Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 5, **Number** 1, May 2025
https://doi.org/10.47709/brilliance.v5i1.5179

Analysis of Most Significant Bit (MSB) Steganography Method in an IoT-Based Keyboard Learning System

Darinda Sofia Tanjung[1], **Yunita Sari Siregar**[2]*, **Juliana**
[1,3]Universitas Katolik Santo Thomas, Indonesia, [2]Politeknik Negeri Medan, Indonesia
[1]darinda_tanjung@ust.ac.id, [2]yunitasarisiregar@polmed.ac.id, [3]anna.jait@gmail.com

open access

***Corresponding Author**

Article History:
Submitted: 25-12-2024
Accepted: 25-01-2025
Published: 08-04-2025

Keywords:
Steganography; Most Significant Bit (MSB); Internet of Things (IoT); Data Security; Learning System; Sensor.

ABSTRACT

The advancement of digital technology has driven the development of Internet of Things (IoT)-based learning systems to enhance efficiency and interactivity in the teaching and learning process. IoT implementations is the development of a sensor-based keyboard learning system. This system is designed to help users improve their typing skills through monitoring parameters such as typing speed, keypress patterns, and usage duration. This data becomes an important asset that requires protection due to its connection to user privacy and personal performance evaluation. In IoT-based systems, data is typically transmitted through networks that are potentially vulnerable to security threats, such as eavesdropping, modification, or theft of data or information. The development of this system is carried out in a sensor-based keyboard learning application with IoT integration, which is equipped with data security features using steganography techniques. Steganography is a technique used to hide data or information within a digital medium so that it cannot be accessed by others. The method used is the Most Significant Bit (MSB), a simple steganography algorithm that utilizes significant bits from image files to embed secret messages. This research aims to analyze the effectiveness of the MSB method in protecting user data stored and transmitted through IoT system. The implementation is carried out on a sensor-based keyboard learning system, where the data collected from the sensors is used to track typing speed, keypress patterns, and typing duration. This data is encrypted using MSB steganography and embedded into a cover image before being transmitted through the IoT network.

INTRODUCTION

The Internet of Things (IoT) technology has rapidly advanced and opened up new possibilities in various sectors, including education. One of the applications of IoT in education is the use of connected devices to support the learning process in a more efficient and interactive way. One example of IoT implementation is in a sensor-based keyboard learning system. This system allows users, especially students, to improve their typing skills by utilizing sensors that can measure typing speed, accuracy, keypress patterns, and the duration of each practice session. With the advancement of this technology, the data generated becomes highly valuable and crucial to keep confidential. The development of knowledge to secure data is continually improving to ensure that technology users always feel safe. Data storage activities and digital information exchange come with risks and must, of course, be accompanied by an information security system. Various methods are used to protect data, such as hiding data through steganographic techniques or encoding data into codes that are difficult to understand. This way, if the data is stolen or intercepted by others, it will be difficult for them to know and understand the actual information (Rahmatsyah, Siregar, & Khairunnisa, 2024).

The word "steganography" comes from the Greek words "steganos," meaning "hidden or covered," and "graphein," meaning "to write." Therefore, it can be concluded that steganography is the science and art of writing or hiding a message within a medium in such a way that the existence of the message is unknown or unnoticed by anyone other than the sender and the recipient of the message (Wahyuningsih, Pandex, & Stefanny, 2017). Steganography is a technique used to hide information within data without revealing the existence of that information or causing suspicion that the data containing the hidden message has been altered. The current implementation of steganography uses digital media as a container or concealment medium for messages, one of which is image media (digital image) (Febriani & Irawati, 2016). Steganography requires two conditions: a carrier medium to hide the data and the secret data that will be concealed. The carrier media can include audio, images, video, or text. The use of steganography is one of the methods that can be employed to protect highly sensitive data, as such data is confidential and needs to be kept secure (Ro'isa & Suartana, 2019). Steganography involves the combination of three primary colors: Red, Green, and Blue (RGB). In a 24-bit color image, each pixel is represented by three bytes, where each byte corresponds to one of the primary colors: Red, Green, and Blue. The process of embedding a message into a cover image is called encoding, while extracting the message from the stego image is called decoding. Both processes require a secret key (stego key), ensuring that only the

