

A deep learning-based surveillance system for enhancing public safety through internet of things and digital technology using Raspberry Pi

Shreedevi Kareppa Sanapannavar, Chayadevi Mysore Lakshmanagowda, Geetha Sundararajan

Department of Computer Science and Engineering, BNM Institute of Technology, Bengaluru, India

Article Info

Article history:

Received Jun 5, 2024

Revised Jul 31, 2024

Accepted Aug 6, 2024

Keywords:

Cloud storage

Elliptic curve digital signature algorithm

Internet of things

Public spaces

VGG16

ABSTRACT

In public spaces, individuals encounter challenges due to the prevalence of malicious activities like theft and kidnapping. As the internet of things (IoT) and digital technology continue to expand rapidly, efforts to create safe environments are becoming increasingly sophisticated. To address these security concerns, a proposed solution involves the utilization of video-capturing technology with the help of a Raspberry Pi web camera. Videos of the surroundings are recorded, a digital signature algorithm is applied to protect the videos, and they are then transmitted to authorized individuals who use them for forensic analysis. This process allows for the identification and investigation of any suspicious or criminal activities. The captured video data is compared with a standard dataset using a deep learning process. By analyzing the content of the videos and identifying the potential threat objects, we can allow for prompt intervention or further investigation by relevant authorities.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Shreedevi Kareppa Sanapannavar

Department of Computer Science and Engineering, BNM Institute of Technology

27th cross, 12th Main, BSK II stage, Bengaluru 560070, India

Email: shreedevivr@bnmit.in

1. INTRODUCTION

This work utilizes the internet of things (IoT), which denotes to a network of interconnected devices capable of communicating and exchanging data across the Internet without human intervention. The range of such devices can extend from simple sensors and actuators to complex industrial machinery and consumer electronics. Cloud storage is a service that allows you to store data securely on remote servers accessed via the internet, rather than storing it on your local hard drive or other physical storage devices. When utilizing cloud storage, your data is stored on servers maintained by a third-party provider, which are often located in data centers around the world. The elliptic curve digital signature algorithm (ECDSA) is a cryptographic approach employed to create digital signatures, relying on elliptic curve cryptography. It is a variation of the digital signature algorithm (DSA) standardized by NIST. ECDSA offers a high level of security with relatively smaller key sizes compared to other traditional digital signature schemes. The core of ECDSA relies on the properties of elliptic curves over finite fields. An elliptic curve is defined by,

$$y^2 = x^3 + ax + b$$

where a and b are constants, and the curve is defined within a finite field. ECDSA functions within specific elliptic curve domain parameters, which encompass the curve equation, base point, order, and other related parameters.

VGG16 is a deep convolutional neural network architecture proposed by the visual geometry group (VGG) at the University of Oxford. It finds extensive application in image classification and related tasks in computer vision. The “16” in VGG16 denotes the number of weight layers in the network, excluding the pooling and activation layers. It is characterized by its simplicity, using small 3×3 filters and deeper networks. The architecture has no complex elements like residual connections or inception modules, making it straightforward to understand and implement. It is computationally intensive compared to more modern architectures like residual network (ResNet) and Inception due to its deeper design. This article proposes a system that establishes a confident and secure public space within a smart city.

Our research aims to create an intelligent surveillance security system capable of detecting weapons, specifically guns or knives. In pursuit of this objective, we utilized diverse computer vision techniques and deep learning algorithms to identify weapons from captured images. Recent advancements in machine-learning and deep learning, particularly with models like VGG16, have shown significant progress in detecting objects and recognition, especially in images. Object identification and classification are crucial initial steps in any video surveillance system, as they pave the way for further object-tracking tasks. To accomplish this, we trained a classifier model using the COCO, *i.e.*, “Common objects in context” dataset.

This paper aims to create a system capable of swiftly detecting guns and knives while utilizing minimal computational resources. With the rapid advancement of technology, it has become increasingly crucial to detect any potential threats in public areas. Therefore, our proposed system has the potential to be effectively integrated into the surveillance system. The proposed security measures to promptly identify firearms or edged weapons are shown in Figure 1.

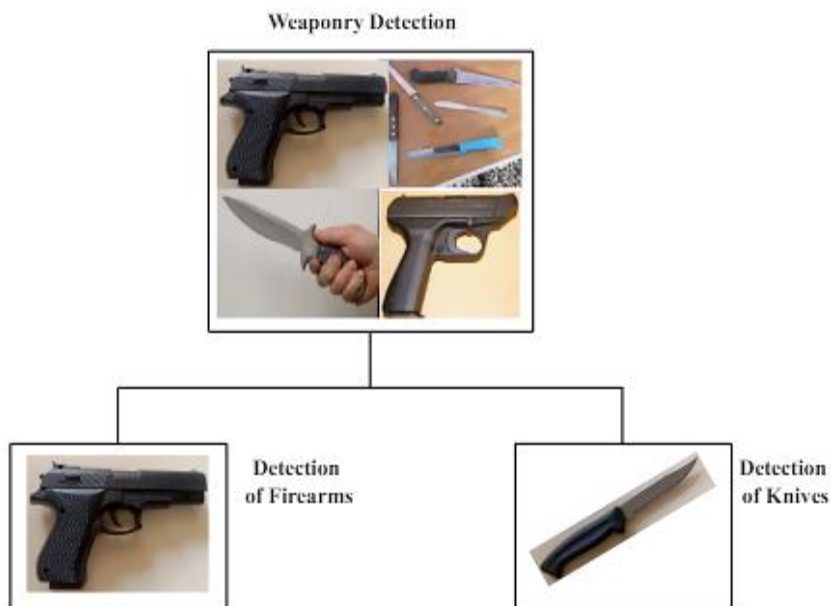


Figure 1. Weapons detection types

2. LITERATURE SURVEY

Parking scarcity in urban areas, including university campuses, necessitates a smart parking system. Jabbar *et al.* [1] introduces an IoT-based Raspberry Pi parking management system (IoT-PiPMS) that efficiently matches drivers with vacant spots, reduces congestion and saves costs. The system monitors parking occupancy, guides drivers via a smartphone app, and updates data in real time. It enhances accessibility and user experience, with forthcoming intentions for expansion and improved privacy measures. Studies [2] presents a novel unified method for automatically detecting vehicles and license plates in urban surveillance systems. Our approach identifies high-energy frequency areas in digital camera images, reducing data volume. We introduce a novel filter for this purpose and show its effectiveness in IoT and smart city applications. Technology has revolutionized our world, including home security [3]. We propose a Raspberry Pi-based home security system using hardware like Raspberry Pi, a camera, a touch screen, and an Android mobile. Our system, a smart mirror, serves both informational and security purposes, responding to touch and mobile commands. It sends alert messages with clear intruder photos to the owner's mobile upon detection. Trained with you only look once (YOLO) and Haar techniques, our system achieves 96% accuracy in

