaly_weight_tuple = []
for i in range(cluster_num):
    print("current calculate cluster  {0}  ...".format(i))
    tmp = np.where(cluster_assment == i)
    if tmp[0].size == 0:
        continue
    else:
        class_data = win_matrix[:, :, tmp[0]]
        cd_rows, cd_cols, cd_n = class_data.shape
        dictionary = class_data[:, int((win_size * win_size + 1)/2), :]
        dic_rows, dic_cols = dictionary.shape
        class_alpha = np.zeros((K, cd_cols, cd_n))
        class_index = np.zeros((K, cd_n))
        for j in range(cd_n):
            X = class_data[:, :, j]
            dictionary[:, (j * cd_cols): (j*cd_cols + cd_cols - 1)] = 0
            alpha, index, chosen_atom, resdiual = hyper.somp(dictionary, X, K)
            class_alpha[:, :, j] = alpha
            class_index[:, j] = index.transpose()
            resdiual_stack[:, :, save_num + j] = resdiual

        save_num = save_num + cd_n
        class_index = class_index.astype('int')
        class_global_alpha = np.zeros((dic_cols, cd_cols, cd_n))
```

```
    class_global_frequency = np.zeros((dic_cols, cd_cols, cd_n))
    for n_index in range(cd_n):
        class_global_alpha[class_index[:, n_index], :, n_index] = class_alpha[:, :, n_index]
        class_global_frequency[class_index[:, n_index], :, n_index] = 1

    posti_class_global_alpha = np.fabs(class_global_alpha)
    data_frequency = class_global_frequency[:, 0, :]
    frequency = np.sum(data_frequency, axis=1)
    sum_frequency = np.sum(frequency)
    norm_frequency = frequency/sum_frequency
    data_mean_alpha = np.mean(posti_class_global_alpha, axis=1)
    sum_alpha_2 = np.sum(data_mean_alpha, axis=1)
    norm_tmp = np.linalg.norm(sum_alpha_2)
    sparsity_score = sum_alpha_2 / norm_tmp
    anomaly_weight = norm_frequency
    anomaly_weight[frequency > 0] = sparsity_score[frequency > 0] / frequency[frequency > 0]
    # sparsity_score = sparsity_score * norm_frequency
    sparsity_sort_index = np.argsort(- sparsity_score)
    sparsity_sort_index = sparsity_sort_index.astype('int')
    frequency_sort_index = np.argsort(- norm_frequency)
    frequency_sort_index = frequency_sort_index.astype('int')
    tmp_class_dic_label = np.array(tmp[0])
    class_order_data_index_tuple.append(tmp_class_dic_label)
    selected_dic_num = np.round(selected_dic_percent * cd_n)
    selected_dic_num = selected_dic_num.astype('int')
    bg_dic_ac_tuple.append(tmp_class_dic_label[sparsity_sort_index[0: selected_dic_num]])
    bg_dic_fc_tuple.append(tmp_class_dic_label[frequency_sort_index[0: selected_dic_num]])
    anomaly_weight_tuple.append(anomaly_weight)
    bg_dic_tuple.append(dictionary[:, sparsity_sort_index[0: selected_dic_num]])

    # sio.savemat(result_path + "dic_{0}_frequency.mat".format(i), {'dic_frequency':
frequency})
    # sio.savemat(result_path + "dic_{0}_reflect.mat".format(i), {'dic_reflect': sum_alpha_2})
```

```
bg_dic = np.column_stack(bg_dic_tuple)
bg_dic_ac_label = np.hstack(bg_dic_ac_tuple)
bg_dic_fc_label = np.hstack(bg_dic_fc_tuple)
anomaly_weight = np.hstack(anomaly_weight_tuple)
class_order_data_index = np.hstack(class_order_data_index_tuple)
norm_res = np.zeros((wm_n, win_size * win_size))
for i in range(wm_n):
    norm_res[i, :] = np.linalg.norm(resdiual_stack[:, :, i], axis=0)
mean_norm_res = np.mean(norm_res, axis=1) * anomaly_weight.transpose()
anomaly_level = mean_norm_res/np.linalg.norm(mean_norm_res)
tg_sort_index = np.argsort(- anomaly_level)
tg_dic = data2d[:, class_order_data_index[tg_sort_index[0: target_dic_num]]]
print ("successs!!")