E-ISSN : 2807-9035
Volume 5, **Number** 1, May 2025
https://doi.org/10.47709/brilliance.v5i1.5179

party with the secret key can perform the embedding and extraction of the message (Ramadhani, Siregar, & Siregar, 2018). Steganography is a method used to hide information or messages within other media such as digital images, text, sound, or video (Rosmala & K, 2018).. A digital image is an image that has been stored in a file format, allowing it to be processed using a computer. Digital images are used in various fields to assist humans in their work (Setyansyah, Siregar, & Khairani, 2019). The web is a system related to documents used as a medium to display text, images, multimedia, and other content on the internet (Syahputra, Khairani, & Siregar, 2023).

LITERATURE REVIEW

Some previous studies that underlie this research include a study titled *Method of Digital Image Steganography*. The conclusion of this study is that steganography is a method where digital images are the most commonly used media for transmitting data over the internet. Based on the analysis, the advantage of the Most Significant Bit (MSB) method is its resilience against contrast changes in the image, making the message readable even with minor distortions. However, its weakness is that it has a large Mean Squared Error (MSE), meaning that the quality of the image before and after embedding the message is significantly different (Ratnasari & Dwiyanto, 2020). The next research, titled *"Monitoring of Tilapia Fish Pond Based on IoT with Ammonia, Temperature, Height, and pH Sensors"*, concluded that this system is capable of monitoring the conditions of the pond and the surrounding environment that affect the development of the fish. Additionally, the system can be monitored via the internet using the Thingspeak API. (Soambaton, Djuniadi, & Al-Azhari, 2024). The Internet of Things (IoT) is a technological revolution that represents the future of computing and communication, with its development relying on the dynamic innovation of technology across various fields, from wireless sensors to nanotechnology. With IoT, previously disconnected objects can now communicate and exchange data automatically, opening new opportunities in various sectors, including education. It has the potential to enhance the learning experience by connecting students with the learning environment.(Zaini, 2024)

There are two common/popular types of keyboards used in computer devices: the QWERTY keyboard and the DVORAK keyboard. The difference between these two types lies in the arrangement of the letters on the keyboard. While the QWERTY layout is the standard used in the United States (US), the DVORAK layout is more commonly used in Europe. Generally, divides the keyboard structure into four categories: Typing Keys, Number Keys, and Control Keys. This study will use the QWERTY keyboard as it is widely used in Indonesia. In terms of the keys used, the development of the keyboard has not progressed significantly since its invention, with the primary changes being the addition of some auxiliary keys that help speed up the opening of application programs. There are two main principles in mastering the 10-finger typing technique (Al Rozi, Indrawati, & Widodo, 2018):

1. Each Finger Handles Specific Letters: Each finger is assigned to handle certain letters, ensuring an efficient and accurate typing process.
2. Proper Positioning: The fingers should be placed in a standby position. In the 10-finger typing technique, the keyboard layout consists of five rows, starting from the top row and going down to the bottom row. Proper finger placement on these rows is crucial for optimal typing efficiency.

