

OBJECT DETECTION IN AN IMAGE UNDER RAINY CONDITIONS USING YOLO V7

*A project report submitted in partial fulfilment of the requirements for the
award of the degree of*

**BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING**

Submitted by

K.L.V.Satyanarayana (319126512156)

K.Jyothirmayi (319126512153)

E.Havila (319126512142)

P.Padmaja (318126512095)

Under the guidance of

Dr. B. Jagadeesh

Professor & HOD



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING
ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)**

*(Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC With 'A'
Grade Sangivalasa, Bheemili mandal, Visakhapatnam dist. (A.P)*

2022-2023

DEPARTMENT OF ELECTRONICS AND COMMUNICATION

ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA &


NAAC With 'A' Grade)

Sangivalasa, Bheemili mandal, Visakhapatnam dist. (A.P)

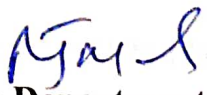


CERTIFICATE

This is to certify that the project report entitled "OBJECT DETECTION IN AN IMAGE UNDER RAINY CONDITIONS USING YOLO V7" submitted by K.L.V.Satyanarayana (319126512156), K.Jyothirmayi (319126512153), E.Havila (319126512142), P.Padmaja (318126512095) in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Electronics & Communication Engineering** of Anil Neerukonda Institute of technology and Sciences (A), Visakhapatnam is a record of bonafide work carried out under my guidance and supervision.


Project Guide
Dr. B. Jagadeesh
Professor & HOD
Department of E.C.E
ANITS

Professor
Department of E.C.E.
Anil Neerukonda
Institute of Technology & Sciences
Sangivalasa, Visakhapatnam-531162


Head of the Department
Dr. B. Jagadeesh
Professor & HOD
Department of E.C.E
ANITS

Head of the Department
Department of E C E
Anil Neerukonda Institute of Technology & Sciences
Sangivalasa - 531 162

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our project guide **Dr. B. Jagadeesh** Department of Electronics and Communication Engineering, ANITS, for his guidance with unsurpassed knowledge and immense encouragement.

We are grateful to **Dr. B. Jagadeesh**, Head of the Department, Electronics and Communication Engineering, for providing us with the required facilities for the completion of the project work.

We are very much thankful to the **Principal and Management, ANITS, Sangivalasa**, for their encouragement and cooperation to carry out this work. We express our thanks to all **teaching faculty** of the Department of ECE, whose suggestions during reviews helped us in accomplishment of our project. We would like to thank all **non-teaching staff** of the Department of ECE, ANITS for providing great assistance in the accomplishment of our project.

We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last, but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

PROJECT STUDENTS

K.L.V.Satyanarayana (319126512156)

K.Jyothirmayi (319126512153)

E.Havila (319126512142)

P.Padmaja (318126512095)

ABSTRACT

This project's main objective is to employ advanced computer vision technologies to spot things in a picture. Intelligent transportation systems substantially rely on object detection. Weather factors that restrict camera performance by limiting visibility, such hazardous conditions include the presence of heavy snow, fog, rain, dust, or sandstorms. To make the best strategic decisions and offer the necessary safety, effective object detection is regarded as a key stage in traffic monitoring or intelligent.

Numerous individual's raindrops with a variety of sizes and intricate forms make up rain, which obstructs the light that is reflected off the objects. It alters the intensity of the images and video frames. This results in low contrast and a significant amount of whitening of the visual data. As a result, visibility is insufficient for effectively detecting objects in the image, resulting in video collisions. The achievement of clear visibility would be aided by the development of efficient image enhancing techniques to provide good visual appearance or discriminative characteristics.

The purpose of the project is to minimize the impact of rain on visual data and to detect the required item. Machine Learning techniques are used in surveillance to eliminate noise. Noise can be efficiently removed using machine learning. Hybrid systems that can predict and apply the optimal combination of detectors and trackers are expected to become more prevalent in the future.

TABLE OF CONTENTS

CERTIFICATE	ii
ACKNOWLEDGEMENT	iii
ABSTRACT	iv
LIST OF FIGURES	viii
1. INTRODUCTION	2
1.1 MOTIVATION	2
1.2 PROBLEM IDENTIFICATION	2
1.3 OBJECTIVE	2
2. REQUIREMENT ANALYSIS	8
2.1 LITERATURE SURVEY	8
2.2 RELATED TECHNOLOGY	10
PYTHON	10
MACHINE LEARNING	10
DEEP LEARNING	11
OPEN CV	11
SCIKIT IMAGE	12
PYTORCH	12
NUMPY	12
GOOGLE COLAB	13
VGG	13
MOBILENETS	13

TENSORFLOW	14
3. DERAISING ALGORITHM	16
3.1 GENERATIVE ADVERSARIAL NETWORK (GAN)	16
3.1.1 INTRODUCTION	16
3.1.2 How GAN's WORK?	17
3.1.3 WHY GAN's?	18
3.2. ATTENTIVE GENERATIVE ADVERSIAL NETWORK	19
3.2.1 INTRODUCTION	19
3.2.2 WHY AGAN?	19
3.3 RAINDROP REMOVAL USING AGAN	21
3.3.1 GENERATIVE NETWORK	22
3.3.2 DISCRIMINATIVE NETWORK	25
3.3.3 DERAINDROP DATASET	26
4. OBJECT DETECTION ALGORITHM	28
4.1 YOLO	28
4.1.1 INTRODUCTION	28
4.2 YOLO V2	29
4.3 YOLO V3	29
4.4 YOLO V4	30
4.5 YOLO V5	30
4.6 YOLO V6	31
4.7 YOLO V7	31
4.7.1 YOLOV7 ARCHITECTURE	32
4.7.2 FUNCTIONALITY	32

4.8 APPLICATIONS OF OBJECT DETECTION	33
5. RESULTS AND CONCLUSION	35
5.1 RESULTS	36
5.2 CONCLUSION	38
6. FUTURE SCOPE	39
REFERENCES	41

LIST OF FIGURES

Fig No	Figure Name	Page no
1	GAN Schema	17
2	AGAN Architecture	21
3	Contextual Autoencoder Architecture	24
4	YOLO V7 Architecture	32
5.1	Input Rainy Image	36
5.2	Output Derained image	36
5.3	Object Detection Output in rain image	37
5.4	Object Detection Output in derain image	37

ACRONYMS

R-CNN	:	Region based Convolutional Neural Network
YOLO	:	You Only look Once
MAP	:	Mean Average Precision
SRTV	:	Slice Rotational Total Variation
WTNN	:	Weighted Tensor Nuclear Norm
TTV	:	Tensor Total Variation
FSLs-MOD	:	Full Spectrum Light Source Modulation
WCC-PM	:	Weather Condition Classification Prediction Model
GPU	:	Graphic processing Unit
GAN	:	Generative Adversarial Network
MSE	:	Mean Squared Error
AGAN	:	Attentive Generative Adversarial Network
Res-Net	:	Residual Network
LSTM	:	Long Short-Term Memory
CNN	:	Convolutional Neural Network

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

1.1 MOTIVATION

To allow connected automobiles to travel safely in their surroundings, high level auto dynamic security frameworks rely heavily on visual information to group and limit objects such as humans, other nearby vehicles, road signs, and lights. However, under adverse weather conditions, such as during storms, the display of article identification algorithms may suffer greatly. Despite great progress in the development of Deraining techniques, little is known about how rain affects object placement, especially when it comes to independent driving. Artificial intelligence (AI) programs employ object classification techniques to identify specific objects in a class that are compelling items.

1.2 PROBLEM DEFINITION

In order to make the most effective decisions and offer adequate safety, object detection is regarded as a vital step in traffic monitoring or intelligent surveillance. Weather may at times influence object detection. The project's goals include lessening the impact of rainfall on visual information and identifying and monitoring the required item.

