SEMINAR REPORT

ON

"DEEP LEARNING IN INTRUSION DETECTION SYSTEMS"

BY

MR. PREM PAWAR

Exam No:71815521B

Under the guidance of

Mr. PRASHANT SADAPHULE

DEPARTMENT OF COMPUTER ENGINEERING

ALL INDIA SHRI SHIVAJI MEOMRIAL SOCIETY'S

INSTITUTE OF INFORMATION TECHNOLOGY

PUNE 411001

SAVITRIBAI PHULE PUNE UNIVERSITY

2019-2020

AISSMS

INSTITUTE OF INFORMATION TECHNOLOGY

ADDING VALUE TO ENGINEERING

Approved by AICTE New Delhi, Recognized by the Government of Maharashtra
and Affiliated to Savitribai Phule Pune University.
Accredited by NAAC with A grade

Department of Computer Engineering

CERTIFICATE

This is to certify that **Mr. PREM PAWAR Exam No.:71815521B** from **Third Year I^{st} Shift Computer Engineering** has successfully complited his/her seminar work titled

"AIR POLLUTION HOTSPOT DETECTION"

at All India Shri Shivaji Memorial Society's Institute of Information Technology, Pune in the partial fulfillment of the Bachlors Degree in Computer Engineering

Mr. Prashant Sadaphule
Internal Guide

Mrs. Prajwal Gaikwad
Seminar Coordinator

Seal/Stamp of the college

Dr. S. N. Zaware
Head of the Department
Computer Engineering

Place: PUNE
Date:

ABSTRACT

Spatial data mining is the extraction of implicit knowledge, spatial relations or other patterns not explicitly stored in spatial database. The focus of this paper is placed on the information derivation of spatial data. Geographical coordinates of hot spots in forest fire regions, which are extracted from the satellite images, are studied and used in the detection of likely fire points. False alarms can occur in the derived hotspots. While this false information can be identified by comparing the radiance detected at several bands, we introduce a different approach to remove some of the false alarms. We use clustering and Hough transformation to determine regular patterns in the derived hotspots and classify them as false alarms on the assumption that fires usually do not spread in regular patterns such as in a straight line.

Machine Learning is one of the major popular research topic and its relay with the evaluation of techniques and methods which enable the data processor to learn and execute activities. The basic idea of AQI is mining the data sets from different satellites parameters and providing the final output with moderate spatial resolution on pollution information. Hence information will be useful for Indian ministry to change current plans.

By leveraging a socially meaningful air quality information system with high-density, high-accuracy, nearreal- time data, Indian Ministry can use this information to help make pollution mitigation decisions, change current plans and announce preventive public alerts. An approach for gathering accurate air pollution information can use end user applications in handheld or wearable devices and navigation systems. With this valuable information, quality of life improvements can be made such as when to plan sporting or other outdoor activities or routing traffic to lesser polluted areas.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who gave me the possibility to complete this report. I pay my deep sense of gratitude to **Dr. S.N. Zaware(HOD)** of Computer Engineering department,AISSMS IOIT,Pune. A special thanks to our Third year project coordinator **Mrs. Prajwal Gaikwad**, whose help, stimulating suggestions and encouragement ,helped me to coordinate my project.

Many thanks to the project guide, **Mr. Prashant Sadaphule** who have given her full effort in guiding the team in achieving the goal as well as her encouragement to main our progress on track. I would like to appreciate the guidance given by other supervisors as well as staff members in our project presentation that has improved my presentation skills by their tips.

Mr. Prem Pawar
AISSMS IOIT, Pune.

INDEX

List of Figures

List of Tables

Chapter 1

INTRODUCTION

Our planet is plagued by environmental problems that deplete natural resources and strain livelihoods, many of which are exacerbated by poor industrial practices. If left unchecked, it can affect livelihoods. In 2018's Environmental Performance Index India is ranked 177 out of 180. Air pollution in India is caused by fuel wood and biomass burning, burning of crop residue in agriculture fields on a large scale, emission from vehicles and traffic congestion etc. India is the third largest in the emission of greenhouse gases after China and the United States. The severity of air pollution is so much that life expectancy among Indians on an average reduces by 3.4 years while among the residents of Delhi it reduces by almost 6.3 years. According to UN reports 55 of the world's population lives in urban areas in 2018 and one in eight people live in one of 33 mega-cities each with more than 10 million inhabitants. By 2050, 58 of the global population will live in hyper-dense urban areas, resulting in greater energy consumption, more waste, and more traffic congestion which will adversely affect air quality and increase the pollution levels of cities. By leveraging a socially meaningful air quality information system with high-density, high-accuracy, nearreal-time data, Indian Ministry can use this information to help make pollution mitigation decisions, change current plans and announce preventive public alerts. An approach for gathering accurate air pollution information can use end user applications in handheld or wearable devices and navigation systems. With this valuable information, quality of life improvements can be made such as when to plan sporting or other outdoor activities or routing traffic to lesser polluted areas. Efficiencies in mitigating air pollution can also be realized as connected air purifier systems receiving highly accurate pollution data can save vast amount of energy working always at the optimal load.

Spatial data mining is the extraction of implicit knowledge, spatial relations or other patterns not explicitly stored in spatial database. The focus of this paper is placed on the information derivation of spatial data. Geographical coordinates of hot spots in forest fire regions, which are extracted from the satellite images, are studied and used in the detection of likely fire points. False alarms can occur in the derived hotspots. While this false information can be identified by comparing the radiance detected at several bands, we introduce a different approach to remove some of the false alarms. We use clustering and Hough transformation to determine regular patterns in the derived hotspots and classify them as false alarms on the assumption that fires usually do not spread in regular patterns such as in a straight line. This project demonstrates the application of spatial data mining to reduce false alarm from the set of hotspots derived from NOAA images.

Geostatistical analysis considers the concentration of a potentially hazard in soil as a regionalized variable in space. Geostatistics was developed as a means to describe spatial patterns of soil pollution by semivariograms and to predict the values of soil attributes at unsampled locations [10]. Geostatistical models could be used to estimate the spatial patterns of soil contaminant without measuring soil data in an entire area. The degree of contamination and hotspot areas for soils may vary with the methods used. For delineating hazardous areas, indicator kriging (IK) determines the spatial probability distribution of soil pollution in fields [6,11-16]. IK provides a non-parametric distribution estimated at an unsampled location directly using fixed thresholds and qualifies the spatial patterns of a hazardous risk. Moreover, stochastic simulation methods such as sequential indicator simulation (SIS), have been recently proposed to overcome the inherent limitations of IK [17-19]. The stochastic simulation method is based on a probabilistic model, and does not require any assumption for the shape of the conditional distribution and the systematically adds a stochastic noise component into the kriging model. Simulation with multiple realizations offers significant improvements over kriging techniques at sites with high data variations. Hotspot mapping is used to help identify where soil pollution exists and comes from. Recently, Kernel density estimation (KDE) is one of the methods for analyzing the first order properties of a point event distribution [20-22], in part because it is easy to understand and implement. KDE has been widely used for hotspot analysis and detection. The objective of KDE is to produce a smooth density surface of

point events over space by computing event intensity as density estimation [22-24]. Moreover, Schnabel and Tietje [23] applied the KDE method to spatially distributed heavy metal soil data and compared it with ordinary kriging. The results represent the interdependence between various heavy metal concentrations and additional site characteristics. Furthermore, the method could be a valuable supplement for the geostatistical uncertainty assessments.