Here is the finger placement on the key rows from the "10-Finger Typing Skills Module" by Depdiknas (Al Rozi et al., 2018) :

1. Left Hand:
   a. Pinky Finger:
      First row: ’ and 1
      Second row: Tab and Q
      Third row: Caps Lock and A
      Fourth row: Left Shift and Z
      Fifth row: Ctrl and Windows
   b. Ring Finger:
      First row: 2
      Second row: W
      Third row: S
      Fourth row: X
   c. Middle Finger:
      First row: 3
      Second row: E
      Third row: D
      Fourth row: C
   d. Index Finger:
      First row: 4 and 5
      Second row: R and T
      Third row: F and G

E-ISSN : 2807-9035
Volume 5, **Number** 1, May 2025
https://doi.org/10.47709/brilliance.v5i1.5179

Fourth row: V and B

e. Thumb:
Fifth row: Left Alt and Spacebar

2. Right Hand:
   a. Thumb:
   Fifth row: Right Alt and Spacebar
   b. Index Finger:
   First row: 6 and 7
   Second row: Y and U
   Third row: H and J
   Fourth row: N and M
   c. Middle Finger:
   First row: 8
   Second row: I
   Third row: K
   Fourth row: ,
   d. Ring Finger:
   First row: 9
   Second row: O
   Third row: L
   Fourth row: .
   e. Pinky Finger:
   First row: 0, -, =, and Backspace
   Second row: P, [, and]
   Third row: ;, ', and Enter
   Fourth row: / and right Shift

Steganography is a technique aimed at hiding secret messages or writings in such a way that the hidden information cannot be identified by others (third parties). In other words, only the sender and the receiver are able to know the message (Yanti & Budayawan, 2023). The steganography technique continues to evolve along with the advancements in time and technology. One example of this is the use of watermarking. In the present day, steganography also involves computer technology. In computer technology, data security through steganography can be achieved in two ways [5]. The first method involves a single file as the media or carrier file. The second method involves two files: one containing the secret data to be hidden and the other being the media or carrier file (Anwar, 2017). The media used for embedding secret messages in digital steganography techniques include (Permana & Amna, 2022):

1. Text. In steganography algorithms that use text as the medium, techniques such as Natural Language Processing (NLP) are often used to ensure that the text, after being embedded with the secret message, does not appear suspicious to the reader. Examples of text formats include text files, HTML, PDF, etc.
2. Audio. This format is frequently chosen because audio files are generally large in size, allowing them to store a substantial amount of secret data. Common audio formats include WAV, VOC, MP3, etc.
3. Image. The image format is the most commonly used because it is one of the most frequently exchanged file formats on the internet. Additionally, many steganography algorithms are available for image-based media. Examples of image formats include BMP, GIF, PCX, JPEG, etc.
4. Video. Although video files are very large, they are rarely used for steganography due to their size, which reduces practicality, as well as the lack of sufficient algorithms to support this format. Examples of video formats include MPEG, AVI, etc.

In general, there are two processes in steganography: the embedding process, which involves inserting a message into the cover-object, and the decoding process, which involves extracting the message from the stego-object. Both processes may require a secret key called the stego-key to ensure that only authorized parties can perform the embedding and extraction of the message.The hiding of secret data within digital media alters the quality of the media. The criteria that must be considered when hiding data include (Permana, 2021):

1. Fidelity: The quality of the cover image should not change significantly. After embedding the secret data, the stego-image should still appear clearly, and the observer should not be aware that there is secret data hidden within the image.
2. Robustness: The hidden data must be resistant to manipulation of the cover image, such as changes in contrast, sharpening, cropping, rotation, enlargement, encryption, and other image processing operations. If any image processing operations are performed, the hidden data should not be corrupted.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 5, **Number** 1, May 2025
https://doi.org/10.47709/brilliance.v5i1.5179

3. Recovery: The hidden data must be recoverable. Since the goal of steganography is data hiding, the secret data within the cover image must be retrievable at any time for further use.

IoT technology has empowered environmental monitoring systems to become more flexible, efficient, and cost-effective (Setiawan, Hiswara, & Muthmainah, 2023). The Internet of Things (IoT) is a technological revolution that represents the future of computing and communication, with its development relying on the dynamic innovation of technology across various fields, from wireless sensors to nanotechnology. With IoT, previously disconnected objects can now communicate and exchange data automatically, opening new opportunities in various sectors, including education. It has the potential to enhance the learning experience by connecting students with the learning environment (Zaini, 2024).

METHOD

In this study, the methods used to secure user fingerprint data are the Most Significant Bit (MSB) insertion method. This example aims to explain the step-by-step process for securing fingerprint data using the MSB steganography method

Fingerprint Data**:** E3 F7 03 A5 F2 BD 23 57 F8 C9 1A 3B C2 F7

To embed the data into a digital image, the fingerprint data must first be converted into binary format. The binary conversion of the fingerprint data is shown below:

E3 = 11100011
F7 = 11110111
03 = 00000011
A5 = 10100101
F2 = 11110010
BD = 10111101
23 = 00100011
57 = 01010111
F8 = 11111000
C9 = 11001001
1A = 00011010
3B = 00111011
C2 = 11000010
F7 = 11110111

Next, arrange the fingerprint data into a single row:

11100011 11110111 00000011 10100101 11110010 10111101 00100011 01010111 11111000 11001001 00011010 00111011 11000010 11110111

Next, prepare the digital image that will be used to embed the fingerprint data. It is important to note that the digital image must meet the minimum pixel size requirement. In the case mentioned above, there are 112 bits of data to be embedded. In MSB (Most Significant Bit) steganography, one bit of data is embedded into each RGB (Red, Green, Blue) component of a pixel. Since each pixel consists of three color components (R, G, B), a single pixel can store 3 bits of data. To determine the minimum number of pixels required, the following steps can be followed:

Original data : 112 bit
End Marker : 8 bit
Total bits : 112 + 8 = 120 bit
Number of pixels required : 40 pixels

So, from the calculation above, it is determined that the minimum resolution of the image used must have at least 40 pixels. The resolution that can accommodate 40 pixels is 6 x 7 pixels (42 pixels). It is assumed that the digital image to be used as the image cover is as follows. It can be seen in table 1.

Table 1. Pixel Values of the Image Cover

Pixel	Red	Green	Blue
1	10001100	11011010	01101111
2	01011101	10101101	11010011
3	00100101	10011110	01000111
…	…	…	…
And so on until pixel 40			

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 5, **Number** 1, May 2025
https://doi.org/10.47709/brilliance.v5i1.5179

After obtaining the pixel values for the cover image, the next step is to insert each bit of the fingerprint data into the first bit of each pixel. The detailed calculation is as follows:

1. Pixel 1:
   Fingerprint data bits:
   11100011 11110111 00000011 10100101 11110010 10111101 00100011 01010111 11111000 11001001 00011010 00111011 11000010 11110111
   Take the first 3 bits (bolded): 1, 1, 1. Replace the Most Signoficant Bit of each color component:
   Red : 10001100 → 11001100
   Green : 11011010 → 11111010
   Blue : 01101111 → 11101111

The value on the left represents the original pixel value, while the value on the right represents the pixel value after the fingerprint data has been embedded.

2. Pixel 2:
   Fingerprint data bits:
   111**000**11 11110111 00000011 10100101 11110010 10111101 00100011 01010111 11111000 11001001 00011010 00111011 11000010 11110111
   Take the next 3 bits: **0, 0, 0**. Replace the Most Siginificant Bit of each color component:
   Red : 01011101 → 01011101
   Green : 10101101 → 00101101
   Blue : 11010011 → 01010011

3. Pixel 3:
   Fingerprint data bits:
   111000**11** **1**1110111 00000011 10100101 11110010 10111101 00100011 01010111 11111000 11001001 00011010 00111011 11000010 11110111
   Take the next 3 bits: 1, 1, 1. Replace the MSB of each color component:
   Red : 00100101 → 10100101
   Green : 10011110 → 10011110
   Blue : 01000111 → 11000111

Continue the process as described above until pixel 40. The result of the data embedding using the Most Significant Bit (MSB) method can be seen in table 2.

