A Triple Hybrid Algorithm for an Efficient Data Security Over Wireless Data Transfer

Dr. M. V. Karthikeyan, M.E., P. Arunkumar, P. Balamurugan
Department of Electronics and Communication Engineering,
St. Joseph's Institute of Technology, Chennai, India.

Abstract:- **Data security refers to the process of protecting data from unauthorized access and data corruption throughout its lifecycle. Data security includes data encryption, hashing, tokenization, and key management practices that protect data across all applications and platforms. The security system used nowadays uses Data encryption software to effectively enhance data security by using an algorithm (called a cipher) and an encryption key to turn normal text into encrypted ciphertext. To an unauthorized person, the cipher data will be unreadable. That data can then be decrypted only by a user with an authorized key. Whereas with the improving data insecurity nowadays leads to loss of confidential data as the key is easily hackable because of a single algorithm usage. To overcome this problem this project presents an Efficient Data Security System where triple hybrid algorithm will be used to secure the data stored and accessed in cloud. The multiple sensors such as Heart rate sensor, blood oxidation sensor, temperature sensor, Blood pressure sensor and respiration sensors are used to collect the data. Then, the triple hybrid algorithm which is a combination of JWT token, AES encryption and Hash encoding are used to secure the data using the multiple sensors collected and store in to the cloud. Thus, this project successfully provides an end to end data security and access in to cloud storage.**

Keywords:- ***JWT GENERATION, AES ENCRYPTION, MQTT PROTOCOL***

I. INTRODUCTION

Data security is the process of protecting corporate data and preventing data loss through unauthorized access. This includesprotecting your data from attacks that can encrypt or destroy data, such as ransomware, as well as attacks that can modify or corrupt your data. Data security also ensures data is available to anyone in the organization who has access to it. Some industries require a high level of data security to comply with data protection regulations. For example, organizations that process payment card information must use and store payment card data securely, and healthcare organizations in the USA must secure private health information (PHI) in line with the HIPAA standard. But even if your organization is not subject to a regulation or compliance standard, the survival of a modern business depends on data security, which can impact both the organization's key assets and private and data belongings to its customers.

Data privacy is the distinction between data in a computer system that can be shared with third parties (non-private data), and data that cannot be shared with third parties (private data). There are two main aspects to enforcing data privacy:**Access control**—ensuring that anyone who tries to access the data is authenticated to confirm their identity, and authorized to access only the data they are allowed to access

Data protection—ensuring that even if unauthorized parties manage to access the data, they cannot view it or cause damage to it. Data protection methods ensure encryption, which prevents anyone from viewing data if they do not have a private encryption key, and data loss prevention mechanisms which prevent users from transferring sensitive data outside the organization.Data security has many overlaps with data privacy. The same mechanisms used to ensure data privacy are also part of an organization's data security stratergy.

The primary difference is that data privacy mainly focuses on keeping data confidential, while data security mainly focuses on protecting from malicious activity. For example, encryption could be a sufficient measure to protect privacy, but may not be sufficient as a data security measure. Attackers could still cause damage by erasing the data or double-encrypting it to prevent access by authorized parties.

II. LITERATURE REVIEW

[1] Proposed SPADE, an encrypted data deduplication scheme that resists compromised key servers and frees users from the key management problem. Specifically,proposed a proactivization mechanism for the servers-aided message-locked encryption (MLE) to periodically substitute key servers with newly employed ones, which renews the security protection and retains encrypted data deduplication. A servers-aided password-hardening protocol has been proposed to resist dictionary guessing attacks. Based on the protocol, a password-based layered encryption mechanism and a password-based authentication mechanism has been proposed and integrated them into SPADE to enable users to access their data only using their passwords. Provable security and high efficiency of SPADE are demonstrated by comprehensive analyses and experimental evaluations.

[2] Data, a key asset in our data-driven economy, has fueled the emergence of a new data trading industry. However, there are a number of limitations in conventional data trading platforms due to the existence of dishonest buyer/data broker. To mitigate these limitations, we posit the importance of a data processing-as-a-service model, which complements conventional data hosting/exchange-as-aservice model. Specifically, in this paper, we introduce a secure data trading ecosystem and present a new blockchain-based data trading ecosystem (hereafter

referred to as SDTE). In the ecosystem, both data broker and buyer are not able to obtain access to the seller's raw data, as they are only getting access to the analysis findings that they require. In other words, we reduce the challenge of securing the dataset to the challenge to secure the data processing. We also build a security model to analyze the data trading market, and describe a new set of trading protocols for the entire data trading market. To demonstrate utility, we implement our proposed secure data trading platform (SDTP) on Ethereum & Intels Software Guard Extensions (SGX) and perform an in-depth analysis

