iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VI **Month of publication:** June 2022

DOI: https://doi.org/10.22214/ijraset.2022.44581

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

Anomalous Object Detection with Deep Learning

L. Poojitha[1], Y. Shravani[2], A. Satyavani[3], Dr. Syed Jahangir Badashah[4], Dr. C. N. Sujatha[5]
[1, 2, 3]B Tech ECE Students, Sreenidhi Institute of Science and Technology
[4]Associate Professor ECE, Sreenidhi Institute of Science and Technology
[5] Professor ECE, Sreenidhi Institute of Science and Technology

***Abstract:** In many computers vision systems, object identification and monitoring are crucial characteristics. Object identification and monitoring is a difficult job in the fields of computer vision that attempts to identify, recognize and track things across a video series of pictures. It aids in the understanding and description of object behaviour rather than relying on human operators to monitor the computers. Its goal is to find moving things in a video clip or a security camera. On the other hand, rely heavily on computers, on the other hand many respect attributes approaches. The system collects a snapshot from the camera, process it as per the models requirements and the outputs the data to the tensorflow framework which provides a list of the frames discoveries and the objects dependability points and connections to their binding containers. Before exposing the object to the user, the software takes those connections and creates a rectangle adjacent to it. This research detects the presence of anomalous items in camera-captured sequences, with anomalies being things that correspond to categories that should not be present in a given scene.*
***Keywords:** Convolutional Neural Network (CNN), Faster R-CNN, Network on Convolution Feature Map (NoC), Deep Expectation (DEX), Deep Residual Conv-Deconv Network, A-ConvNet.*

I. INTRODUCTION

Anomalous is of uncertain nature or abnormal things. These abnormal objects are detecting by using the deep learning. Object detection and tracking are important features in many computer vision systems and are used in autonomous vehicles, surveillance, medical diagnostics and automation and more. It is one of the research center that has made great strides in recent years. Conventional based Neural Network (CNN) methods have proven to be very effective in this regard. As the integration of modern computers and equipment increases and the availability of data are also increases, complex models can achieve both high levels of accuracy and short processing times. Get real time performance with the right model and powerful hardware. The aim of this project is to improve an anomalous object detecting, tracking and also gives the sound indication of the object to the user it is run by the Raspberry Pi 4 model in real time. It also provided the system is easy-to use image detection and tracking. The Raspberry-Pi is an inexpensive and low power usage computer. Its flexibility and price have made it popular with automation setups, especially among hobbyists. However, due to the low compression capabilities, obtaining objects using the limited current architecture can result in very long processing time.

II. LITERATURE SURVEY

We have done a comprehensive survey of available literature to capture the theme and design confusing features acquisition strategies in deep learning. We adopted a methodical strategy to gather research on the topic of perplexing discovery, and we divided the recommended approaches into the parts below. Because this was a research piece, we began by looking at research at top conference proceedings in domains like computer vision and machine learning. The conferences we selected are related papers are ICC, CIKM, SPW, KDD, AAAI, CS Cloud, ICBDA, DMIN, SMC, ICML, CEUR, ICCV, AINS AND CVPR. In addition, journals from selected pages related by the International Journal for data analytics and data science, performance Intelligence analysis, data mining and data acquisition, Ambient Journal Intelligence and IEEE transactions, Humanized computing. In Processing Processes, IEEE IoT, IEEE communication letters. The mysterious discovery in the deep reading is a fast grow research flatform, also we have included a few arXivs preprints as well.

III.RESEARCH METHODOLOGY

The anomalies cannot always be considered an attack yet it tends to be an amazing act that is not before known. It may be different object. Anomaly Detection of significant and emerging problem in many different fields such as information theory, in-depth study, computer theory, machine learning, and mathematical calculations within a variety of applications from different categories are included banking, security purpose, agriculture, education, health care and transportation confusing discovery.

Now, at a time when information must be checked for relationship or prediction whether unknown or known, segregated, deprecated, and consolidated, based on data mining, machine learning, and in-depth reading Supported strategies and strategies are used. Gaining a high level of accuracy and get Specification the level of discovery of a confusing action from information or data or event. Currently a hybrid based on deep learning and methods are also being developed and tensor flow of model starts by collecting data from the video and converting they begin to become independent.

A. System Design

In this System Detecting an object involves both saying that the object of the said category exists, as well as the image local action. The location of the item is usually represented by a rectangle box. Objects or things are capture by the USB Camera. This camera is used as an input to the system and it captures different items. In the device we used Raspberry Pi 4 B model because it has ethernet port and also it consumes less power. Adaptor is used to supply the power to the device. When adaptor is connected to the device then the device is ON. It also connected one Bluetooth. This Bluetooth is used to give sound voice when anomalous objects are detected and also it gives voice when non anomalous object detected. This system is used deep learning technology. Nowadays most people used this technology because it gives most accurately results and it is easy to recongize or detect the objects.