sio.savemat("bg_dic.mat", {'bg_dic': bg_dic})
sio.savemat("bg_dic_ac_label.mat", {'bg_dic_ac_label': bg_dic_ac_label})
sio.savemat("bg_dic_fc_label.mat", {'bg_dic_fc_label': bg_dic_fc_label})
sio.savemat("tg_dic.mat", {'tg_dic': tg_dic})
tg_dic_label = class_order_data_index[tg_sort_index[0:target_dic_num]]
sio.savemat("tg_dic_label.mat", {'tg_dic_label': tg_dic_label})
return data2d, bg_dic, tg_dic, bg_dic_ac_label, tg_dic_label
```

7.1.4 ROC_AUC. Py

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import metrics
import HyperProTool as hyper

def ROC_AUC(target2d, groundtruth):
    """
    :param target2d: the 2D anomaly component
    :param groundtruth: the groundtruth
    :return: auc: the AUC value
    """

    rows, cols = groundtruth.shape
    label = groundtruth.transpose().reshape(1, rows * cols)
    result = np.zeros((1, rows * cols))
    for i in range(rows * cols):
        result[0, i] = np.linalg.norm(target2d[:, i])

    # result = hyper.hypernorm(result, "minmax")
    fpr, tpr, thresholds = metrics.roc_curve(label.transpose(), result.transpose())
    auc = metrics.auc(fpr, tpr)
    plt.figure(2)
    plt.plot(fpr, tpr)
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.show()
    return auc
```

CHAPTER 8: TESTING

8.1 TESTING METHODOLOGIES

There are different types of testing methods or techniques used as part of the software testing methodology. Some of the important testing methodologies are:

8.1.1 WHITE BOX TESTING

White box testing (clear box testing, glass box testing, and transparent box testing or structural testing) uses an internal perspective of the system to design test cases based on internal structure. It requires programming skills to identify all paths through the software. The tester chooses test case inputs to exercise paths through the code and determines the appropriate outputs. While white box testing is applicable at the unit, integration and system levels of the software testing process, it is typically applied to the unit. While it normally tests paths within a unit, it can also test paths between units during integration, and between subsystems during a system level test.

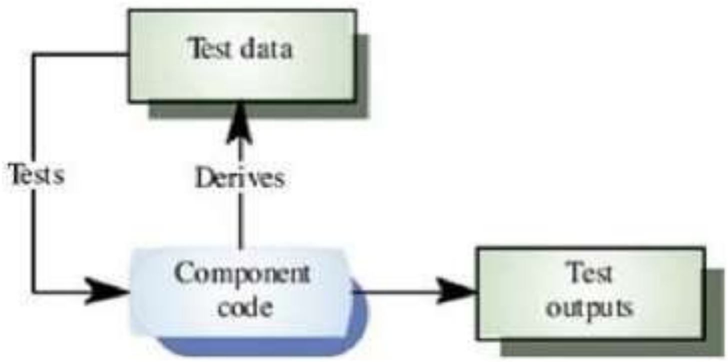

Fig. 8: White box Testing

Though this method of test design can uncover an overwhelming number of test cases, it might not detect unimplemented parts of the specification or missing requirements, but one can be sure that all paths through the test object are executed. Using white box testing we can derive test cases that:

- Guarantee that all independent paths within a module have been exercised atleast once.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.
- Execute internal data structure to assure their validity.

ADVANTAGES OF WHITE BOX TESTING

- To start the white box testing of the desired application there is no need to wait for user face (UI) to be completed. It covers all possible paths of code which will ensure a thorough testing.

- It helps in checking coding standards.
- Tester can ask about implementation of each section, so it might be possible to remove unused/deadlines of codes helps in reducing the number of test cases to be executed during the black box testing.
- As the tester is aware of internal coding structure, then it is helpful to derive which type of input data is needed to test the software application effectively.
- White box testing allows you to help in code optimization.