[3] Memory resistor or memristor is the forth fundamental circuit element that has attained considerable attention due to its unique characteristics and possible extensive applications in future generation nanoscale circuits and systems. In this brief, the contribution that memristor-based circuits may offer to the evolution of cryptographic hardware and embedded systems is discussed. Specifically, it will be shown how memristor-based implementation of security algorithms can mitigate the danger of differential power analysis attacks (DPA) at the technology level with lower cost and energy compared to conventional existing algorithmic countermeasure techniques. A 128-bit Advanced Encryption Standard (AES) cryptoprocessor was designed and implemented in both CMOS and hybrid CMOS/memristor technology. The robustness of the CMOS- based implementation against power analysis attacks was evaluated on Side-Channel Attack User Reference Architecture (SAKURA-GII) while the nanoscale counterpart system was evaluated by using a customized simulation and attack environment which was developed for extracting power traces using Synopsys and Cadence tools along with a DPA attack software implemented in MATLAB. It was observed that hybrid CMOS/memristor-based implementation provides considerable improvement over implementation with regular CMOS architectures in terms of energy consumption and attack tolerance and demonstrates good potential in mitigating DPA attacks without having to apply costly countermeasures such as masking or hiding.

[9] Object detection in streaming images is a major step in different detection-based applications, such as object tracking, action recognition, robot navigation, and visual surveillance applications. In most cases, image quality is noisy and biased, and as a result, the data distributions are disturbed and imbalanced. Most object detection approaches, such as the faster region-based convolutional neural network (RCNN), single shot multibox detector with 300CE300 inputs (SSD300), and you only look once version 2 (YOLOv2), rely on simple sampling without considering distortions and noise under real- world changing environments, despite poor object labeling. In this paper, we propose an incremental active semi-supervised learning (IASSL) technology for unseen object detection. It combines batch-based active learning (AL) and bin-based semi- supervised learning (SSL) to leverage the strong points of AL's exploration and SSL's exploitation capabilities. A collaborative sampling method is also adopted to measure the uncertainty and diversity of AL and the confidence in SSL. Batch-based AL allows us to select more informative, confident, and representative samples with low cost. Bin-based SSL divides streaming image samples into several bins, and each bin repeatedly transfers the discriminative knowledge of convolutional neural network deep learning to the next bin until the performance criterion is reached. The IASSL can overcome noisy and biased labels in unknown, cluttered data distributions. We obtain superior performance, compared with the state-of-the-art technologies, such as Faster RCNN, SSD300, and YOLOv2.

EXISTING SYSTEM

Proposed SPADE, an encrypted data deduplication scheme that resists compromised key servers and frees users from the key management problem. Specifically, proposed a proactivization mechanism for the servers-aided message-locked encryption (MLE) to periodically substitute key servers with newly employed ones, which renews the security protection and retains encrypted data deduplication. Presented a servers-aided password-hardening protocol to resist dictionary guessing attacks. Based on the protocol, further propose a password- based layered encryption mechanism and a password-based authentication mechanism and integrate them into SPADE to enable users to access their data only using their passwords. Provable security and high efficiency of SPADE are demonstrated by comprehensive analyses and experimental evaluations.

DISADVANTAGES OF EXISTING SYSTEM

In this system, it can be easily hacked and it cannot secure highly confidential data.The theoretical treatment on such a type of encryption algorithm is still lacked. A single encryption algorithm is insufficient to transmit data securely .

III. METHODOLOGY

In this project, we primarily focuses to provide an efficient data security system to secure the data stored and accessed in cloud. To achieve this multiple data from sensors such such as Heart rate sensor, blood oxidation sensor, temperature sensor, Blood pressure sensor and respiration sensors are being used to collect the data from patients. In order to secure this data we use a triple hybrid algorithm which is a combination of JWT token, AES encryption and Hash encoding are used to secure the data in the cloud. JSON Web Token (JWT) is a standard used to create access tokens for an application. It works this way: the server generates a token that certifies the user identity, and sends it to the client. The client will send the token back to the server for every subsequent request, so the server knows the request comes from a particular identity. AES includes three block ciphers. Each cipher encrypts and decrypts data in blocks of 128 bits using cryptographic keys of 256 bits, respectively. Symmetric, also known as secret key, ciphers use the same key for encrypting and decrypting. Once JWT is generated and AES encryption is processed finally hash encoding is processed for providing more security to the data. Then this secured data is stored in the cloud for future use. And a wireless gateway is used to transmit the data wirelessly to the cloud server. In addition to this patient's care taker will receive a SMS alert whenever any input parameter like blood pressure goes abnormal. Thus, this project successfully provides an end to end data security and access the secured data from the cloud storage.