1.1 MOTIVATION OF THE REPORT

The evolution of intrusion detection systems is motivated by some important facts as:

- Our planet is plagued by environmental problems that deplete natural resources and strain livelihoods, many of which are exacerbated by poor industrial practices. If left unchecked, it can affect livelihoods
- Air pollution is the main problem in india we all are facing. The cities like delhi are always have pollution level very high.
- Today india not have system to detect and predict pollution hotspot. So the motivation behind this topic is to detect air pollution hotspot and predict the future trajectory of hotspot using satellite data.
- Today mining the datasets from different satellites parameters and providing the final output with moderate spatial resolution on pollution information. Hence information will be useful for change detection analysis to predict future pathways.

Chapter 2

LITERATURE SURVEY

A number of approaches based on traditional machine learning, including SVM [10], [11], K-Nearest Neighbour (KNN) [12], ANN [13], Random Forest (RF) [14], [15] and others [16], [17], have been proposed and have achieved success for an intrusion detection system. In recent years, deep learning, a branch of machine learn- ing, has become increasingly popular and has been applied for intrusion detection; studies have shown that deep learning completely surpasses traditional methods. In [18], the authors utilize a deep learning approach based on a deep neural net- work for flow-based anomaly detection, and the experimental results show that deep learning can be applied for anomaly detection in software defined networks.

According to [20], RNNs are considered reduced-size neural networks. In that paper, the author proposes a three- layer RNN architecture with 41 features as inputs and four intrusion categories as outputs, and for misuse-based IDS. However, the nodes of layers are partially connected, the reduced RNNs do not show the ability of deep learn- ing to model high-dimensional features, and the authors do not study the performance of the model in the binary classification. With the continuous development of big data and comput- ing power, deep learning methods have blossomed rapidly, and have been widely utilized in various fields. Following this line of thinking, a deep learning approach for intrusion detection using recurrent neural networks (RNN-IDS) is pro- posed in this paper. Compared with previous works, we use the RNN-based model for classification rather than for pre- training. Besides, we use the NSL-KDD dataset with a sep- arate training and testing set to evaluate their performances in detecting network intrusions in both binary and multiclass classification, and we compare it with J48, ANN, RF, SVM and other machine learning methods proposed by previous researchers.

Chapter 3

Data Sources

The data is procured fromtwo sources namely Sentinel- 5p and Giovanni. The data from sources is different in terms of spatial resolution and temporal resolution. Solution consists of 2 different models which will work on both types of data respectively to identify major emission regions, trends and seasonal cycles over India.

daily, monthly basis, hovmoller plots and extract the coordinates, columnar concentrations and timestamp of the respective gases. This data is then converted into json and csv formats for further processing. The JSON data can be use to visualise the map in the front-end whereas the csv data will then be used in clustering algorithms to detect the hot spots.

3.1 Sentinel 5P

3.2 Information

Sentinel-5 Precursor is the first mission of the Copernicus Programme dedicated to monitoring air pollution. Its instrument is an ultraviolet, visible, near and short-wavelength infrared spectrometer called Tropomi. The Satellite is built on a hexagonal Astrobus L 250 satellite bus equipped with S- and X-band communication antennas, three foldable solar panels generating 1500 watts and hydrazine thrusters for station-keeping.[1][2]

The satellite operates in an 824 km Sun-synchronous orbit with a Local Time of Ascending Node of 13:30 hours.

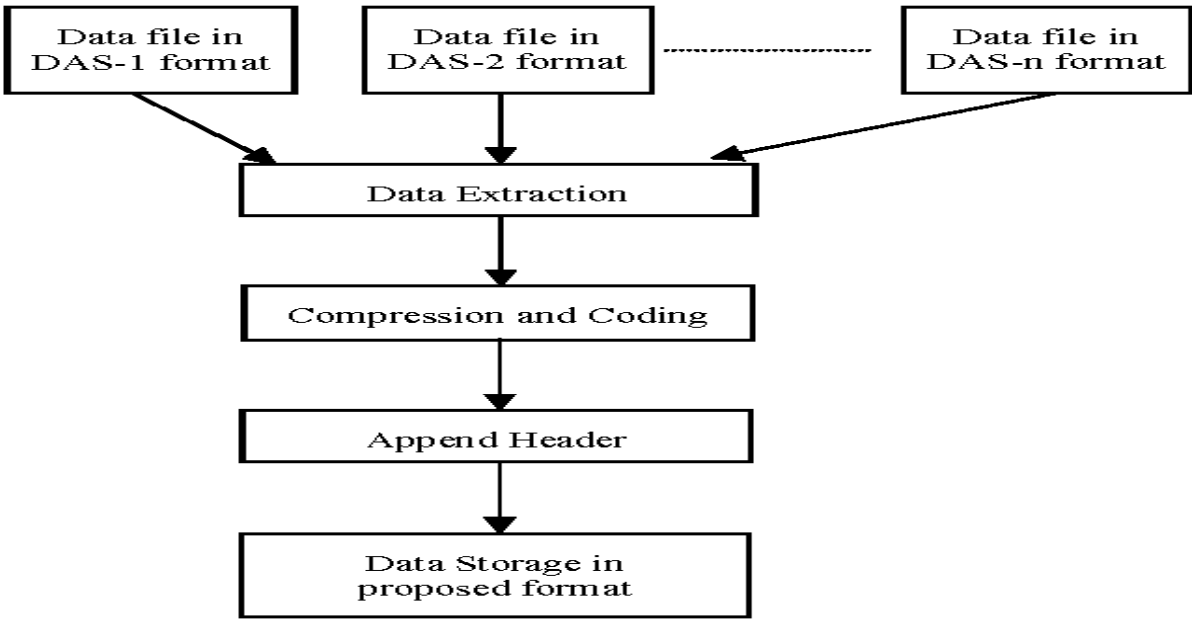


Figure 3.1: satellite data storage.

3.3 Tropomi

Tropomi (TROPOspheric Monitoring Instrument) is a spectrometer sensing ultraviolet (UV), visible (VIS), near (NIR) and short-wavelength infrared (SWIR) to monitor ozone, methane, formaldehyde, aerosol, carbon monoxide, NO2 and SO2 in the atmosphere. It extends the capabilities of the OMI from the Aura satellite and the SCIAMACHY instrument from Envisat.[5]

Tropomi will be taking measurements every second covering an area of approximately 2600 km wide and 7 km long in a resolution of 7 x 7 km. Light will be separated into different wavelengths using grating spectrometers and then measured with four different detectors for respective spectral bands. The UV spectrometer has a spectral range of 270-320 nm, the visible light spectrometer has a range of 310-500 nm, NIR has a range of 675-775 nm, and SWIR has a range of 2305-2385 nm.[6]

The instrument is split into four major blocks: UV-VIS-NIR spectrometers and a calibration block, SWIR spectrometer with its optics, instrument control unit and a cooling block. The total mass of Tropomi will be 200 kg with a power consumption of 170 watts on average and a data output of 140 Gbit per orbit.[6][1]

The instrument was built by a joint venture between the Netherlands Space Office, Royal Netherlands Meteorological Institute, Netherlands Institute for Space Research, Netherlands Organisation for Applied Scientific Research and Airbus Defence and Space Netherlands.