detecting human intruders. It can potentially authenticate users by comparing captured faces with stored photos, but this presents challenges in clear image capture during authentication. As urban parking challenges [4] grow with increasing vehicle numbers, this article introduces a dual-lens millimeter wave (MMW) radar antenna for IoT-based smart parking systems. It features a flat dielectric punch lens to boost transmitting antenna gain and a dielectric rod lens for wide beamwidth and stable performance. The combined dual-lens design enhances radar accuracy and stability at 24 GHz. Transmitting antenna gain measures 15.8 dBi, receiving antenna gain 7.9 dBi, with a 3 dB beamwidth of 65 degrees. The antenna offers stable performance, suitable for MMW radar smart parking systems, and is cost-effective compared to array-based solutions.

Cloud computing is mainstream but with urbanization, there is more surveillance data [5]. Edge computing offers a solution. Our study investigates using it for smart parking surveillance. We achieved over 95% accuracy in real-world tests. This system could be crucial for smart cities and transportation. Our paper describes the development and testing of this smart parking surveillance system, which is among the first to use edge computing in real-world parking surveillance. We achieved 95.6% accuracy in diverse conditions, offering advantages over existing systems and potential for smart cities and transportation. Kumbhar [6] introduces an IoT-based security system for residences utilizing Raspberry Pi 3, Pi camera, and passive infrared (PIR) sensor. It sends email alerts to homeowners and members upon detecting intrusion, and can also activate a security alarm. People have the ability to monitor remotely their residences and receive alerts on their devices, preventing unauthorized access. The internet of things (IoT) merges devices into networks for advanced services, but privacy and security are critical [7]. This paper explores IoT attack models and ML-based security solutions, focusing on authentication, access control, malware detection, and secure offloading. Challenges remain in implementing these techniques in real-world IoT systems. Kieu-Do-Nguyen *et al.* [8] introduces a multi-functional elliptic curve cryptography (ECC) hardware design supporting ECDSA and Edwards-curve digital signature algorithm (EdDSA) algorithms. It is efficient, low-cost, and does not rely on digital signal processor (DSP). The core runs up to 112.2 MHz with Virtex-7 devices, using only 10,259 slices. It accommodates multiple ECC algorithms (Ed25519, ECDSA with NIST P-256, P-384, and P-521) in one design, enabling its compatibility with diverse hardware platforms without necessitating redesign.

Puthiyidam *et al.* [9] suggests a timestamp-based approach to improve ECDSA for IoT device authentication. It prevents signature reuse and fake signature generation by generating unique signatures each time. This enhances security and efficiency, outperforming most ECDSA variants. Our experiments confirm its effectiveness with minimal overhead. Future work includes improving execution efficiency and addressing quantum computing security. He *et al.* [10] proposes a low-latency ECDSA architecture, significantly reducing double point multiplication (DPM) calculation to five clock cycles per loop. Our design outperforms existing ones, achieving DPM times of 3.584, 5.656, and 7.453 μ s over GF(2163), GF(2233), and GF(2283), respectively, with improved throughput efficiency.

Smart homes improve living standards but face data security challenges [11]. We propose a blockchain-based multi-cloud storage model and an efficient identity-based proxy aggregate signature (IBPAS) scheme to enhance security and conserve resources. Our experiments demonstrate notable enhancements in communication expenses and storage effectiveness when juxtaposed with conventional methods. This methodology guarantees the integrity and dependability of data in smart home environments. Rajashree *et al.* [12] introduces a Raspberry Pi-based security system for entrance visitor validation, utilizing ECDSA with Wi-Fi. It captures, timestamps, and authenticates visitor images, sending them for verification via email on Windows or Linux. Components include Raspberry Pi 3, Raspberry Pi Camera, MIRACL software library, a tactile switch, and a breadboard.

The latest trends like IoT, smart cities, and digital transformation are fueling huge data growth, making cloud storage crucial [13]. Yet, security concerns persist. Our paper reviews cloud storage security comprehensively, covering challenges, encryption methods, and open research topics. Weeds are a substantial hurdle for corn production. Yang *et al.* [14] proposed SE-VGG16, a deep learning model, to accurately identify weeds in corn fields. SE-VGG16 improves on VGG16 by incorporating the SE attention mechanism, reducing convolutional kernel size, and using Leaky rectified linear unit (Leaky ReLU) activation functions. It achieves 99.67% accuracy, outperforming other models. The studies [15], [16] assessed Timoho leaf extract (TLE) as a corrosion inhibitor for 304 stainless steel (304SS) in 0.5 M H₂SO₄. TLE, effective at 0–6g/L, reduced corrosion by 99.37% at 3g/L. Mixed adsorption was identified as the inhibition mechanism. Machine learning (CNN-VGG16) predicted corrosion with 96% accuracy. Phenol in TLE played an essential function in adsorption on 304SS. Quantum chemistry improved comprehension of the adsorption process. Advancements in sensor tech and broadband networks are driving the emergence of the internet of everything (IoE) and mobile IoT [17]. However, mobile IoT networks face challenges from complex communication environments and energy-intensive data processing. In response to this, we suggest an improved CNN-based algorithm for predicting outage probability (OP) in mobile IoT networks, integrating transmit antenna selection (TAS) and cooperative schemes.