DISADVANTAGES OF WHITE BOXING TESTING

- To test the software application a highly skilled resource is required to carry out testing whohas good knowledge of internal structure of the code which will increase the cost.
- Updating the test script is required if there is change in requirement too frequently.
- If the application to be tested is large in size, then exhaustive testing is impossible.
- It is not possible for testing each and every path/condition of software program, which mightmiss the defects in code.
- White box testing is a very expensive type of testing.
- To test each path or conditions may require different input conditions, so in order to test full application, the tester need to create range of inputs which may be a time consuming.

8.1.2 BLACK BOX TESTING

Black box testing focuses on the functional requirements of the software. It is also known asfunctional testing. It is a software testing technique whereby the internal workings of the item being tested are not known by the tester. For example, in a black box test on software design the tester onlyknows the inputs and what the expected outcomes should be and not how the program arrives at those outputs.

The tester does not ever examine the programming code and does not need any further knowledge of the program other than its specifications. It enables us to derive sets of inputs that will

Fig. 9: Black Box Testing

fully exercise all functional requirements for a program.

Black box testing is an alternative to white box technique. Rather it is a complementary approach that is likely to uncover a different class of errors in the following categories: -

- Incorrect or missing function.
- Interface errors.
- Performance errors.
- Initialization and termination errors.
- Errors in objects.

ADVANTAGES OF BLACK BOX TESTING

- The test is unbiased as the designer and the tester are independent of each other.
- The tester does not need knowledge of any specific programming languages.
- The test is done from the point of view of the user, not the designer.
- Test cases can be designed as soon as the specifications are complete.

DISADVANTAGES OF BLACK BOX TESTING

- The test inputs need to be from large sample space. That is, from a huge set of data this willtake time.
- Also, it is difficult to identify all possible inputs in limited testing time. So, writing test casesis slow and difficult.
- Chances are more that there will be unidentified paths during this testing.

8.2 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application.This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/orsystem configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

8.3 SYSTEM TESTING

This information contributes towards reducing the ambiguity about the system. For example, when deciding whether to release a product, the decision makers would need to know the state of the product including aspects such as the conformance of the product to requirements, the usability of the product, any known risks, the product's compliance to any applicable regulations.

Software testing enables making objective assessments regarding the degree of conformance of the system to stated requirements and specifications. System testing checks complete end-end scenarios, as a user would exercise the system. The system has to be tested for correctness of the functionality by setting it up in a controlled environment. System testing includes testing of functional and nonfunctional requirements. It helps to verify and validate the system. All components system should have been successfully unit tested and then checked for any errors after integration.

8.4 QUALITY ASSURANCE

Quality assurance consists of the auditing and reporting functions of management. The goal of quality assurance is to provide management with the data necessary to be informed about product quality, thereby gaining insight and confident that the product quality is meeting its goals. This is an "umbrella activity" that is applied throughout the engineering process. Software quality assurance encompasses: -

- Analysis, design, coding and testing methods and tools.
- Formal technical reviews that are applied during each software engineering.
- Multi-tiered testing strategy.
- Control of software documentation and the change made to it.
- A procedure to ensure compliance with software development standards.
- Measurement and reporting mechanisms.

Quality Factors

An important objective of quality assurance is to track the software quality and assess the impact of methodological and procedural changes on improved software quality. The factors that affect the quality can be categorized into two broad groups:

- Factors that can be directly measured.
- Factors that can be indirectly measured.

These factors focus on important aspects of a software product.

