DEPARTMENT OF COMPUTER SCIENCE

MINOR PROJECT

DISASTER DETECTION

GROUP MEMBERS:

1) **Sankalp Agrawal 18103022 B1**
2) **Shyam Agarwal 18103025 B1**
3) **Ria Ahlawat 18803004 B13**

SUPERVISOR : Dr. Indu Chawla

Date : 08-12-2020

Table of Contents

	Page No.
Acknowledgement……	**3**
Introduction……	**4**
Current State of Art	**4**
Approach	**4**
Objectives……	**5**
Literature Survey……...	**6**
Methodology……...	**12**
CNN(Convolutional Neural Network)	**12**
Pseudocode	**15**
Results……	**17**
Conclusion……..	**19**
Future Scope……..	**19**
References……..	**20**

Acknowledgement

The satisfaction that accompanies the successful completion of this project would be incomplete without the mention of people who made it possible, without whose constant guidance and encouragement would have made efforts go in vain. We would sincerely like to express our gratitude and respect towards all those who guided us through the completion of this project.

We would like to thank our project supervisor **Dr. Indu Chawla, Computer Science and Engineering Department, Jaypee Institute of Information Technology** for providing encouragement, constant support and guidance which contributed largely in completion of this project.

We also thank the Computer Science and Engineering Department, Jaypee Institute of Information Technology who gave us this opportunity to implement a practical project.

Last but not the least, Our gratitude to the great almighty and our parents without whose concern and devoted support and blessings, the project would not have been the way it is.

1. INTRODUCTION

Current state of Art : Across the globe Natural disasters cause ecological, social, and economic damage. To minimize these losses, early detection of these disasters and an autonomous response are important and helpful to disaster management systems. Therefore, in this article, we propose an early detection framework using fine-tuned convolutional neural networks for CCTV surveillance cameras, which can detect flood, cyclones, earthquakes and wildfires.

All around the world we use sensors to monitor for natural disasters as:

Seismic sensors (seismometers) and vibration sensors (seismoscopes) are used to monitor for earthquakes (and downstream tsunamis).“Hook echo” of tornado is measured by radar maps. Flood sensors are used to measure moisture levels while water level sensors monitor the height of water along a river, stream, etc. Wildfire sensors are still in their infancy but hopefully will be able to detect trace amounts of smoke and fire. Each of these sensors is *highly specialized* to the task at hand — detect a natural disaster early, alert people, and allow them to get to safety.

Using machine learning and computer vision we can *augment* existing sensors, thereby increasing the accuracy of natural disaster detectors, and most importantly, allow people to take precautions, stay safe, and prevent/reduce the number of deaths and injuries that happen due to these disasters.

Approach : By Problem Identification to develop a system to find a solution to detect a Natural Disaster by capturing the images from the CCTV footage and several studies that we can find about the various factors that contribute towards the detection of Natural Disaster. Hence a system can be developed to detect these Natural Calamities (flood, cyclone, earthquake and wildfires).

Our first objective is To build an application that takes data in the form of images captured from the CCTV footage. This image is saved as data in our Sql database model on Xampp Local server.

Next, To detect whether the image is of a natural disaster and if yes, then what kind of natural disaster is it, we found machine learning to be an effective tool to train the data and hence we developed a machine learning model. For this we collected information on various models that can be used for categorical classification of a dataset. Among them we decided to test on a Convolutional Neural Network(CNN) classifier. Finally we wanted to build a Web Application that takes data from a database and prints the result after successful detection by a Machine learning model for ease of user. Hence, the collected data from the database and the machine learning model was imported into web application assets. The final results were then printed on the User's screen.

Next, after the successful detection of the disaster the application sends warning signals to the desired authorities and the news channels along with the image and location of the disaster so that necessary actions can be taken.

2. OBJECTIVES

a) **To build and train a machine learning model to detect a Natural Disaster from the image.**

b) **To build a Web Application that takes data in the form of an image, stores it in the database and displays the result after successful detection by Machine learning model.**

c) **To build an effective warning system to relay the information about the type and location of the disaster to the concerned departments**

3. LITERATURE SURVEY

S. No.	Paper Title	Year and Publication	Objective
1.	Disaster Management in India	February 2012 Indian Research Journal of Extension Education	To Study Disaster Profile of India · and To study Disaster Management in India
2.	Understanding of a Convolutional Neural Network	Aug. 2017 IEEE - 017 International Conference on Engineering and Technology (ICET)	This paper explains and define all the elements and important issues related to CNN, and how these elements work. In addition, it also state the parameters that effect CNN efficiency.
3.	Convolutional Neural Network Based Flood Detection Using Remote sensing images	2019, EasyChair preprints	The proposed system uses pre-existing CNN, namely Alex Net, for mapping flooding regions using remote sensing images and the application of high-level spatial features for classification of satellite imagery.

