DETECTING THE MOVEMENT OF OBJECTS WITH WEBCAM

Mrs. SYED SHAHEEN [1], DEVAGUPTAPU VAISHNAVI DEVI [2], BADDEM USHA SREE [3], PIDUGU JAYA MADHURI [4]

[1] Asst. Professor, Department Of Computer Science And Engineering,
RAGHU INSTITUTE OF TECHNOLOGY(AUTONOMOUS)
Affiliated to JNTU GURAJADA, VIZIANAGARAM
Email: shaheensyed@raghuenggcollege.com

[2] B.Tech, Department Of Computer Science And Engineering - Cyber Security,
RAGHU INSTITUTE OF TECHNOLOGY(AUTONOMOUS)
Affiliated to JNTU GURAJADA, VIZIANAGARAM
Email: v114284d@gmail.com

[3] B.Tech, Department Of Computer Science And Engineering - Cyber Security,
RAGHU INSTITUTE OF TECHNOLOGY(AUTONOMOUS)
Affiliated to JNTU GURAJADA, VIZIANAGARAM
Email: ubaddem@gmail.com

[4] B.Tech, Department Of Computer Science And Engineering - Cyber Security,
RAGHU INSTITUTE OF TECHNOLOGY(AUTONOMOUS)
Affiliated to JNTU GURAJADA, VIZIANAGARAM
Email: jayamadhuripidugu@gmail.com

ABSTRACT

A key technique for improving our comprehension of the development of the threat scenario is the analysis of data sets related to cyber incidents. Numerous investigations need to be conducted because this is a relatively new study issue. In this work, we present a statistical study of a data collection of breach incidents covering 12 years (2005–2017) of malware-related cyberattacks. We demonstrate that, in contrary to the results published in the literature, because hacker breach incidents display autocorrelations, stochastic processes rather than distributions should be used to represent both breach sizes and event inter-arrival periods. Next, specific models of stochastic processes are suggested to suit the breach sizes and the inter-arrival periods, respectively. Furthermore, we demonstrate that the breach sizes and the inter-arrival periods can be predicted by these models. We do both qualitative and quantitative trend studies on the data set to provide deeper insights into the evolution of cyber breach occurrences. We deduce a number of cybersecurity insights, such as the fact that while the frequency of cyberattacks is increasing, the extent of their harm is not.

Keywords: Machine Learning, Support Vector Machine, Django, Masquerader, Cyber Breaches, Scrape data, Interpretation, Authentication, Sequential Query Language, Wamp Server, Regression, Neural Networks, Adding a dataset , classifying using SVM algorithm , Admin access, Analyzing the type of Breaches, Malware Analysis, Un-malware Analysis, Graphical representation of Dataset.

1. INTRODUCTION

Breach scenarios might arise from inappropriate data collection, information loss, or sufficient data leakage. One reason for a data leak might be a program error or lax, non-standard security. Since we can find patterns from these breach instances that happen at specific intervals, our study focuses mostly on identifying and detecting patterns related to cyber hacking breaches. At both the classification and clustering stages, these patterns are identified by utilizing machine learning methods. Given the emphasis on two-way classification and instantaneous trigger action, classification will be chosen over clustering. Because they are easy to comprehend, classification techniques like logistic regression, decision tree learning, support vector machines, and neural networks are frequently used to identify masqueraders or users who are not verified.

To concentrate on the algorithm's efficacy, we keep a vast collection of website logs for our machine learning algorithms to analyze. Focusing on the model's efficiency is necessary since the issue also maps on time-space tradeoff. It is evident that decision tree learning performs poorly over time, but excels when dealing with outliers. Additionally, the threshold value has a significant bias in logistic regression. The system will fail as a whole if the threshold is lost of control. Although quite sophisticated, neural networks need a large amount of data at first.