Raindrops cause the visual data to be extremely whitening and have low contrast. As a result, there are video collisions since visibility is insufficient for recognizing objects in the display with precision. Clear visibility will be made possible by using efficient image-enhancing techniques to create good visual appearance or discriminative features. The primary objective of our project is to find things in humid environments. Though there have been other attempts, here we aim for greater rapidity and accuracy.

1.3 OBJECTIVE

In order to help the corresponding vehicles, move securely in their environments, sophisticated automobile effective-safety systems in overall, and autonomous vehicles, depend heavily on aesthetic data to categorise and localise things like pedestrians, traffic signs and lights, and other related to motorized vehicles. The efficiency of object recognition techniques can still substantially decrease in poor weather, particularly rainy ones. Although Deraining methods have made great progress, the effects of rain on object detection have gotten less attention, particularly when it comes to self-driving automobiles. The key objective of the study is to

propose innovative approaches for reducing the impact of rain on an autonomous vehicle's capacity to detect objects. Our goal is to assess the effectiveness of object detection systems that have been evaluated and developed with imagery collected in both clear and wet weather.

Artificial intelligence and machine learning have experienced rapid development in the current digital era. These methods are utilized to develop machines that imitate human recognizing abilities. Finding what and where (many) items exist in a picture is known as object identification. Object detection is based on fundamental idea of artificial intelligence.

Based on improved object detection efficiency and object detection technology, surveillance has made many discoveries. Artificial intelligence is increasingly being used. The rain, which is comprised of countless raindrops of various sizes and complicated forms, blocks the light that is reflected off the items. Images and video frames will change in intensity as a result. Identifying and discovering every known object in a scene is the objective of object detection.

Real-time searching and identification are immensely difficult tasks. The problem has yet to be addressed in an effective way. The techniques that have been created thus far, despite the extensive study in this field, are ineffective, need a lengthy training period, are unsuitable for real-time use, and cannot scale to many courses.

Object recognition is somewhat simpler when a machine is looking for out something specific. Recognizing all the items, on the other hand, necessitates the ability to distinguish one object from another, regardless of whether they are of the same type. If computers are ignorant of all the object possibilities, this becomes an extremely difficult issue.

1.4 DIGITAL IMAGE PROCESSING

The broad range of computerised creation of images has been shown by a need for thorough evaluation in order to assess the viability of suggested remedies for a specific problem. A crucial aspect concealed in the design of photo processing frameworks is the sizeable amount of testing and experimenting often required to arrive at a satisfactory solution. Because to this feature, it often requires a good deal of preparation and rapid modelling to decrease the time and cost needed to arrive at a viable structure for implementation.

WHAT IS DIP?

A picture can be thought of to possess the two-dimensional capacity $f(x, y)$, where both axes represent the two spatial directions. The picture's power or dark level is thus accepted as the capability off at any pair of directions (x, y) . We define an image as a programmed picture since x, y , and the abundance estimation are all attached discrete amounts. DIP corresponds to digital PC technologies used to create rich graphics. A smart image consists of a small number of elements, each of which has a specific purpose and value. These components are the pixel values.

Since that seeing is our most evolved sense, it is not surprising that images have the greatest impact in direct observation. The distinction between image acquisition and other related topics, such as analysis of images and computer vision, is still disputed among developers. It is sometimes helpful to describe image handling as a teaching technique where the input and output are both images. It is an artificial constraint, which is somewhat constricting. Picture research falls between image processing and computer vision in terms of its application.

There are no obvious borders between the range of picture handling on one side and finished eyesight on the other. In any case, viewing three different categories of automated processes—low, mid, and abnormal state forms—as parts of this spectrum can be helpful. Low-level processes are exemplified by primitive acts like preparing images to reduce noise during improvement and image improvement. Because images act as the process's information inputs and outputs, it is classified as a low-level operation.

Even if majority of its data is obtained from photos, a middle level process may be defined by the fact that outcomes are traits that aren't present in the photographs. Lastly, advanced quantity handling includes "Knowing a source of imagined things, as in picture evaluation, and at the most extreme point of the spectrum acting out the cognitive abilities generally attributed to human vision.

WHAT IS AN IMAGE?

Where x and y are coordinates, an image is characterized by a two-dimensional structure of the format $f(x, y)$. Hence, the appropriate value of "T" at each pair of directions determines the strength of the photo(x, y).

Processing on image:

There are three various kinds of image processing:

low-level, mid-level, and high-level

Low-level Processing:

- Noise elimination via pre-processing.
- Enhancing contrast
- sharpening of image

Middle Level Processing:

- Segmentation
- detection of edge
- extraction of object

High Level Processing:

- interpretation of image
- Scene interpretation

Why Image Processing?

The digitised information must be changed for it to be displayed on one or more output devices since it is unreadable. By enhancing visual appeal of the structures within the digital image, it could be made more appropriate for the application.

Image processing is performed in three modes, Which as

- Image to Image transformation
- Image to Information transformations
- Information to Image transformations

Pixel:

Pixels are tiniest constituent of picture. One value could be indicated by each pixel. In an 8-bit grayscale photo, a pixel's value ranges[0,255]. Each pixel stores an organization value to the quantity of light that is present at that particular spot.

Resolution :

Numerous definitions exist for the resolution. Some examples of resolutions include pixel resolution, resolution of space, temporal resolution, and spectral resolution. Resolution in terms of pixels refers to how many pixels are included in an entire digital image. An illustration. The resolution of an image is described as $M \times N$ if it includes N columns & M rows. The image's visual appeal enhances as you raise the pixel resolution. An image's resolution may be classified as follows:

- Low Resolution image
- High Resolution

Higher resolution is an inefficient use of resources. It is not always possible to get pictures of excellent quality at a reasonable price. Consequently, imaging is desirable. With the use of specific methods and algorithms, we can generate pictures with high resolution using super resolution imaging from low resolution imagery.

CHAPTER 2
REQUIREMENT ANALYSIS

2. REQUIREMENT ANALYSIS

2.1 LITERATURE SURVEY

Hayder Radha et. al [1]: performed a survey and provided an overview of the most inventive and imaginative ways to prevent severe weather conditions from impairing an automated vehicle's ability to distinguish things. They looked into and assessed the efficiency of object detection techniques being explored for integration with autonomous cars.

Using both Faster R-CNN & YOLO, mean Average Precision (mAP) for recognizing objects decreased under wet circumstances compared to clear weather. When rainy circumstances serve as inputs to the detection frameworks, deraining algorithms perform less well in terms of detection. This is valid for YOLO as well as Faster R-CNN. The Deraining method tends to smooth down the input image, reducing the useful data.

George. Set. al [2]:In this article, one structure is developed for simultaneous moving recognition of objects and video deraining. The purpose is to create successful video deraining model that clear the unnecessary rain streaks and rebuild the clean data by combining the missing features from this data in a single phase. A few algorithms have been offered for deraining and moving object recognition.

According to the proposed technique, the wet video is viewed as a third order tensor with three components, $V = B + F + R$. The three components which represent the backdrop, the foreground, and the raining elements, respectively, are right away recognizable from each other.

Slice rotational total variation, or SRTV, is a new technique proposed in the article to combine various rain patterns into a single pattern by converting the rainy data into upfront pieces by rotating an angle of 90 degrees. This results in a dot-like pattern for each rainy data, which is then removed using the Total Variation (TV) operator, completely removing the rain streaks.

Rain streaks are eliminated from the backdrop by combining SRTV and WYNN. Separating the foreground from the background: Moving objects undergo a prominent and continuous shift in intensity in both spatial and temporal dimension. To identify foreground components, l_1 minimization and Tensor Total Variation (TTV) are used along with SRTV.

When compared to other approaches now in use, this method delivers results which are quite good. It offers peak signal-to-noise ratios of about 59.18 dB in light rain and 48.05 dB in severe rain. When compared to other approaches previously employed, the average of the f0 and f1 scores that is utilised for analysis of moving object identification is also excellent.