The SWIR spectrometer was designed and built by the Optical Payloads Group of Surrey Satellites (SSTL); it employs an immersed grating design in which light impinges upon an etched grating from within a high-index substrate (silicon in this case). The reduced wavelength within the refractive medium permits an efficient, space-saving design. The SWIR grating was provided by SRON (Netherlands), who also provided the Front-End Electronics (FEE). The SWIR spectrometer receives light from the main instrument via an intermediate pupil, and directs this - via a telescope - towards a slit which defines the along-track footprint of the instrument on the ground. Light from the slit is re-collimated, diffracted by the immersed-grating at high-order and finally imaged onto a two-dimensional detector by a high aperture relay lens. The SWIR detector (furnished by Sofradir, France) has 256 elements in the across-track direction and 1024 elements in the spectral direction (the element pitch is 30 microns); it is operated cold (typically 140 K). The SWIR spectrometer optics are mounted on a cooled optical bench (approximately 200K) and the instrument is insulated by a multiple-layer insulation (MLI) blanket. The SWIR instrument was aligned, focussed and characterised at the Mullard Space Science laboratory thermal vacuum facility in Surrey, UK.

The Tropospheric Monitoring Instrument provides the most detailed methane emissions monitoring available. It has a resolution of about 50 square kilometres.

3.4 Copernicus Program

Copernicus is the European Union's Earth observation programme coordinated and managed by the European Commission in partnership with the European Space Agency (ESA), the EU Member States and EU Agencies.[1]

It aims at achieving a global, continuous, autonomous, high quality, wide range Earth observation capacity. Providing accurate, timely and easily accessible information to, among other things, improve the management of the environment, understand and mitigate the effects of climate change, and ensure civil security.[2]

The objective is to use vast amount of global data from satellites and from ground-

based, airborne and seaborne measurement systems to produce timely and quality information, services and knowledge, and to provide autonomous and independent access to information in the domains of environment and security on a global level in order to help service providers, public authorities and other international organizations improve the quality of life for the citizens of Europe. In other words, it pulls together all the information obtained by the Copernicus environmental satellites, air and ground stations and sensors to provide a comprehensive picture of the "health" of Earth.

One of the benefits of the Copernicus Programme is that the data and information produced in the framework of Copernicus are made available free-of-charge [3] to all its users and the public, thus allowing downstream services to be developed.

The services offered by Copernicus cover six main interacting themes: atmosphere, marine, land, climate, emergency and security.[4]

Copernicus builds upon three components:

The space component (observation satellites and associated ground segment with missions observing land, atmospheric and oceanographic parameters) This comprises two types of satellite missions, ESA's five families of dedicated Sentinel (space missions) and missions from other space agencies, called Contributing Missions, In-situ measurements (ground-based and airborne data-gathering networks providing information on oceans, continental surface and atmosphere), Services developed and managed by Copernicus and offered to its users and public in general.

3.5 Giovanni

Giovanni (meteorology) - Web interface that allows users to analyze NASA's gridded data from various satellite and surface observations.

Giovanni provides researchers with the capability to examine data on atmospheric chemistry, atmospheric temperature, water vapor and clouds, atmospheric aerosols, precipitation, and ocean chlorophyll and surface temperature. The primary data consist of global gridded data sets with reduced spatial resolution. Basic analytical functions performed by Giovanni currently are carried out by the Grid Analysis and Display System (GrADS).

The GES-DISC Interactive Online Visualization ANd aNalysis Infrastructure (Giovanni) allows to explore satellite data using sophisticated analyses and visualizations.

Giovanni allows access to data from multiple remote sites, supports multiple data formats including Hierarchical Data Format (HDF), HDF-EOS, network Common Data Form (netCDF), GRIdded Binary (GRIB), and binary, and multiple plot types including area, time, Hovmoller, and image animation.

Chapter 4

Clustering

Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including pattern recognition, image analysis, information retrieval, bioinformatics, data compression, computer graphics and machine learning.

Cluster analysis itself is not one specific algorithm, but the general task to be solved. It can be achieved by various algorithms that differ significantly in their understanding of what constitutes a cluster and how to efficiently find them. Popular notions of clusters include groups with small distances between cluster members, dense areas of the data space, intervals or particular statistical distributions. Clustering can therefore be formulated as a multi-objective optimization problem. The appropriate clustering algorithm and parameter settings (including parameters such as the distance function to use, a density threshold or the number of expected clusters) depend on the individual data set and intended use of the results. Cluster analysis as such is not an automatic task, but an iterative process of knowledge discovery or interactive multi-objective optimization that involves trial and failure. It is often necessary to modify data preprocessing and model parameters until the result achieves the desired properties.

4.1 Clustering Algorithms

following are the types of clustering algorithms:

- Region Growing Method: Region growing method helps in forming the clusters as follows: • Step 1: Choose seed point from available data set and compare

with the remaining points in the data set. • Step 2: Cluster is then grow from chosen seed point and add neighbouring points which are close to it. • Step 3: Once the growth of one cluster stops, choose another seed point which does not belong to any cluster, and follow same step 1 and step 2 for the formation of other clusters. • Step 4: The above process is repeated till all points in the data sets have been covered in formation of clusters.

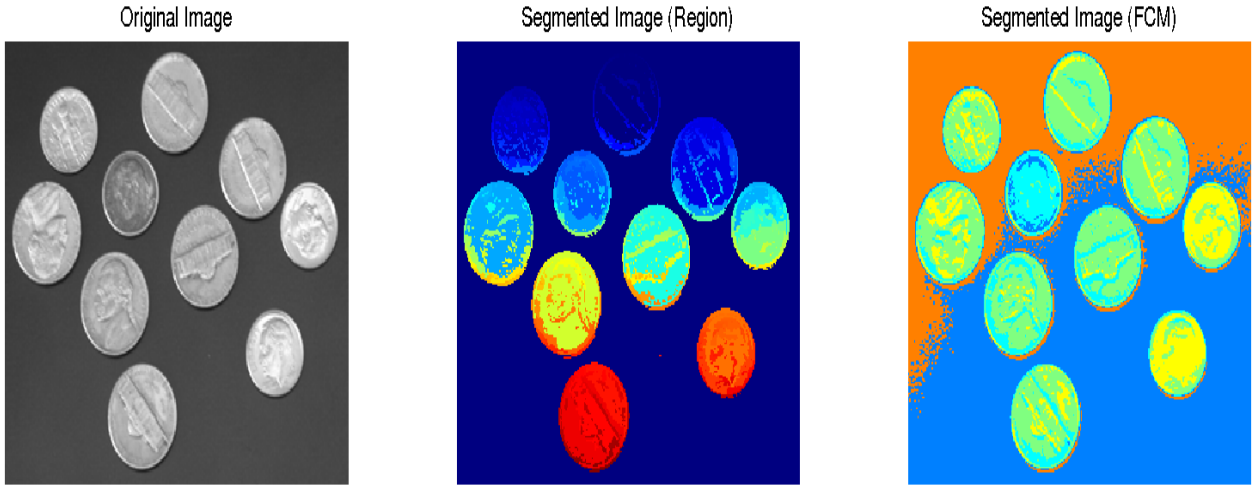


Figure 4.1: region growing method.

- Connectivity-based clustering (hierarchical clustering): Connectivity-based clustering, also known as hierarchical clustering, is based on the core idea of objects being more related to nearby objects than to objects farther away. These algorithms connect "objects" to form "clusters" based on their distance. A cluster can be described largely by the maximum distance needed to connect parts of the cluster. At different distances, different clusters will form, which can be represented using a dendrogram, which explains where the common name "hierarchical clustering" comes from: these algorithms do not provide a single partitioning of the data set, but instead provide an extensive hierarchy of clusters that merge with each other at certain distances. In a dendrogram, the y-axis marks the distance at which the clusters merge, while the objects are placed along the x-axis such that the clusters don't mix.