Object detection is crucial in autonomous driving, utilizing algorithms like YOLOv3 on datasets like COCO to identify objects accurately. This technology is essential for safe driving, as seen in innovations like Tesla's self-driving cars, which aim to reduce pollution and traffic [18]. Object detection ensures these vehicles can accurately perceive their surroundings, detecting vehicles, pedestrians, and traffic signs to prevent accidents and ensure safe travel. This paper offers a comprehensive survey of state-of-the-art weapon detection methods, focusing on specific challenges and applications [19]. In response to increasing crime, Kaya *et al.* [20] introduces a superior deep learning model, achieving 98.40% accuracy in detecting seven weapon types. Tested on diverse backgrounds, it enhances security efficacy and inspires further research in real-time monitoring and concealed gun detection. Nagappan and Rani [21] enhances security through real-time weapon detection using closed-circuit television (CCTV) [22], [23]. By employing advanced algorithms like YOLOv5, we achieved a high F1-score of 91%, ensuring safety and attracting investment. Global gun violence prompts action [24] YOLOv3 system detects firearms efficiently, enhancing surveillance. Outperforming traditional methods, it offers cost-effective solutions, revolutionizing safety through robotics. Our combined [25] DeepEAST and Keras optical character recognition (OCR) model enhances text detection and recognition in videos. While effective on datasets like ICDAR 2015, improvements are needed, particularly for RoadText-1K. Subsequent efforts aim to integrate pre-trained models to enhance performance across diverse conditions. Bhatti *et al.* [26] refers to testing various algorithms including VGG16, Inception-V3, and Inception-ResNetV2. Chayadevi *et al.* [27] created an automated security system for automated teller machines (ATMs), using image processing. It recognizes faces with Viola-Jones and detects weapons with support vector machine (SVM) and random forests. TensorFlow drives the machine learning. An Android app alerts authorities, and there is a speech alert for the visually impaired.

3. SYSTEM ARCHITECTURE

This section outlines the methodology used, research materials employed, system design, and fabrication process. The idea explored in this paper involves combining IoT with the idea of smart public spaces. Our main focus is to ensure the system operates effectively in both the prototype and real-world environments. This comprehensive system ensures both the security of the captured footage and proactive measures in response to potential threats, which is as shown in Figure 2.

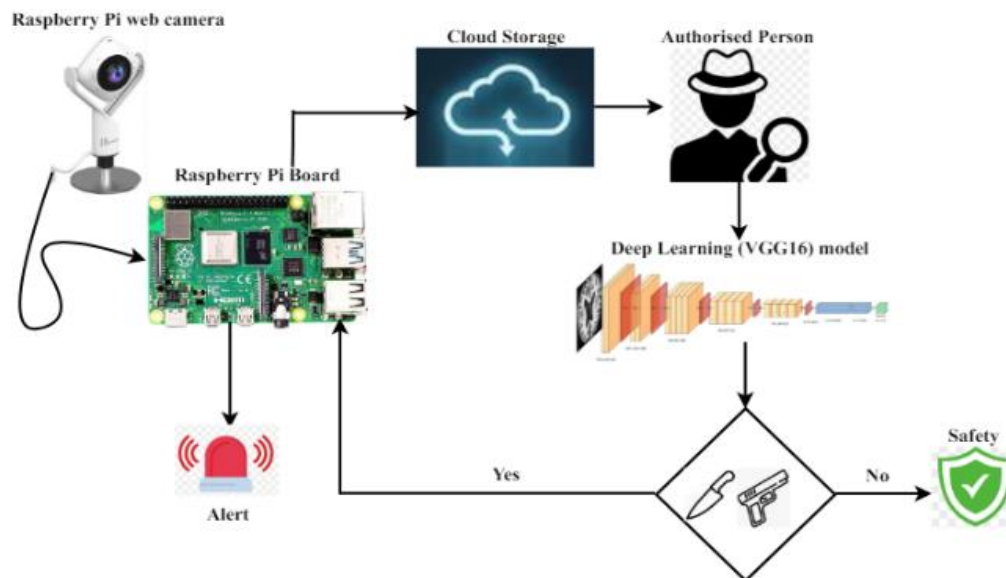


Figure 2. Raspberry Pi based smart public spaces management system architecture

3.1. Selection of components

The materials and components are selected based on needs and compatibility, determined through a thorough literature review. During this review, we examined various design variations and components used, as well as their functions in smart public premises systems. Hardware and software choices are made during the component selection phase. The selected hardware components include Raspberry Pi 3 model V1.3, Raspberry Pi web camera with USB cable, 128-bit Micro SD card and LED lights. Software tools chosen

include the Raspberry Pi operating system, Python, multi-precision integer rational arithmetic crypto library (MIRACL), and others. After completing the component selection, the system architecture is established. This subsection offers a summary of the chosen components. The interconnection of the components is depicted in Figure 3.



Figure 3. Connection between Raspberry Pi and Pi web camera

3.1.1. Raspberry Pi 3 Model V1.3

The Raspberry Pi 3 model V1.3 was chosen for this project primarily for its robust processing capabilities as a single-board embedded computer. It excels in connecting to the internet, offering options via either an Ethernet port or wirelessly through Wi-Fi or Bluetooth. Its ease of internet connectivity stands out as one of Raspberry Pi's advantages over Arduino. Additionally, Raspberry Pi supports a diverse range of programming languages, including Python, Java, C, C++, Perl, Ruby, and BASIC. In contrast, Arduino is limited to Arduino or C/C++. It has Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @1.4 GHz 1 GB LPDDR2 SDRAM. The Pi web camera needs substantial processing power to capture and process videos effectively. Therefore, the Raspberry Pi 3 is typically better suited for our system and provides better performance for video capturing and streaming.