We may draw the conclusion that the suggested technique can effectively derain and simultaneously detect moving objects. This approach introduces an entirely novel technology called SRTV, significantly improving its effectiveness. The presently suggested method may be used to films with static backgrounds, and it can be modified further to accommodate videos with dynamic backgrounds.

Mohamed Hammami et. al [3]:We can infer that the given method is capable of derain and concurrently identify moving objects. This strategy significantly improves the effectiveness by introducing SRTV, a whole new technology. The approach that is now presented may be used to films with dynamic backgrounds in addition to films with fixed backgrounds.

Classification Prediction Model (WCC-PM) that presents challenges due to inadequate lighting and weather. They wanted to show the complexity of moving object identification algorithms on the IR spectrum & VIS spectrum independently in their study, which primarily concentrated on two issues. Their subsequent goal was to evaluate the success of their approach for shifting between full-spectrum cannabis light sources.

Qian Rui et. al [4]: applied a generative network with attention through adversarial training. The basic goal is to activate the generative and discriminative networks with visual attention. The introduction of visual focus into dual generative and discriminative networks is the primary benefit of this investigation.

Generative Network: A contextual auto encoder and an attentive-recurrent network are its two sub-networks. Attentive recurrent network job is to detect portions of the input picture that deserve more attentive. Models of visual attentions have been used to isolate and classify certain areas of a picture. The idea has been used to the creation of raindrop-free background images for visual classification and identification.

Discriminative Network: In the discriminative phase, only some GAN-based approaches exploit local as well as global image-content integrity to distinguish between fake and true pictures. Although the local discriminator concentrates on specific areas, the global classifier examines the entire image for abnormalities. They made advantage of the attentive-recurrent network's attention map. The features of the discriminator's internal layers, in particular, are collected and sent to CNN. Based on the output from CNN and the attention map, they come up with a losses estimation. The completely linked layer represents whether the input picture is true or false at the end layer.

2.2 RELATED TECHNOLOGY :

PYTHON

Python is an object-orient, higher level, dynamic semantic scripting language. It has special appeal for Rapid Application Development as well as for use as a scripting or glue language for getting existing components together because of its higher-level built-in data models, dynamic kind, and dynamic binding capabilities. Python's small syntax fosters readability and makes it simple to learn, lowering the costs associated with creating software. Python's support for modules and packages aids program modularity and reuse. Both the extensive standard library and the Python interpreter are freely distributable and available in source or binary form for all major operating systems. Python is a great option for machine learning and artificial intelligence (AI) projects given its simplicity, consistency, and ease of use. Python is an attractive option for machine learning and artificial intelligence (AI) uses due to its simplicity of use, consistency, accessibility to excellent modules and platforms, mobility, system autonomy, and big community. This enhances the language's general acceptance.

MACHINE LEARNING

Computers may now learn without being explicitly programmed thanks to the science of machine learning. Machine learning has been one of the more intriguing technologies in mankind. Itself provides the computer the ability to learn, which makes it more like human minds, as the name implies. Machine learning is being utilized extensively now, perhaps in a lot more fields than one might carry out.

Machine learning (ML) is the process of automating and enhancing how computers learn from their experiences without the need for programming or human oversight. The process begins with delivering accurate information to our tools (computers), which are then trained through the creation of machine learning models using the data and other approaches. The algorithms we use are defined by the data types we have to hand and the operations we need to automate.

DEEP LEARNING

A branch of machine learning known as "deep learning" employs numerous layers of neural networks to analyse enormous amounts of data and perform computations on it. The deep learning algorithm is based on the makeup and functioning of the brain of an individual. The deep learning algorithm is learning by themselves and may be used to both organized and unstructured data sources. Deep learning may help many kinds of businesses.

By continuously evaluating data with a predetermined logical framework, deep learning algorithms try to reach conclusions that are similar to those that people would reach. Deep learning does this via a multi-layered neural network algorithmic framework. The structure of the human mind served as inspiration for the network's architecture. We can train neural networks to recognise patterns and categorise various forms of information, just as how human brains do it naturally.

The likelihood of detecting and delivering a proper result increase by using the different layers of neural networks as a kind of filter that operates from coarse to fine. Similar processes occur in the human brain.

OPENCV

In order to alter and extract data from films and images, we need to understand how they are stored. Technology gives us this ability. Technology is the basis for artificial intelligence or the primary tool used in it. Robotics, autonomous vehicles, and photo-editing software all heavily rely on computer technology.

OpenCV is a huge open-source processing of pictures, machine learning, and vision library. It now plays a large role in real-time operation, which is critical in modern systems. This may be

applied when assessing images and videos for identifying individuals, objects, and even human handwriting. When paired with a variety of sections, such as NumPy, Python can manage the arrangement of the OpenCV arrays for analysis, understanding an image structure and its many features. We do empirical actions on these properties in vector space.

SCIKIT IMAGE

A free, open-source image processing package for Python is called Scikit-image (formerly known as Scikits.image). It contains algorithms for feature identification, morphology, analysis, filtering, colour space alterations, segmentation, geometric transformations, and more.

Features of scikit-image are as follows:

- It is a very simple and compact image processing tool.
- It was built on the foundation of SciPy, matplotlib, and NumPy.
- Anybody may use it and access it.
- It is a highly efficient open-source utility.

PYTORCH

PyTorch is a free machine learning framework built on the Torch library, which is used widely by Facebook's AI research group for tasks that include image recognition and processing natural languages. Torch is the basis of PyTorch. It is free open-source software released under the Revised BSD copyright. PyTorch uses dynamic computation rather than static computation graphs, providing it more liberty in developing intricate models. Document your code that supports mathematical equations.

PyTorch offers two high-level capabilities:

- Tensor computation (similar to NumPy) with significant acceleration offered by graphics processing units (GPU).
- Deep neural networks based on a participate-based differentiation automatic system.

NUMPY

NumPy stands for Numerical Python. The NumPy Python module serves to work with matrices. It additionally includes functions for working with matrices, the Fourier transform, and linear algebra. NumPy arrays, unlike lists, remain in a single uninterrupted place in memory, making it highly easy for scripts to access and modify them. This is the basic cause NumPy is speedier than lists. It was also enhanced to operate with the most current CPU layout. NumPy arrays are necessary to record and handle data with the Python extension of the renowned computer vision programs OpenCV. Because photos with multiple channels can be saved as three-dimensional arrays, accessing certain pixels in a picture by indexing, cutting, or filtering with other arrays is extremely effective. The adoption of the NumPy to array as a universal data structure in OpenCV for photos, extracted feature points, filter kernels, and other data structures permits programming and debugging tremendously.

GOOGLE COLAB

Following are the features offered by Google Colab:

- Python code development and execution.
- Notebooks can be created, posted, and distributed.
- Notebooks are able to loaded and saved from/to Google Drive.
- GitHub notebooks can be imported and published.
- Import external datasets, for example, from Kaggle.
- Integrate PyTorch, TensorFlow, Keras, plus OpenCV.
- Free cloud storage with GPU.

VGG

Another convolution neural network design utilised for image classification is the VGG network.

MOBILENETS

MobileNets are utilized to construct small deep neural networks. MobileNet is based on a condensed design that use 3-3 productive-wise separated convolutions, which need as much as eight times lower compute than a standard convolution while maintaining extremely high accuracy.

TENSOR FLOW

Tensor Flow is an open-source data processing computer library. Its adaptability allows for simple calculating deployment across a wide range of systems including personal computers, client groups, handheld devices, and edge devices. Tensor flow was created by Google Brain scientists and engineers in collaboration with Google's AI group. It includes strong deep learning and machine learning assistance, and its responsive numerical computing core is utilized in an array of scientific parts.

TensorFlow was chosen because it is simple and clear to generate, train, and deploy Item Detection Models and because it gives a library of Detection Models that have been trained on COCO, Kitti, Open Illustrations datasets. One of the few Detection Models is made up of Single Shot Detector (SSDs) plus Mobile Nets architecture because it is fast, efficient, and doesn't need a large number of CPU cores to identify objects.