4.	Wireless sensor network for AI-based flood disaster detection	2020, Annals of Operations Research	This study proposes a wireless sensor network decision model for the detection of flood disasters by observing changes in weather conditions compared to historical information at a given location from data such as air pressure, wind speed, water level, temperature and humidity and precipitation from sensors located at several points in the area under consideration and obtained sea level air pressure and rainfall from the Google API. The developed support vector machine (SVM) model includes a number of coordinators responsible for a number of sectors (locations). The SVM model sends the binary decisions (flood or no flood) with an accuracy of 98%.
5.	Smart Disaster Detection and Response System for Smart Cities	2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)	A review of the state-of-the-art of solutions for disaster discovery and response, The design of an innovative smart system for disaster prediction, discovery and response for smart cities
6.	Analysis of satellite images for disaster detection	IGARSS 2016 IEEE International Geo science and Remote Sensing Symposium	The study proposes an automatic disaster detection system by implementing , convolutional neural

			network (CNN), to analysis satellite images.
7.	Disaster detection system using Arduino	2008, International Conference on Information Communication and Embedded Systems	The author aims to reduce the number of disasters drastically in order to come up with a safer and secure environment. This paper describes a system which detects the possible disasters that one can face in a household or work-space. It is an Arduino based Disaster Detection System that contains sensors for detecting the disasters. This system is new in the sense that it incorporates detection of more than one disaster with one device and still proves to be as cheap as possible. It is also unique in the sense that it automatically informs the emergency services when a disaster is detected.

8.	IoT Based Disaster Detection and Early Warning Device	2017, International Journal of MC Square Scientific Research Vol.9	.Create a smart IoT device that acts as an alarm and monitoring system during natural disasters that operates by communicating over internet.. The ultimate aim of the project is to spread the disaster warning information quickly through internet and make it available to those who need it as early as possible. The fact that internet is faster than the seismic waves of an earthquake, and much faster than a flood or tsunami, helps the device to deliver the alert message much before the actual calamity reach the user's location giving that vital extra time to take those precautionary emergency measures.
9.	Early fire detection using convolutional neural networks	2017, Neurocomputing Volume 288, Pages 30-42	The early fire detection framework made using fine-tuned convolutional neural networks for CCTV surveillance cameras, which can detect fire in varying indoor and outdoor environments. Finally, a dynamic channel selection algorithm for cameras based on cognitive radio networks, ensuring reliable data dissemination .

10.	An integrated approach to natural disaster management	May 2006 Disaster Prevention and Management	First it provides a framework for effective natural disaster management from a public project management perspective. Second, it proposes an integrated approach for successfully and effectively managing disaster crisis. Third, it specifies a set of critical success factors for managing disaster related public projects..
11.	Optimized CNN Based Image Recognition	March 2018 OptikVolume 156,Pages 772-777	Traditional CNN model don't consider the weights of learning instances which reduces the accuracy of image recognition. Aiming at the problems above,this paper proposed one optimized CNN based image recognition model. Firstly, target region selected by bottom-up region proposals contributes to retrieve the target region of each learning instance. Secondly, enhancement weight based model is used to optimize the CNN model contributing to make full use of different learning instances. At last, adequate experiments show our method's superiority, especially compared to some other traditional methods.

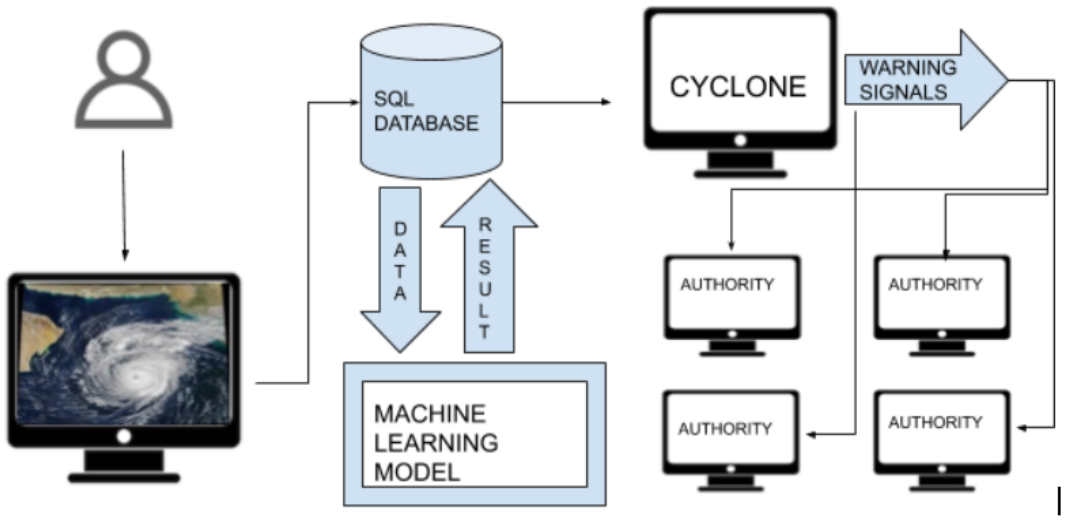
12.	A Real Time Video Processing Based Surveillance System for Early Fire and Flood Detection	June 2007, IEEE Instrumentation and Measurement Technology Conference	This paper presents a simple and effective method for automatic disaster detection in the early stage by real time video analysis. Via analyzing the specific feature vectors including color/grayscale histogram concentration, a fully automatic process is developed to substantially improve the performance, especially in early detection thus to reduce the loss caused by natural disaster, of the existing security surveillance systems