3.1.2. Pi web camera

The Pi web camera is chosen for the proposed system due to its compatibility and purpose-built design for Raspberry Pi. It is ideal for taking video recordings. After plugging the camera into the Pi board and giving it a few commands, users can utilize the camera immediately. It also has the capability of capturing images with a resolution of 5 MP. Regarding the prices, this camera comes at a very low cost, which is excellent value considering the features and performance it offers. The command that needs to capture video is:

$$fswbcam -r 1280 \times 1024 -v -S 10 - --set brightness = 100\% image1.jpg$$

3.1.3. Raspbian

Raspbian is a free operating system tailored for customizing or programming all Raspberry Pi models. It comes pre-installed with a variety of applications for general use, programming, and educational purposes. Raspbian supports programming languages such as Python, Java, Scratch, and more. To use Raspbian, one would install it onto a personal computer (PC) or Micro SD card, and then insert it into the Raspberry Pi. After connecting a monitor to the Pi board, it functions like a normal PC. Raspbian allows users to easily install numerous software packages from its open-source software repository at no cost. In this project, Python is employed to develop the necessary code for gathering data from various sensors to the Raspberry Pi and uploading it to the IoT cloud.

3.1.4. MIRACL

MIRACL, which stands for multiprecision integer and rational arithmetic cryptographic library, is a software library crucial for cryptographic operations involving large numbers. It efficiently handles arithmetic operations on large integers and rationals, making it suitable for implementing algorithms like ECC and Rivest–Shamir–Adleman (RSA) encryption. MIRACL is optimized for resource-constrained environments and supports various cryptographic schemes while maintaining security through regular updates and audits. It is a vital tool for implementing high-precision cryptographic algorithms efficiently and securely.

3.1.5. ECDSA algorithm

There are 3 steps:

– Key generation

First, a user generates a key pair consisting of a private key (d) randomly selected integer in the range $[1, n-1]$.

$$d \in [1, n - 1] \quad (1)$$

and a corresponding public key (Q) which is calculated by multiplying the base point G by the private key d .

$$Q = dG \quad (2)$$

Here, the multiplication refers to scalar multiplication on the elliptic curve, meaning d times the point G .

– Signing

To sign a message m , the signer first computes a unique value e from the message using a hash function.

$$e = HASH(m) \quad (3)$$

Then, a random number k is chosen and used to compute a point (x_1, y_1) on the elliptic curve. The x -coordinate of (x_1, y_1) modulo the order of the curve gives the r component of the signature. The s component is calculated as (4),

$$s = k^{-1}(e + dr) \text{ mod } n \quad (4)$$

where d is the private key, n is the order of the curve, and k^{-1} is the modular multiplicative inverse of k modulo n .

– Verification

To verify the signature, the verifier uses the signer's public key Q , the message m , and the signature (r, s) . First, (3) is calculated from the message. Then,

$$w = s^{-1} \text{ mod } n \quad (5)$$

$$u_1 = ew \text{ mod } n, u_2 = rw \text{ mod } n \quad (6)$$

are calculated. Finally, the point

$$(x_1, y_1) = u_1G + u_2Q \quad (7)$$

is computed, where G is the base point of the curve. If (x_1, y_1) equals (r, y_1) , the signature is valid; otherwise, it is invalid.

3.2. System architecture

To acquire public space footage, a Raspberry Pi webcam is utilized, and the captured videos are safeguarded using the elliptic curve digital signature algorithm (ECDSA) to prevent unauthorized access by hackers or hijackers. These secured videos are then transmitted to an authorized individual who utilizes them for forensic analysis. In the event of any malicious activities, the videos are scrutinized to ascertain the types of sharp objects involved, facilitated by deep learning models such as VGG16. The VGG16 model employs convolutional neural networks (CNNs) to identify whether the perpetrator is wielding a gun or a knife. If such objects are detected, a notification is sent to the Raspberry Pi, prompting it to activate an alarm system.

3.3. Research in deep learning

Our research employs a thorough methodology, which is systematically divided into three distinct sections. This structured approach allows for a clear and organized presentation of our research process. Figure 4 visually illustrates these sections, providing a detailed overview of how each component contributes to our overall research strategy.

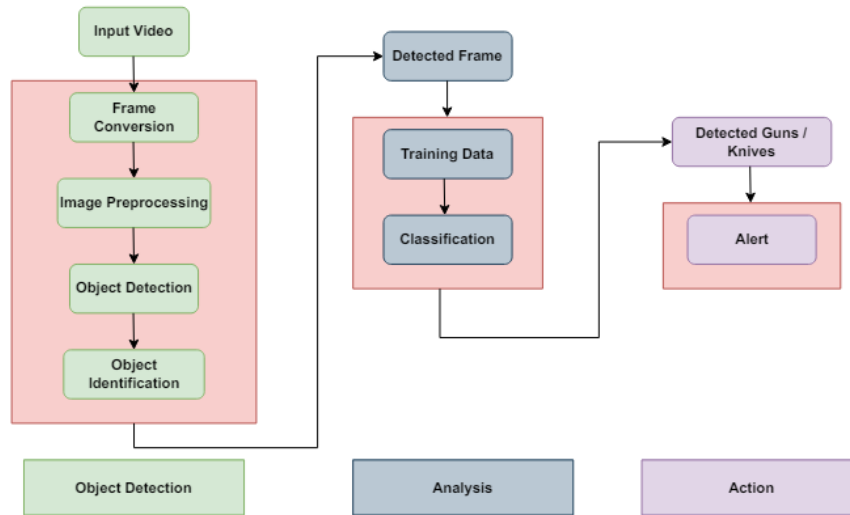


Figure 4. The flow of research in deep learning

3.3.1. Object detection

Initially, the videos obtained from the Raspberry Pi web camera are fed into the system. These videos are then decomposed into multiple frames, which allows for a more granular analysis of the visual data. Once the videos have been transformed into frames, a series of image preprocessing steps are applied. This includes techniques such as image translation, which involves shifting the pixels of the image to a certain direction, shearing, which distorts the image along an axis, normalization, which adjusts the pixel values to a standardized range, and edge detection, which identifies the edges or boundaries of objects within the image as depicted in Figure 5.