CHAPTER - 3
DERAINING ALGORITHM

3. DERAINING ALGORITHM

3.1 GENERATIVE ADVERSARIAL NETWORK (GAN)

3.1.1 INTRODUCTION

GANs are algorithmic designs that create new, synthetic instances of data which may be misinterpreted for real data by using two neural networks that are in fight with one another (thus the name "adversarial"). They are widely used to produce images, films, and vocals.

For example, a discriminative algorithm could decide whether or not an email as spam by looking at every one of the phrases used in the message. One of the labels is "spam," and features make the input data are words gathered from email in bag. Mathematically, the label is denoted by the letter y , and the features are denoted by the symbol x . In the formula $p(y|x)$, the term "probability for y given x " is employed, which in the present instance implies "the chance that a message is spam given the content that it contains".

Differentiation algorithms then associate label with features. Only that correlation is of concern to them. Generative algorithms can be thought of as doing the reverse. They seeks to predict characteristics gives certain label rather than predicting a label given specific features.

How likely are these features, assuming this email is spam, is the issue a generative algorithm is trying to answer. Generative models are concerned with "how you acquire x ", while discriminative models are concerned with the relationship between y and x . A label is determined via the probability of x given y or the probability of characteristics, $p(x|y)$, may be captured using them. Classifiers can also be generated using generative algorithms. Fortunately, they are capable of more than simply classifying raw data. Even generation models mimic the distribution of different classes, discriminative models learn the boundaries among various classes.

3.1.2 HOW GANs WORK?

The discriminator assesses the fact that every instance of data evaluated by the generator truly comes from its training dataset, whereas the generator includes an artificial neural network which produces new data items.

The offender gets new, synthetic visuals from the generator, which itself is making images in the background. Even though they are phoney, it does this in the hopes that they too would be seen as genuine. The generator's mission is to create plausible handwritten digits, enabling you to cheat without obtaining caught. The discriminator goal is to recognise fake images produced through the generator.

The generator model develops images based on unpredictable noise(z) and learns how to build realistic images. (As shown in Figure 1). Input unpredictability is selected using a standard or regular distribution and then fed into a generator, which creates a final output image.

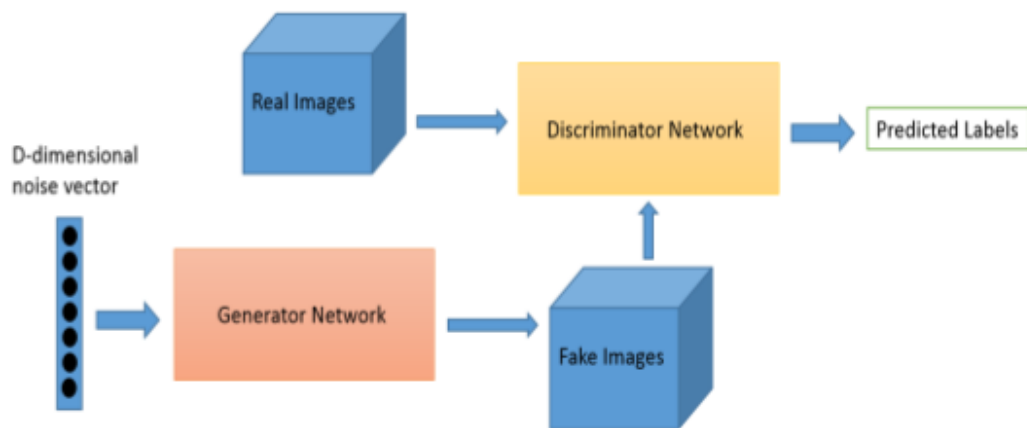


Figure 1: GAN Schema

The discriminator had to learn how to tell between truly and fake pictures using the output of the generator, incorporating both parts of the training set.

Conclusion $D(x)$ indicates the chances that the data being provided is real. If the input is false, $D(x)$ should be 0, else it should be 1.

The generator produces a picture after accepting random numbers. This established image, together with a stream of shots from the real, ground-truth dataset, is passed into the discriminator. The discriminator receives both true and illegal images and allows probabilities, with 1 indicating authenticity and 0 indicating fraud images.

3.1.3 WHY GANs?

The best characteristic of GANs is nature of learning, that typically follows robust unsupervised learning. As a consequence, GANs do not require assigned data, which makes them extremely effective and simple to utilize since they eliminate the laborious job of tagging and marking the data.

Second, a generative algorithm was outlined that can generate realistic-looking data by mixing it with an adversarial network that generates good natural photos. This framework may be used to improve pixels in photos, create pictures from input text, transfer images from one domain to another, and modify the style of the image, in addition to producing degree of assurance synthetic data.

Secondly, imagine a situation in which we lack adequate data for the topic at hand. In that situation, rather than adopting tactics like data augmentation, we may use adversarial networks to "generate" new data. Furthermore, there are numerous activities that, by necessity, need the realistic creation of samples from a range.

A substantial number of generation models, and GANs in particular, enable machine learning to deal with multi-level outputs. A single intake may be closely associated with a number of acceptable outputs, each of that is acceptable, when we have several tasks to accomplish.

Not to mention, every other industry has taken note of the never-ending GAN research because it's so fascinating. Many key technological improvements have been made by GANs during their development, leading to their expanding popularity.

3.2 ATTENTIVE GENERATIVE ADVERSARIAL NETWORK (AGAN)

3.2.1 INTRODUCTION

The brightness of the background image can be significantly decreased and images may be harmed by raindrops bonding to a pane window or cam sensor. In this study, we quickly eliminate raindrops from the problem, resulting in an unaltered picture from a raindrop-damaged photo. We are able to tackle the issue at hand through the use of an adversarial training attention generating network. Our fundamental idea is to activate the generate and discriminatory network via visual focus. Our visual attention centered on the positions of the raindrops and their surroundings throughout the workout. The information input will lead the generative network to concentration more on the raindrop areas and the structures around it while also allowing the discriminative network to evaluate the local consistency of the restored regions. This incorporation of visual information into both generator and discriminatory networks is the primary outcome of this study. Our study showed how both quantitatively and qualitatively, our approach outperform the most current techniques.

3.2.2 WHY AGAN?

Many approaches have been put out to solve the problems with raindrop detection and removal. Some methods are aimed at discovering raindrops but not to eliminate them. As a result, they cannot use a single input image acquired by a standard camera. Instead, various technologies are developed to identify and eliminate raindrops, such as stereoscopic, video or an especially made optical shutter. It can only tolerate light showers, though and the images are hazy. It is discovered that the procedure does not work with raindrops that are comparatively huge and thick. This effort, aims to address the significant occurrence of rainfall. In general, the issue of removing raindrops is overwhelming since it is impossible to determine beforehand which sections are covered in them.

Second, the majority of the background details in the occluded regions are entirely lost. When the raindrops are widely spaced and quite big, the issue becomes more difficult. This approach

employs a generative adversarial network to handle the issue with the produced outputs being assessed by the discriminative network to make sure they match actual images. The generative network makes a first effort to create an attention map in order to address the problem's complexity. Because it guides the following procedure within the network's generative phase to concentrate on raindrop spots, the focus map is the biggest element of the network. A recurrent network made up of shallow residual neural networks, a layer of convolutional LSTM, and a couple of simple convolutional layers created it. An attentive-recurrent network is used to achieve this. An autoencoder, is second component of the generating network, accepts the attention map together with the input picture. Multi-scale losses are used at decoder side of autoencoder to get more contextual information. Each of the of these losses assesses the variation between the output of the convolutional layer and the associated, appropriately downscaled ground truth.

The convolutional layers are the given features from decoder layer. To get a more comprehensive likeness to reality, perceptual loss is applied to the autoencoder's finished product in addition to these losses. This end result is also the generative network's contribution.

