International Journal of Safety and Security Engineering
Vol. 14, No. 5, October, 2024, pp. 1539-1545
Journal homepage: http://iieta.org/journals/ijsse

Protected Park: A Cornerstone of Trust in the Smart City

Shreedevi Kareppa Sanapannavar[1*], Rajashree Sridhar[2], Kusuma Prakash[2]

[1] Department of Information Science and Engineering, B N M Institute of Technology, Bengaluru 560070, India
[2] Department of Computer Science and Engineering, B.N.M. Institute of Technology, Bengaluru 560070, India

Corresponding Author Email: shreedevisuresh@gmail.com

https://doi.org/10.18280/ijsse.140521

Received: 19 July 2024
Revised: 28 September 2024
Accepted: 10 October 2024
Available online: 31 October 2024

Keywords:
smart park, smart city, Internet of Things, digital signature algorithm, public alert system, secure cloud storage

ABSTRACT

Smart cities include parks where children can play with their friends and elderly people can spend a good amount of time peacefully by watching kids playing and chatting with their age group people. Due to the potential threats posed by malicious activities, including theft and kidnapping, there arises a requirement for fortified smart park infrastructure ensuring a secure environment for children and elderly persons to enjoy recreational activities without any fear. This research proposes a secure system that tries to mitigate the risks associated with malicious activities through the integration of the Internet of Things domain with digital image processing components, triggering immediate alerts to authorized personnel upon detection of any such activities. The proposed system employs a dataset-driven approach to identify potential threats, specifically focusing on metallic weapons, and is a cost-effective solution for monitoring and managing smart parks.

1. INTRODUCTION

The Internet of Things (IoT) enables the integration of physical devices into the digital realm by granting them Internet connectivity and analytical capabilities. When IoT is combined with other domains can make an efficient system that can help humans. The Internet of Things (IoT) is revolutionizing [1] daily life by connecting smart devices to the internet. While promising, IoT still faces critical challenges. This paper reviews its key issues, architecture, and applications, offering insights into its real-world potential. In the past two decades, IoT devices have become integral across industries, yet their security remains a major challenge due to vulnerabilities [2] that cyberattacks exploit, causing widespread damage. This work examines the IoT security landscape, identifying key challenges, security goals, common threats, and necessary countermeasures for securing IoT systems. One such system is proposed in this article that creates a fearless and secure park as part of a smart city. A smart city uses technology and innovation to improve efficiency, sustainability, and inclusivity, tackling issues like pollution, population growth, and healthcare. IoT-based sensor networks, enhanced by machine learning, offer promising solutions in healthcare. Ghazal et al. [3] explores their application and lays the groundwork for future research on IoT's impact in smart cities. Smart cities have advanced rapidly, driven by IoT innovations that enable new services and applications. Bellini et al. [4] in their work reviews the research on IoT-enabled smart cities, highlighting key trends and challenges in building sustainable, efficient urban environments. It also surveys IoT technologies and classifies smart city approaches into eight domains, expanding on the traditional six-domain model.

2. RELATED WORK

Babangida et al. [5] examines the present state of sensors and sensing devices designed to discern human activity within smart home environments, with a specific emphasis on mitigating computational burdens and enhancing the precision of activity recognition mechanisms. License Plate Recognition (LPR) stands as a pivotal facilitator for Intelligent Transportation Systems (ITS) and the development of smart cities [6]. The suggested architecture incorporates a three-tiered image processing framework comprising pre-processing, segmentation, and character recognition stages. Employing advanced techniques such as Canny edge detection, contour detection, and masking, the system adeptly identifies the edges of vehicles and their corresponding license plates. The wireless connectivity of the ESP32 camera to the Raspberry Pi (RPI) ensures seamless communication, enabling the system to function with high efficiency and smooth operation. The envisaged SAVP (Smart Autonomous Vehicle Parking) system [7] offers intelligent parking solutions tailored for Autonomous Vehicles (A.V.s), employing fog nodes as an intermediary layer. A streamlined, unified blockchain and cryptography module is incorporated to authenticate and authorize A.V.s, facilitating seamless access during parking maneuvers. The system demonstrates exceptional efficiency by diminishing average throughput while concurrently addressing apprehensions related to security and privacy. This pioneering approach is applicable for the adept administration of forthcoming scalable Self-Parking (S.P.) systems tailored for Autonomous Vehicles (A.V.s) [8]. The presented case study employs the Analytic Hierarchy Process (AHP) to systematically rank security risks associated with smart home consumer devices. The AHP model, founded on empirical data, exhibits internal consistency and robustness, underscoring the significance of security factors within contemporary devices. The incorporation of empirical security studies into future research endeavours can significantly enrich the model's insights. In the realm of the Internet of Things (IoT), wherein computers and sensors interconnect, applications in smart homes involve functions like temperature monitoring, smoke detection, and lighting regulation, as elucidated in the research problem as discussed in the article [9]. However, the pervasive adoption of IoT in such contexts introduces pronounced security and privacy challenges, including potential breaches through surveillance apparatus or the occurrence of false alarms. This survey delves into the intricacies of IoT design, objects, and standards, with a specific focus on the tiered framework of the Internet of Things and associated security considerations.

The investigation meticulously scrutinizes critical components requiring protection and aspires to formulate a resilient solution to fortify security in smart home environments. Quality management is dependent on advanced technologies such as industrial image processing and cyber-physical cloud systems, which leverage sophisticated machine learning algorithms and Support Vector Machines (SVMs), as outlined in the proposal presented in the study [10]. Cutting-edge cryptographic algorithms with lightweight characteristics have the potential to elevate data security in forthcoming scenarios. Geetha and Deepalakshmi [11] introduces a novel void avoidance mechanism, strategically eliminating void nodes, thereby augmenting the energy efficiency of sensor nodes and fortifying data security in cloud environments through the implementation of a B+ tree index structure. The storage and retrieval of secured data are executed using the DB+ algorithm for preservation and the RQP method for acquisition, respectively. Joseph et al. [12] explores the identification of security breaches in both open and encrypted networks through the utilization of the ESP8266 Wi-Fi module. The study focuses on revealing the victim's BSSID during the detection process.

Table 1. Literature survey

SI	Author	Title	Methodology	Conclusion
1.	Abdellatif et al. [6]	A low cost IoT-based Arabic license plate recognition model for smart parking systems	OCR Technology to detect the license plate characters	Increase the efficiency of the license plate characters
2.	Shahzad et al. [7]	Enabling Fog- Blockchain Computing for Autonomous-Vehicle-Parking System: A Solution to Reinforce IoT–Cloud Platform for Future Smart parking	SAVP system for managing the parking services.	The use of emerging fog-blockchain technologies are being explored to enhance the existing IoT-cloud platform.
3.	Tavana et al. [8]	Analytical Hierarchy Process: Revolution and Evolution	The Analytic Hierarchy Process (AHP) is utilized to categorize security risks in smart home consumer devices.	The AHP model specifically and statistically emphasized the importance of different security aspects in contemporary home consumer products.
4.	Aldahmani et al. [9]	Cyber-Security of Embedded IoTs in Smart Homes: Challenges, Requirements, Countermeasures, and Trends		Temperature control, smoke detection, automated lighting control, and the installation of smart locks are all possible with smart home technology.
5.	Vineetha and Madhumala [10]	Providing Security and Managing Quality Through Machine Learning Techniques for an Image Processing Model in the Industrial Internet of Things, in Smart IoT for Research and Industry	Utilizing machine learning techniques and SVMs, quality management relies on cyber-physical cloud systems and industrial image processing.	
6.	Geetha and Deepalakshmi [11]	Enhanced Energy in Sensors by Avoiding Voids and Saving Sensitive Data on Cloud Using B+ Tree Index with Retrieval of Query Predicates	Agglo clustering algorithm Dynamic Avoidance Algorithm	Less energy utilization.
7.	Joseph et al. [12]	Detection of DoS Attacks on Wi-Fi Networks Using IoT Sensors. in Sustainable Advanced Computing	The ESP8266 Wi-Fi module is used to implement the suggested concept.	Extracting the victim's BSSID, it may identify assaults on wireless LANs (WLANs).
8.	Kumar et al. [13]	Smart Face Recognition Using IOT and Machine Learning	Finding all the faces using an oriented gradient histograms (HOG)	Utilizing the Raspberry Pi Infra-Red camera module further, the system may be used as a security surveillance system with an improved identification rate.
9.	Manikantha et al. [14]	Smart Worker Monitoring System Using Facial Recognition and Deep Learning Techniques	Using Deep Learning Techniques	Keeping tabs on them identities and how long background programs were active.
10.	Yang et al. [15]	Cryptanalysis and Improvement of a Blockchain-Based Certificateless Signature for IIoT Devices	A blockchain-based certificateless signature (CLS) mechanism for IIoT devices to achieve safe and lightweight communication	

The research presented by Sharma and team endeavors to amalgamate Internet of Things (IoT), Web of Things (WoT), and Big Data technologies within the context of smart cities in India, with a focus on achieving elevated scalability and adaptability through the utilization of dependable embedded devices [16]. The convergence of IoT and machine learning facilitates the aggregation of extensive data sets obtained from cameras and sensors, empowering the deployment of real-time identification algorithms [17]. This advanced technological solution finds widespread adoption in diverse domains, particularly in the realms of security, access control, and various applications encompassing attendance monitoring, residential security, and parking management. Supplementary functionalities, such as I.P. address tracking and alert systems, can be seamlessly incorporated following due approval. As posited i, remote work not only provides the advantage of geographical flexibility but also contributes to an enhanced work-life equilibrium and diminished stress associated with commuting. Advanced job monitoring and tracking software play a pivotal role in enhancing business productivity by monitoring background applications, URLs, and the duration of applications. The methodology employed and conclusions drawn from the literature survey section are detailed in Table 1. The escalating presence of active adversaries, posing a deliberate threat to society, underscores the growing imperative for security in smart parks.

3. SYSTEM DESIGN

Incorporating hardware components and deploying requisite software constitutes integral facets of the system design workflow. The ensuing section delineates the system design prerequisites essential for the envisaged project.

3.1 Hardware setup

The project mandates the installation of the official Raspberry Pi Operating System onto a microSD card. Subsequently, an apt Raspberry Pi web camera must be chosen, followed by the establishment of a connection between the Raspberry Pi 3 B+ and the selected web camera through a micro USB cable, interfacing with a designated USB port [18].

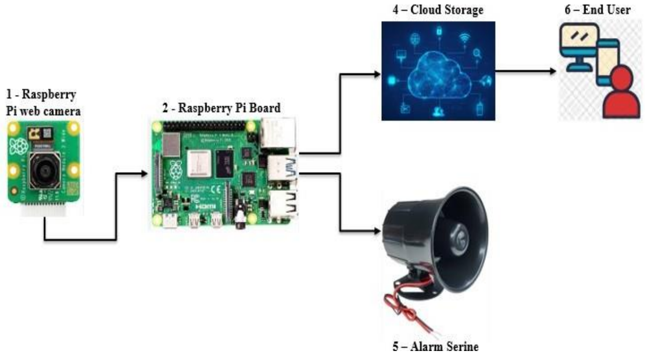


Figure 1. Overall system view

The project is delineated by a sequence of five steps, as depicted in Figure 1.

The steps include:

1. Acquisition of data through a webcam.
2. Generation of a digital signature utilizing the ECDSA algorithm.
3. Activation of an alarm system in the event of sharp object detection.
4. Storage of data in either a local drive or a cloud environment, contingent upon the client's specifications.
5. Retrieval of data and subsequent verification.

The initial two steps are executed on a Raspberry Pi platform, while the third step involves data storage. The fourth step is carried out on end devices such as mobile devices, laptops, or desktops. This paper proposes utilizing the Raspberry Pi Model 3B V1.3 along with a Raspberry Pi-compatible USB webcam for image capture. The USB webcam offers greater reliability compared to the Pi Camera, which uses a serial connection and is more fragile. A 64GB microSD card was employed to ensure faster OS performance. For end-user interaction, devices such as laptops, desktops, tablets, or smartphones can be used. The detailed configuration specifications for each device can be found in Table 2.

Table 2. Hardware components and configurations

SI No.	Components	Configuration
1	Raspberry Pi	Raspberry Pi Model V1.3
2	Web Camera	Raspberry Pi webcamera with USB cable
3	Micro SD card	64 bits
4	End Users	Smart Phones, Tabs, Desktops, and Laptops

3.2 Web camera configuration and connection

For the execution of this project, the utilization of a USB-connected web camera is recommended, as it exhibits superior performance in capturing both video streams and still images. The inherent advantage of USB-connected cameras lies in their seamless plug-and-play functionality, ensuring straightforward integration into the system. The Raspberry Pi device is appropriately configured to accommodate the requirements of the project, leveraging the fswebcam software package. This software facilitates the capture of both high-quality photos and video frames. The operational command employed for capturing images through the fswebcam software is exemplified below:

$$fswebcam - r\ 1280x720 - -no - banner\ /smart\_park$$
$$/image1.jpg$$

The acquired image of the smart park sample, obtained via a webcam, is illustrated in Figure 2.

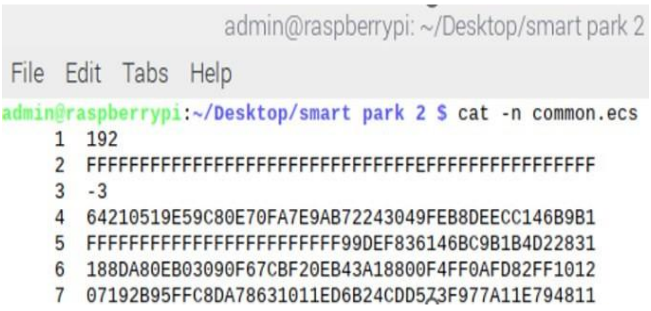
Figure 2. Smart park

In the project workflow, when timestamping is required for images, ImageMagick is utilized to embed temporal metadata, including both time and date, onto the images. This preprocessing step precedes the archival of the images in the cloud storage system, ensuring temporal context is preserved for future retrieval and reference [19]. In this endeavor, the protection of image data captured from the webcam is executed by leveraging the MIRACL (Multi-precision Integer Rational Arithmetic Crypto Library) cryptographic library package. This entails employing advanced cryptographic algorithms and protocols provided by MIRACL to ensure the confidentiality, integrity, and authenticity of the acquired image data. The library facilitates multi- precision arithmetic operations, enabling robust cryptographic computations necessary for securing sensitive visual information. The utilization of MIRACL enhances the overall security posture of the system, fortifying against potential threats and unauthorized access to the webcam-derived image data [20].

3.3 ECDSA algorithm

Digital signatures serve as a means of data authentication, with the utilization of elliptic curve cryptography representing an advanced technique in this domain. Elliptic Curve Cryptography (ECC) stands out as one of the most efficient cryptographic methodologies, demanding lower processing overhead. Within ECC, the Elliptic Curve Digital Signature Algorithm (ECDSA) [21, 22] is a prominent mathematical algorithm employed to validate the integrity and authenticity of digital information. Public-key cryptography, exemplified by ECDSA, operates on key pairs, consisting of public keys distributed openly and private keys maintained in secrecy, enhancing the overall security of cryptographic processes.

Point arithmetic procedures like point doubling and point addition are used to implement ECC protocols (ECDSA). The ECC provided by is a curve equation $y^2 = x^3 + Ax + b \ mod \ p$. This equation uses variables like [15A, B]. It is provided by NIST P192 elliptical curves with prime field. Below is information on the variables used in the equation (NIST P192) as described in Figure 3.



```
admin@raspberrypi:~/Desktop/smart park 2 $ cat -n common.ecs
     1  192
     2  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEFFFFFFFFFFFFFFFF
     3  -3
     4  64210519E59C80E70FA7E9AB72243049FEB8DEECC146B9B1
     5  FFFFFFFFFFFFFFFFFFFFFFFF99DEF836146BC9B1B4D22831
     6  188DA80EB03090F67CBF20EB43A18800F4FF0AFD82FF1012
     7  07192B95FFC8DA78631011ED6B24CDD573F977A11E794811
```

Figure 3. NIST P192 curve parameters

The elliptical curve digital signature algorithm consists of three phases.

- Phase 1: Creation of keys
- Phase 2: File signatures (R, S)
- Phase 3: Validation of the signed document

Phase 1: Creation of keys:

Random seed: A random seed serves as an initial input for the initialization of a pseudorandom number generator (PRNG). In the context of computational algorithms, particularly those involving stochastic processes, a random seed is a numeric value that kickstarts the deterministic sequence of seemingly random numbers generated by the PRNG. The selection of an appropriate random seed is crucial for ensuring reproducibility in computational experiments and simulations, as the same seed will consistently produce the same sequence of pseudorandom numbers, enabling researchers to replicate and verify their results. During the random seed generation phase, temporal information, notably the precise time and date of the image, is incorporated.

ECDSA generates the key using the basis point G and a significant prime number (d_{sp}) as described in Eq. (1).

$$Q_{sp} = d_{sp} \times G \tag{1}$$

- "d_{sp}" - private key which is derived from the random seed.
- "Q_{sp}" - public key that is shared in addition to the signatures.
- "G" is NIST P192's definition of the base point.

This random seed is utilized as a crucial input parameter in the key generation module, which resides within most of the cryptographic library. This module is subject to modifications to enable the generation of both public and private keys. Following this cryptographic process, the resultant keys are stored as public key and private key value, representing the public and private aspects of the cryptographic key pair, respectively.

Phase 2: File signatures (R, S):

A cryptographic process generates a 40-byte signature, which is subsequently divided into two 20-byte components, constituting the pair (R, S). This binary pair is then stored in a file and is contemporaneously disseminated alongside the corresponding image. Simultaneously, the indispensable public key file is shared in tandem with the image. The public key file assumes a crucial role in the image verification procedure.

- Finding the value of R

Utilizing a pseudo-random number generator (PRNG) or an initial seed, generate a random 20-byte value denoted as 'k'. Subsequently, employ this generated 'k' value to perform scalar multiplication with the elliptic curve's base point, thereby determining a new point on the curve.

$$(x, y) = k \times G \tag{2}$$

'R' will represent that point p's 'x' coordinate value.

- Finding the value of S:

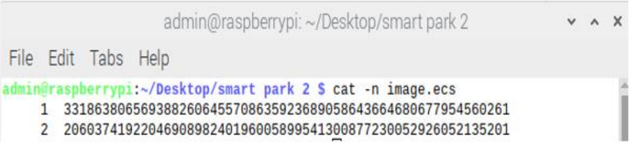
The equation used to find the part of signature value S is described in Eq. (3).

$$S = K^{-1}\left(Z + d_{sp} \times R\right) mod \ P \tag{3}$$

where, P represents the finite field in the Ed25519 curve, which is given by 2^{255}-19. K^{-1} is the multiplicative inverse of K. The x-coordinate value (R) of point P is determined by employing Eq. (2). The respective R and S values corresponding to Smart Park, as depicted in Figure 2, are illustrated in Figure 4.

- Z Computation:

The SHA-1 algorithm generates a 128-bit hash independent of the size of the input. The SHA-1 function inflates to a 128-bit hash when it receives an input of 100 bits. A 128-bit hash will be produced from the output even if an input of one gigabyte is given.



```
admin@raspberrypi:~/Desktop/smart park 2 $ cat -n image.ecs
     1	3318638065693882606455708635923689058643664680677954560261
     2	2060374192204690898240196005899541300877230052926052135201
```

Figure 4. R and S values generated for smart park

This basic SHA1 hashing can be represented by the following Eq. (4):

$$\text{Hash1} = \text{SHA1 (ln)} \qquad (4)$$

In the above equation, more than one input that leads to Hash1 may exist in the domain space. To avoid the hash value collision, this article proposes the use of a 4-byte SALT extra information that is added to SHA1. After that, the equation would be as follows (Eq. (5)):

$$Hash_1 = SHA1(ln) + SALT \qquad (5)$$

Once after generating a SHA1 hash of 128 bits for more collision resistance, this will be given to SHA2 resulting in a 256-bit hash value. The Eq. (6) that would result from passing the 128-bit Hash1 value to SHA-2 is as follows:

$$Z = SHA2(Hash_1) \qquad (6)$$

There is less chance of a collision in this scenario because Hash1 is smaller than Z.

The SALT is extra information to better protect an image. The function of SALT in the equation is to protect Z's uniqueness from collision assaults. “Z” is a combination of two hash functions of different output sizes. To guarantee uniqueness, this method first compresses the result of the SHA1 procedure using a reduced size before appending the SALT. Finally, the whole equation used to compute Z is as described in Eq. (7).

$$\text{Hash1} = \text{SHA1 (ln)} \qquad (7)$$

Z stores 256 bits of output.

3.4 Alarm in case of sharp object detection

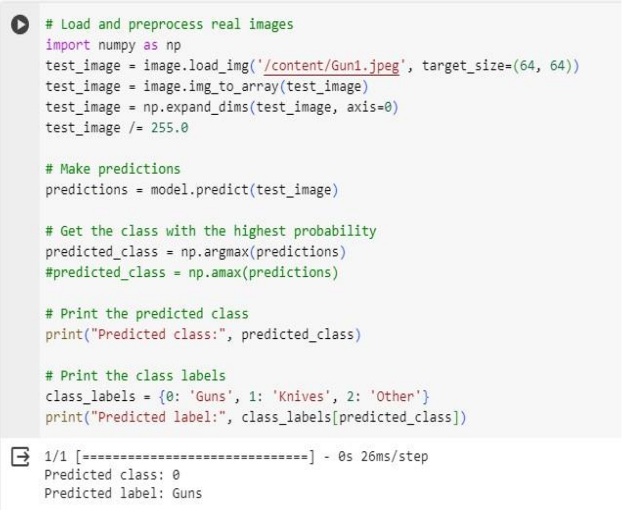
The alarm system utilizes sophisticated image analysis algorithms to expeditiously alert authorized personnel upon identifying anomalous objects with high acutance within an image dataset. This functionality enhances forensic investigation capabilities in cases of theft, allowing for swift and targeted responses. The park, functioning as a pivotal recreational enclave for children, serves a dual purpose. It not only acts as a deterrent against potential abduction scenarios through the implementation of the advanced alarm system but also plays a vital role in fostering the overall societal well-being.

This study analyzes biometric protocols for securing wireless sensor networks (WSNs), focusing on their characteristics and limitations. It emphasizes the importance of unique biometric authentication and reviews attack defense schemes [23]. The proposed WSN-MAS enhances WSN security and is applicable to smart environments such as healthcare and intelligent transportation.

This paper presents a new automated model for anomaly detection and localization in crowd videos using optimized Self-Organizing Maps (SOM) [24]. The model generates three video frame patterns and employs feature extraction techniques such as HOG, LGP, and PCA for dimensionality reduction.

The dataset for the detection of weapons is chosen from the study [25]. The dataset consists of 4794 images, with 4409 images assigned to the training class and 385 images assigned to the test class. Machine learning models that input or output data sequences are known as sequence models. In this paper, we will use a sequential model. The Sequential model is a simple and convenient way to build a linear stack of layers for a neural network. which can create a Sequential model and add layers to it one by one, specifying the input shape and the number of neurons in each layer.

The Dense layer represents a fully connected layer, where each neuron is connected to every neuron in the previous and next layers. The input shape parameter is specified only for the first layer, as the subsequent layers can automatically infer their input shapes. MaxPooling is a downsampling operation that reduces the spatial dimensions of the input data. Here, the activation functions are 'relu' and 'sigmoid'. Finally, train the model on a dataset using the fit method.



```python
# Load and preprocess real images
import numpy as np
test_image = image.load_img('/content/Gun1.jpeg', target_size=(64, 64))
test_image = image.img_to_array(test_image)
test_image = np.expand_dims(test_image, axis=0)
test_image /= 255.0

# Make predictions
predictions = model.predict(test_image)

# Get the class with the highest probability
predicted_class = np.argmax(predictions)
#predicted_class = np.amax(predictions)

# Print the predicted class
print("Predicted class:", predicted_class)

# Print the class labels
class_labels = {0: 'Guns', 1: 'Knives', 2: 'Other'}
print("Predicted label:", class_labels[predicted_class])
```

```
1/1 [==============================] - 0s 26ms/step
Predicted class: 0
Predicted label: Guns
```

Figure 5. Model to predict weapons

If the model predicts the 'Gun' or 'Knife' then it signals the alarm in the park and notifies concerned security officers (see Figure 5).

The performance metrics accuracy, precision, and recall, were evaluated across multiple deep learning models. A detailed comparison is provided in Table 3.

Table 3. Performance evaluation of various models using a real dataset

SI No.	Algorithms	Accuracy	Precision	Recall
1	VGG19	97.29%	91.18%	79.01%
2	VGG16	97.10%	87.23%	81.16%
3	ResNet 50	86.11%	75.62%	72.19%

3.5 Storage of image data in the drive

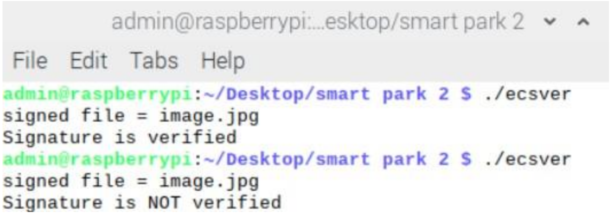
The acquired images, accompanied by their corresponding date and time stamps, are systematically archived in cloud storage or a designated storage drive, facilitating their integration into image detection applications. Typically, these images are encoded in the JPEG (Joint Photographic Experts Group) file format to optimize data compression, achieving a judicious compromise between maintaining satisfactory image quality and minimizing overall file size. Users are afforded the flexibility to tailor parameters such as compression levels during the image storage process.

3.6 Verifying the sign

The preliminary communication phase encompasses the transmission of the primary message, concomitant with the public key (Q_{sp}) and the signature (R, S). During the verification procedure, the message is employed to derive the hash value, and subsequently, the value p is computed using Eq. (8) in accordance with Eq. (3). The acceptance criterion for the authenticity of the message is satisfied only if both the value R and the computed value are in concordance.

$$P = S^{-1}\, Z \times G + S^{-1} \times R \times Q_{sp} \tag{8}$$

The value of Z is calculated again for the received image at the receiver side and then the value of P is found out using the public key. If a signature is valid then the data is authenticated else the data is modified by some external malicious activity.



```
admin@raspberrypi:~/Desktop/smart park 2 $ ./ecsver
signed file = image.jpg
Signature is verified
admin@raspberrypi:~/Desktop/smart park 2 $ ./ecsver
signed file = image.jpg
Signature is NOT verified
```

Figure 6. Validation for images

The verification of image authenticity involves a meticulous comparison between the provided image and the corresponding frame captured in the original event video (see Figure 6). It is imperative to recognize that relying solely on a single image may not yield conclusive results, as the outcome is contingent upon the video's frame rate, denoted as frames per second (fps). In essence, if the video is recorded at 33 fps, there will be 33 individual frames to scrutinize for every second of footage. Table 4 delineates the public key, R, and S values associated with diverse images from the smart park dataset.

Table 4. The R and S values of the image's public key

Pictures	Public Key	R Value	S Value
	3318638065693882606455708635923689058643664680677954560261215	1184661037838522723347023061088546266949753134157049206621	5492873726759280974013235040949879660646875108678525692463
	3867505862548321950403856914789627247313830325443194074052	2331166343038353669475372546793345784885373921764329518959	1954677835897677742064401629292101347089889162257559924289
	5922009533165673920784734375054242618321920578134591477558	1983582759970146136624261750664636318493147765595072761196	3706882835897790607040743850156389123612370475245400615996
	4941152402187364020324256881693447122085866965204301818840	5574044859009408164420069839415342926819107218543741978939	1563411350322933790840163963447626854778121620698336613009
	3242838646291286492504434017823212573355657689327773830138	4580920925858784010480059952364017010428620391780936267915	2073232607295099081137985193509657293089626841138656589569

4. CONCLUSION

The document introduces a robust security mechanism employing the ECDSA algorithm tailored for a smart park system, which acquires timestamped image data and archives it for subsequent analysis, accompanied by an authentication value. This dataset proves instrumental in identifying and addressing potential malicious activities within park premises. The implementation leverages a combination of a microprocessor and a webcam designed for operation within IoT environments, thereby augmenting both reliability and cost efficiency. The security aspects of the project are fortified using the MIRACL crypto library, incorporating digital signatures for authentication and encryption to ensure data confidentiality.

REFERENCES

[1] Mouha, R.A.R.A. (2021). Internet of Things (IoT). Journal of Data Analysis and Information Processing, 9(2): 77. https://doi.org/10.4236/jdaip.2021.92006

[2] Schiller, E., Aidoo, A., Fuhrer, J., Stahl, J., Ziörjen, M., Stiller, B. (2022). Landscape of IoT security. Computer Science Review, 44: 100467. https://doi.org/10.1016/j.cosrev.2022.100467

[3] Ghazal, T.M., Hasan, M.K., Alshurideh, M.T., Alzoubi, H.M., Ahmad, M., Akbar, S.S., Al-Kurdi, B., Akour, I.A. (2021). IoT for smart cities: Machine learning approaches in smart healthcare—A review. Future Internet, 13(8): 218. https://doi.org/10.3390/fi13080218

[4] Bellini, P., Nesi, P., Pantaleo, G. (2022). IoT-enabled smart cities: A review of concepts, frameworks and key technologies. Applied Sciences, 12(3): 1607. https://doi.org/10.3390/app12031607

[5] Babangida, L., Perumal, T., Mustapha, N., Yaakob, R. (2022). Internet of Things (IoT) based activity recognition strategies in smart homes: A review. IEEE Sensors Journal, 22(9): 8327-8336. https://doi.org/10.1109/JSEN.2022.3161797

[6] Abdellatif, M.M., Elshabasy, N.H., Elashmawy, A.E., AbdelRaheem, M. (2023). A low cost IoT-based Arabic license plate recognition model for smart parking systems. Ain Shams Engineering Journal, 14(6): 102178. https://doi.org/10.1016/j.asej.2023.102178

[7] Shahzad, A., Gherbi, A., Zhang, K. (2022). Enabling fog–blockchain computing for autonomous-vehicle-parking system: A solution to reinforce IoT-cloud platform for future smart parking. Sensors, 22(13): 4849. https://doi.org/10.3390/s22134849

[8] Tavana, M., Soltanifar, M., Santos-Arteaga, F.J. (2023). Analytical hierarchy process: Revolution and evolution. Annals of Operations Research, 326(2): 879-907. https://doi.org/10.1007/s10479-021-04432-2