Figure 5. Preprocessed images

After preprocessing, the system proceeds to perform object detection and identification in the initial section. This involves detecting and recognizing objects within the frames using techniques such as convolutional neural networks (CNNs) or other machine learning algorithms. Object detection identifies the presence and location of objects in the frames, while object identification determines what those objects are, often by matching them to predefined categories or classes. This process enables the system to extract meaningful information from the video data, facilitating tasks such as surveillance, tracking, or analysis.

Table 1 provides a comprehensive breakdown of the video samples utilized for testing purposes. Specifically, a total of seven videos were dedicated to knife detection, collectively segmented into approximately 35,600 frames, each exclusively depicting knives. Additionally, eight videos were allocated for gun detection, segmented into approximately 42,340 frames, each solely containing guns.

Table 1. Number of annotated frames and number of videos

Weapons	No. of frames	No. of videos
Knife	35,600	7
Gun	42,340	8
Total	77,940	15

3.3.2. Analysis

After object identification, the frames containing the detected objects are subjected to a training phase. During this phase, the system learns from the detected objects to improve its ability to recognize and classify similar objects in the future. The training data, comprising the detected objects, is used to refine the system's understanding of object features and characteristics. The training data is then fed into a classification model, such as VGG16, recognized for its efficacy in image classification endeavors. VGG16 has multiple layers of convolution and pooling operations that learn hierarchical characteristics extracted from the input images, enabling accurate classification. In the training process, VGG16 learns to associate the detected objects with specific classes or categories.

3.3.3. Action

Upon detection of objects such as guns or knives by the model, a notification signal is generated and transmitted to the Raspberry Pi. Upon receiving this notification, the Raspberry Pi initiates a response mechanism, which includes activating a buzzer to sound an alarm. This alarm serves as an alert to notify relevant parties about the presence of potentially dangerous objects.

3.4. VGG16 model

VGG16 is a convolutional neural network (CNN) architecture specifically designed for image classification and recognition tasks. This architecture excels at identifying and categorizing objects within images. Figure 6 displays a range of sample images from the standard common objects in context (COCO) dataset, which are formatted as ".jpg" files. Specifically, Figure 6(a) through Figure 6(e) show sample images of guns, while Figure 6(f) through Figure 6(j) present sample images of knives.



Figure 6. Sample images from gathered dataset

The architecture consists of five distinct layers, each serving a specific function within the system. These layers work together to ensure the seamless operation and integration of the overall design. Figure 7 provides a detailed illustration of these layers, highlighting their individual roles and interactions.

- Input layer: VGG16 takes an input image of fixed size (usually 224×224 pixels), but we take 300×300 pixels.
- Convolution layers: VGG16 comprises of 13 convolutional layers, each succeeded by a ReLU activation function and 3×3 convolutional filters. These convolutional layers are structured to detect various features in the input image, such as edges, textures, and shapes, at different levels of abstraction. ReLU is mathematically defined as,

$$f(x) = \max(0, x)$$

where x is the input to the function.

- Pooling layers: After some convolutional layers, VGG16 employs max-pooling layers with 2×2 filters and a stride of 2. Pooling layers reduce the spatial dimensions of the feature maps, making the network more computationally efficient and reducing the risk of overfitting.
- Fully connected layers: VGG16 contains three fully connected layers followed by ReLU activation functions. The first two fully connected layers have 4,096 neurons each, and the third fully connected layer has 1,000 neurons, which correspond to the 1,000 classes in the dataset VGG16 was originally trained on.
- Output layer: The output layer of VGG16 uses the softmax activation function, which converts the final layer's outputs into probabilities. This convolutional layer is responsible for the detection of weapons within the input images. The resultant output feature maps, which highlight the presence of detected weapon features, are depicted in Figures 8(a) and 8(b).

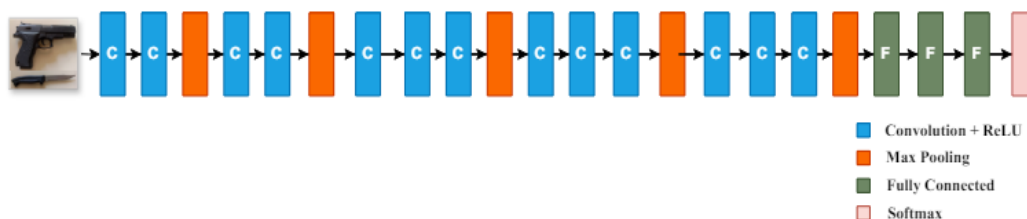


Figure 7. VGG 16 convolution neural network

```

1/1 [=====] - 0s 135ms/step
1/1 [=====] - 0s 38ms/step
1/1 [=====] - 0s 28ms/step
[[0.03472205]]
[[0.11089072]]
The image contains a knife.
1/1 [=====] - 0s 24ms/step

1/1 [=====] - 0s 121ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
[[0.07120285]]
[[0.08352856]]
The image contains a gun.
1/1 [=====] - 0s 16ms/step

```

Figure 8. Output of malicious object detected (a) knife and (b) gun

3.5. Interfacing systems: transmitting packets to Raspberry Pi

Wireshark can indeed be used to analyze the network traffic between a system and a Raspberry Pi. By capturing and inspecting the packets sent between these devices, you can troubleshoot connectivity issues, analyze protocols, and ensure the security of the communication. This can be particularly useful when setting up networked applications or diagnosing network-related problems in projects involving Raspberry Pi. Receiving the simple service discovery protocol (SSDP) packets from the system IP address to Raspberry Pi, as illustrated in Figure 9. When the system detects guns or knives, it triggers a notification to be sent to the Raspberry Pi, which in turn activates an alert buzz to the public.