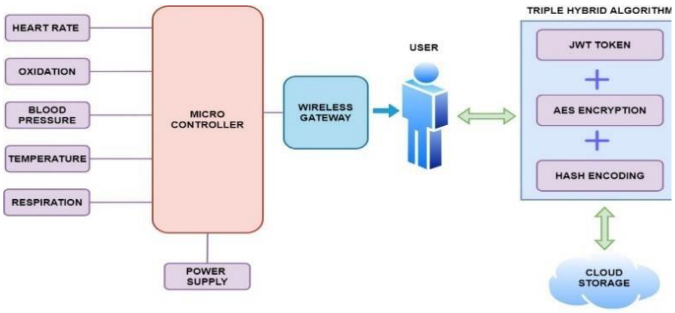
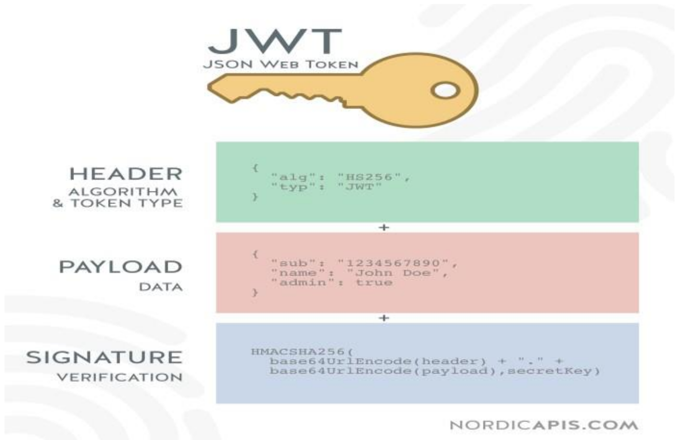


Figure 3.1 Proposed System Architecture

IV. DESIGN AND IMPLEMENTATION

JSON Web Tokens are an open, industry standard RFC 7519 method for representing claims securely between two parties. JSON Web Token (JWT) is an open standard (RFC 7519) that defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed. JWTs can be signed using a secret (with the **HMAC** algorithm) or a public/private key pair using **RSA** or **ECDSA**. Although JWTs can be encrypted to also provide secrecy between parties, we will focus on *signed* tokens. Signed tokens can verify the *integrity* of the claims contained within it, while encrypted tokens *hide* those claims from other parties. When tokens are signed using public/private key pairs, the signature also certifies that only the party holding the private key is the one that signed it.



```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

```
{
  "sub": "1234567890",
  "name": "John Doe",
  "admin": true
}
```

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),secretKey)
```

Fig 4.13 JWT GENERATION

AES ENCRYPTION

The Advanced Encryption Standard (AES), also known by its original name Rijndael is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001 .AES is a variant of the Rijndael block cipher developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, who submitted a proposal to NIST during the AES selection process.[6] Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits. AES has been adopted by the U.S. government. It supersedes the Data Encryption Standard (DES), which was published in 1977. The algorithm described by AES is a symmetric- key algorithm, meaning the same key is used for both encrypting and decrypting the data. In the United States, AES was announced by the NIST as U.S. FIPS PUB 197 (FIPS 197) on November 26, 2001. This announcement followed a five-year standardization process in which fifteen competing designs were presented and evaluated, before the Rijndael cipher was selected as the most suitable (see Advanced Encryption Standard process for more details). AES is included in the ISO/IEC 18033-3 standard. AES became effective as a U.S. federal government standard on May 26, 2002, after approval by the U.S. Secretary of Commerce. AES is available in many different encryption packages, and is the first (and only) publicly accessible cipher approved by the U.S. National Security Agency (NSA) for top secret information when used in an NSA approved cryptographic module (see Security of AES, below).