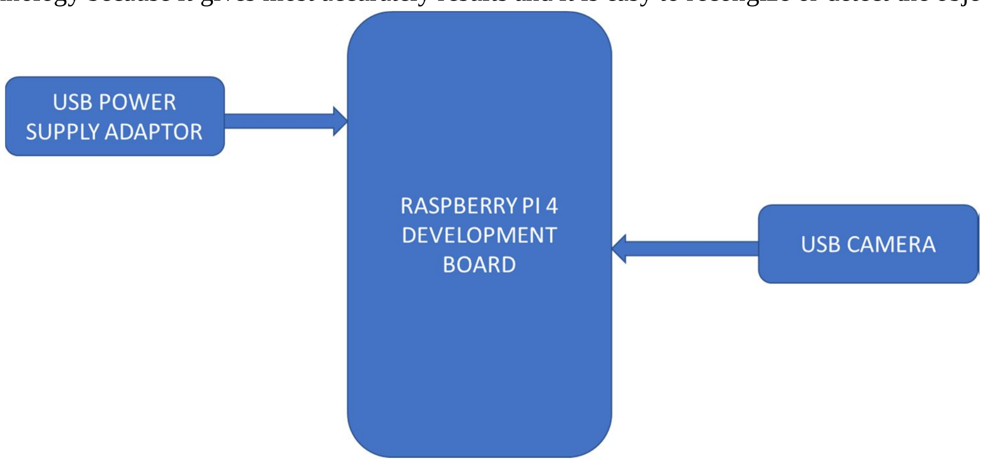


Fig. 1. Block Diagram Of System Device

B. System Analysis

1) Hardware Requirements

- RASPBERRY Pi 4GB RAM DEVELOPMENT BOARD
- 16GB MICRO SD CARD CLASS 10
- TYPE C USB 5V 3AMPS ADAPTOR
- USB CAMERA

2) Software Requirements

- Operating system: Raspberry Pi
- Technology: Deep learning.
- Language: Python Programming Language.
- Platform: Tensor Flow Frame Work, CNN’s.

C. System Implementation

Generate A Model

From collecting information to evaluating our recently constructed object detector, these are the procedures we used to create our model.

The model is implemented in the following steps:

1) *Gathering data (Coco Dataset):* The MS COCO **(**Microsoft Common Objects in Context**)** dataset is used which is a large-scale object detection, segmentation, key-point detection, and captioning dataset. The dataset consists of 328K images.
2) *Labelling Data:* Label Img is a free, open-source tool for graphically labelling images. It's written in Python and uses QT for its graphical interface

```
pip install labelImg
```

D. For Training Generating Tf Records

To store the sequence of binary records the simple format is TF Records. It have some advantages by converting your data into the TF Record, including:

- *Storage Efficient is More:* TF Record data occupies less space than the original data. That can be split into multiple files.
- *Fast I/O:* The TF Record format can also read as parallel I/O, which is useful for multiple hosts or TPUs.
- *Self-contained Files:* The TF Record data can read from the any one source. For example, the COCO dataset initially stores two folders of data ("images" and "annotations")

E. Training Configuration

For customizing the training and scoring process the TensorFlow object retrieval API uses protobuf files. At high level, the configuration file has divided into five parts.

- *Model Configuration:* This sets the model type which will be learned (eg: feature extractor, Meta architecture).
- *Train_config:* Determining which variables, such as preprocessing inputs, which variables, such as, preprocessing inputs, and feature extractor, SGD parameters initialization, should be utilized to train the model parameters.
- *Config eval_config:* Identifies which indicators should be used for evaluations.
- *Train-input_config:* This indicates which database should be used to train the model.
- *Eval-input_config:* The database around which the model would be assessed is determined by this parameter. This should, in general, be distinct from the training input dataset.