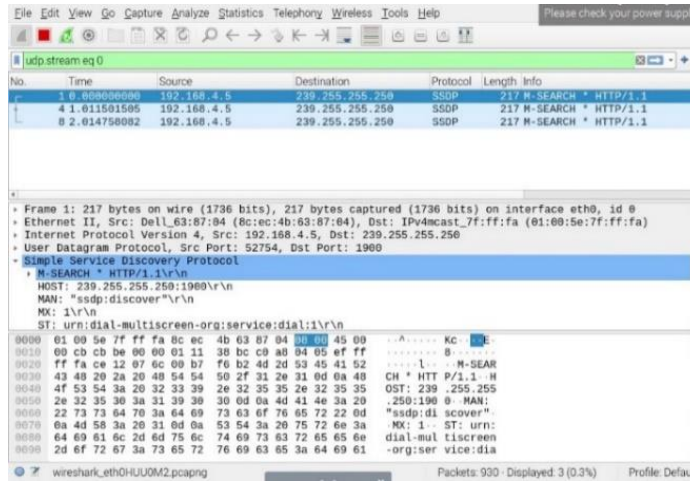


Figure 9. Packets from the system to Raspberry Pi

4. RESULTS

The output from the VGG16 model is automatically saved as a .txt file on the local device as shown in Figure 9. Subsequently, this text file is transmitted from the client side, running Python 3.12.3 on a local device, to the server, which is hosted on a Raspberry Pi. If the server receives data indicating an integer value of 1, signifying the detection of a Gun or Knife, it triggers an alert via a buzzer connected to the Raspberry Pi. Otherwise, it displays the message “YOU ARE IN SAFE.” The results are shown below.

When firearms are detected, the results will be displayed on the client-side interface, providing immediate feedback to users. Simultaneously, this information will also be shown on the server-side interface, ensuring that the data is available for backend processing and monitoring. Figure 10 demonstrates how the results appear on the client-side interface, while Figure 11 illustrates their presentation on the server-side interface.

If no firearms are detected, a notification message will be displayed on the client-side interface to inform users of the absence of threats. At the same time, this status message will also be shown on the server-side interface, ensuring that backend systems are updated accordingly. Figure 12 illustrates how this message appears on the client-side interface, while Figure 13 shows its presentation on the server-side interface.

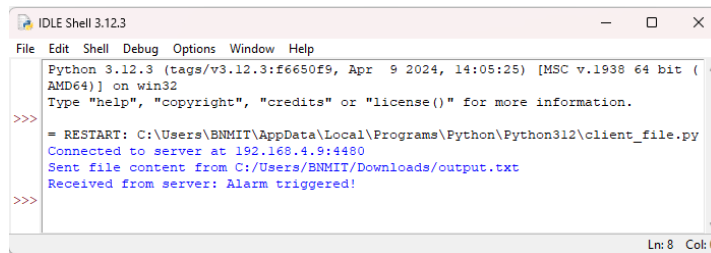


Figure 10. Detected firearms on client-side output

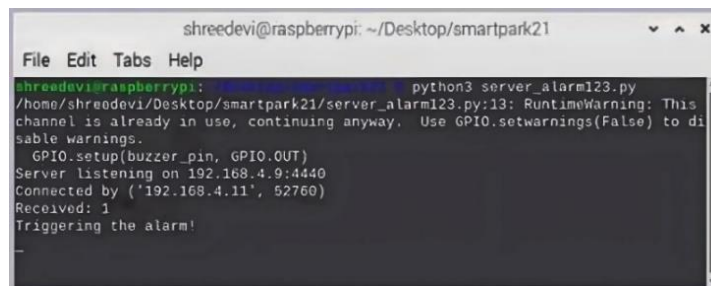
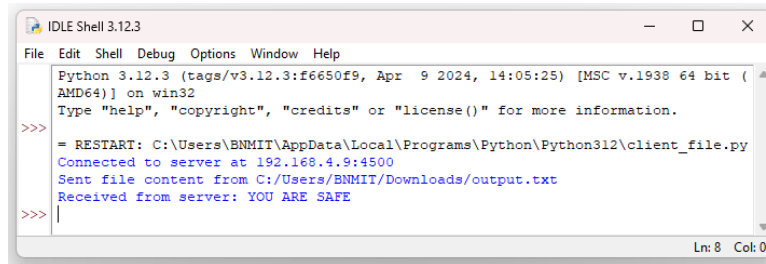


Figure 11. Detected firearms on server-side output

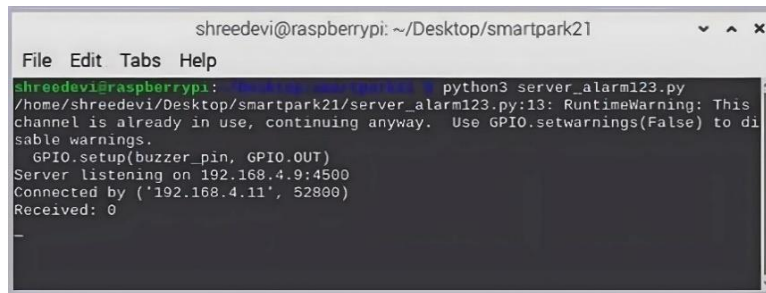


```

Python 3.12.3 (tags/v3.12.3:f6650f9, Apr 9 2024, 14:05:25) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\BNMIT\AppData\Local\Programs\Python\Python312\client_file.py
Connected to server at 192.168.4.9:4500
Sent file content from C:/Users/BNMIT/Downloads/output.txt
Received from server: YOU ARE SAFE
>>>

```

Figure 12. Weapons are not detected on the client-side



```

shreedeivi@raspberrypi: ~/Desktop/smartpark21
File Edit Tabs Help
shreedeivi@raspberrypi: ~$ python3 server_alarm123.py
/home/shreedeivi/Desktop/smartpark21/server_alarm123.py:13: RuntimeWarning: This channel is already in use, continuing anyway. Use GPIO.setwarnings(False) to disable warnings.
  GPIO.setup(buzzer_pin, GPIO.OUT)
Server listening on 192.168.4.9:4500
Connected by ('192.168.4.11', 52800)
Received: 0
_

```

Figure 13. Weapons are not detected on the server-side

The performance metrics, including accuracy, precision, recall, and F1-score, of various deep learning models were evaluated. Among the models assessed, VGG16 demonstrated superior performance, achieving the highest values in accuracy, precision, and F1-score. This comparison is comprehensively detailed in Table 2.