After the generated image output has been obtained, the discriminative network will determine whether it is real enough. The discriminative network evaluates the picture both globally and locally, similar to some inpainting techniques. In contrast to the example of inpainting, the target raindrop locations are not specified in this issue, particularly during the testing stage. As a consequence, due to a lack of information on the local areas, the discriminative network cannot focus about these. In order to tackle this problem, attention maps are used to direct the discriminative network towards key target regions.

The attentive map is injected into generative and discriminative networks, which is an efficient method for raindrop removal. In addition to providing a unique technique for accomplishing this, this is the model's second key contribution.

3.3 RAINDROP REMOVAL USING AGAN

The network's general design is seen in Figure 2. The generative and discriminative networks are the two essential parts of the network, a system based on the theory of adversarial networks that are generative.

The generating network is interested to producing an image that is as real and raindrop-free as possible given an input image that has been harmed by raindrops. The discriminative network will verify if the generative network's produced picture looks to be real.

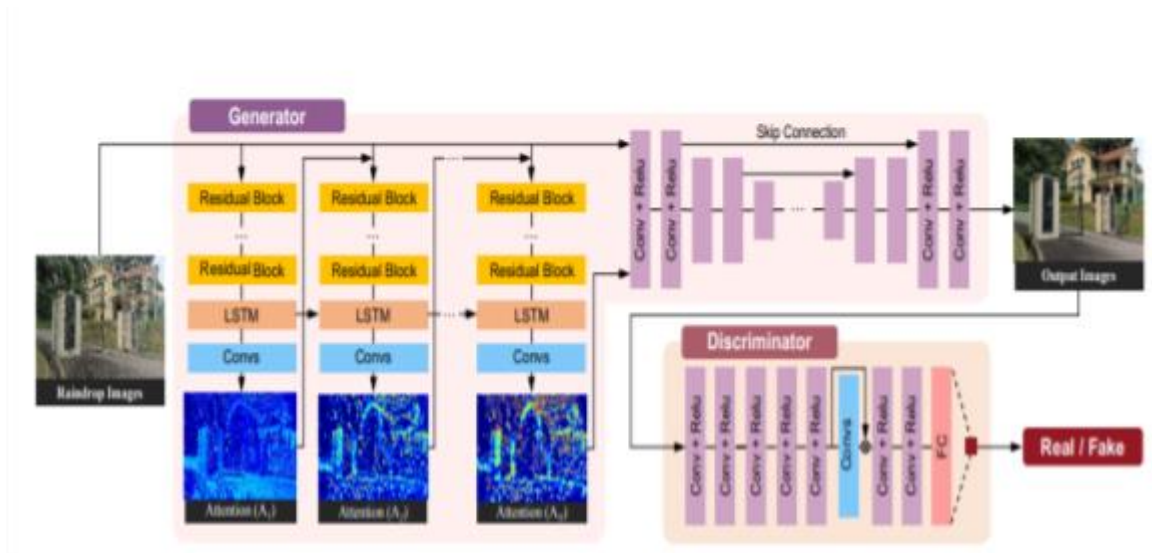


Figure 2: AGAN Architecture

The Generative Adversarial Loss expressed as:

$$\text{Min}(G)\text{Max}(D)E_{R \sim P_{\text{clean}}}[\log(D(R))] + E_{I \sim P_{\text{raindrop}}}[\log(1-D(G(I)))]$$

where G stands for generating network, D for discriminative network. It is sample taken from the water droplet image pool that serves as the generative network's input. R is an example taken from a collection of clear, genuine photos.

3.3.1 GENERATIVE NETWORK

One of the two distinct sub-networks of the generative network is known as the attention-recurrent network, and the other one is the contextual autoencoder. Figure 2 serves as an illustration. The attention recurrent network objective is recognizing areas of the input that require more attention. It is essential for the discriminatory network to focus its assessment on those regions, which are mainly the raindrop regions and the structures surrounding them, and for the contextual autoencoder to give better local restoration of the image in those parts.

Attentive-Recurrent Network :

Similar to this, it is believed that visual attention is vital for creating the background pictures that lack raindrops since it helps the network to decide on the regions for removal or restoration. The visual focus is offered via a recurrent network, as shown in the construction in Figure 2. Each block (of each time step) of the recurrent network comprises a convolutional LSTM unit, layers of convolution and layers of convolution for building the 2D attention levels. These types of layers help by obtaining data derived from the input photograph and the block's preceding block's masks. A larger number implies a higher level of attention on the attention map, resulting in a matrix with numbers spanning 0 to 1. At each time step, it is learnt. Compared to the binary mask M , the attention map is non-binary and shows how attention travels from non-raindrop areas to raindrop regions, with values affecting even within raindrop sections. This extra focus makes appropriate given that the spaces around droplets need attention and that different raindrop zones show parameter shades of transparency. Included are an input gate, a forget gate, an output gate, a cell state, and an LSTM convolution unit. The following definition explains how states and gates interact along the time dimension. Included are an input gate, a forget gate, an output gate, a cell state and an LSTM convolution unit. Relationship between states an input gate, a forget gate, an output gate, a cell state and a convolution LSTM unit are all included. The interaction between states and gates along time dimension is defined as follows. An input gate, a forget gate, an output gate, a cell state and a convolution LSTM unit are all included.

The interaction between states and gates along time dimension is defined as follows:

$$\begin{aligned}
i_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \odot C_{t-1} + b_i) \\
f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \odot C_{t-1} + b_f) \\
C_t &= f_t \odot C_{t-1} + i_t \odot \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c) \\
O_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} \odot C_t + b_o) \\
H_t &= O_t \odot \tanh(C_t)
\end{aligned}$$

where X_t represents features produced by ResNet. The cell state that will be passed to the following LSTM is encoded by C_t . The output characteristics of LSTM unit is represented by H_t . The operation of convolution is represented by the operator. The convolutional layers then receive the LSTM's output feature, which creates a 2D attention map. We set the attention map's values to 0.5 during the training procedure. Each time step, the incoming picture and the current attention map are joined, and they are then fed into the successive stages of the recurrent network.

To train generative network, identical background scene pairs of photos with and without raindrops are employed. The mean squared error (MSE) between the binary mask M and output attentive map at time step t or A_t , is the definition of the loss function in each recurrent block. N steps are involved in this procedure. By nearing the N th time step, the earlier attention maps' values, grow suggesting a rise in confidence. Loss function is stated as follows:

$$L_{ATT}(\{A\}, M) = \sum_{t=1}^N \theta^{N-t} L_{MSE}(A_t, M)$$

where A_t represents attentive map formed at time step t by attentive-recurrent network.

$$A_t = ATT_t(F_{t-1}, H_{t-1}, C_{t-1})$$

with F_{t-1} is attention map from the previous time step paired with the input image. When $t = 1$, F_{t-1} is the input image concatenation with an initial attention map with values of 0.5. Function ATT_t , the attentive-recurrent network at time step is denoted by t . N is set to 4 and θ to 0.8. Highest N values are expected to end in best attention maps, but larger memory is also required. The network looks for structures surrounding the raindrop regions in addition to the

raindrop regions themselves. As the time step grows, the system emphasizes increasingly on the raindrop locations and relevant components.