- Centroid-based clustering: In centroid-based clustering, clusters are represented by a central vector, which may not necessarily be a member of the data set.

When the number of clusters is fixed to k, k-means clustering gives a formal definition as an optimization problem: find the k cluster centers and assign the objects to the nearest cluster center, such that the squared distances from the cluster are minimized.

The optimization problem itself is known to be NP-hard, and thus the common approach is to search only for approximate solutions. A particularly well known approximate method is Lloyd's algorithm, often just referred to as "k-means algorithm" (although another algorithm introduced this name). It does however only find a local optimum, and is commonly run multiple times with different random initializations. Variations of k-means often include such optimizations as choosing the best of multiple runs, but also restricting the centroids to members of the data set (k-medoids), choosing medians (k-medians clustering), choosing the initial centers less randomly (k-means++) or allowing a fuzzy cluster assignment (fuzzy c-means).

- Distribution-based clustering: The clustering model most closely related to statistics is based on distribution models. Clusters can then easily be defined as objects belonging most likely to the same distribution. A convenient property of this approach is that this closely resembles the way artificial data sets are generated: by sampling random objects from a distribution.

  While the theoretical foundation of these methods is excellent, they suffer from one key problem known as overfitting, unless constraints are put on the model complexity. A more complex model will usually be able to explain the data better, which makes choosing the appropriate model complexity inherently difficult.

- Density-based clustering: In density-based clustering, clusters are defined as areas of higher density than the remainder of the data set. Objects in sparse areas - that are required to separate clusters - are usually considered to be noise and border points.

  The most popular density based clustering method is DBSCAN. In contrast to many newer methods, it features a well-defined cluster model called "density-reachability". Similar to linkage based clustering, it is based on connecting points within certain distance thresholds. However, it only connects points

that satisfy a density criterion, in the original variant defined as a minimum number of other objects within this radius.

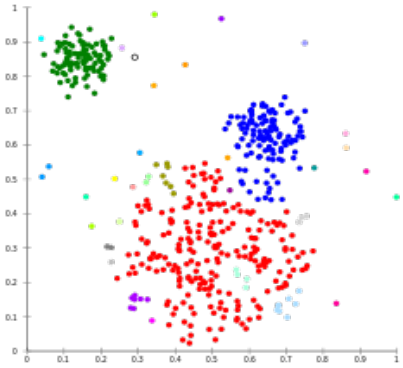


Figure 4.2: before clustering.

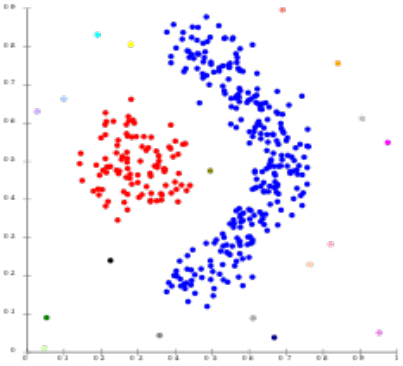


Figure 4.3: after clustering.

4.2 How algorithm works:

Kmeans algorithm is an iterative algorithm that tries to partition the dataset into Kpre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group.

The approach kmeans follows to solve the problem is called Expectation-Maximization. The E-step is assigning the data points to the closest cluster. The M-step is computing the centroid of each cluster. Below is a break down of how we can solve it mathematically (feel free to skip it). The objective function is:

$$J = \sum_{i=1}^{m} \sum_{k=1}^{K} w_{ik} \left\| x^i - \mu_k \right\|^2 \qquad (1)$$

Figure 4.4: objective function

where wik=1 for data point xi if it belongs to cluster k; otherwise, wik=0. Also, k is the centroid of xi's cluster. It's a minimization problem of two parts. We first minimize J w.r.t. wik and treat k fixed. Then we minimize J w.r.t. k and treat wik fixed. Technically speaking, we differentiate J w.r.t. wik first and update cluster assignments (E-step). Then we differentiate J w.r.t. k and recompute the centroids after the cluster assignments from previous step (M-step). Therefore, E-step is:

$$\frac{\partial J}{\partial w_{ik}} = \sum_{i=1}^{m} \sum_{k=1}^{K} \left\| x^i - \mu_k \right\|^2$$

$$\Rightarrow w_{ik} = \begin{cases} 1 & \text{if } k = argmin_j \left\| x^i - \mu_j \right\|^2 \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

Figure 4.5: E-Step

In other words, assign the data point xi to the closest cluster judged by its sum of squared distance from cluster's centroid. And M-step is:

$$\frac{\partial J}{\partial \mu_k} = 2 \sum_{i=1}^{m} w_{ik} \left(x^i - \mu_k \right) = 0$$

$$\Rightarrow \mu_k = \frac{\sum_{i=1}^{m} w_{ik} x^i}{\sum_{i=1}^{m} w_{ik}} \qquad (3)$$

Figure 4.6: M-step

Which translates to recomputing the centroid of each cluster to reflect the new assignments. Few things to note here: Since clustering algorithms including k-means use distance-based measurements to determine the similarity between data points, it's recommended to standardize the data to have a mean of zero and a standard deviation of one since almost always the features in any data-set would have different units of measurements such as age vs income. Given k-means iterative nature and the random initialization of centroids at the start of the algorithm, different initialization may lead to different clusters since k-means algorithm may stuck in a local optimum and may not converge to global optimum. Therefore, it's recommended to run the algorithm using different initialization of centroids and pick the results of the run that that yielded the lower sum of squared distance. Assignment of examples isn't changing is the same thing as no change in within-cluster variation: We'll use simple

$$\frac{1}{m_k}\sum_{i=1}^{m_k}\left\|x^i-\mu_{c^k}\right\|^2 \qquad (4)$$

Figure 4.7: Variation

implementation of kmeans here to just illustrate some concepts. Then we will use sklearn implementation that is more efficient take care of many things for us kmeans algorithm is very popular and used in a variety of applications such as market segmentation, document clustering, image segmentation and image compression, etc. The goal usually when we undergo a cluster analysis is either: Get a meaningful intuition of the structure of the data we're dealing with. Cluster-then-predict where different models will be built for different subgroups if we believe there is a wide variation in the behaviors of different subgroups. An example of that is clustering patients into different subgroups and build a model for each subgroup to predict the probability of the risk of having heart attack.

As the graph below shows that we only ended up with two different ways of clustering based on different initialization. We would pick the one with the lowest sum of squared distance.