Table 2 Results of Data Embedding Using the MSB Method

Pixel	**Red**	**Green**	**Blue**
1	10001100	11011010	11101111
2	01011101	00101101	01010011
3	10100101	10011110	11000111
…	…	…	…
And so on until pixel 40.			

RESULT

In conducting this research, the researcher first analyzed the currently existing system. Existing applications generally focus on typing training by emphasizing speed and accuracy but rarely consider the proper and technical use of fingers. The available systems only assess typing speed and the number of typing errors without evaluating whether users are using the correct fingers for each key pressed. This causes users to rely more on their own comfort, often involving improper finger use, which prevents them from achieving optimal ten-finger typing skills. Several typing training applications do provide basic instructions on finger usage. However, in practice, validation systems that effectively ensure the correct fingers are used for each key have yet to be implemented. For example, users can complete typing exercises using fingers that do not conform to the standard ten-finger typing rules but still achieve high scores as long as speed and accuracy criteria are met. This highlights a gap in the validation system, which should also prioritize proper finger use according to the ideal positions on the keyboard.

Without a feature to validate proper finger use, users tend to develop inefficient habits that can lead to finger fatigue and reduced productivity in the long term. Relying on systems that only measure typing speed and accuracy without addressing proper technique also fails to provide adequate feedback for users to correct their finger positions and usage. As a result, users may type quickly but not efficiently or correctly, which could lead to errors or discomfort in the future. From this analysis, it is clear that existing systems have yet to address the need to ensure users optimize their ten-finger typing skills according to the proper function of each finger on the keyboard. Therefore, a more advanced and comprehensive system is needed—one that not only evaluates typing speed and accuracy but also automatically validates finger use for each key pressed. Such a system would help users improve their typing technique

E-ISSN : 2807-9035
Volume 5, **Number** 1, May 2025
https://doi.org/10.47709/brilliance.v5i1.5179

and achieve truly effective ten-finger typing skills. As a result, users would not only type quickly but also with proper and efficient techniques.

After analyzing the existing system, a more in-depth problem analysis was conducted. The system analysis revealed that many users are unable to utilize all ten fingers correctly when typing on a keyboard. This issue arises from various factors, one of which is the development of bad habits from the early stages of learning to type. Users tend to focus more on improving their typing speed rather than paying attention to proper technique. As a result, although several typing applications are available, they have not been entirely effective in teaching the correct ten-finger typing method. This problem is exacerbated by the lack of a validation system to ensure that the fingers used for typing correspond to their proper positions on the keyboard. Even when users attempt to use all ten fingers, their technique often lacks consistency and efficiency. Users who type inconsistently—whether in terms of the sequence or the positioning of their fingers—end up creating suboptimal typing patterns. This leads to overexertion of certain fingers while others remain underutilized, potentially causing long-term issues such as muscle fatigue or even injury.

One of the primary challenges identified is the lack of user awareness regarding the importance of proper typing techniques, particularly the balanced use of all ten fingers. Users often feel satisfied with their typing performance as long as speed and accuracy are achieved, without considering the long-term effects of improper finger usage. For example, many users frequently rely on their index fingers and thumbs while neglecting the other fingers during typing. This is a serious issue because such incorrect typing methods, while possibly quick, prevent users from realizing their full potential in typing more efficiently and without the risk of injury. The inability of existing applications to validate finger usage is another critical concern. Although many typing training applications provide instructions on finger usage, they lack concrete feedback on whether users are actually employing the correct fingers as per the rules. These applications typically measure only speed and typing errors but fail to validate whether the typing is performed with proper techniques. As a result, users may repeatedly make the same mistakes without realizing they are not typing efficiently or ergonomically using all ten fingers. Therefore, it is crucial to develop a system capable of automatically validating finger usage during typing. Such a system would not only help users refine their typing techniques but also provide clear feedback on incorrect finger usage and areas for improvement. Consequently, users would receive more comprehensive guidance in enhancing their ten-finger typing skills effectively and efficiently.

DISCUSSION

In conducting a needs analysis for the system to be developed, the system requirements are divided into two main parts: functional requirements analysis and non-functional requirements analysis. The functional requirements analysis focuses on the features that the system must provide to achieve the desired outcomes, while the non-functional requirements analysis covers technical aspects such as the hardware and software needed to support the system's implementation. Both aspects must be fulfilled to ensure the system operates effectively and aligns with its objectives. The functional requirements of the system are as follows:

1. The system can validate the correct finger usage when pressing keyboard keys: The system must be able to detect which finger presses a particular key on the keyboard and ensure that the finger used complies with the ten-finger typing rules. This validation is crucial to provide feedback to users on their typing techniques. The system should also identify incorrect finger usage and provide corrective suggestions so users can improve their typing techniques in real-time.
2. The system can validate user typing accurately: The system must have the capability to monitor and validate the accuracy of users' typing in terms of both speed and precision. As users type, the system will record typing errors and issue alerts based on expected typing standards. Additionally, the system should present statistics on users' progress, such as error rates during typing quizzes.
3. The system provides real-time feedback: This feature allows users to receive immediate feedback whenever they press the wrong key or use the incorrect finger. The feedback includes notifications indicating that the finger used does not comply with proper typing rules.
4. The system can track and store user data: The system must be capable of storing data related to users' typing sessions, such as the number of errors and typing speed during quizzes. This data can be used to generate progress reports for users and help them identify areas that need improvement.
5. The system provides training mode and quiz mode: To facilitate the improvement of typing skills, the system should have two main modes. In training mode, users are guided through tutorials on proper ten-finger usage with clear instructions. In quiz mode, users can test their abilities by typing text while still receiving feedback on speed, accuracy, and finger usage.

CONCLUSION

The conclusion of this research is that the Most Significant Bit (MSB) steganography method can be effectively implemented to protect user data in a sensor-based IoT keyboard learning system. The application of this method provides sufficient basic protection, although it still requires enhancement to address more complex data manipulation threats. Thus, the integration of IoT, steganography, and additional security technologies has great potential to create a

E-ISSN : 2807-9035
Volume 5, **Number** 1, May 2025
https://doi.org/10.47709/brilliance.v5i1.5179

secure, modern, and innovative learning system.

ACKNOWLEDGMENT

We would like to express our sincere gratitude and pride to the Ministry of Education, Culture, Research, and Technology (Kemdikbudristek) for their invaluable support in providing research funding through the 2024 Fundamental Research Scheme facilitated by the Directorate of Research, Technology, and Community Service.

REFERENCES

Al Rozi, U., Indrawati, C. D. S., & Widodo, J. (2018). Pemanfaatan program aplikasi typing master untuk meningkatkan kecepatan mengetik siswa SMK Wikarya Karanganyar Tahun 2018/2019. *Jikap*, *2*(4), 91–100. Retrieved from https://jurnal.uns.ac.id/JIKAP/article/download/38165/25178

Anwar, S. (2017). Implementasi Pengamanan Data Dan Informasi Dengan Metode Steganografi LSB Dan Algoritma Kriptografi AES. *Jurnal Format*, *6*(1), 65–74.

Febriani, S. R., & Irawati, D. C. (2016). Implementasi Digital Watermarking Pada Citra Menggunakan Metode Least Significant Bit. *Jurnal Informatika Dan Komputer*, *21*(3), 8–18. Retrieved from https://ejournal.gunadarma.ac.id/index.php/infokom/article/download/1522/128

Permana, A. A. (2021). Pengamanan Pesan Menggunakan Teknik Steganografi Dengan Algoritma Least Significant Bit (Lsb). *Jurnal Algoritma, Logika Dan Komputasi*, *3*(2), 291–298. https://doi.org/10.30813/j-alu.v3i2.2477

Permana, A. A., & Amna, H. (2022). Implementasi Steganografi File Citra Digital Menggunakan Metode Least Significant Bit. *Jurnal Teknik*, *11*(1), 62–72. https://doi.org/10.31000/jt.v11i1.6161

Rahmatsyah, I., Siregar, Y. S., & Khairunnisa. (2024). Proteksi Keamanan Data dengan Menerapkan Algoritma Bacon Cipher dan ROT128. *Journal of Computer Science and Information Technology*, *4*(1), 34–41. Retrieved from https://journal.fkpt.org/index.php/Explorer/article/view/1099

Ramadhani, R., Siregar, D., & Siregar, Y. S. (2018). Implementasi Steganografi Menggunakan Algoritma Diversity Pada Citra Digital. *Jurnal Teknologi Dan Ilmu Komputer Prima (JUTIKOMP)*, *1*(1), 102–114. https://doi.org/10.34012/jutikomp.v1i1.337

Ratnasari, A. P., & Dwiyanto, F. A. (2020). Metode Steganografi Citra Digital. *Sains, Aplikasi, Komputasi Dan Teknologi Informasi*, *2*(2), 52. https://doi.org/10.30872/jsakti.v2i2.3300

Ro'isa, F., & Suartana, I. M. (2019). Implementasi Steganografi dengan Menggunakan Metode Masking and Filtering untuk Menyisipkan Gambar ke dalam Citra Digital. *Journal of Informatics and Computer Science (JINACS)*, *1*(01), 9–15. https://doi.org/10.26740/jinacs.v1n01.p9-15

Rosmala, D., & K, A. K. (2018). Perbandingan Metode Most Significant Bit dan Least Significant Bit pada Steganografi untuk Keamanan Data Media Digital. *MIND Journal*, *3*(2), 36–46. https://doi.org/10.26760/mindjournal.v3i2.36-46

Setiawan, Z., Hiswara, A., & Muthmainah, H. N. (2023). Mengoptimalkan Jaringan Sensor Nirkabel dalam Aplikasi Monitor Lingkungan dengan Teknologi IoT di Indonesia. *Jurnal Multidisiplin West Science*, *2*(10), 858–867. https://doi.org/10.58812/jmws.v2i10.704

Setyansyah, R., Siregar, Y. S., & Khairani, M. (2019). Noise Removal Pada Citra Digital Dengan Menggunakan Metode Active Contour. *ALGORITMA: Jurnal Ilmu Komputer Dan Informatika*, *5*(1), 978–979. Retrieved from http://www.seminar.ilkom.unsri.ac.id/index.php/ars/article/view/2130

Soambaton, M. F., Djuniadi, D., & Al-Azhari, A. H. (2024). MONITORING KOLAM IKAN NILA BERBASIS IoT DENGAN SENSOR AMONIA, SUHU, KETINGGIAN, DAN PH. *Jurnal Informatika Dan Teknik Elektro Terapan*, *12*(2), 919–926. https://doi.org/10.23960/jitet.v12i2.4021

Syahputra, A., Khairani, M., & Siregar, Y. S. (2023). *PERANCANGAN WEB SERVER DAN PROXY SERVER DENGAN SISTEM OPERASI*. *9*(4), 34–43.

Wahyuningsih, S., Pandex, T. V. D., & Stefanny, V. (2017). Implementasi Visible Watermarking Dan Steganografi Least Significant Bit Pada File Citra Digital. *Jurnal TELEMATIKA MKOM*, *8*(2), 140–145.

Yanti, F., & Budayawan, K. (2023). Implementasi Steganografi Menggunakan Metode Least Significant Bit (LSB) dalam Pengamanan Informasi pada Citra Digital. *Voteteknika (Vocational Teknik Elektronika Dan Informatika)*, *11*(1), 63. https://doi.org/10.24036/voteteknika.v11i1.121968

Zaini, N. (2024). *Implementasi Internet of Things Pada Prototype*. *2*(1), 357–362.