4. METHODOLOGY

We Find the dataset of various disasters (earthquake, flood, wildfire, cyclone) which were split into test and train dataset. Next, To classify the natural disaster we found machine learning to be an effective tool to train the data and hence we developed an machine learning model. For this we collected information on various models that can be used for image classification of a dataset among them we decided to test on CNN classifier.

Next we built a web application which takes the data input in the form of an image and stores in the database which is taken by our machine learning model. The result is printed about the type of disaster on the screen after successful prediction by the model.

Finally , after the disaster is successfully detected the information about the type and location of the disaster is relayed to the concerned authorities so that the necessary actions could be taken.

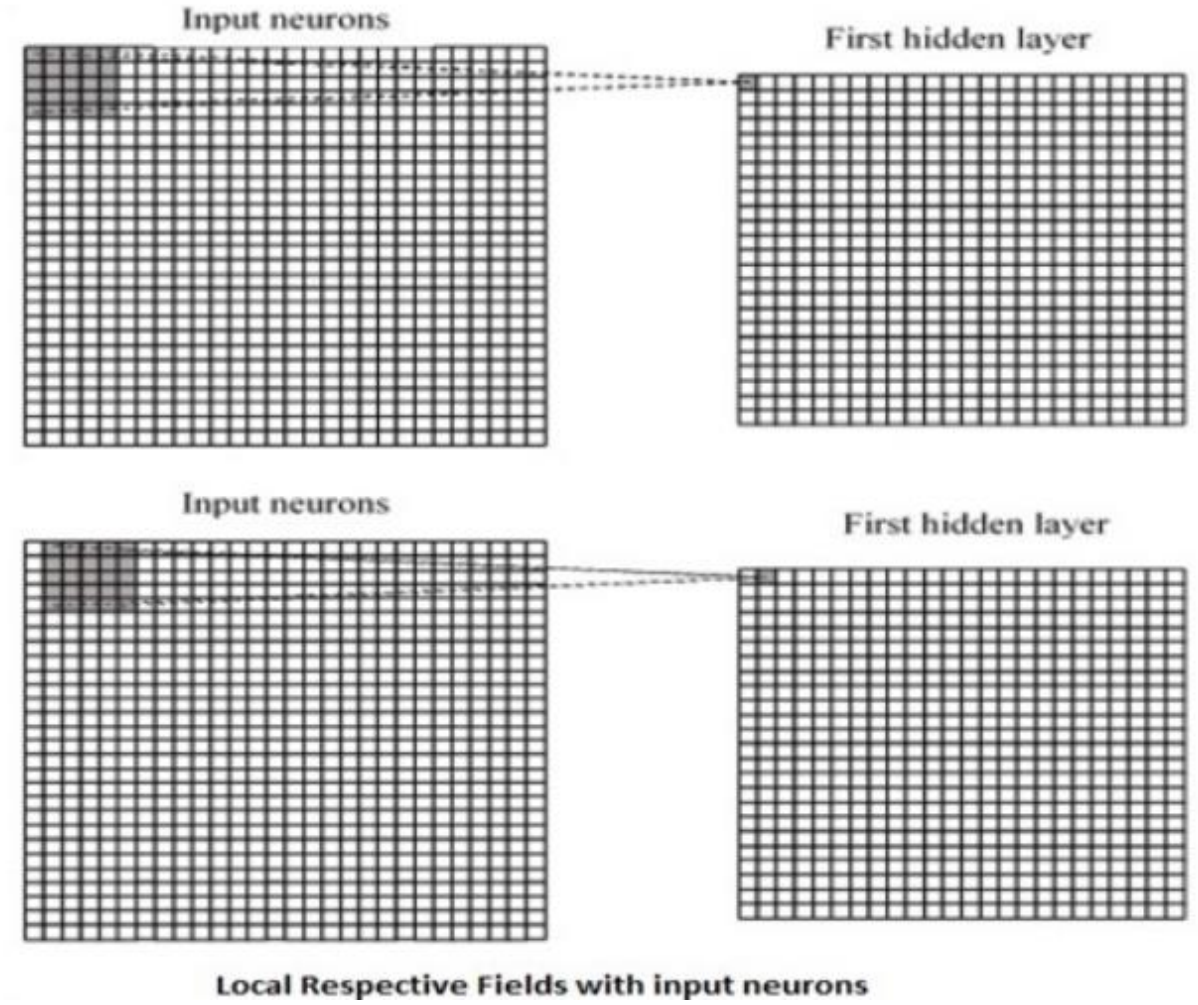


OVERVIEW OF THE PROJECT

CNN(Convolution Neural Network): Convolutional Neural networks are used in image or face recognition. It is designed to process data through multiple layers of arrays. CNN does not extract the feature of the image just like ordinary neural networks rather it takes the image as a two dimensional array and operates on them directly. The dominant approach of CNN includes solutions for problems of recognition. Many multinational companies have invested towards the development and research of various recognition techniques with CNN being one of them.