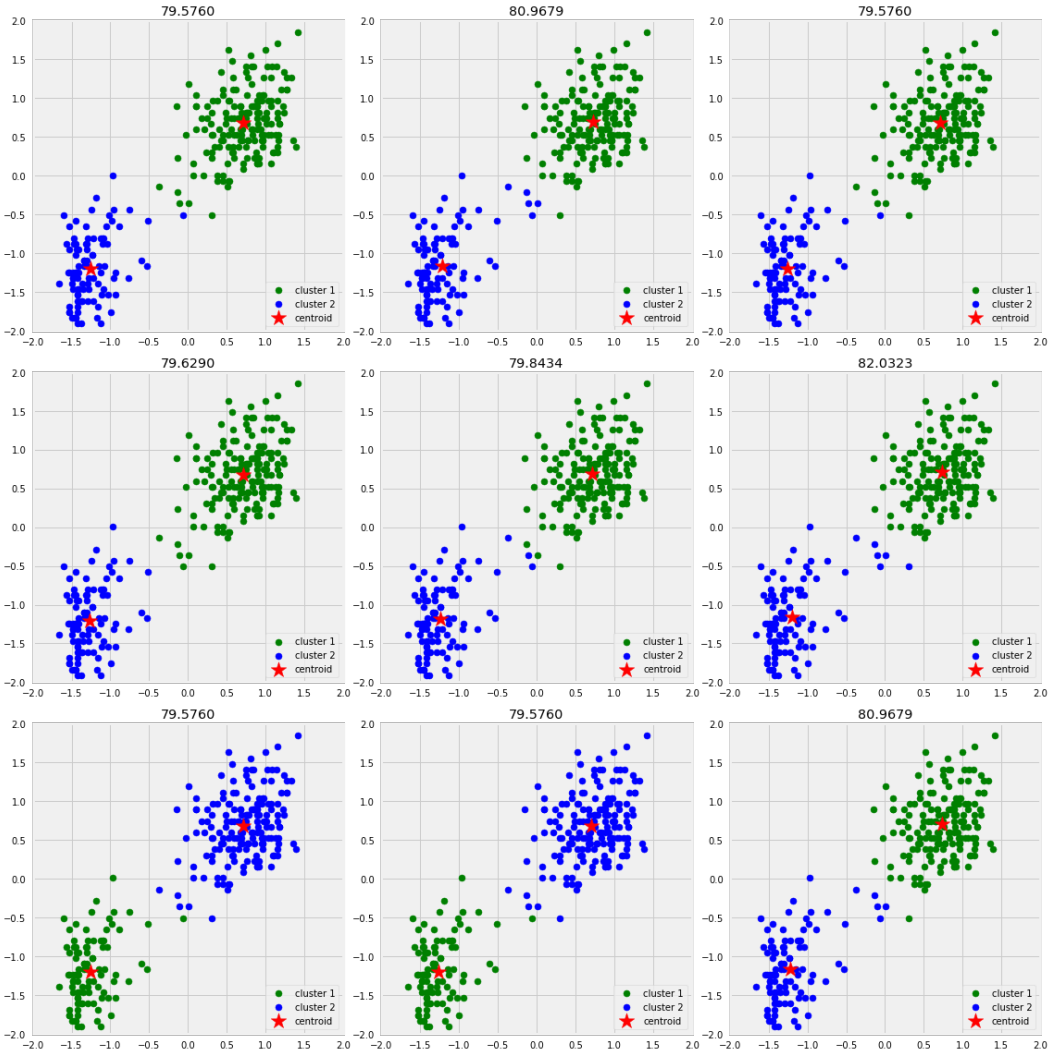


Figure 4.8: output of algorithm

Chapter 5

Hough Trasnsform

The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing.[1] The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform.

The classical Hough transform was concerned with the identification of lines in the image, but later the Hough transform has been extended to identifying positions of arbitrary shapes, most commonly circles or ellipses. The Hough transform as it is universally used today was invented by Richard Duda and Peter Hart in 1972, who called it a "generalized Hough transform"[2] after the related 1962 patent of Paul Hough.[3][4] The transform was popularized in the computer vision community by Dana H. Ballard through a 1981 journal article titled "Generalizing the Hough transform to detect arbitrary shapes".

Once the clusters are formed, we can proceed to detect the false alarm on the assumption of points that got from clustering. Hough Transform is an algorithm which is used to link points on edge by determining whether they lie of a cluster curve of specified shape. Data mining is use with this algorithm to verify if there is something on the derived hot spots location from SPOT satellite images. The flagged hot spots is then fed to time series module to trace the source trajectories, and based on that pattern, we predict future pathways

5.1 theory

In automated analysis of digital images, a subproblem often arises of detecting simple shapes, such as straight lines, circles or ellipses. In many cases an edge detector can be used as a pre-processing stage to obtain image points or image pixels that are on the desired curve in the image space. Due to imperfections in either the image data or the edge detector, however, there may be missing points or pixels on the desired curves as well as spatial deviations between the ideal line/circle/ellipse and the noisy edge points as they are obtained from the edge detector. For these reasons, it is often non-trivial to group the extracted edge features to an appropriate set of lines, circles or ellipses. The purpose of the Hough transform is to address this problem by making it possible to perform groupings of edge points into object candidates by performing an explicit voting procedure over a set of parameterized image objects (Shapiro and Stockman, 304).

The simplest case of Hough transform is detecting straight lines. In general, the straight line y = mx + b can be represented as a point (b, m) in the parameter space. However, vertical lines pose a problem. They would give rise to unbounded values of the slope parameter m. Thus, for computational reasons, Duda and Hart proposed the use of the Hesse normal form

$r = x\cos\ theta + y\sin\ thetar = x\cos\ theta + y\sin\ theta, where r is the distance from the origin$

R theta line.GIF It is therefore possible to associate with each line of the image a pair $(r,\ theta)(r,\ theta)$. $The(r,\ theta)(r,\ theta)$ plane is sometimes referred to as Hough space for the $dimensional Radon transform. In fact, the Hough transform is mathematically equivalent to the Rad$

Given a single point in the plane, then the set of all straight lines going through that point corresponds to a sinusoidal curve in the (r,) plane, which is unique to that point. A set of two or more points that form a straight line will produce sinusoids which cross at the (r,) for that line. Thus, the problem of detecting collinear points can be converted to the problem of finding concurrent curves.

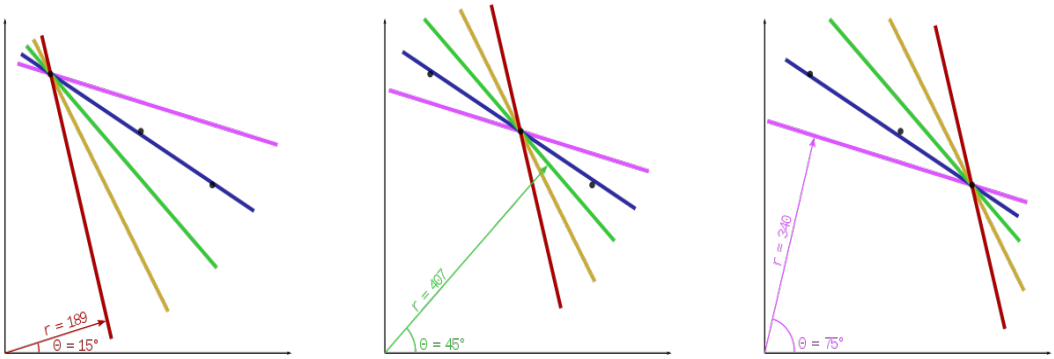
5.2 Implementation

The linear Hough transform algorithm uses a two-dimensional array, called an accumulator, to detect the existence of a line described by

$r = x\cos\ theta + y\sin\ thetar = x\cos\ theta + y\sin\ theta. The dimension of the accumulator equals$

trivialtofindtheappropriatepeaks, andthustheappropriatelines.

The final result of the linear Hough transform is a two-dimensional array (matrix) similar to the accumulator—one dimension of this matrix is the quantized angle and the other dimension is the quantized distance r. Each element of the matrix has a value equal to the sum of the points or pixels that are positioned on the line represented by quantized parameters (r,). So the element with the highest value indicates the straight line that is most represented in the input image.