F. Model Training

The existence and placement of various feature classes are detected using a feature collection model that has been trained. A model, for example, can be trained using photographs. containing different pieces of fruit and on labels that represent the categories of fruits being displayed (eg apple, banana, or strawberry), and data that specifies where each item came from the picture. If the image is provided later in the model, it will release a list of items we find, the location of the binding box containing each item, and a score indicating confidence that the acquisition was right. The following command is followed to train the model:

```
# Change into the models directory
$ cd tensorflow/models
# Make directory for storing training progress
$ mkdir train
# Make directory for storing validation results
$ mkdir eval
# Begin training
$ python research/object_detection/train.py --logtostde --train_dir=train \
  --pipeline_config_path=ssd_mobilenet_v1_coco.config
```

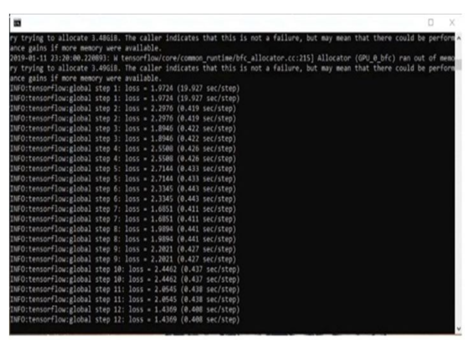


Figure: Training Model

Tensor Board is a web application that provides the essential visual tools and machine learning testing tools for analyzing and comprehending graphs and TensorFlow. The current loss is recorded on the tensor board every five minutes or so.

Tensor board- logdir = training

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

FIG: Monitoring Loss using Tensor Board

G. Exporting Inference Graph

To predict about the training ,after end of the training the following file will be generated. When training is complete, the below file is submitted as a model.

frozen_inference_graph.pb
graph.pbtxt
model.ckpt.data-00000-of-00001
model.ckpt.index
model.ckpt.meta
ssd_mobilenet_v1_coco.pbtxt

H. Running Model on Pi

Perhaps the biggest problem with this project is the low integration capabilities of the Raspberry Pi. High quality detectors work with high accuracy and can handle high frame images. These features are achieved using dedicated multi component Raspberry Pi hardware. These detectors are available on the Raspberry Pi, but the guessing time is much longer than on a regular computer.

The Follow Steps are:
The Update of Raspberry Pi

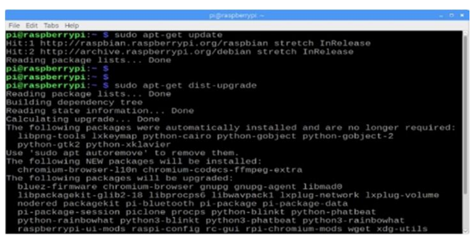


Fig: Raspberry Pi Update

You need to fully update your Raspberry Pi. Some commands are given in below. To type these commands we want open a terminal block.

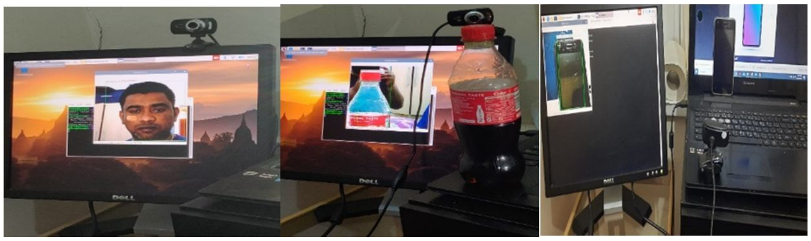
Sudo apt-get update
Sudo apt-get dist-upgrade

I. Install Necessary Libraries

sudo apt-get -y install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev libavcodec-dev libavformat-dev libswscale-dev libv4l-dev libxvidcore-dev libx264-dev
sudo apt-get -y install qt4-dev-tools libatlas-base-dev libhdf5-10
sudo pip3 install opencv-python
sudo pip3 install opencv-contrib-python==4.1.0.25
sudo pip3 install tensorflow
sudo pip3 install keras
sudo pip3 install opencv-python
sudo pip3 install tensorflow

IV. RESULTS

Anomalous Object Detection Class Outputs:



[a] [b] [c]