- Its operational characteristics
- Its ability to undergo changes
- Its adaptability to a new environment.
- Effectiveness or efficiency in performing its mission

CHAPTER 9: RESULT & PERFORMANCE ANALYSIS

9.1 OUTPUT SNAPSHOTS

Fig.10. Original Image

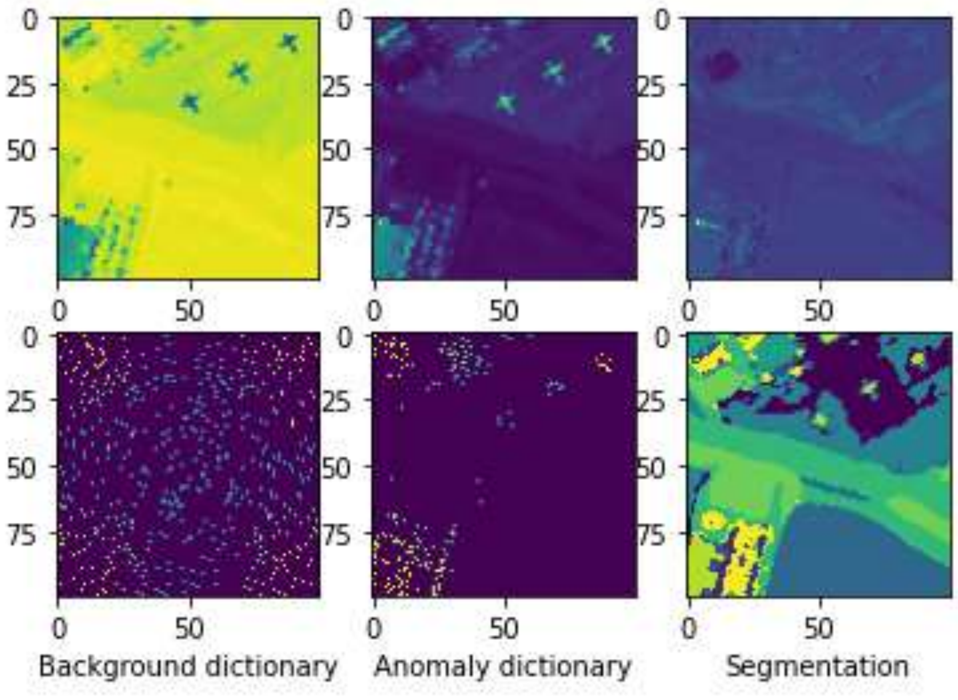

Fig. 11. Dictionaries images

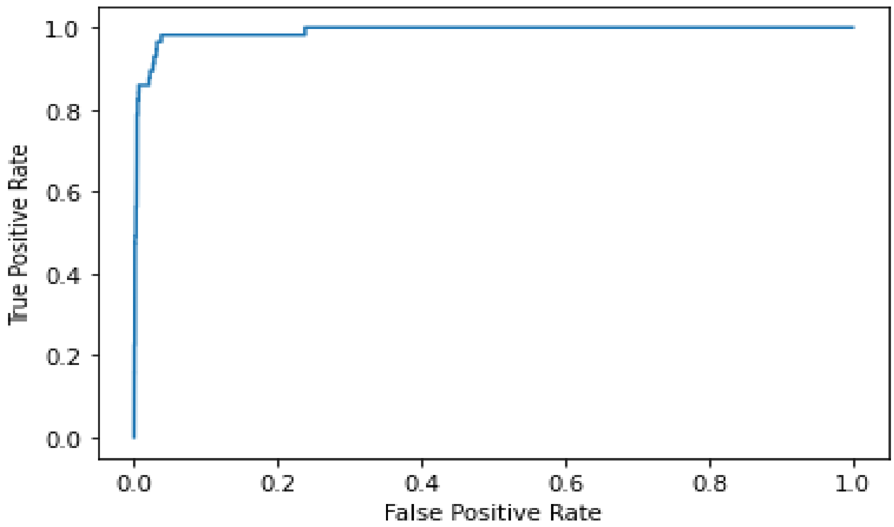

Fig. 12. ROC Curve

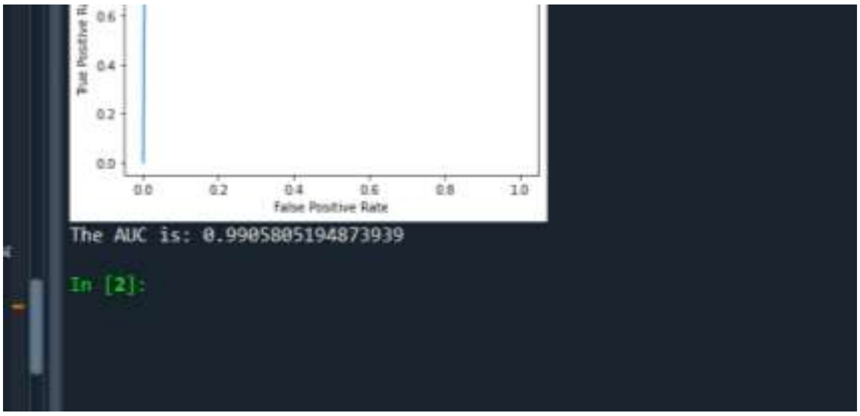

Fig. 13. AUC Value

Fig. 14. Groundtruth

9.2 RESULT ANALYSIS

By Briefly examining the output generated by the model we build, we noticed that model with ROC value 0.99 . It means, that model detect the anomaly with an accuracy 99.05%.

CHAPTER 10: CONCLUSION

Having the models and research work based on the low rank approximation and sparse representation regarding the detection of anomalies in hyperspectral images, here in this project, we tried the combination of low rank approximation and joint sparse representation with Dictionary learning concept to yield the results of anomaly detection with better accuracy and we had reached it at our best during project development.

CHAPTER 11: FUTURE WORK

On the basis of future scope, we can integrate the work that done now with the deep learning and enhanced deep learning in order to classify the pixels more accurately according to similar classes to provide the ideal accuracy. Not only with DL, Auto-Encoding techniques might be applied to transform the data as convenient to implement the model on data.

REFERENCES

1. D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.
2. Q. Shi, X. Tang, T. Yang, and R. Liu, "Hyperspectral image denoising using a 3D attention denoising network," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2020.3045273.
3. D. He, Q. Shi, X. Liu, and Y. Zhong, "Deep subpixel mapping based on semantic information modulated network for urban land use mapping," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2021.3050824.
4. Q. Shi *et al.*, "A deeply supervised attention metric-based network and an open aerial image dataset for remote sensing change detection," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2021.3085870.
5. N. M. Nasrabadi, "Hyperspectral target detection: An overview of current and future challenges," *IEEE Signal Process Mag.*, vol. 31, no. 1, pp. 34–44, Jan. 2014.
6. A. Plaza *et al.*, "Recent advances in techniques for hyperspectral image processing," *Remote Sens. Environ.*, vol. 113, pp. S110–S122, 2009.
7. Y. Dong, T. Liang, Y. Zhang, and B. Du, "Spectral-spatial weighted Kernel manifold embedded distribution alignment for remote sensing image classification," *IEEE Trans. Cybern.*, vol. 51, no. 6, pp. 3185–3197, Jun. 2021.
8. H. Fabelo *et al.*, "In-vivo hyperspectral human brain image database for brain cancer detection," *IEEE Access*, vol. 7, pp. 39098–39116, 2019.
9. D. M. Varade, A. K. Maurya, and O. Dikshit, "Development of spectral indexes in hyperspectral imagery for land cover assessment," *IETE Tech. Rev.*, vol. 36, no. 5, pp. 475–483, 2019.
10. B. Du and L. Zhang, "Random-selection-based anomaly detector for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 5, pp. 1578–1589, May 2011.
11. Chein-I Chang, "Anomaly Detection and Classification for Hyperspectral Imagery", *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 6, june2002
12. Manel Ben Salem, "Anomaly detection in hyperspectral imagery: an overview", *IEEE Image Processing Applications and System Conference, 2014.*
13. Yang Xu, Zebin Wu, "A Novel Hyperspectral Image Anomaly Detection Method Based On Low Rank Representation", IGARSS 2015.
14. Shaoyu Wang, Xinyu Wang, "Auto-AD: Autonomous Hyperspectral Anomaly Detection Network Based on Fully Convolutional Autoencoder", *IEEE Trans. Geosci. Remote Sens.*