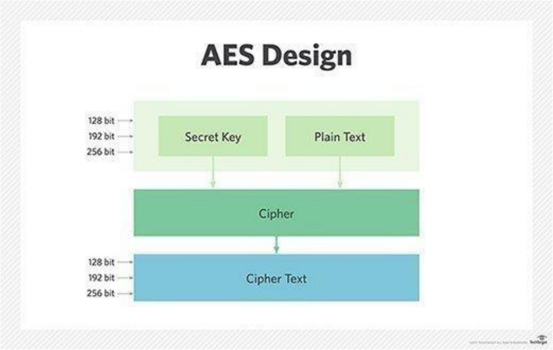
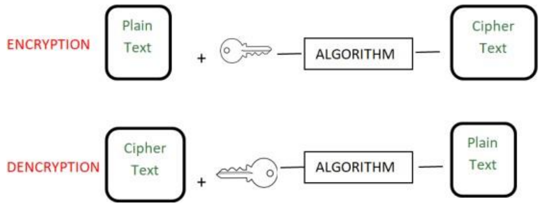


Fig. 4.14 AES ENCRYPTION

HASH ENCODING

Encryption, encoding and hashing, these terms are commonly interchanged and used incorrectly; knowing the differences, when and why to use each is important. Organizations have had breaches that sourced back to using the wrong data transforming method, and have gotten flack when using the terms incorrectly in their press releases as it indicates they are not knowledgeable and potentially careless with user data. A prime example was a breach Adobe suffered.

Encoding Encoding data is a process involving changing data into a new format using a scheme. Encoding is a reversible process; data can be encoded to a new format and decoded to its original format. Encoding typically involves a publicly available scheme that is easily reversed. Encoding data is typically used to ensure integrity and usability of data and is commonly used when data cannot be transferred in its current format between systems or applications. Encoding is not used to protect or secure data because it is easy to reverse. An example of encoding is: Base64 Take a scenario where a user wants to upload a resume to a job application website and the web server stores Hashing Hashing involves computing a fixed-length mathematical summary of data, the input data can be any size. In contrast to encoding, hashing cannot be reversed. It is not possible to take a hash and convert it back to the original data. Hashing is commonly used to verify the integrity of data, commonly referred to as a checksum. If two pieces of identical data are hashed using the same hash function, the resulting hash will be identical. If the two pieces of data are different, the resulting hashes will be different and unique. As an example, say Alice wants to send Bob a file and verify that Bob has the exact same file and that no changes occurred in the transferring process. Alice will email Bob the file along with a hash of the file. After Bob downloads the file, he can verify the file is identical by performing a hash function on the file and verify the resulting hash is the same as Alice provided. An example of a hash function is: SHA512 In addition to verifying the integrity of data, hashing is the recommended data transformation technique in authentication processes for computer systems and applications. It is recommended to never store passwords and instead store only the hash of the "salted password". A salt is a random string appended to a password that only the authentication process system knows; this guarantees that if two users have the same password the stored hashes are different. When a user inputs a password to a web application, the password is sent to the web server. The web server then appends the salt to the password and performs a hash function on the password and a salt and compares this output hash with the hash stored in the database for the user. If the hashes match for that user, the user is granted access. Hashing ensures in the event of a breach, or malicious insider the original passwords can never be retrieved. Salting ensures that, if a breach does occur, an attacker cannot determine which users have the same passwords. Encryption Encryption is the process of securely encoding data in such a way that only authorized users with a key or password can decrypt the data to reveal the original. There are two basic types of encryption; symmetric key and public key. In symmetric key, the same key is used to encrypt and decrypt data, like a password. In public key encryption, one key is use to encrypt data and a different key is used to decrypt the data. Encryption is used when data needs to be protected so those without the decryption keys cannot access the original data. When data is sent to a website over HTTPS it is encrypted using the public key type. While encryption does involve encoding data, the two are not interchangeable terms, encryption is always used when referring to data that has been securely encoded. Encoding data is used only when talking about data that is not securely encoded. An example of encryption is: AES 256.