A convolutional neural network uses three basic ideas −

- Local respective fields
- Convolution
- Pooling
- Flattening

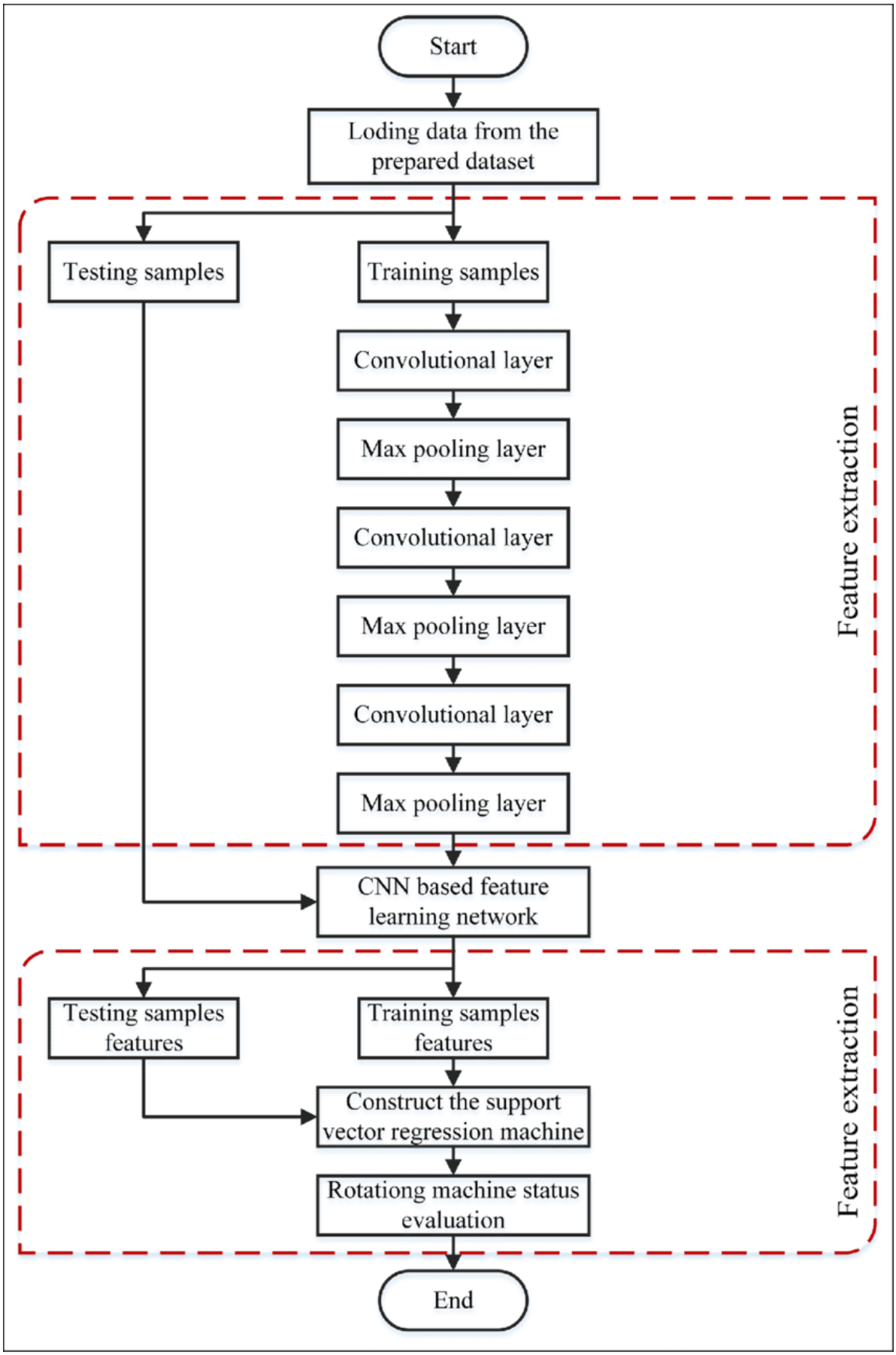


Local Respective Fields with input neurons

If we observe the above representation, each connection learns a weight of the hidden neuron with an associated connection with movement from one layer to another. Here, individual neurons perform a shift from time to time. This process is called “convolution”.

The mapping of connections from the input layer to the hidden feature map is defined as “shared weights” and bias included is called “shared bias”.

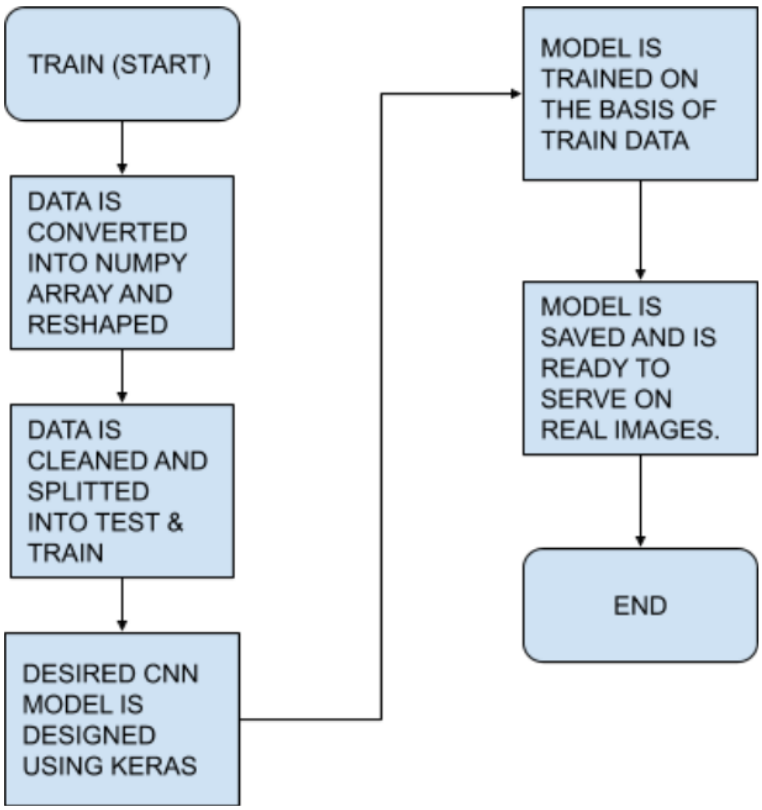
CNN or convolutional neural networks use pooling layers, which are the layers, positioned immediately after CNN declaration. It takes the input from the user as a feature map that comes out of convolutional networks and prepares a condensed feature map. Pooling layers helps in creating layers with neurons of previous layers.


Start
Loding data from the prepared dataset
Testing samples
Training samples
Convolutional layer
Max pooling layer
Convolutional layer
Max pooling layer
Convolutional layer
Max pooling layer
Feature extraction
CNN based feature learning network
Testing samples features
Training samples features
Construct the support vector regression machine
Rotationg machine status evaluation
Feature extraction
End

Pseudocodes –

1. CNN-

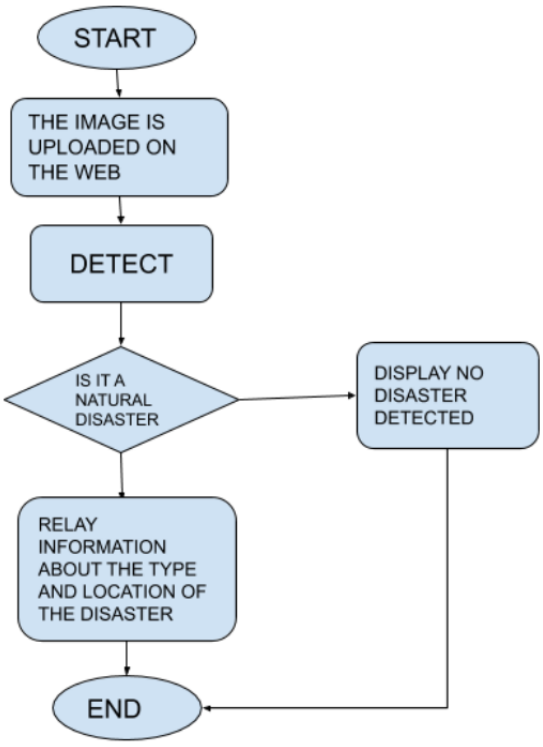
```
classify = Sequential()
classify.add(Convolution2D(256, 3, 3, input_shape = (224, 224, 3), activation = 'relu'))
classify.add(MaxPooling2D(pool_size = (2, 2)))
classify.add(Flatten())
classify.add(Dense(units = 512, activation = 'relu'))
classify.add(Dense(units = 4, activation = 'softmax'))
classify.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = ['accuracy'])
from keras.preprocessing.image import ImageDataGenerator
train_data = ImageDataGenerator(rescale=1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)
test_data = ImageDataGenerator(rescale=1./255)
training_set = train_data.flow_from_directory('dataset/train', target_size=(224, 224), batch_size=32,
class_mode='categorical')
test_set = test_data.flow_from_directory('dataset/test', target_size=(224, 224),
                                         batch_size=32,
                                         class_mode='categorical')
classify.fit(training_set,
               steps_per_epoch=3543//32,
               epochs=30,
               validation_data=test_set,
               validation_steps=885//32)
classify.save('model_saved2')
```



TRAINING THE MODEL FLOW CHART

2. WEB APPLICATION

```
$sql = "SELECT * FROM prediction_probability ORDER BY timestamp DESC LIMIT 1";
if($result = $link ->query($sql))
{
    while($rows = $result->fetch_array(MYSQLI_ASSOC))
    {
        $cyclone = $rows['cyclone'];
        $earthquake = $rows['earthquake'];
        $flood = $rows['flood'];
        $wildfire = $rows['wildfire'];
    }
    $disaster = array("Cyclone"=>$cyclone,"Earthquake"=>$earthquake,"Flood"=>$flood, "Wildfire"=>$wildfire)
    $max = max(array_values($disaster));
    $key = array_search($max, $disaster);
    echo $key;
}
$folder_path = "uploads";
$files = glob($folder_path.'/*');
foreach($files as $file) {
    if(is_file($file))
        unlink($file);
}
$target_dir = "uploads/";
$target_file = $target_dir . basename($_FILES["fileToUpload"]["name"]);
$uploadOk = 1;
$imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));
if(isset($_POST["submit"])) {
  $check = getimagesize($_FILES["fileToUpload"]["tmp_name"]);
  if($check !== false) {
    echo "File is an image - " . $check["mime"] . ".";
    $uploadOk = 1;
  } else {
    echo "File is not an image.";
    $uploadOk = 0;
  }
```



WORKING OF WEB FLOW CHART

5. RESULTS

a) PC Specifications:

OS Used : Windows 10 (64 bits)
Processor : Intel(R) Core(TM) i5-8250 CPU@1.60GHz 1.80GHz
Ram : 8.00 GB

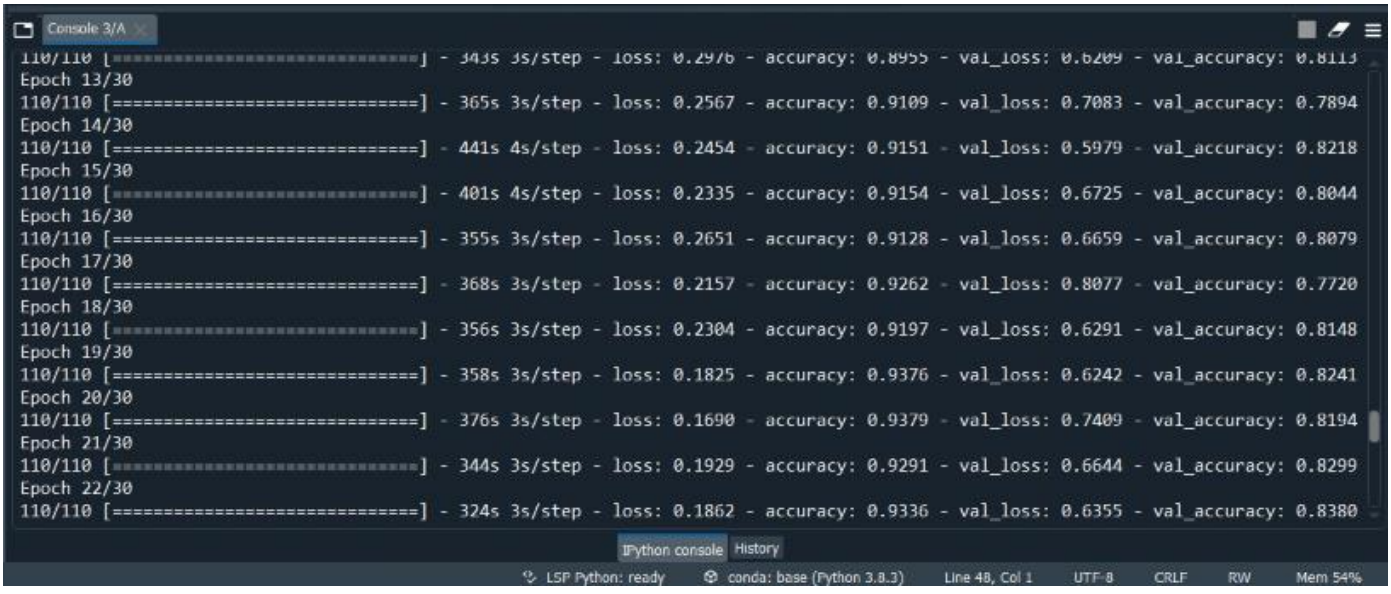
b) Softwares Used:

1) VISUAL STUDIO CODE : Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git!
2) XAMPP Control Panel : XAMPP stands for cross-platform, Apache, MySQL, PHP and Perl, . This open source software provides a local web server on one's computer.
3) SPYDER : Spyder is an open source cross-platform integrated development environment for scientific programming in the Python language.

c) Languages Used:

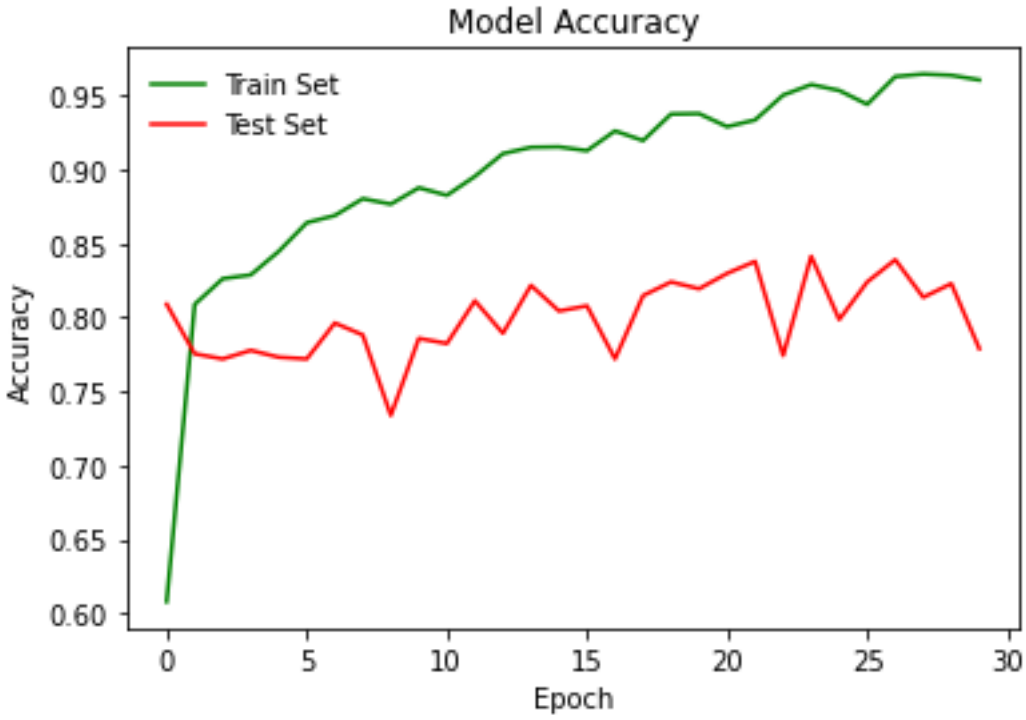
1) Python :Used for training the model and sending warning signals.
2) PHP : Connecting to the database and fetching the data.
3) MySQL : Used for maintaining the database..
4) JavaScript : Used for checking validations while uploading, fetching and displaying the results.
5) HTML/CSS : Used for creating layout, interface and designing the web application.