[a] Person Recognized

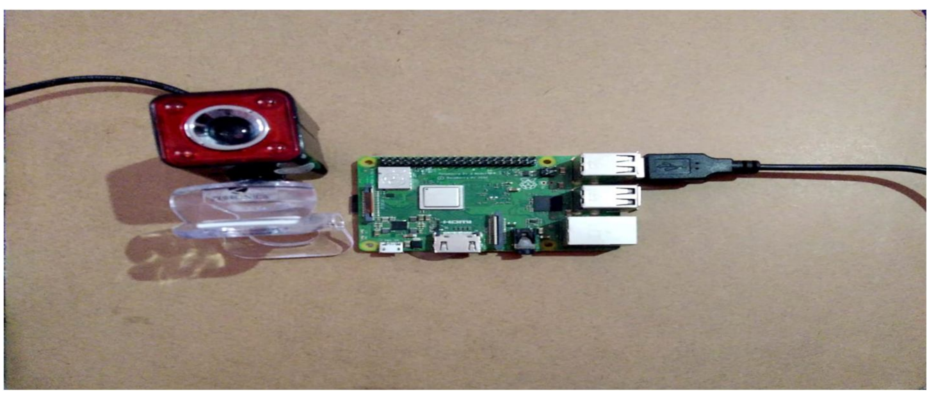
[b] Bottle Recognized

[c] Cell Phone Recognized

In the images, the objects are places in front of the camera and that camera is detected the objects. These detected objects and that object is anomalous or not are display on the computer desktop and also it gives voice indication. In first picture, person is capture and that display on the monitor it is a person and gives sound indication as anomalous. In second picture, bottle is detected and third picture, the phone is detected.

V. CONCLUSION

In this paper, we tried to improve the real time object and anomalous objects capture, and tracking and gives a voice indication system by using Raspberry Pi 4 B Model. An object model and was cognitive database were used. Based on the position and separation of the object, tracking is performed that links the frame-to-frame detection of the USB camera. The proposed framework made it possible to achieve frames of up to 7.9 FPS with 1 element and up to 2.9 FPS with 6 elements. This figure is much higher than obtained using a qualitative model, which is 7x for 1 subject and 2.6x for 6 subjects. Although the proposed structure appears to be effective in most cases, it has its drawbacks. Occlusion issues, inability to distinguish among surrounding objects and wrong positives whenever illumination conditions change are just a few examples. Furthermore, the suggested framework's processing speed is unknown because the processing time is greatly reliant on the price of the content collected. This applies to the more costly versions to a lesser extent.



VI. FUTURE SCOPE

In many programs, we may be interested in finding common ground. The discovery of "backgrounds," such as rivers, walls, mountains, has not yet been addressed by most of the methods mentioned here. Typically, this type of problem is dealt with first by separating the image and then labelling each part of the image. Of course, in order to successfully capture all objects in the scene, and to fully understand the scene, we will need to find the pixel level of objects, and more, a 3D model of such a scene. Therefore, sometimes object acquisition methods and image classification methods may need to be integrated. We are still a long way from gaining a comprehensive understanding of the world, and to achieve this, we may need effective visual perception.

REFERENCES

[1] R. Longbottom, “Roy Longbottom's Raspberry Pi, Pi 2 and Pi 3 Benchmarks,” 5 2017. [Online]. Available: http://www.roylongbottom.org.uk/Raspberry%20Pi%20Benchmarks.htm#anchor24b. [Accessed 3 4 2020].

[2] M. Larabel, “Raspberry Pi 3 Model B+ Benchmarks,” 22 3 2020. [Online]. Available: https://www.phoronix.com/scan.php?page=article&item=raspberrypi-3-bplus&num=1. [Accessed 20 4 2020].

[3] E. Upton, “Raspberry Pi 4Model B+ on sale now at $35,” The Raspberry Pi Foundation, 14 3 2020. [Online]. Available: https://www.raspberrypi.org/blog/raspberry-pi-3-model-bplus-sale-now-35/. [Accessed 20 4 2020].

[4] S. M. Kuo, B. H. Lee and W. Tian, Real-Time Digital Signal Processing: Fundamentals, Implementations and Applications, Chichester: John Wiley & Sons, Incorporated , 2013.

[5] R. B. Fisher, T. P. Breckon, K. Dawson-Howe, A. Fitzgibbon, C. Robertson, E. Trucco and C. K. I. Williams, Dictionary of Computer Vision and Image Processing, Chichester: John Wiley & Sons Ltd, 2014.

[6] A. Krizhevsky, I. Sutskever and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” 2012. [Online]. Available: https://www.cs.toronto.edu/~fritz/absps/imagenet.pdf. [Accessed 7 2 2018].

[7] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, A. Weyand, M. Andreetto and H. Adam, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” 17 4 2017. [Online]. Available: https://arxiv.org/pdf/1704.04861. [Accessed 5 2 2020].

[8] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, “Going deeper with convolutions,” 17 9 2014. [Online]. Available: https://arxiv.org/abs/1409.4842. [Accessed 7 2 2014].

[9] K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition,” 10 12 2015. [Online]. Available: https://arxiv.org/abs/1512.03385. [Accessed 8 2 2020].

[10] R. Girshick, J. Donahue, T. Darrell and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” 22 10 2014. [Online]. Available: https://arxiv.org/abs/1311.2524. [Accessed 13 4 2021].