The SVM's picture classifications and overall accuracy outperform Decision Trees, and its data needs during early stages of analysis are typically not met. Therefore, we would rather support vector machines with kernels for efficient access pattern categorization on open-access websites for scraping or information extraction. In addition to being easier to grasp than neural networks, support vector machine (SVM) models allow us to convert our data with potential outputs by identifying optimal boundaries through the use of a kernel technique.

1.1 Purpose

The following three contributions are ours. Firstly, we demonstrate that stochastic processes, rather than distributions, should be used to describe both the hacking breach incident sizes and interarrival durations, which indicate the frequency of incidents. It has been observed that the inter-arrival periods of hacking breach instances may be well explained by a certain point procedure.

It is our aim that this study would stimulate more research that will provide an in-depth understanding of alternative strategies for risk minimization. Insurance businesses, governmental organizations, and regulators can benefit from these insights as they must have a thorough understanding of the characteristics of data breach risks.

To help enterprises successfully anticipate, avoid, and reduce cybersecurity threats, cyber hacking breaches are modelled and predicted. Organizations may fortify their defences against cyber threats and shield their vital assets and data from illegal access and exploitation by utilizing predictive analytics, threat intelligence, and proactive risk management techniques.

1.2 SCOPE

Cybersecurity researchers must prioritize modelling and forecasting cyber hacking incidents. To foresee and stop future breaches, the current scope of work entails using a variety of methods and tools to evaluate prior breach data, spot patterns and trends, and create prediction models. The current state of modelling and cyber hacking breach prediction is summarized as follows:

Data Collection and Analysis Organizations collect vast amounts of data related to past hacking breaches, including attack vectors, compromised systems, and outcomes. Advanced analytics tools are used to analyze this data and identify patterns, such as common vulnerabilities exploited by hackers, typical attack timelines, and targeted industries or sectors Constant observation and modification Cyber dangers are always changing, thus in order for prediction models to continue to be useful, they must be continuously monitored. To continually improve prediction models based on fresh data and new threats, organizations use a combination of automated monitoring systems, security analytics platforms, and human experience.

Mitigation and Risk Assessment: By estimating the possibility and possible consequences of cyberattacks, predictive models help companies manage risk. Organizations may reduce risks and lessen the effect of possible breaches by developing incident response plans, implementing targeted security controls, and prioritizing security investments based on the results of these evaluations.

Using threat intelligence, automation, behavioral analytics, advanced analytics, machine learning, and continuous monitoring, cyber threat modeling and prediction encompasses real-time anticipation, detection, and mitigation of cyber attacks. Through the implementation of a proactive and data-driven cybersecurity strategy, entities may fortify their barriers and proficiently guard against constantly changing cyber hazards.

1.3 Motivation

To fortify cybersecurity defenses, mitigate risk, cut expenses, guarantee regulatory compliance, improve incident response capabilities, safeguard sensitive data, and preserve business continuity in the face of changing cyberthreats, modeling and forecasting cyberhacking breaches is done.

Organizations may strengthen their defenses against the constantly evolving threat landscape by embracing a proactive and predictive approach to cybersecurity.

The present study is motivated by several questions that have not been investigated until now, such as: Are data breaches caused by cyber attacks increasing, decreasing, or stabilizing? A principled answer to this question will give us a clear insight into the overall situation of cyber threats. This question was not answered by previous studies.

More research that provides in-depth understanding of alternative risk reduction strategies is what we hope this study will spur. Since insurance firms, governmental organizations, and regulators must have a thorough understanding of the nature of data breach risks, these insights are helpful.

1.4 Fundamental concepts Techniques

Data Collection: Gathering relevant data is crucial for modeling cyber hacking breaches. This may include network traffic logs, system event logs, user authentication logs, firewall logs, etc. The data should cover a sufficient time period and include both normal and anomalous activities.

Malware: Malware attacks involve malicious software designed to disrupt, damage, or gain unauthorized access to computer systems or networks. Types of malware include viruses, worms, Trojans, ransomware, spyware, adware, and rootkits.