θ	r
15	189.0
30	282.0
45	355.7
60	407.3
75	429.4

θ	r
15	318.5
30	376.8
45	407.3
60	409.8
75	385.3

θ	r
15	419.0
30	443.6
45	438.4
60	402.9
75	340.1

Figure 5.1: output of algorithm

5.2.1 Hough transform in image processing

The NSL-KDD dataset [21], [13] generated in 2009 is widely used in intrusion detection experiments. In the latest liter- ature [23]–[25], all the researchers use the NSL-KDD as the benchmark dataset, which not only effectively solves the inherent redundant records problems of the KDD Cup 1999 dataset but also makes the number of records reasonable in the training set and testing set, in such a way that the classifier does not favour more frequent records. The dataset covers the KDDTrain + dataset as the training set and KDDTest + and KDDTest 21 datasets as the testing set, which has different normal records and four different types of attack records, as shown in Table 1. The KDDTest 21 dataset is a subset of the KDDTest + and is more difficult for classification.

There are 41 features and 1 class label for every traf- fic record, and the features include basic features (No.1- No.10), content features (No.11 - No.22), and traffic features (No.23 - No.41) as shown in Table 2. According to their characteristics, attacks in the dataset are categorized into four attack types: DoS (Denial of Service attacks), R2L (Root to Local attacks), U2R (User to Root attack), and Probe (Probing attacks). The testing set has some specific attack types that disappear in the training set, which allows it to provide a more realistic theoretical basis for intrusion detection.

Table 5.1: Different classifications in the NSL-KDD dataset

	Total	Normal	DOS	Probe	R2L	U2R
$KDDTrain^{+}$	125073	67343	45927	11656	995	52
$KDDTest^{+}$	22544	9711	7458	2421	2754	200
$KDDTest^{-21}$	11850	2152	4342	2402	2754	200

5.2.2 DATA PREPROCESSING

NUMERICALIZATION

There are 38 numeric features and 3 nonnumeric fea- tures in the NSL-KDD dataset. Because the input value of RNN-IDS should be a numeric matrix, we must convert some nonnumeric features, such as ‘protocol_type’, ‘service’ and ‘flag’ features, into

Table 5.2: Features of NSL-KDD dataset

No.	Features	Types	No.	Features	Types
1	duration	Continuous	22	is_guest_login	Symbolic
2	protocol_type	Symbolic	23	count	Continuous
3	service	Symbolic	24	srv_count	Continuous
4	flag	Symbolic	25	serror_rate	Continuous
5	src_bytes	Continuous	26	srv_serror_rate	Continuous
6	dst_bytes	Continuous	27	rerror_rate	Continuous
7	land	Symbolic	28	srv_rerror_rate	Continuous
8	wrong_fragment	Continuous	29	same_srv_rate	Continuous
9	urgent	Continuous	30	diff_srv_rate	Continuous
10	hot	Continuous	31	srv_diff_host_rate	Continuous
11	num_failed_logins	Continuous	32	dst_host_count	Continuous
12	logged_in	Symbolic	33	dst_host_srv_count	Continuous
13	num_compromised	Continuous	34	dst_host_same_srv_rate	Continuous
14	root_shell	Continuous	35	dst_host_diff_srv_rate	Continuous
15	su_attempted	Continuous	36	dst_host_same_src_port_rate	Continuous
16	num_root	Continuous	37	dst_host_srv_diff_host_rate	Continuous
17	num_file_creations	Continuous	38	dst_host_serror_rate	Continuous
18	num_shells	Continuous	39	dst_host_srv_serror_rate	Continuous
19	num_access_files	Continuous	40	dst_host_rerror_rate	Continuous
20	num_outbound_cmds	Continuous	41	dst_host_srv_rerror_rate	Continuous
21	is_host_login	Symbolic			

numeric form. For example, the feature 'protocol_type' has three types of attributes, 'tcp', 'udp', and 'icmp', and its numeric values are encoded as binary vectors (1,0,0), (0,1,0) and (0,0,1). Similarly, the feature 'service' has 70 types of attributes, and the feature 'flag' has 11 types of attributes. Continuing in this way, 41- dimensional features map into 122-dimensional features after transformation.

NORMALIZATION

First, according to some features, suchas 'duration[0,58329]', 'src_bytes[0,1.3 109]' and 'dst_bytes[0,1.3 109]', where the difference between the maximum and minimum values has a very large scope, we apply the logarithmic scaling method for scaling to obtain the ranges of 'dura- tion[0,4.77]', 'src_bytes[0,9.11]' and 'dst_bytes[0,9.11]'. Second, the value of every feature is mapped to the [0,1] range linearly according to (1), where Max denotes the maximum value and Min denotes minimum value for each feature.

$$x_i = \frac{x_i - Min}{Max - Min}$$

5.2.3 METHODOLOGY

It is obvious that the training of the RNN-IDS model consists of two parts - Forward Propagation and Back Propagation. Forward Propagation is responsible for calculating the out- put values, and Back Propagation is responsible for passing the residuals that were accumulated to update the weights, which is not fundamentally different from the normal neural network training.

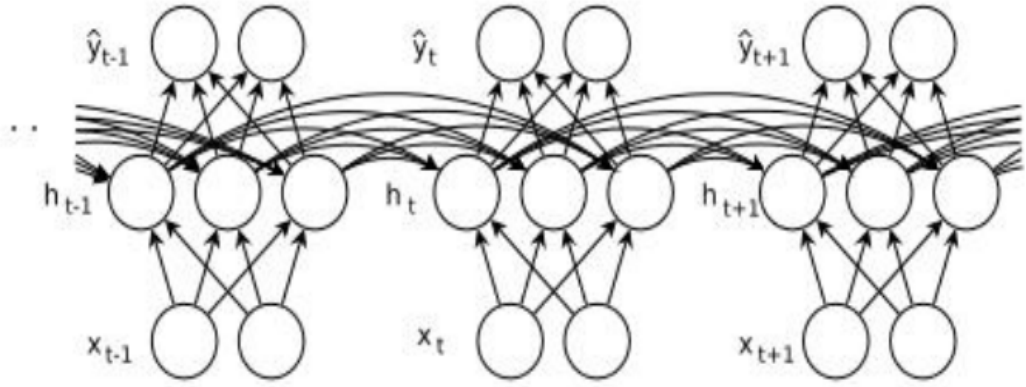


Figure 5.2: The unfolded Recurrent Neural Network.

An unfolded recurrent neural network is presented. The standard RNN is formalized as fol lows: Given training samples x_i (i = 1, 2, . . ., m), a sequence of hidden states h_i(i = 1, 2, . . ., m), and a sequence of predictions $\hat{y}_i$ (i = 1, 2, . . ., m). W_{hx} input-to-hidden weight matrix, W_{hh} is the hidden-to-hidden weight matrix, W yh is

the hidden-to-output weight matrix, and the vectors b_h and b_y are the biases [13]. The activation function e is a sigmoid, and the classification function g engages the SoftMax function. Forward Propagation Algorithm and Weights Update Algorithm are described as Algorithms 1 and 2 respectively. The objective function associated with RNNs for a single training pair (x_i , y_i) is defined as f(θ) =L(y_i : $\hat{y}_i$) [26], where L is a distance function which measures the deviation of the predictions $\hat{y}_i$ from the actual labels y_i . Let be the learning rate and k be the number of current iterations. Given a sequence of labels y_i (i = 1, 2, . . ., m).