[9] Aldahmani, A., Ouni, B., Lestable, T., Debbah, M. (2023). Cyber-security of embedded IoTs in smart homes: Challenges, requirements, countermeasures, and trends. IEEE Open Journal of Vehicular Technology, 4: 281-292. https://doi.org/10.1109/OJVT.2023.3234069

[10] Vineetha, B., Madhumala, R. (2022). Providing security and managing quality through machine learning techniques for an image processing model in the Industrial Internet of Things. In Smart IoT for Research and Industry, pp. 161-177. https://doi.org/10.1007/978-3-030-71485-7_10

[11] Geetha, S., Deepalakshmi, P. (2019). Enhanced energy in sensors by avoiding voids and saving sensitive data on cloud using B+ tree index with retrieval of query predicates. Mobile Networks and Applications, 24: 234-247. https://doi.org/10.1007/s11036-018-1203-z

[12] Joseph, I., Honnavalli, P.B., Charanraj, B. (2022). Detection of DoS attacks on Wi-Fi networks using IoT sensors. In Sustainable Advanced Computing: Select Proceedings of ICSAC 2021, pp. 549-558. https://doi.org/10.1007/978-981-16-9012-9_44

[13] Kumar, A., Kalumbi, S., RaoMV, P.P. (2023). Smart face recognition using IOT and machine learning. International Journal of Research Publication and Reviews, 4(5): 4161-4171. https://doi.org/10.55248/gengpi.4.523.42625

[14] Manikantha, K.M., Mishra, B., Abhinav, T.L. (2023). Smart worker monitoring system using facial recognition and deep learning techniques. International Journal of Computer Applications Technology and Research, 12(1): 60-62. https://ijcat.com/archieve/volume12/issue1/ijcatr12011010.pdf.

[15] Yang, X., Wang, W., Tian, T., Wang, C. (2023). Cryptanalysis and improvement of a blockchain-based certificateless signature for IIoT devices. IEEE Transactions on Industrial Informatics, 20(2): 1884-1894. https://doi.org/10.1109/TII.2023.3282317

[16] Sharma, A., Reddy, S., Patwal, P.S., Gowda, D. (2022). Data analytics and cloud-based platform for internet of things applications in smart cities. In 2022 International Conference on Industry 4.0 Technology (I4Tech), Pune, India, pp. 1-6. https://doi.org/10.1109/I4Tech55392.2022.9952780

[17] Tetali, D.R., Kumar, K., Ramana, L. (2017). A Python tool for evaluation of subjective answers (ApTeSa). IJMET, 8(7): 247-255.

[18] Raspberry Pi HQ. (2018). Using a push button with Raspberry Pi GPIO. https://raspberrypihq.com/use-a-push-button-with-raspberry-pi-gpio/.

[19] Patil, P., Patil, S., Miniyar, V., Bandal, A. (2018). Subjective answer evaluation using machine learning. International Journal of Pure and Applied Mathematics, 118(24): 1-13.

[20] Yuan, X.H., Li, J.F., Wang, D.X., Chen, Y.F., Mao, X.F., Huang, L.T., Xue, H., Wang, W.H., Ren, K., Wang, J.Y. (2024). S-Eval: Automatic and adaptive test generation for benchmarking safety evaluation of large language models. arXiv preprint arXiv:2405.14191. https://doi.org/10.48550/arXiv.2405.14191

[21] Genç, Y., Afacan, E. (2021). Design and implementation of an efficient elliptic curve digital signature algorithm (ECDSA). In 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, pp. 1-6. https://doi.org/10.1109/IEMTRONICS52119.2021.9422589

[22] Sghaier, A., Zeghid, M., Machhout, M. (2016). Fast hardware implementation of ECDSA signature scheme. In 2016 International Symposium on Signal, Image, Video and Communications (ISIVC), Tunis, Tunisia, pp. 343-348. https://doi.org/10.1109/ISIVC.2016.7894012

[23] Shivanna, P., Venkatesiah, S.S. (2021). Secure multimodal authentication scheme for wireless sensor networks. International Journal of Safety and Security Engineering, 11(6): 653-661. https://doi.org/10.18280/ijsse.110605

[24] Naik, A.J., Thimmaiah, G.M. (2021). Detection and localization of anamoly in videos using fruit fly optimization-based self organized maps. International Journal of Safety and Security Engineering, 11(6): 703-711. https://doi.org/10.18280/ijsse.110611

[25] Kaggle. Guns-knives object detection dataset. https://www.kaggle.com/datasets/iqmansingh/guns-knives-object-detection.