Phishing: Phishing attacks involve sending deceptive emails, messages, or websites that impersonate legitimate entities to trick recipients into providing sensitive information, such as passwords, credit card numbers, or personal data.

Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) Attacks: DoS and DDoS attacks aim to disrupt the availability of online services by overwhelming target systems or networks with a flood of traffic, rendering them inaccessible to legitimate users.

Man-in-the-Middle (MitM) Attacks: MitM attacks involve intercepting and altering communication between two parties without their knowledge or consent. Attackers can eavesdrop on sensitive information or manipulate data exchanged between the parties.

SQL Injection: SQL injection attacks exploit vulnerabilities in web applications' input fields to inject malicious SQL queries into the underlying database, allowing attackers to extract, modify, or delete data stored in the database.

Cross-Site Scripting (XSS): XSS attacks involve injecting malicious scripts into web pages viewed by other users, typically through vulnerable web applications. These scripts can steal session cookies, redirect users to malicious websites, or deface web pages.

Network Security Measures:

- Use firewalls to monitor and control incoming and outgoing network traffic, blocking unauthorized access and malicious content.
- Implement intrusion detection and prevention systems (IDPS) to detect and respond to suspicious activities and potential cyber threats in real-time.
- Employ virtual private networks (VPNs) to encrypt data transmitted over public networks, protecting sensitive information from interception by unauthorized parties.
- Segment networks and apply access controls to limit the exposure of critical assets and minimize the impact of a potential breach.

Endpoint Security Solutions:

Install and regularly update antivirus, anti-malware, and anti-spyware software on all endpoints, including desktops, laptops, servers, and mobile devices, to detect and remove malicious software.

Use endpoint detection and response (EDR) solutions to monitor and analyze endpoint activities for signs of compromise, anomalous behaviour, or potential security incidents.

Patch Management:

Regularly update and patch operating systems, applications, and firmware to address known vulnerabilities and security weaknesses, reducing the risk of exploitation by cyber attackers.

Establish a formal patch management process to prioritize and deploy patches promptly, ensuring timely protection against emerging threats and vulnerabilities.

Secure Configuration and Access Controls:

Implement strong password policies, enforce multi-factor authentication (MFA), and use role-based access controls (RBAC) to limit privileges and restrict access to sensitive systems and data.

Configure devices, applications, and services securely by disabling unnecessary features, ports, and protocols, and enabling encryption, logging, and auditing mechanisms to enhance security posture.

Data Encryption and Backup:

Encrypt sensitive data at rest and in transit using strong encryption algorithms and cryptographic protocols to protect against unauthorized access and data breaches.

Regularly back up critical data to secure and offsite locations, ensuring data availability and integrity in the event of a ransomware attack, data corruption, or system failure.

Security Awareness Training:

Educate employees, contractors, and users about common cyber threats, social engineering tactics, and best practices for cybersecurity hygiene through regular training sessions, awareness campaigns, and simulated phishing exercises.

Foster a culture of cybersecurity awareness and accountability, encouraging employees to report suspicious activities, adhere to security policies, and follow incident response procedures.

2. LITERATURE SURVEY

Akanksha Soni et al. (Soni, 2020) developed a model that detects whether a person is wearing a helmet in real time thereby, detecting any violations. This project was also implemented with the help of TensorFlow, Keras and OpenCV. Their proposed model showed major improvements when compared to some previous models that gave wrong predictions whenever a rider wears clothes over their face. They achieved an overall accuracy of 98% when tested. S Chen et al. (Chen, 2020) implemented a model with the help of TensorFlow to identify ID card numbers. With the help of OpenCV the image of an ID card is preprocessed and the number on the ID card is recognized and given as output with the help of a trained CNN model. When tested it was observed that training speed is fast and the accuracy is high. Emily Caveness et al. (Caveness, 2020) developed TensorFlow Data Validation (TFDV) which offers a scalable solution for data analysis and validation for machine learning. It is deployed in production which is integrated with TensorFlow Extended (TFX), which is an end-to-end ML platform. Their system has gained a lot of traction ever since they open sourced their project. Other open-source data validation systems such as Apache Spark were also heavily inspired from their project. Apache Spark packs with built-in modules for streaming and has a fast, easy to use system for big data processing.(Nair, 2018) Yonghui Lu et al. (Lu, 2020) proposed an efficient YOLO Architecture, YOLO-compact for a real time single category detection. As we know in most practical applications, the number of categories in object detection is always single and the authors aimed to make detections faster and more efficient for these scenarios. By performing a series of experiments, the authors were able to come up with an efficient and compact network with the help of YOLOv3. It was observed that YOLO-compact is only of 9MB size, about 26 times smaller than YOLOv3, 6.7 times smaller than tiny-yolov2 and 3.7 times smaller than tiny-yolov3. The average precision of YOLO-compact is 86.85% which is significantly higher than other YOLO models. M. B. Ullah (Ullah, 2020), proposed a CPU-based YOLO object detection model that is intended to run on nonGPU computers. In the proposed method, the author optimized YOLO with OpenCV in a way that real time object detection can be much faster on CPU based computers. Their network architecture comprises 2 Convolutional layers each followed by pooling layers and 3 fully connected layers. Their model detects objects from videos in 10.12 to 16.29 FPS with 80-99% confidence in CPU-based computers. J.Redmon et al.(Redmon,2016),introduced YOLO, a unified model for object detection. This model is simple to construct and can be trained directly on full images. Unlike classifier-based approaches, YOLO is trained on a loss function that directly corresponds to detection performance and the entire model is trained jointly. Fast YOLO is the fastest general-purpose object detector in the literature and YOLO pushes the state-of-the-art in

real-time object detection. YOLO also generalizes well to new domains making it ideal for applications that rely on fast, robust object detection. Daniele Gratterola et al.(Daniele,2021), presented Spektral, an open-source Python library for building graph neural networks with TensorFlow and the Keras application programming interface. Spektral implements a large set of methods for deep learning on graphs, including message-passing and pooling operators, as well as utilities for processing graphs and loading popular benchmark datasets. The purpose of this library is to provide the essential building blocks for creating graph neural networks, focusing on the guiding principles of user-friendliness and quick prototyping on which Keras is based. Spektral is, therefore, suitable for absolute beginners and expert deep learning practitioners alike

3. EXISTING SYSTEM

Many academics employ the well-known machine learning technology TensorFlow with Keras for data classification. Dataflow graphs serve as the representation of computation in TensorFlow. A complex calculation can be carried out relatively easily by describing it as a graph and effectively transferring the graph's component components to a machine in the form of a cluster using Google's framework TensorFlow. This real-time object identification system, called You Only Live Once (YOLO), is really good. The class probabilities are determined for each of the bounding boxes while object detection is taken into account as a regression problem. Over the past century, technology has made enormous strides, from the Internet of Things (IoT) to machine learning and deep learning. CNN is utilised in numerousMachine learning has become a well-known application area in numerous industries, including medical, marine science, and many more. Implementing object detection techniques using TensorFlow, Keras, and YOLO results in increased accuracy, robustness, and detection speed.

4. PROPOSED SYSTEM

After vectorization and flattening, our matrix is transmitted to the fully connected layer. When all the neurons in a layer are related to all the neurons in the layer above, the layer is said to be fully linked. Spatial information is lost in the completely linked layers. Next to the last fully linked layer is the output layer. Once all the neurons are joined, the whole neural network is seen. Because it generates an accurate probability distribution of the outputs to categorize as dog, cat, vehicle, truck, etc., softmax regression is often used for classification issues. TensorFlow computes using dataflow graphs as its representation.