Algorithm 1 Forward Propagation Algorithm

Require: $x_i(i = 1, 2, ..., m)$ OUTPUT $\hat{y}_i$

1: for i from 1 to m do
2: $t_l = W_{hxxi} + W_{hhhi} - 1 + bh$
3: $h_i = sigmoid(t_i)$
4: $s_i = W_{yhhi} + b_y$
5: $\hat{y}_i = SoftMax(s_i)$
6: end for

5.3 EVALUATION METRICS

In our model, the most important performance indica- tor (Accuracy, AC) of intrusion detection is used to measure the performance of the RNN-IDS model. In addition to the accuracy, we introduce the detection rate and false positive rate. The True Positive (TP) is equivalent to those correctly rejected, and it denotes the number of anomaly records that are identified as anomaly. The False Positive (FP) is the equivalent of incorrectly rejected, and it denotes the number of normal records that are identified as anomaly. The True Negative (TN) is equivalent to those correctly admitted, and it denotes the number of normal records that are identified as normal. The False Negative (FN) is equivalent to those incor- rectly admitted, and it denotes the number of anomaly records that are identified as normal. Table 3 shows the definition of confusion matrix. We have the following notation:

Accuracy: the percentage of the number of records classi- fied correctly versus total the records shown in (2).

$$AC = \frac{TP + TN}{TP + TN + FP + FN}$$

True Positive Rate (TPR): as the equivalent of the Detec- tion Rate (DR), it shows the percentage of the number of records identified correctly over the total number of anomaly records, as shown in (3).

$$TPR = \frac{TP}{TP + FN}$$

False Positive Rate (FPR): the percentage of the number of records rejected incorrectly is divided by the total number of normal records.

$$FPR = \frac{FP}{FP + TN}$$

Precision: It estimates the ratio of the correctly iden- tified attack connection records to the number of all identified attack connection records. If the Preci- sion is higher, the machine learning model is better (Precision [0, 1]).

$$Precision = \frac{TP}{TP + FP}$$

Chapter 6

EVALUATION RESULT

The clustering on the hotspot datasets using region growing algorithm and the removal of some false alarms in derived hotspots using pattern recognition algorithm are demonstrated. An interface is designed to integrate all functions into a useful application. This piece of work can be enhanced to recognize other regular patterns such as rectangle and triangle so as to remove more false alarms in hotspots since fires usually do not spread in such regular patterns.

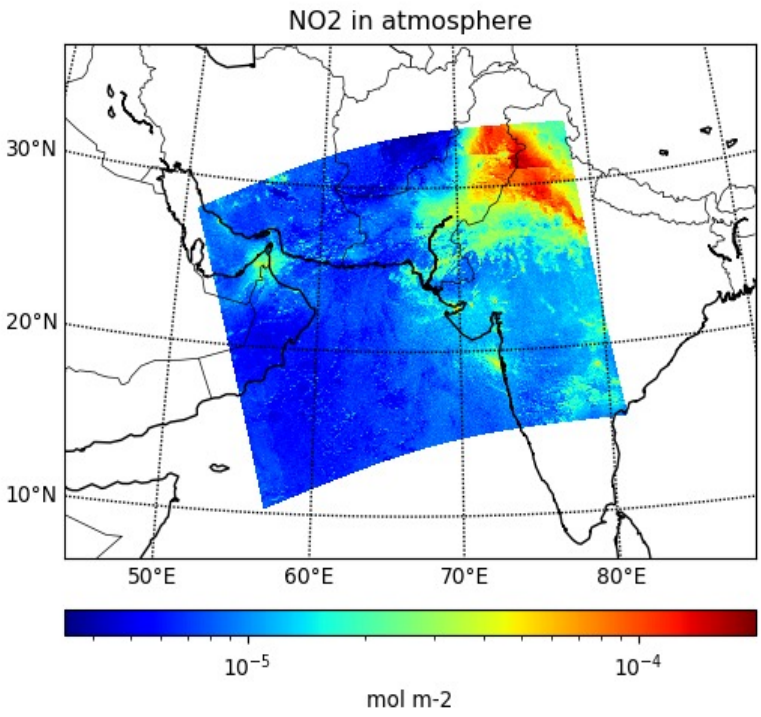


Figure 6.1: output image 1

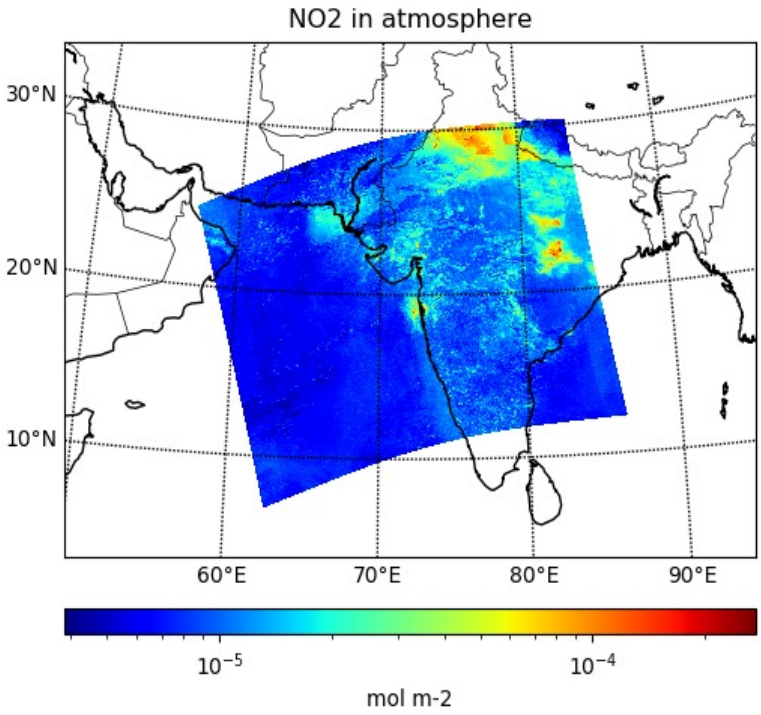


Figure 6.2: output image 1

6.1 Regular pattern visualization and detection

This procedure will be performed when the clustering procedure is completed. The detection for hotspots on straight line is performed based on per-cluster basis. The user can click on the Start Detection button for the program to activate the Hough Transform to detect any straight line of hotspots.

The visualization function buttons provide users with a few options such as to display all the hotspots, to display only the false alarms, to display only the true hotspots, or to display with or without a background map. A snapshot of the visualization for the false alarms is illustrated in Fig. 4. Useful results or information from the pattern detection, such as numbers of detected false alarms and true hotspots

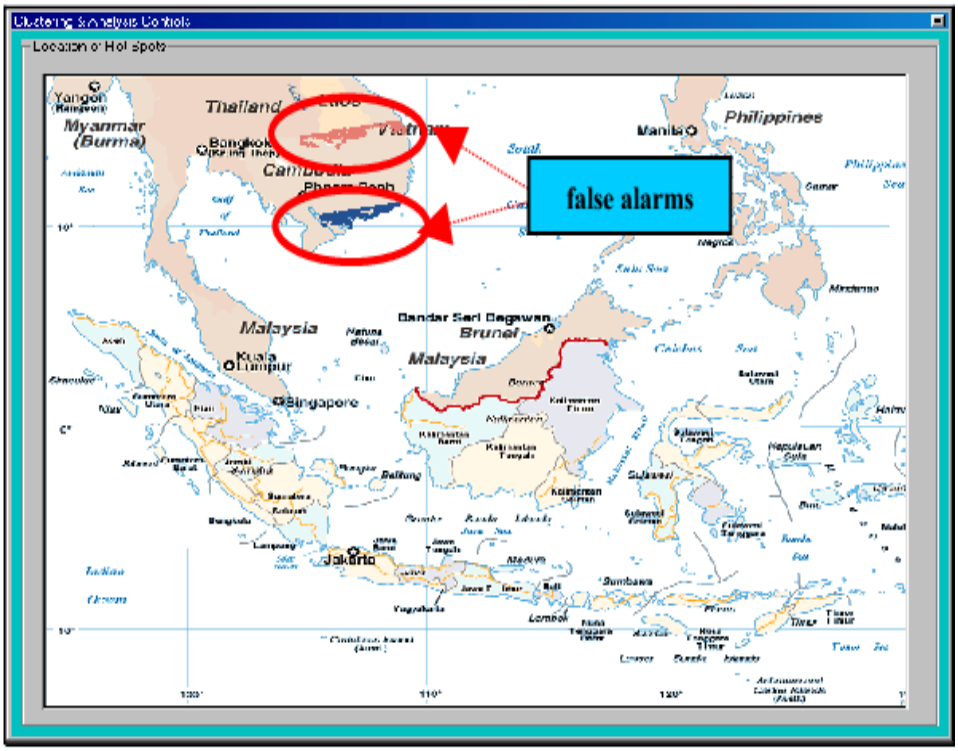


Figure 6.3: output image 1

Chapter 7

CONCLUSION

The clustering on the hotspot datasets using region growing algorithm and the removal of some false alarms in derived hotspots using pattern recognition algorithm are demonstrated. An interface is designed to integrate all functions into a useful application. This piece of work can be enhanced to recognize other regular patterns such as rectangle and triangle so as to remove more false alarms in hotspots since fires usually do not spread in such regular patterns.

GOME and SCIAMACHY measurements of tropospheric NO2 concentration during the 1996 to 2006 period have been used to identify major emission regions, trends and seasonal cycles over India. The geographical distribution of the emission hot spots are observed at the location of large point sources (thermal power plants, large urban and industrial regions). High NO2 concentrations have been observed over the densely populated region (e.g., IG region). High NO2 concentration has also been observed in the industrial regions such as Mumbai-Gujarat industrial corridor (region 1), the Delhi region (region 2) and the east and northeastern India coal mine regions (region 3). The satellite observations clearly detect high NO2 column amount over the individual thermal power plants having large installed capacity. The regional distribution of NO2 concentration observed over India in the satellite measurements is quite similar to the distribution of NOX sources from bottom-up inventories [e.g., Garg et al., 2001; Beig and Brasseur, 2006] and shows a close correspondence with fossil fuel combustion patterns of India.

Chapter 8

FUTURE ENHANCEMENT

Till today, our indian government not have any system to detect the hot spots of their locations. so after some enhacement this module can be used in government organisation.Taking the clustering results into consideration, local Moran's I considered larger areas for each cluster. To compare the detected clusters of two methods meaningfully, the PCI indicator was introduced. PCI compares the spatial pattern homogeneity considering hotspots and cold spots. This index showed that the Getis-Ord statistic reports more polluted air conditions than Moran's I. Clusters, therefore may not be considered as pollution resources, but there are several factors involved in this issue that should be investigated in detail in future works. System is useful for gov organisation to detect hotspots. After some improvements this system can be used in ISRO time.

Bibliography

[1] Aditya C R, Chandana R Deshmukh, Pravin Gandhi, Nayana D K Detection and prediction of air pollution using machine learning International Journal of engineering trends and technology (May 2018)

[2] Dixian Zhu, Changjie Cia, Tianbao Yang, Xun Zhou A machine Learning approach for air quality prediction Article in Big Data and Cognitive Computing

[3] Roya Habibi, Ali Asghar, mohammad sharif A spatial Pattern characterization of air pollution International Journal of geo-Information

[4] Vijayakumar Sajjan, Pramod Sharma Research on an Iot Based Air Pollution Monitoring System International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8, Issue- 9S2, July 2019

[5] Prashant Kumar a,b,, Lidia Morawska c, Claudio Martani d, George Biskos e,f,g, Marina Neophytouh, Silvana Di Sabatino i, Margaret Bell j, Leslie Norfordk, Rex Britter The rise of low-cost sensing for managing air pollution in cities Environment Internationaljournal homepage: www.el sevier .com/locate/envint

[6] R. Udaya Bharathi, M.Seshashayee Weather and Air Pollution real-time Monitoring System using Internet of Things International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-9, July 2019

[7] Sarun Duangsuwan, Aekarong Takarn, and Punyawi Jamjareegulgarn Research on an Iot Based Air Pollution Monitoring System The 18th International Symposium on Communications and Information Technologies (ISCIT 2018)

[8] Kennedy Okokpujie, Etinosa Noma-Osaghae, Odusami Modupe, Samuel John and Oluga Oluwatosin A SMART AIR POLLUTION MONITORING SYSTEM

International Journal of Civil Engineering and Technology (IJCIET) Volume 9, Issue 9, September 2018

[9] G. Karatas and O. K. Sahingoz, "Neural network based intrusion detection systems with different training functions," in Digital Forensic and Security (ISDFS), 2018 6th International Symposium on. IEEE, 2018, pp. 1–6.

[10] Centre for Remote Imaging, Sensing and Processing, "Space Views of Asia", CD-ROM, Second Edition, 2001.

[11] A. Ung, L. Wald, T. Ranchin, C. Weber, J. Hirsch, G. Perron, J. Kleinpeter Satellite data for air pollution mapping over a city – Virtual stations Author manuscript, published in "EARSeL Symposium 2001 "Observing our environment from space: new solutions for a new millenium", Paris : France (2001)"

[12] David Marshall, "Region Growing"-*http* : *//www.cs.cf.ac.uk/Dave/Vision*ı*ecture/node*35.*htn*

[13] Hans-Peter Kriegel, Jorg Sander, Martin Ester and Stefan Gundlach, "Database Primitives for Spatial Data Mining".

[14] Jan Kucera and Yoshifumi Yasuoka, "Regional Monitoring of Forest Disturbances and their Potential Effects to Carbon Cycling".

[15] Liew SC, Lim OK, Kwok LK, Lim H, "A Study of the 1997 Forest fires in South East Asia Using SPOT Quicklook Mosaics", Proc. 1998 International Geoscience and Remote Sensing Symposium, Vol 2, pp. 879-881.

[16] Robert J Fisher and William J Jackson, "Forest Fires in Rural Communities".

[17] Shen CM, Liew SC, Kwok LK, "Spatial and Temporal Pattern of Forest or Plantation Fires in Riau, Sumatra from 1998 to 2000", Proc. ACRS 2001 - 22nd Asian Conference on Remote Sensing, 5-9 November 2001, Singapore. Vol. 1, pp. 520-525.

[18] Akimoto, H. (2003), Global air quality and pollution, Science, 302, 1716– 1719, doi:10.1126/science.1092666.

[19] Badhwar, N., R. C. Trivedi, and B. Sengupta (2006), Air quality status and trends in India, paper presented at Better Air Quality Workshop, Clean Air Initiative for Asian Cities, Yogyakarta, Indonesia, 13– 15 Dec.

[20] Beig, G., and K. Ali (2006), Behavior of boundary layer ozone and its precursors over a great alluvial plain of the world: Indo-Gangetic Plains, Geophys. Res. Lett., 33, L24813, doi:10.1029/2006GL028352.

[21] Beirle, S., U. Platt, M. Wenig, and T. Wagner (2003), Weekly cycle of NO2 by GOME measurements: A signature of anthropogenic sources, Atmos. Chem. Phys., 3, 2225–2232.