Output : CNN Model after run



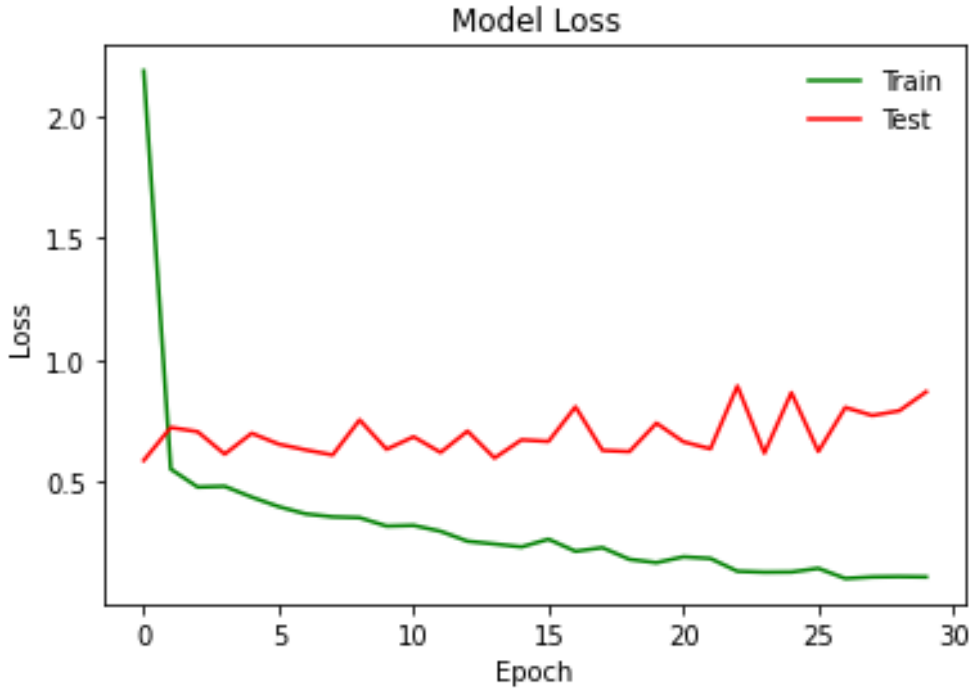
Accuracy Factor

The CNN model gave an accuracy of – 85%(approx.) for the images which were tested and the accuracy of – 95%(approx.) for the images which were used for training the CNN model.



Loss Factor

The CNN Model clearly shows that the loss factor for the training set in increasing for increase in the number of epochs whereas the loss factor for the test set in almost constant.



6. CONCLUSION

By studying the research paper focused around disaster detection and management and the studies related to the significance of early detection of the natural disaster, we developed a system which could detect the natural disaster at the earliest using the machine learning model and send warning signals to the concerned authorities in order to control the consequences. We used the web based approach to create the interface for the system.

7. FUTURE SCOPE

Although the system developed using Machine learning model and web programming techniques in order to detect the natural disaster with remarkable accuracy, there are still some sectors which can be worked upon and improved

1.The Web Application can be improved by using web scraping and mining techniques. The details about the concerned authorities of the particular location can be figured out automatically in order to send the warning signals.

2.The hardware support can be used along with the machine learning model, like the alarm is raised in a particular location as soon as the system detects any disaster.

3.Data from the seismometers, flood sensors, wildfire sensors and radar maps can be taken as an input for the model. This way we can predict the disaster even before the occurrence and take the required preventive measures.

8. REFERENCES

[1] C.L. LaiYang Jie-Ci Y.H. Chen ”A Real Time Video Processing Based Surveillance System for Early Fire and Flood Detection”

[2] Pairote Pathranarakul, Dean Kyne ”An integrated approach to natural disaster management”

[3] Hao, Wua,Rongfang Biea, Junqi Guoa, Xin Meng ,Shenling Wanga”Optimized CNN Based Image Recognition”

[4] http://www.deeplearningbook.org/contents/convnets.html

[5] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” pp. 1–28, 2016.

[6] Y. Guo, Y. Liu, A. Oerlemans, S. Wu, and M. S. Lew, “Author ’ s Accepted Manuscript Deep learning for visual understanding◌ϭπ: A review To appear in◌ϭπ: Neurocomputing,” 2015.

[7] https://www.opendatascience.com/blog/an-intui
(PDF) Understanding of a Convolutional Neural Network. Available from: https://www.researchgate.net/publication/319253577_Understanding_of_a_Convolutional_Neural_Network

[8] Keigo Koizumi, Kenji Hirata, Kazuhiro Oda, Yukishige Fujita, Sadayuki Kamide, Tohru Watanabe “Disaster Detection System Using Sensor Networks and Its Field Experiment Evaluations,” SICE Journal of Control, Measurement, and System Integration, Vol.5, No.1, pp. 041-047, January 2012

[9] Urabe K. Saji H., “Detection of Road Blockage Areas after Earthquake Disaster in Mountainous Districts Using Aerial Images” Journal of Japan Association for Earthquake Engineering, Vol.9, No.4, 2009

THANK YOU