4.1 Proposed Algorithm

TENSORFLOW AND KERAS

Tensorflow and Keras are algorithms that work towards object detection-Algorithms based on classification, and Algorithms based on regression. YOLO falls under TensorFlow an open-source platform that is used for Machine Learning, created by the Google Brain team. It is explicitly used for complex numerical computation, which packs together a bunch of machine learning and deep learning models and algorithms. It can be used for a variety of applications such as classifying handwritten digits, object detection, image recognition, and natural language processing (Natraj, 2019) by training and running deep neural networks. Application Development for Mask Detection and Social Distancing Violation Detection using Convolutional Neural Networks Keras which acts as an interface for TensorFlow is an open - source library that provides an efficient way of implementing neural networks. It consists of useful functions such as activation functions, and optimizers.

How Does TensorFlow Work?

With the help of TensorFlow, developers can create dataflow graphs which are structures that show how data passes through the graph, or a series of nodes. Think of each node as a mathematical operation and each edge representing a multidimensional data array or a tensor. This can be easily implemented in python where these nodes and tensors act as objects. However, the mathematical operations are

performed in C++ binaries which shows an optimal performance. Python takes care of directing the traffic and combines them to work together as a unit. TensorFlow can be run on multiple platforms such as in a cloud, a local machine, CPUs or GPUs, iOS, and Android devices. It can also be run on Google's custom TensorFlow Processing Unit (TPUs). Th e trained models can be run on any system for predicting results. TensorFlow 2.0 which was released in October 2019 made many significant changes from user feedback. It works more efficiently and is more convenient with simple Keras API for training models and better performance. With the help of TensorFlow Lite,it is possible to train models on a wide variety of devices.

YOLO OBJECT DETECTION

You Only Look Once (YOLO) is an effective real time object detection system. It considers object detection as a regression problem and finds the class probabilities for each of the bounding boxes. In one evaluation the neural network predicts bounding boxes and class probabilities from the image, hence the name YOLO. The base model detects images at an astonishing speed of 45 frames per second whereas a smaller version called Fast YOLO detects at 155 frames per second. It performs better than other detection methods such as Deformable Part Models (DPM) and Region-based convolutional neural networks (R-CNN). There are two types of latter category.

4.2 INPUT AND OUTPUT DESIGN

4.2.1INPUT DESIGN

The information system and the user are connected through the input design. In order to transform transaction data into a form that can be processed, it entails developing specifications and procedures for data preparation. These can be completed by having people key the data directly into the system or by inspecting the computer to read data from a written or printed document. The input design is all about minimizing mistakes, limiting the amount of input needed, preventing delays, cutting out unnecessary processes, and simplifying the process. The input has been thoughtfully crafted to maintain privacy while offering security and usability. The following was taken into account by input design:

- What data should be given as input?
- How should the data be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur

OBJECTIVES

1. Contribution The process of turning an input description that is user-oriented into a computer-based system is called design. The purpose of this design is to prevent mistakes in the data input process and to provide management with the proper guidance for obtaining accurate information from the computerized system.

2. To manage massive amounts of data, it is accomplished by designing user-friendly displays for data entry. Easier data entry and error-free design are the objectives of input design. Everything can be done with the data thanks to the way the data entering page is made. Record viewing facilities are also offered.

3. The validity of the data will be checked when it is input. Screens are useful for entering data. In order to prevent the user from becoming stuck in a state of confusion, pertinent messages are sent when necessary. Making an input layout that is simple to understand is, thus, the goal of input design.

4.2.2 OUTPUT DESIGN

An output that satisfies end user criteria and displays information in an understandable manner is considered high quality. Through outputs, processing findings are shared with users and other systems in any given system. In output design, the location of the information for both the hard copy output and the immediate requirement

is specified. It is the user's primary and most direct source of information. The system's ability to support user decision-making is enhanced by clever and efficient output design.

1. When designing computer output, one should follow a methodical and well-planned process to ensure that the desired output is produced and that every output component is made in a way that makes the system easy and efficient for users to use. The precise output required to satisfy the criteria should be identified by analyzing design computer output.

2. Select methods for presenting information.

3. Create documents, reports, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

i. Convey information about past activities, current status or projections of the

ii. Future.

iii. Signal important events, opportunities, problems, or warnings.

iv. Trigger an action.

v. Confirm an action.

4.2 ARCHITECTURE DIAGRAM

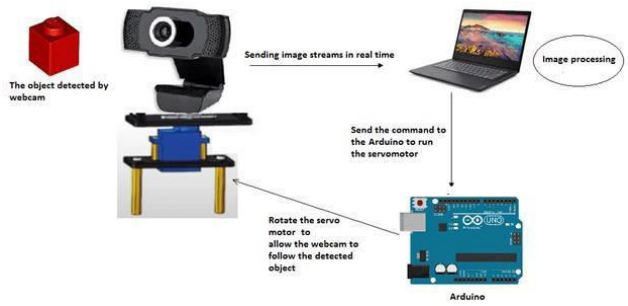

Figure 4.3.1 Architecture Diagram

4. RESULTS

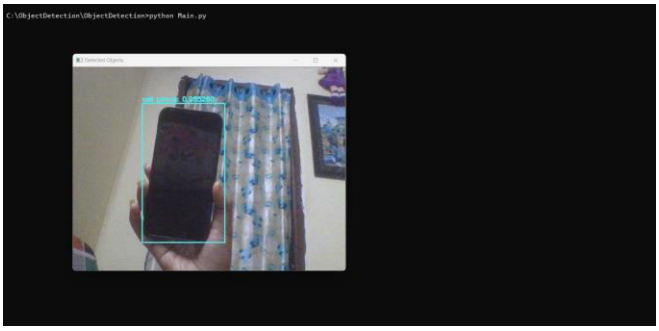

Figure 5.1: The above figure shows detecting the movement of objects with webcam

5. CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

Deep learning-based object identification techniques have received a lot of interest lately from a variety of academic disciplines. While we have made progress in optimizing the algorithms for object detection tasks, there is still potential for improvement in terms of more accurate identification of both single and multiple objects. Thus, a suggested model based on the deep learning-based ImageAI deep learning library's YOLOv3-ResNet detection module is presented in this research. The integration of ResNet architecture with Darknet-53 resulted in an algorithm that can effectively extract features. The experimental findings show that our model obtained an accuracy of 68%, which is an improvement above traditional object detection techniques.

5.2 FUTURE SCOPE

Detecting movement of objects using a webcam has a wide range of potential future applications across various industries and domains. Here are some potential future scopes:

Security and Surveillance: Enhanced security and surveillance systems can be developed using webcam-based object movement detection. This could include applications in home security, office security, public spaces monitoring, and perimeter security for sensitive installations.

Automotive Safety: Integration of webcam-based object movement detection into automotive safety systems can contribute to improving driver assistance systems, collision avoidance, and pedestrian detection in autonomous vehicles.

Healthcare Monitoring: Webcam-based movement detection can be utilized for monitoring patients in healthcare facilities, assisting with fall

detection for the elderly, tracking movements during physical therapy sessions, and even monitoring vital signs remotely.

Retail Analytics: Retailers can use webcam-based movement detection for foot traffic analysis, customer behavior analysis, and optimizing store layouts for better customer engagement and product placement.

Gesture Recognition: By analyzing movements detected by a webcam, gesture recognition systems can be developed for human-computer interaction, virtual reality, and gaming applications.

Environmental Monitoring: Webcams can be deployed in natural environments to monitor wildlife movements, track changes in vegetation, and observe environmental phenomena such as landslides or river flow.

Smart Home Automation: Incorporating webcam-based movement detection into smart home systems can enable functions like automatically turning on lights when movement is detected, adjusting heating and cooling based on occupancy, or even monitoring pets while homeowners are away.

Education and Training: Webcam-based movement detection can be used in educational settings for tracking student engagement, assessing physical activities during online physical education classes, or developing interactive learning experiences.

Industrial Automation: In manufacturing and logistics, webcam-based movement detection can aid in automating processes such as quality control, inventory management, and package sorting.

Agriculture: In agriculture, webcam-based movement detection can be used for monitoring crop growth, detecting pests or diseases, and optimizing irrigation and fertilization processes.

6. REFERENCES

[1] Xia, Z. (2019). An Overview of Deep Learning. Deep Learning in Object Detection and Recognition, 1-18. do:10.1007/978-981-10-5152-4 1

[2] Leung, H., & Haykin, S. (1991). The complex backpropagation algorithm. IEEE Transactions on Signal Processing, 39(9), 2101-2104. doi: 10.1109/78.134446

[3] Real-Time Object Detection for Aiding Visually Impaired using DeepLearning. (2020). International Journal of Engineering and Advanced TechnologyRegularIssue,9(4),1600-1605. doi: 10.35940/ijeat.d8374.049420

[4] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference On Computer Vision and Pattern Recognition (CVPR). doi: 10.1109/evpr.2016.91

[5] Budiharto, W., Gunawan, A. A., Suroso, J. S., Chowanda, A., Patrik, A., & Utama, G. (2018). Fast Object Detection for Quadcopter Drone Using Deep Learning. 2018 3rd International Conference on Computer and Communication Systems (ICCCS). doi: 10.1109/ccoms.2018.8463284

[6] Li, X., Wang, J., Xu, F., & Song, J. (2019). Improvement of YOLOv3 Algorithm in Workpiece Detection. 2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER). doi: 10.1109/cyber46603.2019.9066490

[7] Zhao, L., & Wan, Y. (2019). A New Deep Learning Architecture for Person Detection. 2019 IEEE 5th International Conference on Computer and Communications (ICCC). doi: 10.1109/iccc47050.2019.9064172

[8] Lu, Z., Lu, J., Ge, Q., & Zhan, T. (2019). Multi-object Detection Method based on YOLO and ResNet Hybrid Networks. 2019 IEEE 4th International Conference on Advanced Robotics and Mechatronics (ICARM). doi:10.1109/icarm.2019.8833671

[9] Zhang, D. (2018). Vehicle target detection methods based on color fusion deformable part model. EURASIP Journal on Wireless Communications and Networking, 2018(1). doi: 10.1186/s13638-018-1111-8

[10] Kang, M., Leng, X., Lin, Z., & Ji, K. (2017). A modified faster R-CNN based on CFAR algorithm for SAR ship detection. 2017 International Workshop on Remote Sensing with Intelligent Processing (RSIP). doi: 10.1109/rsip.2017.7958815

[11] Hung, P. D., & Kien, N. N. (2019). SSD-Mobilenet Implementation for Classifying Fish Species. Advances in Intelligent Systems and Computing Intelligent Computing and

Optimization, 399-408. doi: 10.1007/978-3-030-33585-4 40
[12] Castiblanco, C., Rodriguez, J., Mondragon, I., Parra, C., & Colorado, J.(2014). Air Drones for Explosive Landmines Detection. ROBOT2013: First Iberian Robotics Conference Advances in Intelligent Systems and Computing, 107-114. do: 10.1007/978-3-319-03653-3_9
[13] Ma, H., Liu, Y., Ren, Y., & Yu, J. (2019). Detection of Collapsed Buildings in Post-Earthquake Remote Sensing Images Based on the Improved YOLOv3. Remote Sensing, 12(1), 44. doi: 10.3390/rs 12010044
[14] Khong, L. M., Gale, T. J., Jiang, D., Olivier, J. C., & Ortiz-Catalan, M.(2013). Multi-layer perceptron training algorithms for pattern recognition of myoelectric signals. The 6th 2013 Biomedical Engineering International Conference. do: 10.1109/bmeicon.2013.6687665
[15] Andre, T., Neuhold, D., & Bettstetter, C. (2014). Coordinated multi-robot exploration: Out of the box packages for ROS. 2014 IEEE Globecom Workshops (GC Wkshps). doi: 10.1109/glocomw.2014.7063639