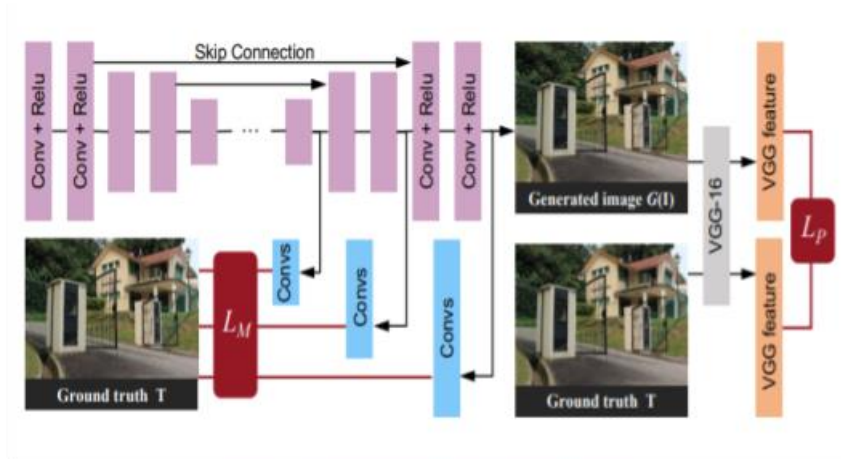


Figure 3: Contextual Autoencoder Architecture

Contextual Autoencoder:

The contextual autoencoder's objective is to generate an image clear of raindrops. The attentive-recurrent network's final attention map is combined with the input picture to form the autoencoder's input. To prevent blurring outputs, 16 conv-relu blocks are used. Figure 3 illustrates the way our contextual autoencoder is created.

Multi-Scale losses and Perceptual losses are 2 types of loss functions in the autoencoder. The characteristics are extracted out of several decoder layers to create outputs in different sized for the multi-scale losses. This allows for the capture of additional contextual data at various scales. Contextual autoencoder is what it is named for this reason as well. As seen below, the loss function is expressed.

$$L_M(\{S\}, \{T\}) = \sum_{i=1}^m \lambda_i L_{MSE}(S_i, T_i)$$

where T_i denotes ground truth, which has the same scale as S_i and S_i denotes i th output obtained from the decoder layers. $\{\lambda_i\}_{i=1}^m$ are weights for different scales. The larger scale has

much more weights placed on it. To be more precise, the sizes of the outputs derived from the previous 1, 3, and 5 layers are utilised. Are 1/4 , 1/2 and 1 of the original size respectively. When the information contained in bigger layers is insignificant. λ 's are set to 0.6, 0.8, 1.0.

Multi-scale losses, which hinge on pixel-by-pixel action and quantify the general disparity among the features of the output of the autoencoder and those of the corresponding ground-truth clean picture, are paired with a perceptual loss. To extract these attributes, a trained CNN could be used.

Example: VGG16 underwent training using the ImageNet data set. As a formula, the perceptual loss function is:

$$L_p(O,T) = L_{MSE}(VGG(O), VGG(T))$$

where VGG creates features from an input image using a CNN that has been pretrained. O is the generative network's overall output picture, or the output of the autoencoder: $O = G(I)$. T is the vision of real world that is not covered in rains.

The loss of our generative can be stated as follows:

$$L_G = 10^{-2} L_{GAN}(O) + L_{ATT}(\{A\},M) + L_M(\{S\}, \{T\}) + L_P(O,T)$$

Where $L_{GAN}(O) = \log(1-D(O))$.

3.3.2. DISCRIMINATIVE NETWORK :

For machines to differentiate between fake and real images, likely GAN-based approaches apply global and local image-content coherence. Whereas the local discriminator focuses on tiny, focused areas, the global discriminator examines the entire picture to check for any inconsistencies. If we are aware of the regions that are most likely to be phoney, the local discriminator technique is especially beneficial. Sadly, the places damaged by raindrops are unknown in this scenario, especially during the testing phase, and no information is provided. Consequently, the local discriminator has to look for those regions on its own.

The concept is to utilise an attentive discriminator to tackle this issue. An attentive map produced by attention-recurrent network is used for this. In particular, the discriminator's internal layers' characteristics are retrieved and given to a CNN. A loss function is produced utilizing the CNN output and the focus map. Also, the result of the discriminative network's CNN is multiplied with the original characteristics before being sent into the next layers. The

basic concept behind this is to direct the discriminator to concentrate on areas that are identified by the attention map. In order to determine if input image is false or real, a completely linked layer is employed at the end layer. The discriminator's overall loss function followed as :

$$L_D(o,R,A_N) = -\log(D(R)) - \log(1-D(o)) + \gamma L_{\text{map}}(O,R,A_N)$$

where L_{map} is the loss between the features extracted from interior layers of the discriminator and the final attention map:

$$L_{\text{map}}(O,R,A_N) = L_{\text{MSE}}(D_{\text{map}}(O),A_N) + L_{\text{MSE}}(D_{\text{map}}(R),0)$$

where $\gamma = 0.05$, D_{map} shows the procedure taken by the discriminative network in producing a 2D map. R is a real-life picture chosen from an array of genuine and precise photos. A value of 0 represents a map having precisely 0 values.

The discriminative network includes seven convolutional layers with a kernel of (3, 3), an entirely connected layer having 1024 connections, and a single neuron having an activation function that is sigmoid. Features from the final third of the convolution layers are fundamentally multiplied backward.

3.3.3 DERAINDROP DATASET

This technique needs a sizable quantity of data with ground truths for training, similar to existing deep learning techniques. Yet, a unique dataset is made because there isn't one for raindrops that are connected to a glass window or lens. In this instance, a series of image pairs are needed, each of which features the exact same background scene, but where one pair is tainted with raindrops and the other is untainted. In this process, two identical pieces of glass are utilized, one of which is cleaned & sprinkled by liquid.

Using two pieces of glassware helps to prevent misalignment since glass refracts light and has a refractive index that varies to air. While capturing the two photographs, it is often a great idea to control other potential sources of misalignment as well, such as camera motion (e.g., sunlight, clouds, etc.)

CHAPTER 4

OBJECT DETECTION ALGORITHM

4. OBJECT DETECTION ALGORITHM

4.1 YOLO

4.1.1 INTRODUCTION

A revolutionary strategy for object detection is known as YOLO. Applying algorithms to detect things has been done previously. However, in this case, object recognition is presented as an issue related to spatially distinct bounding boxes and associated class probabilities. A single neural network can directly predict bounding boxes and class probabilities from entire images in one assessment. Because the entire detection pipeline is made up of a single network, detection performance may be modified simultaneously.

This unified architecture works very quickly. The basic YOLO algorithm processes images in real time at a rate of 45 frames per second. despite the fact that Quick YOLO, a shortened version of the network, only examines 155 frames per second, it still performs two times better than other real-time detectors in MAP. Modern detection techniques make less prediction of false positives on background, but YOLO makes more localization errors than those systems. Last but not least, YOLO catches up extremely wide representations of objects. When deployed to other domains like artwork, it performs better than other detection techniques like DPM and R-CNN when expanding from natural images.

4.1.2 WHY YOLO?

Target detection system YOLO has a speedy detection performance and is perfect for target detection in a real-time circumstance. It provides superior detection accuracy and a quicker detection time when compared to other target detection systems of a similar kind. As according experiment results, the YOLO-based object detection approach offers more resilience and quicker detection speed. The excellent detection accuracy may be ensured even in a complicated setting. The detection speed may also satisfy the demands of real-time detection. On the other side, the YOLO framework (You Only Look Once) approaches object identification differently. Using the entire picture as a single instance, the coordinates of the bounding box and class likelihoods are predicted for these boxes. YOLO's incredible speed, which can process 45 frames per second, is by far its greatest benefit. Another concept that YOLO is comfortable with expands on object representation. One of the most efficient

algorithms for object detection, it has demonstrated performance that is comparable to the R-CNN techniques.

4.2 YOLO V2

The main issues with YOLO, which were the accuracy of geographical location and the detection of small objects in categories, are being addressed with YOLOv2. YOLOv2 uses batch normalisation to raise the system's mean Average Precision. The MAP value is increased by up to 2% by batch norms.

Anchor boxes were a far greater enhancement to the YOLO algorithm as said in YOLOv2. It is well known that YOLO predicts one object every grid cell. The model that results is simpler, however although YOLO can only assign one class to a cell, there are problems when a cell incorporates a lot of goods.