Table 2. Performance metrics for a real dataset in different models

SI No	Algorithms	Accuracy	Precision	Recall	F1-score
1	VGG16	97.27%	90.34%	75.23%	74.37%
2	VGG19	97.14%	88.22%	77.34%	68.01%
3	ResNet18	97.00%	74.89%	71.01%	74.29%

5. CONCLUSION

This work has introduced a novel real-time automatic weapon detection system for monitoring and control purposes. It aims to enhance security and law enforcement, contributing to the betterment and safety of humanity, particularly in nations that have undergone substantial violence. Our primary objective has been to identify the weapon in real-time webcam footage captured by Raspberry Pi. This paper presents a strong security feature that uses the ECDSA method specifically designed for public spaces. It collects the videos and stores them for later examination along with an authentication value. This dataset helps locate and resolve possible harmful activity that may be occurring on public property. The method increases cost-effectiveness and reliability by utilizing a webcam and microprocessor combo made for Internet of Things scenarios. The project's security measures are strengthened by the utilization of the MIRACL crypto library, which integrates digital signatures for authentication and encryption for data secrecy. In this work, the VGG16 model was employed for firearm and knife detection, achieving a peak accuracy of 97.27%, precision of 90.34%, recall of 75.23%, and Fi-score of 74.37%.





REFERENCES

- [1] W. A. Jabbar, C. W. Wei, N. A. A. M. Azmi, and N. A. Haironnazli, "An IoT Raspberry Pi-based parking management system for smart campus," *Internet of Things*, vol. 14, Jun. 2021, doi: 10.1016/j.iot.2021.100387.
- [2] L. Hu and Q. Ni, "IoT-driven automated object detection algorithm for urban surveillance systems in smart cities," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 747–754, Apr. 2018, doi: 10.1109/jiot.2017.2705560.
- [3] R. A. Nadafa, S. M. Hatturea, V. M. Bonala, and S. P. Naikb, "Home security against human intrusion using Raspberry Pi," *Procedia Computer Science*, vol. 167, pp. 1811–1820, 2020, doi: 10.1016/j.procs.2020.03.200.





- [4] Z. Cai, Y. Zhou, Y. Qi, W. Zhuang, and L. Deng, "A millimeter wave dual-lens antenna for IoT-based smart parking radar system," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 418–427, Jan. 2021, doi: 10.1109/jiot.2020.3004403.
- [5] R. Ke, Y. Zhuang, Z. Pu, and Y. Wang, "A smart, efficient, and reliable parking surveillance system with edge artificial intelligence on IoT devices," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 4962–4974, Aug. 2021, doi: 10.1109/tits.2020.2984197.
- [6] D. S. Kumbhar, "IoT based home security system using Raspberry," *International Research Journal of Engineering and Techonology*, vol. 6, no. 1, pp. 305–309, 2019.
- [7] L. Xiao, X. Wan, X. Lu, Y. Zhang, and D. Wu, "IoT security techniques based on machine learning: How do IoT devices use AI to enhance security?," *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 41–49, Sep. 2018, doi: 10.1109/msp.2018.2825478.
- [8] B. Kieu-Do-Nguyen, C. Pham-Quoc, N.-T. Tran, C.-K. Pham, and T.-T. Hoang, "Low-cost area-efficient FPGA-based multi-functional ECDSA/EdDSA," *Cryptography*, vol. 6, no. 2, p. 25, May 2022, doi: 10.3390/cryptography6020025.
- [9] J. J. Puthiyidam, S. Joseph, and B. Bhushan, "Temporal ECDSA: a timestamp and signature mask enabled ECDSA algorithm for IoT client node authentication," *Computer Communications*, vol. 216, pp. 307–323, Feb. 2024, doi: 10.1016/j.comcom.2024.01.016.
- [10] X. He *et al.*, "A universal single and double point multiplications architecture for ECDSA based on differential addition chains," *IEEE Access*, vol. 12, pp. 55434–55447, 2024, doi: 10.1109/access.2024.3390244.
- [11] Y. Ren *et al.*, "Multiple cloud storage mechanism based on blockchain in smart homes," *Future Generation Computer Systems*, vol. 115, pp. 304–313, Feb. 2021, doi: 10.1016/j.future.2020.09.019.
- [12] S. Rajashree, Sukumar R, and B. K. Panduranga, "Security system for visitor validation at entrance using Raspberry Pi and elliptical curve digital signature," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 4, pp. 3777–3781, Nov. 2019, doi: 10.35940/ijrte.D8148.118419.
- [13] P. Yang, N. Xiong, and J. Ren, "Data security and privacy protection for cloud storage: a survey," *IEEE Access*, vol. 8, pp. 131723–131740, 2020, doi: 10.1109/access.2020.3009876.
- [14] L. Yang, S. Xu, X. Yu, H. Long, H. Zhang, and Y. Zhu, "A new model based on improved VGG16 for corn weed identification," *Frontiers in Plant Science*, vol. 14, Jul. 2023, doi: 10.3389/fpls.2023.1205151.
- [15] F. Gapsari *et al.*, "A convolutional neural network -VGG16 method for corrosion inhibition of 304SS in sulfuric acid solution by timoho leaf extract," *Journal of Materials Research and Technology*, vol. 30, pp. 1116–1127, May 2024, doi: 10.1016/j.jmrt.2024.03.156.
- [16] Y. L. Chaitra, R. Dinesh, M. T. Gopalakrishna, and B. V. A. Prakash, "Deep-CNNLT: text localization from natural scene images using deep convolution neural network with transfer learning," *Arabian Journal for Science and Engineering*, vol. 47, no. 8, pp. 9629–9640, Nov. 2021, doi: 10.1007/s13369-021-06309-9.
- [17] L. Xu, J. Wang, X. Li, F. Cai, Y. Tao, and T. A. Gulliver, "Performance analysis and prediction for mobile Internet-of-Things (IoT) networks: a CNN approach," *IEEE Internet of Things Journal*, vol. 8, no. 17, pp. 13355–13366, Sep. 2021, doi: 10.1109/jiot.2021.3065368.
- [18] S. A. Mahaboobunisa, S. Gayathri, Y. K. Babu, and V. S. Kumar, "A novel approach for using common objects in context dataset (Coco) and real-time object detection using ML," *SSRN Electronic Journal*, 2023, doi: 10.2139/ssrn.4379058.
- [19] R. Debnath and M. K. Bhowmik, "A comprehensive survey on computer vision based concepts, methodologies, analysis and applications for automatic gun/knife detection," *Journal of Visual Communication and Image Representation*, vol. 78, Jul. 2021, doi: 10.1016/j.jvcir.2021.103165.
- [20] V. Kaya, S. Tuncer, and A. Baran, "Detection and classification of different weapon types using deep learning," *Applied Sciences*, vol. 11, no. 16, Aug. 2021, doi: 10.3390/app11167535.
- [21] G. Nagappan and V. U. Rani, *Disruptive technologies for sustainable development*, 1st Ed. CRC Press, 2024.
- [22] M. K. Galab, A. Taha, and H. H. Zayed, "Adaptive technique for brightness enhancement of automated knife detection in surveillance video with deep learning," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 4049–4058, Feb. 2021, doi: 10.1007/s13369-021-05401-4.
- [23] D. Berardini, L. Migliorelli, A. Galdelli, E. Frontoni, A. Mancini, and S. Moccia, "A deep-learning framework running on edge devices for handgun and knife detection from indoor video-surveillance cameras," *Multimedia Tools and Applications*, vol. 83, no. 7, pp. 19109–19127, Jul. 2023, doi: 10.1007/s11042-023-16231-x.
- [24] S. Narejo, B. Pandey, D. Esenarro Vargas, C. Rodriguez, and M. R. Anjum, "Weapon detection using YOLO V3 for smart surveillance system," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–9, May 2021, doi: 10.1155/2021/9975700.
- [25] C. Y. Lokkondra, D. Ramegowda, G. M. Thimmaiah, and A. P. B. Vijaya, "DEFUSE: deep fused end-to-end video text detection and recognition," *Revue d'Intelligence Artificielle*, vol. 36, no. 3, pp. 459–466, Jun. 2022, doi: 10.18280/ria.360314.
- [26] M. T. Bhatti, M. G. Khan, M. Aslam, and M. J. Fiaz, "Weapon detection in real-time CCTV videos using deep learning," *IEEE Access*, vol. 9, pp. 34366–34382, 2021, doi: 10.1109/access.2021.3059170.
- [27] M. L. Chayadevi, S. Madhyastha, K. N. Nisarga, H. Charitha, and B. Susharan, "Automated teller machine security with image processing and machine learning techniques," *Journal of Computational and Theoretical Nanoscience*, vol. 17, no. 9, pp. 4473–4481, Jul. 2020, doi: 10.1166/jctn.2020.9100.