IV RESULT

Login page of the web application is shown below

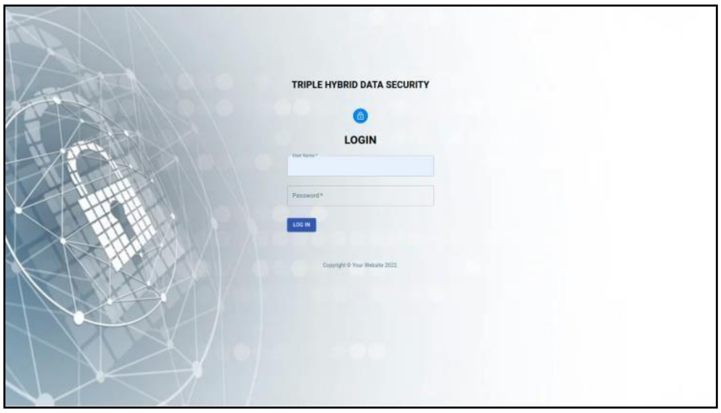
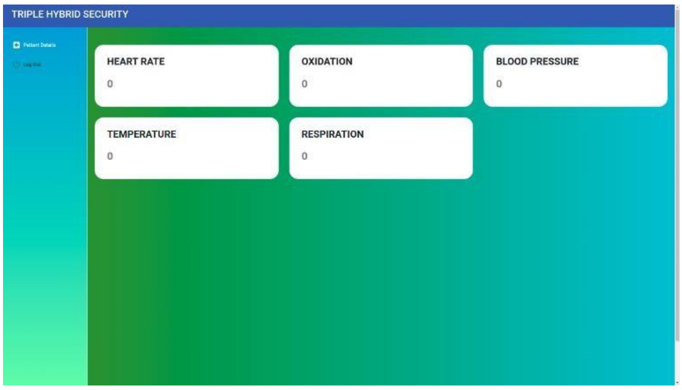


Fig 6.1 Login page

Home page for displaying the data from the sensors are shown in theimage below

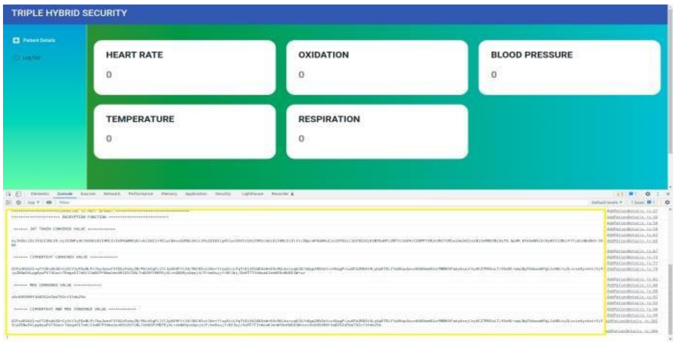


HEART RATE	OXIDATION	BLOOD PRESSURE
0	0	0

TEMPERATURE	RESPIRATION
0	0

Fig 6.2 Home Page

Encryption process using triple algorithm is shown below



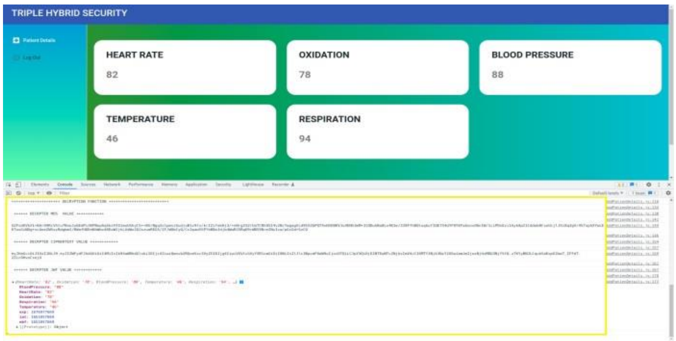
HEART RATE	OXIDATION	BLOOD PRESSURE
0	0	0

TEMPERATURE	RESPIRATION
0	0

```
++++++++++++++++++++ ENCRYPTION FUNCTION ++++++++++++++++++++++++
++++++ JWT TOKEN CONVERED VALUE ++++++++++++
[illegible]
++++++ CIPHERTEXT CONVERED VALUE ++++++++++++
[illegible]
++++++ MD5 CONVERED VALUE ++++++++++++
[illegible]
++++++ CIPHERTEXT AND MD5 CONVERED VALUE ++++++++++++
[illegible]
```