Because many boundary boxes may be predicted from single cell, YOLOv2 removes this constraint. The network is trained to predict five boundaries for each cell with the goal to achieve this. Experimental evidence indicates that the number Five offers a reasonable balance among model complexity and forecast accuracy. DarkNet-19, which consists of 19 convolutional layers in total and 5 max-pooling layers, is the basis upon which the YOLOv2 architecture is created.

4.3 YOLO V3

In complicated and dense crowds, where certain individuals may be entirely or partially covered for variable amounts of time, it can be exceedingly difficult to accurately real-time detect people in pictures or sound recordings. As a result, You Only Look Once version 3 (YOLOv3) is used to train a huge Convolutional Neural Network (CNN) on Google Colab to analyse snapshots in a database and accurately determine people in the pictures.

The image is divided into regions by YOLOv3, which also forecasts bounding boxes and likelihood for each region. The projected probabilities are then utilized to weight these bounding boxes, and the model is then able to carry out a detection according to the final values.

A dataset of 500 high-quality, image-modified Google Open Images images will be utilized in this model. The neural network is capable of offering test data after training, with a mean average precision (MAP) of 78.3% and a final mean loss of 0.6, in addition to correctly recognizing individuals in the pictures.

4.4 YOLO V4

The latest version of the BoF (bag of freebies) and various BoS can be found in YOLO v4. (Bag of specials). Without increasing the inference time, the BoF improves the detector's accuracy.

With an AP value of 43.5 percent on the COCO dataset and a real-time speed of 65 FPS on the Tesla V100, YOLO v4, which is also Darknet-based, surpassed the fastest and most accurate detectors in terms of speed and accuracy. The AP and FPS have risen by 10% and 12%, respectively, in contrast to YOLO v3.

The feature extraction from the input pictures by the YOLO v4 network is based on CSPDarkNet-53. The YOLO v4 network's backbone is composed of five residual block modules, and the characteristic map outputs from these modules are fused at the network's neck. The SPP modules in the neck concatenates the max-pooling outputs of the low-resolution feature map to extract most informative features.

4.5 YOLO V5

The most recent version of YOLO version 5, is used for identifying objects with quick detection speed and pinpoint precision. For the entire Common Objects in Contextual val2017 dataset, it obtains and provides 72% AP 0. In addition, YOLOv5 exists in a variety of forms. The YOLOv5s model, which has a minimum size of 14 megabytes, is the easiest to deploy.

The Efficient Det architecture, which is based on the Efficient Net network design, is used by YOLO v5. In YOLO v5, a more complicated architecture was used to increase accuracy and improve generalisation to a larger variety of object categories. As SPP enables the model to

view the objects at various sizes, it is employed to enhance the detection performance for tiny objects.

4.6 YOLO V6

YOLOv6 is a single-stage object detection framework intended for industrial applications. It has outstanding reliability and a hardware-friendly architecture. As it surpasses YOLOv5 in terms of detection precision and inference speed. It is the most effective OS version of the YOLO architecture for applications in commerce.

4.7 YOLO V7

A single-stage actual time object detector is YOLOV7. For identifying objects with different shapes, anchor boxes are an assortment of established boxes with varying perspective ratios.

YOLO v7 can recognise a wider range of object shapes and sizes than previous generations, which helps to reduce the number of false positives. Also, YOLO v7 has a higher resolution than previous generations. Because of its improved resolution, YOLO v7 can recognise microscopic objects with more precision.

The following characteristics of YOLOV7:

- It is 120 times faster than YOLOv5 and other cutting-edge object detectors.
- Other object detectors cannot match the AP of the COCO dataset.
- Improvement of the architecture and loss function.
- The YOLOv7 repository enables recognition of objects, instance segmentation, categorization, and posture estimation.
- There are multiple YOLOv7 model versions available to meet various speed and accuracy requirements.

4.7.1 YOLO V7 ARCHITECTURE

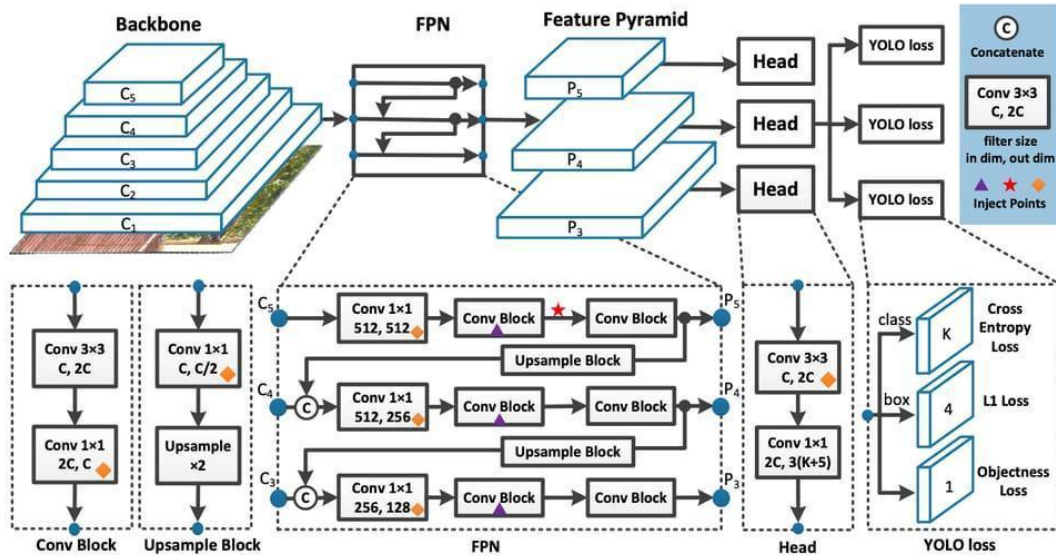


Fig 4 : Architecture of yolov7

Based on ELAN, the YOLOv7 architecture was created. (Efficient layer aggregation network). In order for deeper networks to converge and train efficiently, ELAN take into consideration building an efficient network by controlling the shortest and longest gradient paths.

4.7.2 FUNCTIONALITY

A grid of S*S size has been generated by this architecture inside the picture. If the object's centre falls inside the bounding box of that grid, then this grid is in control of detecting that object. With the assistance of their confidence score, every grid predicts box boundaries. Each confidence score shows how precisely the bounding box coordinates predicted relative to the ground truth prediction along with how well the bounding that predicts involves an object.

We multiply the individual box confidence anticipates and the conditional class probabilities at test time. Our confidence score is stated as follows:

$$P_r(\text{Object}) * IOU_{pred}^{tr}$$

The confidence score must be zero if there are no goods on the grid. The confidence score ought to correspond to the IoU between the ground truth and predicted boxes, if a thing is seen

in the picture. Five predictions are made for each bounding box: (x, y, w, h) and a confidence level. The box's centroid in relationship to the boundaries of the grid cell is represented by the (x, y) coordinates. The enclosing box's height and width can be determined by the h and w coordinates respectively (x, y). The confidence score reveals whether an object is present in the bounding box. This leads to a combination of bounding boxes from each grid, where each grid in addition predicts $\Pr(\text{Class}_i | \text{Object})$, the probability of a C conditional class.

The presence of an item in the grid cell impacted this likelihood. No matter how many boxes there are only one set of class probabilities is predicted by each grid cell. The $S * S * (5 * B + C)$ 3D tensor has these predictions. The individual box confidence forecasts and conditional class probability have now been multiplied.

$$P_r(\text{Class}_i | \text{Object}) * P_r(\text{Object}) * IOU_{pred}^{trut} = P_r(\text{Class}) * IOU_{pred}^{trut}$$

This gives us scores of trusts for each box that are specific to the class. Both the possibility of that class being in the box and how closely the predicted box matches the item is determined by these ratings. After that, we anticipate the input's final result via non-maximal suppression. YOLO does exceptionally well in the test because it predicts outcomes using just one CNN architecture and class is configured in a manner that it sees classification as a regression problem.