BIOGRAPHIES OF AUTHORS







Shreedevi Kareppa Sanapannavar     is a postgraduate student pursuing an MTech in Computer Science and Engineering at BNM Institute of Technology, Bangalore under Visvesvaraya Technological University, Belagavi, India. Her areas of interest include the internet of things, machine learning, deep learning, cryptography, cyber security, and image processing. She can be contacted at email: shreedevivr@bnmit.in.



Chayadevi Mysore Lakshmanagowda     working as professor and head, computer science and engineering, BNMIT Bangalore, Karnataka, India, with a total teaching experience of 25 years. She has been graduated with B.E in computer science and engineering from SJCE, University of Mysore, M.Tech. in computer engineering from SJCE, Visvesvaraya Technological University (VTU) and Ph.D. on “Knowledge discovery from medical image databases: detection and classification of TB and MP smear images with their performance evaluation” from VTU. Her research interests include medical image processing, datamining, knowledge engineering, pattern recognition, speech recognition, biomedical applications, deep learning and ML with agriculture. She has authored 40+ publications in National and International Journals, Conferences along Patents. She has published book chapters at Tata McGraw Hill, Springer and Elsevier Publications in the area of medical image processing and cognitive computing and also Editor for Springer CCIS801 series. She has chaired several National and International Conferences as General Chair and Session Chair. She is also reviewer for National and International Journals and Conferences. Organized several student development programs, workshops, Faculty Development Programs, technical/expert talks and seminars. She has also supervised M.Tech/MCA Projects/Dissertations, BE projects. Also, supervisor for 6 Ph.D. scholars. She is also University (VTU) Nominee for Board of Exam (BOE) and Board of Studies (BOS). Submitted project proposals for Government and Technical boards. Received Project Fund from Karnataka State Council of Science and Technology (KSCST), Department of Science and Technology, Government of India (DST), Innovation and Entrepreneurship Development Centre (IEDC). She can be contacted at: hodcse@bnmit.in.



Geetha Sundararajan     obtained her bachelor’s degree in information technology from Anna University, Chennai. She also obtained his master’s degree in information technology from Sathiyabama University and a Ph.D. in computer science from Kalasalingam University. She is a professor in the CSE Department, at BNM Institute of Technology, Bangalore, India. Her areas of interest are data mining, information retrieval, artificial intelligence, machine learning, and blockchain. She is a life member of I.S.T.E & IE(I). She is currently a research supervisor at Visvesvaraya Technological University (VTU) and a Board of Examiner for affiliated VTU colleges under UG and PG programs. She has published her research work in various reputed indexed journals, Scopus indexed conferences, and book chapters. She is a reviewer in reputed journals and chaired multiple national and international conferences. She published her research work in Indian Patents and Australian Patents. Received a project fund from the Department of Science and Technology (DST) from the Innovation and Entrepreneurship Development Centre (IEDC). She can be contacted at email: geetha.s@bnmit.in.