Fig 6.3 Encryption process

Decryption process using triple algorithm is shown below



TRIPLE HYBRID SECURITY

HEART RATE
82

OXIDATION
78

BLOOD PRESSURE
88

TEMPERATURE
46

RESPIRATION
94

Fig 6.4 Decryption Process

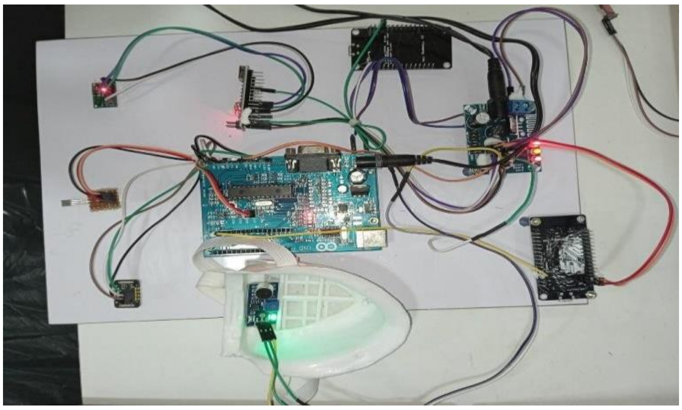


Fig 6.4 Hardware kit

A hardware component has been made with all necessary components and sensors like oxidation sensor, power supply,respiration sensor and blood pressure sensor.

V. CONCLUSION

In this project, we have primarily focused to provide an efficient data security system to secure the data stored and accessed in cloud. In the existing system, they have proposed SPADE, an encrypted data deduplication scheme that resists compromised key servers and frees users from the key management problem. But still it is lagging on theoretical treatment type of encryption algorithms. To overcome this problem this project presents an Efficient Data Security System where triple hybrid algorithm will be used to secure the data stored and accessed in cloud. The multiple sensors such as Heart rate sensor, blood oxidation sensor, temperature sensor, Blood pressure sensor and respiration sensors are used to collect the data.

VI. FUTURE SCOPE

In the coming future, we will review the application of this project and try to fetch more information from the user. It can be promoted for advance medical data security system with advanced features. So, there are more chances to develop or convert this project in many ways.

REFERENCES

[1] Parneet Kaur, Yogesh Kumar, Shakeel Ahmed, Abdulaziz Alhumam, Ruchi Singla and Muhammad Fazal Ijaz “Automatic License Plate Recognition System for Vehicles Using a CNN” IEEE transactions on intelligent transportation systems.

[2] Amir Hossein Ashtari, Graduate Student Member, IEEE, Md. Jan Nordin, and Mahmood Fathy “An Iranian License Plate Recognition System Based on Color Features” IEEE transactions on intelligent transportation systems.

[3] Chao Gou, Kunfeng Wang, Yanjie Yao, and Zhengxi Li “Vehicle License Plate Recognition Based on Extremal Regions and Restricted Boltzmann Machines” IEEE transactions on intelligent transportation systems

[4] G. L. Corneto, F. A. Silva, D. R. Pereira, L. L. Almeida, A. O. Artero, J. P. Papa, V. H. C. de Albuquerqueand H. M. Sapia “A New Method for Automatic Vehicle License Plate Detection” IEEE latin america transactions, vol. 15, no. 1, jan. 2017.

[5] Jingjing Zhang, Yuanyuan Li , Teng Li, Lina Xun, and Caifeng Shan “License Plate Localization in Unconstrained Scenes Using a Two-Stage CNN-RNN” IEEE sensors journal,vol. 19, no. 13, july 1, 2019.

[6] Lele Xie, Tasweer Ahmad, Lianwen Jin , Member, IEEE, Yuliang Liu, and Sheng Zhang “A New CNN-Based Method for Multi-Directional Car License Plate Detection” IEEE transactions on intelligent transportation systems.

[7] Michael D. Kim∗ and Jun Ueda, Member, IEEE “Dynamics-based motion de-blurring improves the performance of optical character recognition during fast scanning of a robotic eye” IEEE/ASME Transactions on Mechatronics.

[8] K. S Raghunandan, Palaiahnakote Shivakumara, Member IEEE, Hamid A. Jalab, Member IEEE, Rabha W. Ibrahim, Member IEEE, G. Hemantha Kumar, Umapada Pal, Senior Member IEEE and Tong Lu, Member IEEE “Riesz Fractional Based Model for Enhancing License Plate Detection and Recognition” IEEE Transactions on Circuits and Systems for Video Technology.

[9] Rahim Panahi, Member, IEEE, and Iman Gholampour, Member, IEEE “Accurate Detection and Recognition of Dirty Vehicle Plate Numbers for High-Speed Applications” IEEE transactions on intelligent transportation systems.

[10] Sergio Montazzolli, Claudio Jung “Real-Time Brazilian License Plate Detection and Recognition Using Deep Convolutional Neural Networks” 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images

[11] MV Karthikeyan,J Manickam and Martin Leo “Security issues in wireless body Area Networks: In Bio- signal Input Fuzzy Security Model : A survey” .RJPBCSresearch journal of pharmaceutical biological & chemical sciences.