4.8 APPLICATIONS OF OBJECT DETECTION:

The major applications of Object Detection are:

FACIAL RECOGNITION

To recognise the face of people in digital images, the "Deep Face" deep learning face recognition system was developed and created by a Facebook research team. In addition, Google Photos has a face recognition instrument that organizes all of the images into groups based on who is in each one.

As various writers have put it, facial recognition focuses on a variety of traits in order to accomplish the aim of face identification.

Perhaps there should be a different class of object detection for face detection. We are curious about how Facebook, Faceapp and other programmes can identify and detect our faces.

In everyday life, this serves as a typical example of object detection. In our daily lives, face detection is already used to unlock our mobile phones and to reduce the rate of other security systems.

INDUSTRIAL QUALITY CHECK

To identify or recognise items, object detection is crucial in industrial operations. A vital role that is frequently used in industrial processes including sorting, stock control, machining, quality control, packaging and others is inspection by sight. Keeping track of inventory may be quite difficult since it is hard to trace goods in real time. Accurate inventory management is made possible by automatic object counting and location.

SELF DRIVING CARS

The most promising technology for the future is autonomous vehicle technology. Yet, since they use so many different kinds of sensors for understanding the world around them. Automated control systems convert sensory data into routing strategies, obstacles, and other knowledge. Given how quickly it happens, this is a significant step towards driverless automobiles.

SECURITY

In the security industry, object detection is significant. It is employed in major apps like Apple's facial ID and the retina scan that is found in any science fiction movies. Authorities frequently utilise this method to obtain access to surveillance feeds and contrast them to their data in order to recognise criminals or objects used in illicit activities, such as vehicle registrations. There are countless opportunities.

CHAPTER 5
RESULTS AND CONCLUSION

5. RESULTS AND CONCLUSION

5.1 RESULTS

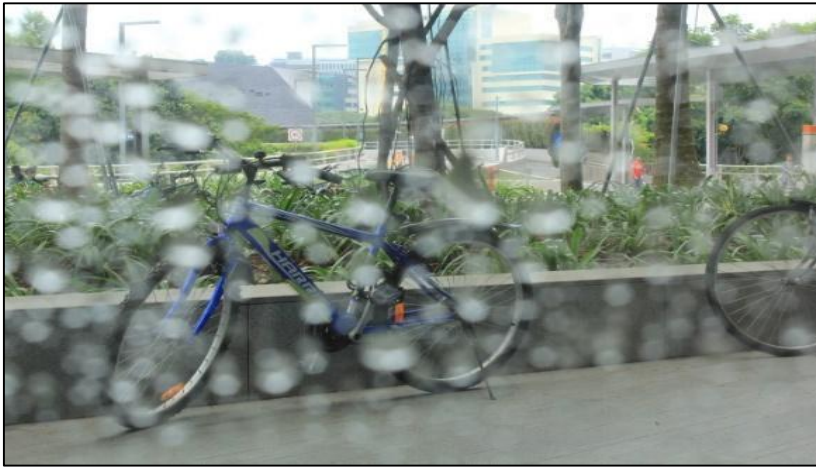


Fig 5 : Input Rainy Image



Fig 6 : Output Derained Image



Fig 7: Object Detection in rain image



Fig 8 : Object Detection in derain image

- Peak signal to noise ratio after de raining: 24.986 dB.
- Structural similarity after de raining: 84.56%.
- Bicycle : 96%.
- Bicycle : 88%.
- Person : 76%.

5.2 CONCLUSION

In this work, we presented an approach for removing raindrops from a single image. The method makes use of a generative adversarial network, in which the generative network creates the attention map using an attentive-recurrent network and applies this map along with the input image to produce a raindrop-free image via a contextual autoencoder. Our discriminative network then evaluates the global and local validity of the output generated. We inject the attention map into the network to enable local validation. We also believe that our approach is the first approach that can effectively deal with the rather heavy presence of raindrops, which the most advanced techniques for raindrop removal are unable to do. The PSNR and SSIM values of the final image are the highest. . The resulting image has the highest PSNR and SSIM values. By running the derained image through the Yolov7 technique for the most accurate identification of the objects in it.

6. FUTURE SCOPE

- Can be further extended to videos
- May try to implement for multiple climatic conditions simultaneously
- Can be extended for tracking the detected objects

REFERENCES

- [1] Hayder Radha and M. Hnewa, "Object Detection Under Rainy Conditions for Autonomous Vehicles: A Review of State-of-the-Art and Emerging Techniques," in *IEEE Signal Processing Magazine*, vol. 38, no. 1, pp. 53-67, Jan. 2021, doi: 10.1109/MSP.2020.2984801.
- [2] Mohamed Hammami, Rania Rebai Boukhriss, Emna Fendri, Moving Object Detection Under Different Weather Conditions Using Full-Spectrum Light Sources, *Pattern Recognition Letters* (2019), doi: <https://doi.org/10.1016/j.patrec.2019.11.004>.
- [3] George S and Baiju P, 2020. An Automated Unified Framework for Video Deraining and Simultaneous Moving Object Detection in Surveillance Environments. *IEEE Access*, 8, pp.128961-128972.
- [4] Qian, Rui, Robby T. Tan, Wenhan Yang, Jiajun Su and Jiaying Liu. "Attentive Generative Adversarial Network for Raindrop Removal from A Single Image." 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018): 2482-2491.
- [5] M. Roser and A. Geiger. Video-based raindrop detection for improved image registration. In *Computer Vision Workshops (ICCV Workshops)*, 2009 IEEE 12th International Conference on, pages 570–577. IEEE, 2009. 1, 2.
- [6] M. Roser, J. Kurz, and A. Geiger. Realistic modeling of water droplets for monocular adherent raindrop recognition using bezier curves. In *Asian Conference on Computer Vision*, pages 235–244. Springer, 2010. 1, 2 .
- [7] H. Kurihata, T. Takahashi, I. Ide, Y. Mekada, H. Murase, Y. Tamatsu, and T. Miyahara. Rainy weather recognition from in-vehicle camera images for driver assistance. In *Intelligent Vehicles Symposium*, 2005. Proceedings. IEEE, pages 205–210. IEEE, 2005. 1, 2.
- [8] Y. Tanaka, A. Yamashita, T. Kaneko, and K. T. Miura. Removal of adherent waterdrops from images acquired with a stereo camera system. *IEICE TRANSACTIONS on Information and Systems*, 89(7):2021– 2027, 2006. 2.
- [9] A. Yamashita, I. Fukuchi, and T. Kaneko. Noises removal from image sequences acquired with moving camera by estimating camera motion from spatiotemporal information. In *Intelligent Robots and Systems*, 2009. IROS 2009. IEEE/RSJ International Conference on, pages 3794–3801. IEEE,2009. 2.

- [10] S. You, R. T. Tan, R. Kawakami, Y. Mukaigawa, and K. Ikeuchi. Adherent raindrop modeling, detection, and removal in video. *IEEE transactions on pattern analysis and machine intelligence*, 38(9):1721–1733, 2016. 2, 3.
- [11] D. Eigen, D. Krishnan, and R. Fergus. Restoring an image taken through a window covered with dirt or rain. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 633–640, 2013. 2, 3, 6, 7.
- [12] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015. 2, 4.
- [13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 4.
- [14] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2011. 2.
- [15] V. Mnih, N. Heess, A. Graves, et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pages 2204–2212, 2014. 3.
- [16] B. Zhao, X. Wu, J. Feng, Q. Peng, and S. Yan. Diversified visual attention networks for fine-grained object classification. *arXiv preprint arXiv:1606.08572*, 2016. 3.
- [17] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015. 3.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, Las Vegas, NV, USA, 2016.
- [19] L. Wang, H. Zhen, X. Fang, S. Wan, W. Ding, and Y. Guo, “A unified two-parallel-branch deep neural network for joint gland contour and segmentation learning,” *Future Generation Computer Systems*, vol. 100, pp. 316–32.