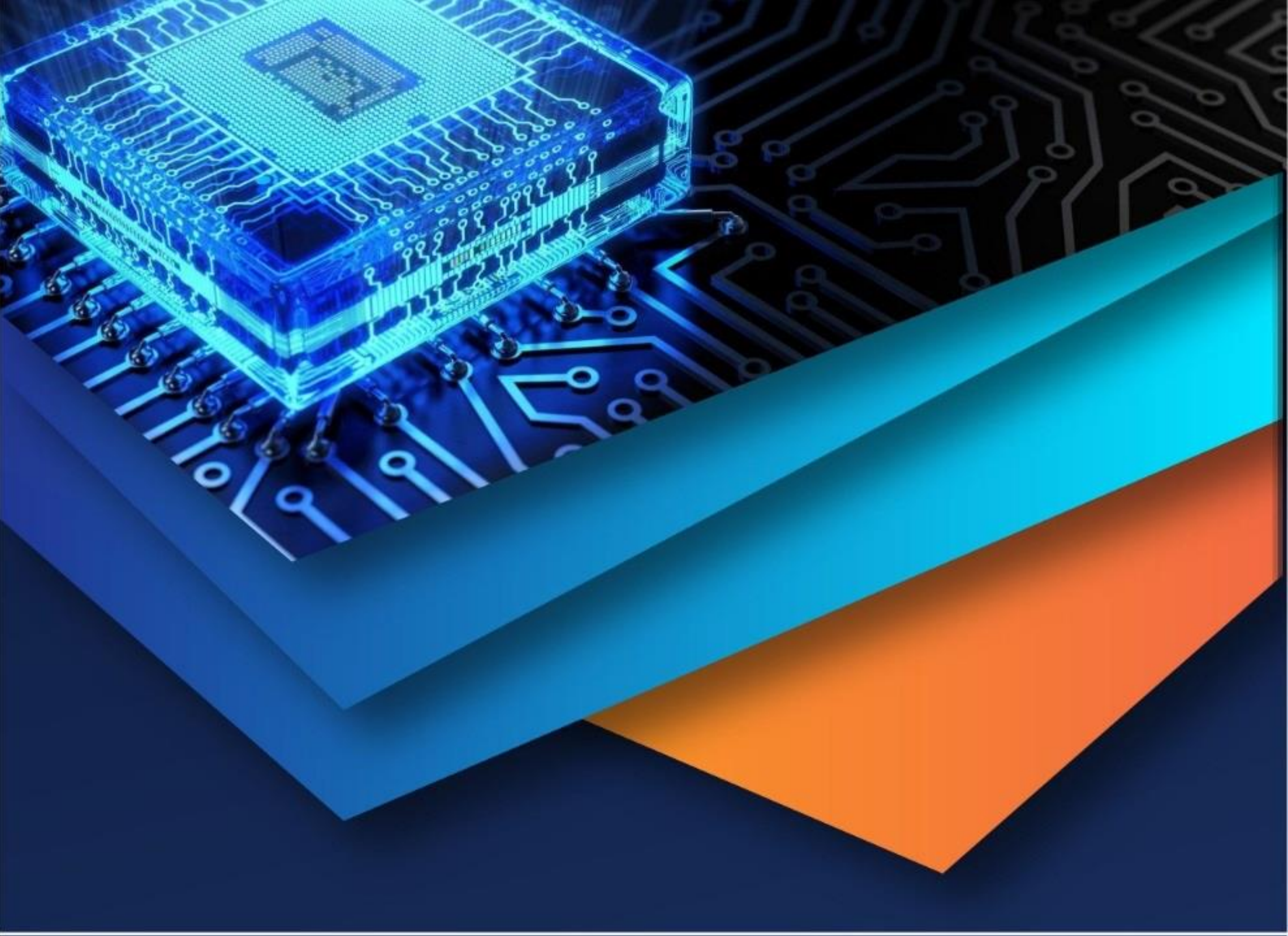
[11] Submitted to University of Glamorgan.

[12] Saikiran Bulusu, Bhavya Kailkhura, Bo Li, Pramod K. Varshney, Dawn Song. “Anomalous Example Detection in Deep Learning: A Survey”, IEEE Access, 2020.

[13] "Recent Advancement in Anomaly Detection in Surveillance Videos", International Journal of Recent Technology and Engineering, 2019.

[14] Ezequiel López-Rubio, Miguel A. Molina-Cabello, Francisco M. Castro, Rafael M. Luque-Baena et al. "Anomalous object detection by active search with PTZ cameras", Expert Systems with Applications, 2021

[15] www.wikipedia.org



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY