

# Investigating the Influence of Temperature on UAV Signal Quality

Ahmed Hussein Abbas <sup>a,1</sup>, Ahmad Taha Abdulsadda <sup>a,2</sup>, Hassanain Ghani Hameed <sup>b,3</sup>

<sup>a</sup> Department of Communication Technical Engineering, Al-Furat Al-Awsat Technical University, Al Najaf, 540001, Iraq

<sup>b</sup> Engineering Technical College/ Najaf, Al-Furat Al-Awsat Technical University, Al Najaf, 540001, Iraq

<sup>1</sup> [ahmed.ms.etcn39@student.atu.edu.iq](mailto:ahmed.ms.etcn39@student.atu.edu.iq); <sup>2</sup> [coj.abdulsad@atu.edu.iq](mailto:coj.abdulsad@atu.edu.iq); <sup>3</sup> [hassanain.hameed@atu.edu.iq](mailto:hassanain.hameed@atu.edu.iq)

\* Corresponding Author

## ARTICLE INFO

### Article history

Received March 21, 2024

Revised June 05, 2024

Accepted June 25, 2024

### Keywords

Drone Communication

Systems;

Temperature Effect;

Signal-to-Noise Ration (SNR);

1DCNN;

RSSI

## ABSTRACT

Advancements in drone technology make them important in many areas. military, industry, and disaster The efficacy of a drone's communication systems can be greatly impacted by temperature fluctuations, either from environmental conditions or mechanical problems in the drone's construction. This study gives an analysis and computational model of the impact of temperature on the performance of drone communication. Utilizing a one-dimensional convolutional neural network, we aim to forecast the signal-to-noise ratio (SNR) and received signal strength indicator (RSSI). Following the initial stage of dataset creation in the drone laboratory, proceed to reprocess the dataset and divide it into a 70% training set and a 30% testing set. Subsequently, a graphical user interface (GUI) was developed using MATLAB App Designer to enhance user friendliness. The outcome suggests that the efficiency of the drone communication system declines with rising temperatures. Using 1DCNN is our contribution to this work; other studies depend only on simulation to assess performance. One benefit of 1DCNN is that the impact may be evaluated by automatically extracting important features from the input dataset. Using 1DCNN is our special addition to this project; other research evaluate the UAV communication system's effectiveness only through simulation. We propose in this work to optimize system characteristics for improved performance, including power transfer, by adding a feedback loop between the CNN result and the communication system. Furthermore, we investigate how different weather conditions, such wind and rain, affect UAV communication systems.

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



## 1. Introduction

Drones, or unmanned aerial vehicles, or UAVs, are now widely used and very significant in many different industries. Monitoring, environmental supervision, and civic security management are examples [1]. Fixed-wing and rotorcraft UAVs are the two primary subtypes of these aircraft. A major and rapid development of fixed-wing unmanned aerial vehicles (UAVs) with vertical takeoff and landing (VTOL) occurred between 2020 and 2022 [2]-[4]. By integrating electric or hybrid propulsion systems, it is possible to combine the capabilities of Aeroplan and multirotor UAVs [5]-[7]. Many unmanned aerial vehicles (UAVs) rely on an onboard autopilot system that utilizes small sensors, including accelerometers, gyroscopes, barometers, and GPS. Accurate determination of a UAV's

orientation and location is the initial stage in achieving precise machine control [8]. Different factors, including weather conditions, obstacles, distance, metal interference, radio interference, and frequency interference, affect the propagation of a drone's signal. Precipitation and high temperatures can affect signal transmission, drone electronics, and battery performance [9].

Fig. 1 [10] shows the use of drones in firefighting operations. We currently use drones to perform tasks like capturing overhead imagery, lifting fire hoses to tall buildings, or releasing fire retardant in distant locations to slow the rapid spread of wildfires. Nevertheless, existing drones designed for firefighting are typically incapable of flying in close proximity due to their inability to withstand the extreme temperatures, which would result in their melting and the failure of their electrical components. Conversations with firefighters revealed that drones, with their ability to approach buildings at a closer distance, could significantly aid in training first responders to access burning structures or wooded areas [11]-[15].

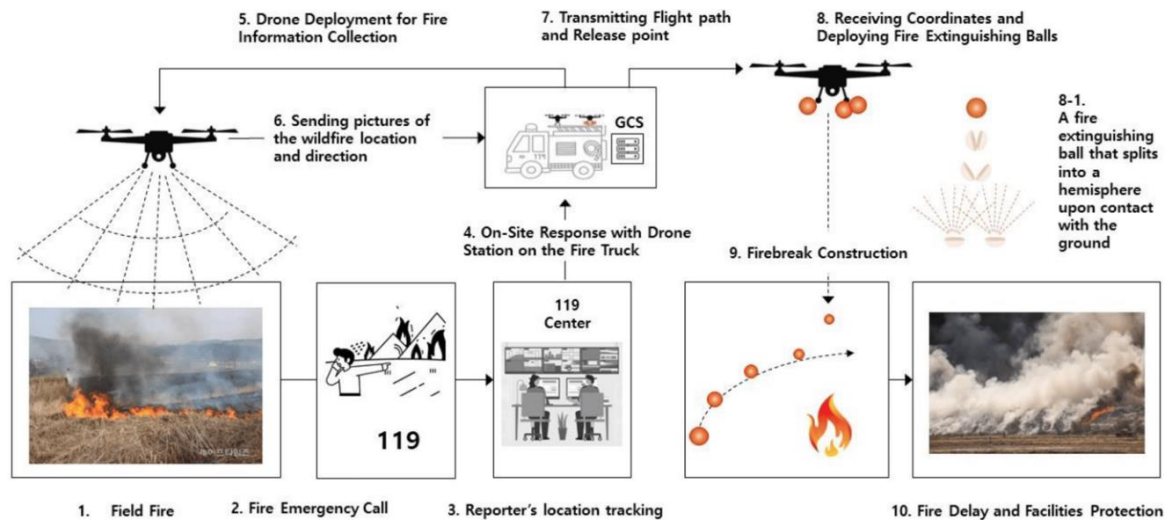


Fig. 1. Use of drones in firefighting

Signal strength is a crucial element. In addition, the drone may move erratically and lose signal strength due to the temperature effect on its communication system. Temperature change affects the drone communication systems, which cause signal attenuation and frequency offset and affect electronic components and batteries. Temperature effects from electromagnetic waves, which change temperature and alter density and refractive index in the atmosphere, Temperatures generate thermal noise, which can decrease the drone communication system's SNR. High temperatures may cause component failure, and low temperatures can decrease battery efficiency. The random motion of electrons generates thermal noise (Johnson-Nyquist), which degrades the drone communication system's performance, including SNR and RSSI. Thermal noise adds an additional noise factor to the received signal. The decrease in SNR can degrade the quality of signal and make it more difficult for the receiver to accurately decode the information from transmitted signal. Thermal noise reduces the SNR for the received signal so that thermal noise causes a limit to the communication system's performance. The increase in the SNR causes a decrease in the received signal strength RSSI.

Weather elements such as wind, air temperature, and atmospheric water content can mix in a variety of ways. Specific combinations can result in the reception of radio signals at distances exceeding the typical range of radio communications by several hundred miles [16]-[17]. Various elements that can impact the propagation of radio signals include geography, structures, plants, carbon dioxide (CO<sub>2</sub>) levels, and weather conditions. Several writers [18]-[20] have conducted studies on weather parameters, including rain, wind, temperature, and humidity. A team of Singaporean researchers focused on analyzing the synergistic impact of wind and rain, a common occurrence in tropical forests [21].

Many scholars examined different temperature sources that affect this unmanned aerial vehicle (UAV). Given that temperature has an impact on the functioning of fundamental components in

electrical and electronic circuits, it is customary in businesses to evaluate its influence on particular devices. Several studies have examined the effect of temperature on network operations. Jari Luomala [22] investigates the effect of temperature on signal intensity and measures the negative linear correlation between temperature and signal strength. Carlo [23] shows temperature has an effect on signal strength and link quality. They found that a lower temperature can reduce the power needed for reliable transmission by up to 16%.

Bannister et al. [24] confirmed by experiment the linear decrease in Received Signal Strength Indication (RSSI), which ranged up to 8 dB, when the temperature increased from 25°C to 65°C. They displayed the impact that this has on the range of communication and connectivity of the network. Wennerstrom et al. [25] showed a correlation between PRR (packet reception ratio) and RSSI with temperature, relative humidity, precipitation, and sunlight. Their findings indicate a strong correlation between PRR and RSSI and temperature. In addition to temperature, other meteorological factors such as humidity, precipitation, and snowfall have also garnered significant interest among academic communities.

The experiment conducted by Thelen et al. [26] determined that radio waves exhibit enhanced propagation in high-humidity conditions, particularly at night and after rainfall. In their study, Capsuto et al. [27] conducted trials and found that cold weather did not have a discernible effect on linkages. However, they noted that rainfall and overall moisture had a significant effect on RSSI. Fluctuations in meteorological conditions impact the strength of radio signals. Received signal intensity is subject to fluctuations due to multipath fading and propagation loss, with temperature being a known influential factor in the performance of radio transceivers. Elevated temperatures have a detrimental effect on signaling. Temperature-dependent transceiver features, like gain, can be utilized as physical mechanisms for measuring fluctuations. Transistors are essentially the crucial components that amplify signals in an amplifier. In CMOS transistors, the primary thermal effect is the reduction in electron mobility as temperature increases.

The inverse relationship between temperature and trans-conductance gain results in a decrease in gain and an increase in noise figure. Experimental findings [28] indicate that variations in signal strength are primarily attributed to changes in gain and noise figures. The decrease in signal strength readings at the transmitter side is a result of the combination of reduced gain and a decrease in the input signal level. On the other hand, the decrease in signal strength at the receiver side is caused by the thermal characteristics of the transmitter's power amplifier and the low-noise amplifier of the receiver. However, the decrease in RSSI is independent of the pace at which the temperature changes [29]. The studies conducted in references [24] and [29] demonstrated that it is possible to reduce the transmission power without compromising the network's performance. A decrease in temperature enhances the transmission of network signals, resulting in the development of an efficient transmission system with minimal power loss.

However, the influence of weather on other communication systems, primarily those operating at 2.4 GHz and frequently used outside, has been extensively researched [30]-[32]. Meteorological circumstances have a significant impact on radio signals, with the extent of this impact varying depending on the frequency range. In addition, empirical evidence from actual implementation in the field contradicts the advice provided by ITU-R. [33]-[35]. Hence, while the weather may not have a direct impact on radio waves operating at 2.4 GHz, it might potentially alter other ambient conditions that can influence the transmission of data in a WLAN. For instance, WLAN transmitters can suffer fluctuations in humidity and temperature during the duration of the day, resulting in varying performance. [36] Examine the effects of elevated temperatures on nodes in an industrial setting. The researchers conclude that it has a direct impact on their connection and that lower transmission power is needed in colder temperatures. As the temperature increases, the quality of the linkages decreases. A laboratory experiment in [37] validated the linear decline of the radio signal strength (RSS) when the temperature increased from 25 °C to 65 °C. The authors assessed the effects of reducing a node's communication range and demonstrated its substantial influence on network connectivity and services through simulations. Several researchers have recognized that humidity plays a crucial role in the performance of communication between sensor nodes. Their findings indicate that radio waves exhibit

enhanced propagation in environments characterized by elevated humidity levels, particularly during nighttime and rainy situations. A separate study [38] suggests that variations in air pressure and precipitation can have an impact on wireless signal propagation.

In this work, we investigate how temperature impacts drone communication systems, which may have an adverse effect on wireless communication system performance. We assess the effectiveness of the drone's communication system by measuring the signal-to-noise ratio (SNR) and the received signal strength indicator (RSSI). The CNN convolutional neural network makes predictions about these parameters. Once we train and test our model on the drone laboratory's generated dataset, the MATLAB App Designer generates an intuitive and user-friendly GUI. Unlike other works that solely rely on simulation to evaluate the effectiveness of the unmanned aerial vehicle UAV communication system, our work incorporates the use of 1DCNN. This study offers a useful technique for assessing a UAV communication system's effectiveness. We see that the performance of the drone communication system decreases with increasing temperatures. The following is the study: An overview of the literature is given in Section 1, along with an explanation of the methodology. The theoretical foundation of the temperature effect on communication systems and convolutional neural network (CNN) layers is then explained in Section 2. Examine the mathematical representations of the SNR (signal-to-noise ratio) and RSSI (received signal strength indicator) in Section 3. Section 4 then describes the techniques employed and explains the procedures and algorithm to experiment with the temperature on a drone using MATLAB 2021b in a convolutional neural network (CNN). The RSSI (Received Signal Strength Indicator) and SNR (Signal to Noise Ratio) findings across a range of temperatures are displayed in Section 5. A comprehensive discussion and recommendations based on the findings are presented in the final part.

## 2. Theoretical Background

This section briefly discusses temperature effect drone communication system and the Convolution Neural Network (CNN) layers.

### 2.1. The Temperature's Impact on Communication System

We employ the widely used basic metrics RSSI (Received Signal Strength Indicator) and SNR (Signal to Noise Ratio) to evaluate the strength of the connection [39]. The Receiving Strength of Signal Indicator (RSSI), a common feature of radio transceivers used in nodes, indicates the power of the received radio signal in a specific radio channel. Thermal noise is defined as random fluctuations in electrical signals caused by the system's thermal energy. Reduce the signal-to-noise ratio (SNR) of the received signal to ensure that thermal noise primarily limits the performance of the communication system. An increase in the signal-to-noise ratio (SNR) results in a reduction in the received signal strength (RSI) [40]-[41].

Weather fluctuations affect the strength of radio transmissions. The temperature has a discernible impact on the strength of the received signal, which fluctuates over time as a result of multipath fade and propagation loss. It significantly influences the performance of radio transmitters. Elevated temperatures have an adverse impact on signals. Temperature-dependent transceiver features, such as gain, can be used as physical mechanisms to measure fluctuations.

Temperature-dependent reductions in electron mobility are the primary factor that affects CMOS transistors. These reductions are critical for providing an amplifier with amplification gain. The negative correlation between temperature and trans-conductance gain leads to a reduction in the latter, resulting in an increase in the noise figure. In general, the signal strength (RSSI) tends to decrease as the temperature increases, and vice versa.

This implies that there is an inverse correlation, or reliance, between signal strength and temperature. Weather fluctuations affect radio signals' power. Temperature is the primary factor that influences changes in signal intensity, and it typically has a negative, linear impact on signal strength. Nevertheless, elevated relative humidity can also exert an influence, particularly when the temperature falls below 0 °C.



Moreover, the relationship between climatic conditions and signal intensity differs depending on the specific radio channel and link. We can alleviate these effects by employing a diverse range of frequencies. Ultimately, reducing the transmit power leads to a stronger correlation with meteorological parameters, minimizing the unpredictable fluctuations in received signal intensity.

## 2.2. Convolution Neural Network (CNN)

In the field of deep learning, the Convolutional Neural Network (CNN) is well recognized as the most renowned and often utilized algorithm [42]-[46]. One significant advantage of CNN over its predecessors is its ability to autonomously detect key elements without the need for human supervision [47]. Convolutional neural networks (CNNs) have been widely utilized in several domains, such as computer vision [48], voice processing [49], and face recognition [50]. CNNs were designed based on the neuronal structure found in animals and human brains, resembling that of a traditional neural network. To be more precise, the visual cortex in a cat's brain is composed of an intricate arrangement of cells, which is replicated by CNN's algorithm [51]. Goodfellow et al. [52] have recognized three primary advantages of the Convolutional Neural Network (CNN): comparable representations, sparse conversations, and sharing of parameters. CNNs utilize weight sharing and local connections to effectively utilize the 2D structures of input data, such as visual signals, in contrast to traditional fully connected (FC) networks. This method employs a minimal number of parameters, which not only simplifies the training process but also accelerates the network. This is identical to the cells seen in the visual cortex. These cells have the ability to perceive only tiny parts of a scene instead of the entire scene. In other words, they extract the local correlations present in the input, similar to local filters over the input.

A commonly used type of CNN, resembling a multi-layer perceptron (MLP), includes many convolution layers followed by sub-sampling (pooling) levels, with the last layers having fully connected (FC) layers. Fig. 2 depicts a case of CNN architecture used [53].

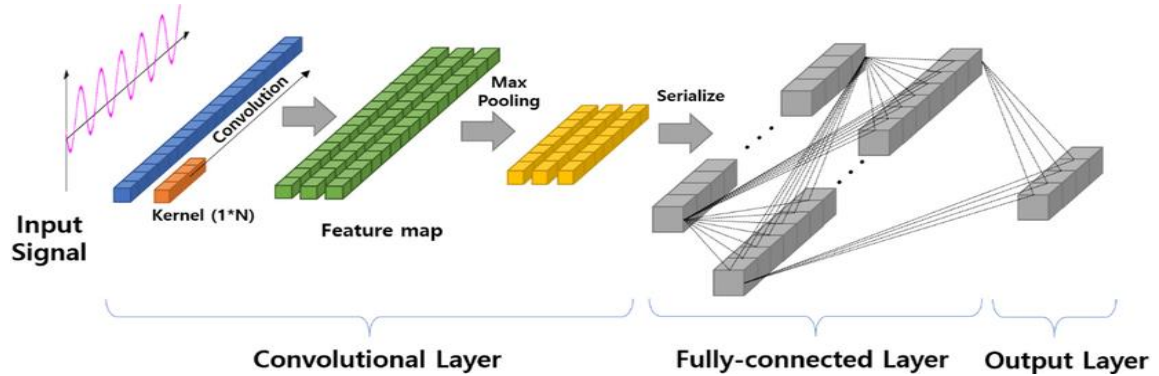


Fig. 2. CNN with four layer

A conventional two-dimensional convolutional neural network (2DCNN) is specifically designed to exploit the spatial characteristics present in 2D images. It achieves this by utilizing locally connected convolutional filters with tied weights, which operate on multiple pixels simultaneously rather than a single pixel [54]-[55]. This approach enhances the ability to detect the interdependencies among pixels. In a 2-dimensional convolutional neural network (2DCNN), the input data is initially transformed into 3-dimensional data, with dimensions for width, height, and depth. The depth is set to 1 for a single-band image and 3 for a three-band image representing the red, green, and blue channels. Next, a feature map is generated through the repeated utilization of convolution operators on sub-regions of the complete picture. This process involves the addition of a bias term, followed by the application of a non-linear activation function. The convolutional process of a convolutional neural network (CNN) is represented by an Equation (1).

$$x_l^j = f \left( \sum_{i \in P_n} x_i^{j-1} * w_{il}^j + b_l^j \right) \quad (1)$$

Where  $l$ : no. of layers;  $j$  component;  $w$ : weight;  $b$ : basis

Adding a pooling layer into the CNN enhances the quality of features obtained from the convolutional layer represent in Equation (2), as discriminant features have an essential role in accurate classification by a classifier.

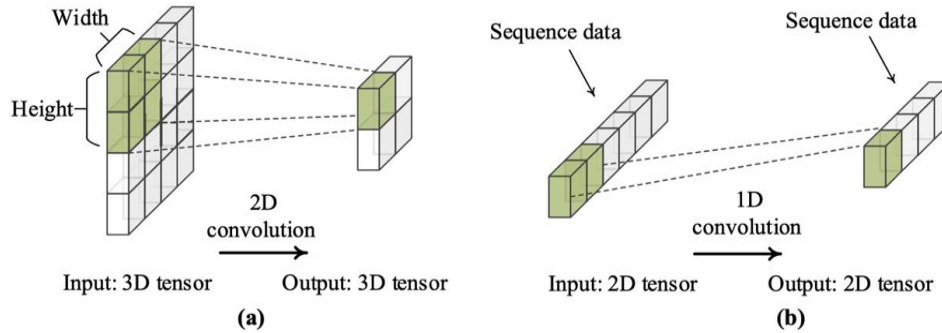
$$x_l^j = f(w_l^j * \max(x_l^{j-1}) + b_l^j) \quad (2)$$

Where  $x_l^j$ : the max pooling operation. CNN utilizes multiple layers of convolution and pooling to improve its capacity to extract localized information from the input data. The CNN employs pooling and convolutional layers to extract latent information and subsequently categorize the data into suitable categories. For this purpose, CNN uses a fully connected layer. Equation (3) represents a fully connected layer that classifies the latent characteristics obtained using the convolution kernel [56].

$$fx^{j+1} = \left( \sum_{i=1}^n w_{il}^j a^{j(i)} + b_l^j \right) \quad (3)$$

The standard deep convolutional neural networks (CNNs) discussed in the preceding section are specifically designed to function primarily. However, when dealing with one-dimensional time-series data, such as the acceleration data mentioned in this research, the 1D Convolutional Neural Network (1DCNN) is typically a more optimal option [57].

Fig. 3 illustrates the distinction between 2DCNN and 1DCNN. When 2DCNN is applied to a 2D image, it produces a 2D image. On the other hand, when 1DCNN is applied to a 1D image, it generates a 1D image. The 1DCNN's convolutional filter is designed to operate on one-dimensional data, allowing it to identify the relationships and connections within the data.



**Fig. 3.** Different between (a) 1DCNN vs (b) 2DCNN

In order to identify temperature impacts, we used a one-dimensional convolutional neural network (1D CNN) in this investigation. The feature extraction process includes the use of CNN. We used the dataset that the drone laboratory provided to create 1DCNN. We employed the subsequent methodology, splitting the original dataset into a training set and a validation set to create a new dataset. We then trained the dataset for regression using 1DCNN in MATLAB. We then calculated performance metrics like mean squared error, root mean absolute error, mean absolute error, and relative error. Finally, we used App MATLAB Designer to transform the code into an accessible and user-friendly platform.

### 3. Mathematical Model

As we mention in previous section there is a relation between RSSI and SNR. The formula of SNR is Equation (4):

$$SNR = \frac{P_r}{P_s} \quad (4)$$

$$P_s = K.T.\nabla F \quad (5)$$

where:

SNR: Signal-to-noise ratio

$P_r$ : Received signal power (W)

$P_s$ : Noise power (W)

$K$ : Boltzmann constant ( $1.38 \times 10^{-23}$  J/K)

$T$ : Temperature (K)

$\nabla F$ : Bandwidth (Hz)

Received Signal Strength Indicator (RSSI) Equation (6):

$$RSSI = P_t - P_l(d) + G_t + G_r \quad (6)$$

where:

RSSI: Received Signal Strength Indicator

$P_t$  Power transmits

$P_l(d)$ : Path loss

$G_t$ : Gain transmits

$G_r$ : Gain received

## 4. Methodology

We will use the following technique for predicting the (RSSI and SNR) temperature impact on the drone's communication system:

### 4.1. One-Dimensional Convolution Neural Networks (1DCNN)

Simulated-based theory models are usually used to find out how temperature impacts drone transmission systems. The data we have collected will be used for training and testing our model using DL (deep learning). Following that, we can utilize it with various types of information. There are many uses for deep learning, which is a type of artificial intelligence. [58]-[59]. As element of the convolutional neural system (CNN), artificial neural networks (ANN) have fewer factors. [60]-[61]. The convolutional neural network (CNN) is a well-known deep learning algorithm that can be learned from data without a person having to separately identify features. [62]-[66].

Once the CNN design has been planned, Follow the steps in the flow chart of 1DCNN in Fig. 4 to process the information. The information that the drone lab gave us was what we used to make 1DCNN. This method was used: To make a new dataset, the old one was split into a training set and a validation set. After that, 1DCNN in MATLAB was used to train the dataset for regression. Once the training was over, success measures such as the mean squared error.

#### 4.1.1. Dataset Collection from Drone Laboratory

We collected a dataset in the drone lab (laboratories of the Industrial Research and Development Directorate/Electronic Manufacturing Centre), as illustrated in Table 1. These parameters (carrier frequency and temperature) have different values. Put a label on each set with its SNR and RSSI, which stand for the set.

The Ministry of Science and Technology's Drone Laboratory conducted the experiment. To collect the dataset, use the all-parameter values from this experiment. As shown in Fig. 5 the drone laboratory heat effect works with the following equipment:

- Drone Quadcopter.
- Temperature Sensor (DHT22).
- Arduino-Uno.
- Noise Figure Meter.
- RF Power Meter.
- PLX-DAQ (data logger).
- Cables.
- Heat Chambers.

**Table 1.** The Sample of dataset for the 1DCNN heat effect

Dataset parameters	
Frequency-Carrier	Temperature
2.4	50
2.4	30
5.8	10
5.8	4

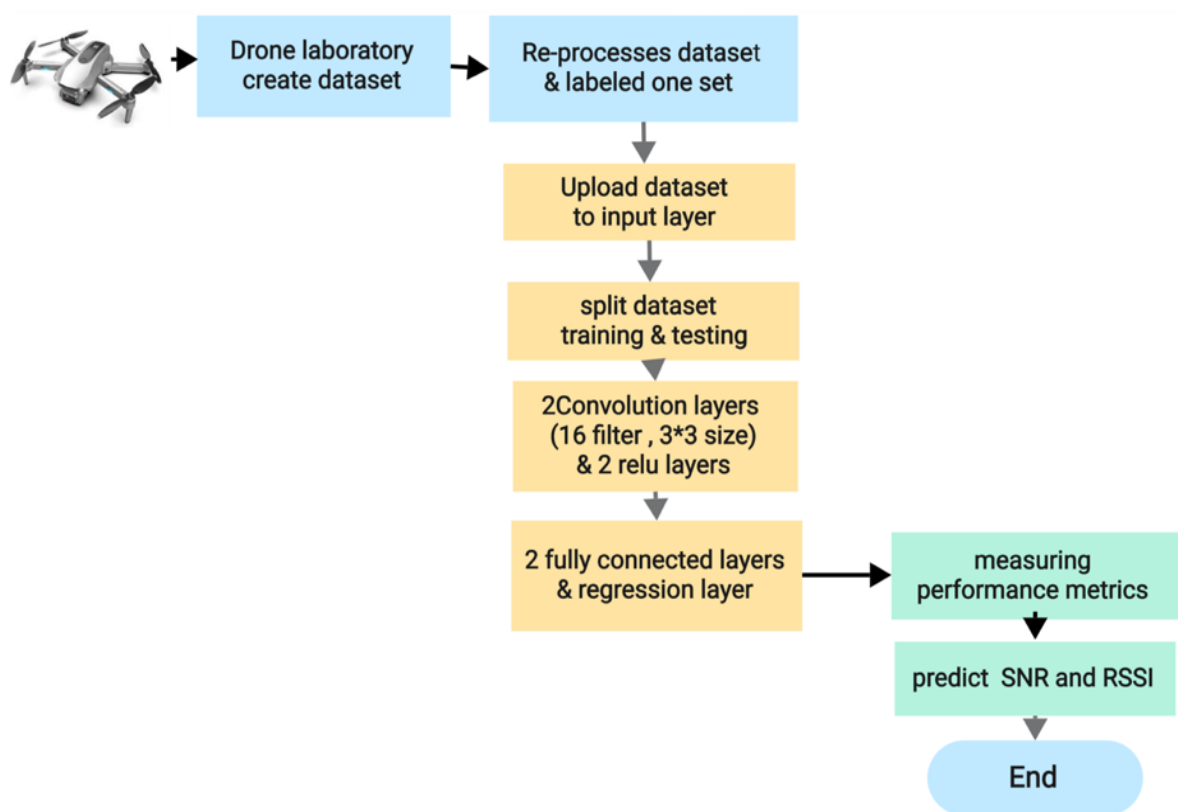
**Fig. 4.** Flow chart of CNN

Fig. 6 depicts the connection between the Arduino and the heat sensor (DHT22). It can accurately measure temperatures ranging from -40 to +125 degrees Celsius. This sensor uses a single digital pin to send data. We connect the VCC and GND pins to the Arduino's 5V pin and ground pin, respectively. Next, connect the Arduino to a data logger (PLX-DAQ) to record the heat value. We then connect the drone antennas to the noise figure meter and RF power meter. with a coaxial cable. Next, adjust the temperature in Heat Chambers to ensure that all values have an effect.



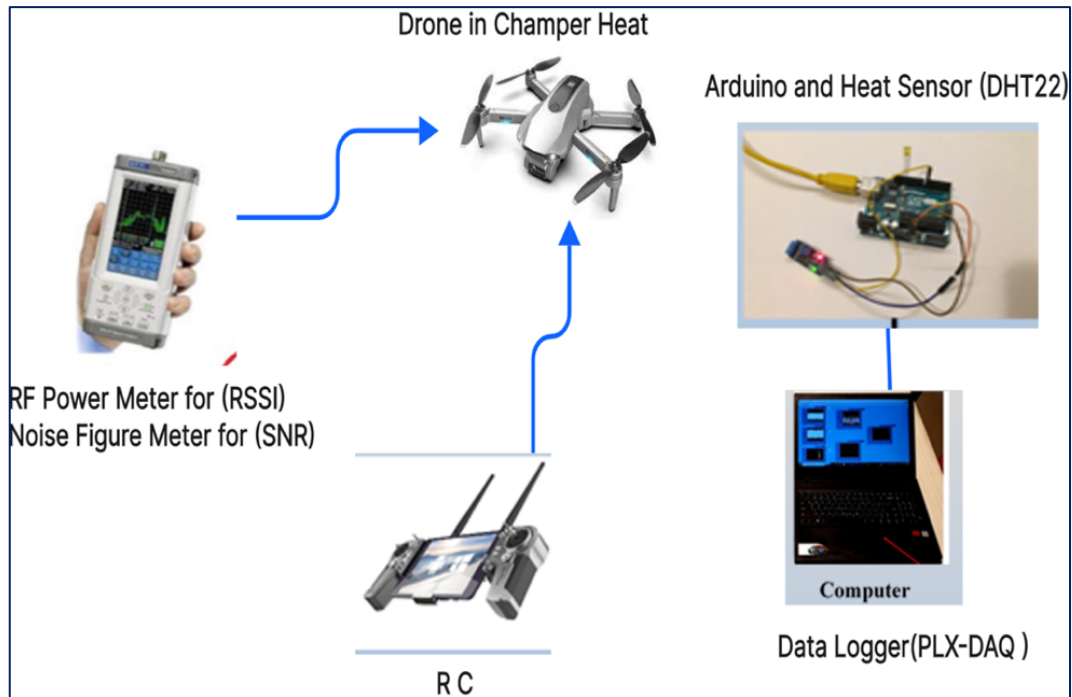


Fig. 5. Diagram for the heat experiment

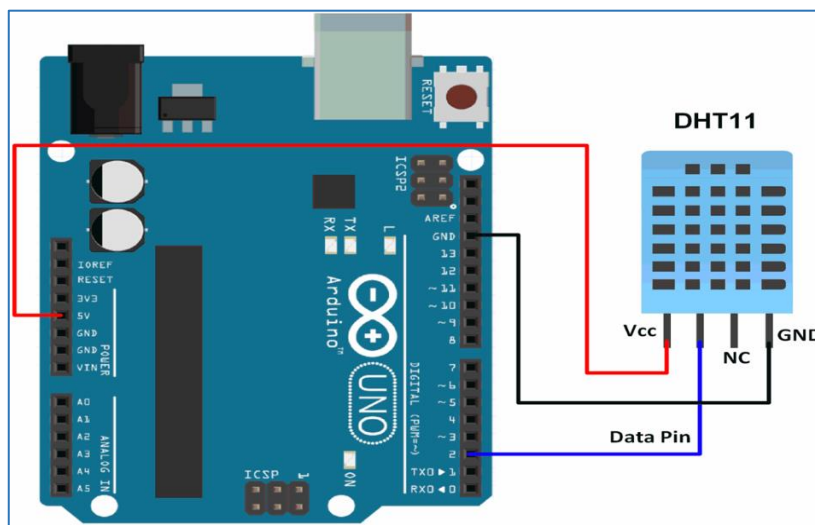


Fig. 6. Diagram for connecting arduino and heat sensor

#### 4.1.2. 1DCNN Architecture

We designed the modal convolution neural networks to be suitable for learning the relationship between parameter input and their SNR and RSSI cross-ponding, following the 1D CNN's structure:

- Input-Layer:** This is where you configure the data's dimensions for entry. There are two separate columns and one row of labels in one set.
- Convolution-Layer:** We pull out important features using a two-layer method and (16, 32) filters with a  $3 \times 3$  size for the kernel. We create feature maps by combining the raw data with weights in the convolution layer's kernel. Real-life Using a  $3 \times 3$  filter size in convolutional layers perfectly balances finding localised features. The output dimensions are adjusted to align with the input dimensions.
- Relu-Layer:** We use two layers and an activation function to take the result of the convolution phase and change it into a random pattern.

- D. Fully Connected-Layer: We use one layer, containing weights, biases, and neurons that link neurons between different layers. In a CNN design, these layers typically come before the final layer. For example, one neuron in this layer gives off a number that displays the anticipated RSSI and SNR.
- E. Regression-Layer: We use one layer to calculate the error between the predicted and actual outputs. This layer is primarily utilised for regression assignments. Including the prediction of continuous values like the RSSI and SNR. The CNN architecture has been planned. To process the dataset, we run the third-step regression.

#### 4.1.3. 1D CNN Regression Process

After loading the dataset through the input layer, see Fig. 4 it illustrates the need to split it into two sets: a 70% training data set and a 30% testing data set. We do this to assess the model's performance and accuracy. We utilise the 1DCNN (convolutional neural network) analysis to estimate the values of the signal-to-noise ratio (SNR) and the RSSI.

1. Mean square error (MSE)

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (7)$$

2. Root mean square error (RMSE)

$$RMSE = \sqrt{MSE} \quad (8)$$

3. Mean Absolute Error (MAE)

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (9)$$

4. Root Mean Absolute Error (RMAE)

$$RMAE = \sqrt{MAE} \quad (10)$$

5. Relative Error (RAE)

$$RAE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (11)$$

Where  $n$  : the number of samples;  $y_i$ : the real value;  $\hat{y}_i$  the predicated value.

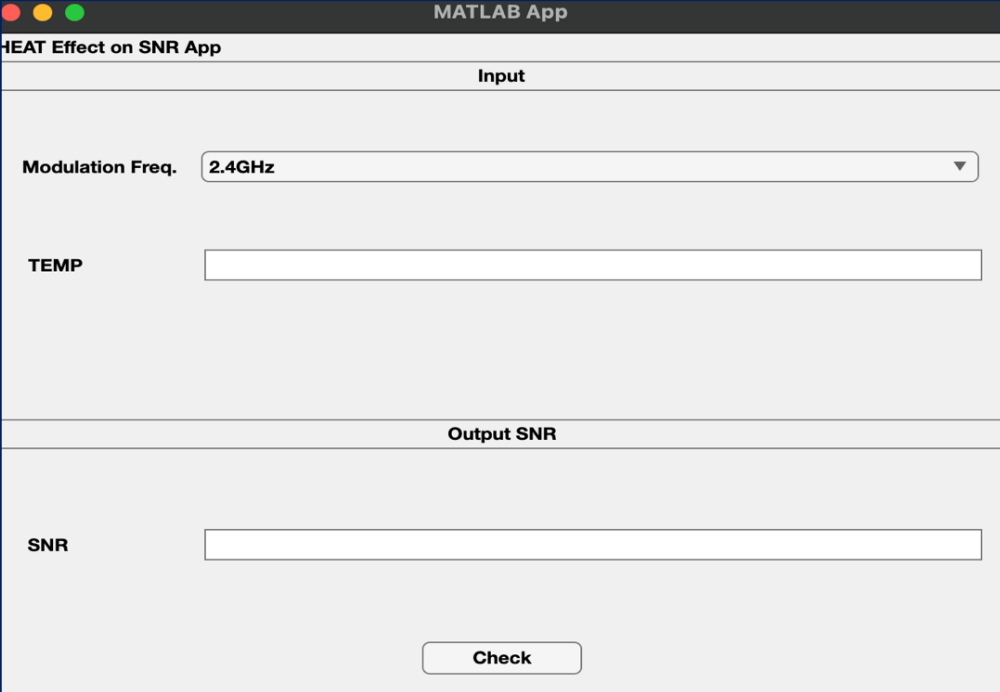
#### 4.2. MATLAB Application Designer

We completed the regression training and identified the factors for system evaluation. We used MATLAB App Designer to transform the code into a user-friendly tool. This is a smart way to make things more accessible and usable. An easy-to-use GUI in App Designer makes it possible to make live MATLAB apps as shown in Fig. 7 to predicate SNR value. and Fig. 8 to predicate RSSI value.

This graphical user interface (GUI) contains the following elements:

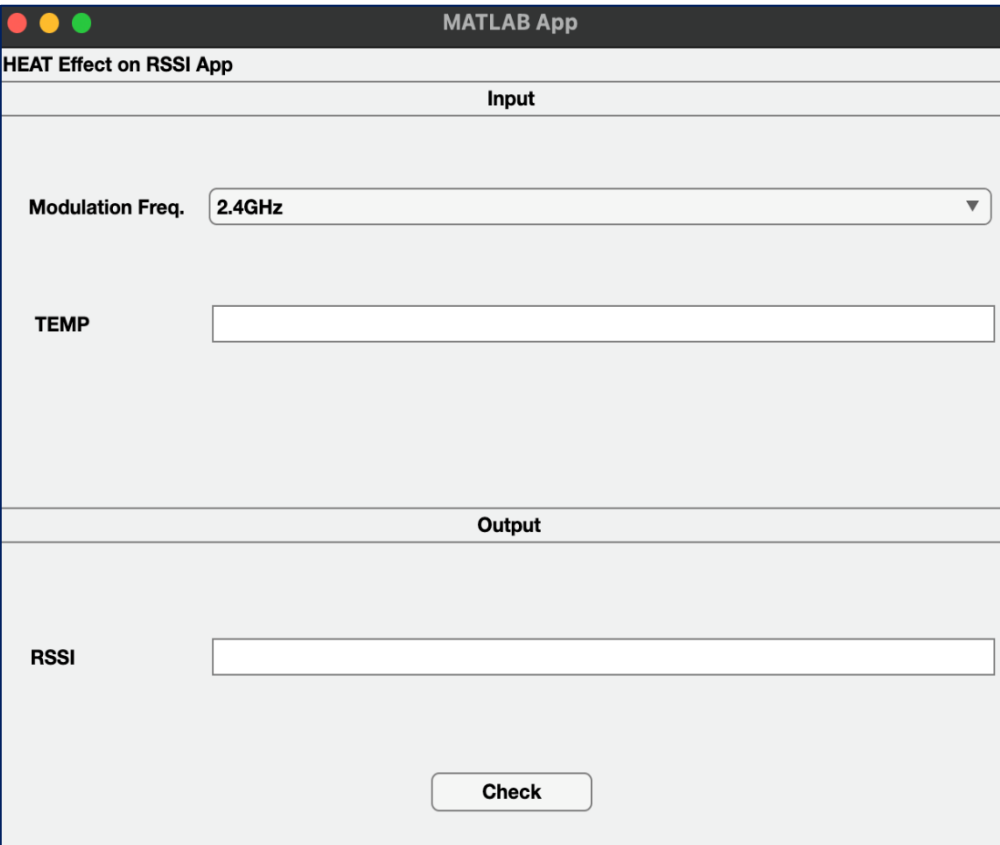
- A. The second field has a drop-down list representing the type of frequency carrier (2.4 GHz, 5.8 GHz).
- B. The second field representing the temperature value.
- C. The third field represents the SNR value.

Pressing the "Check" button displays the SNR value in the SNR field,



The figure shows a MATLAB App window titled "MATLAB App" with a subtitle "HEAT Effect on SNR App". The interface is divided into two main sections: "Input" and "Output SNR". In the "Input" section, there is a "Modulation Freq." label followed by a dropdown menu currently showing "2.4GHz", and a "TEMP" label followed by an empty text input field. The "Output SNR" section contains an "SNR" label followed by an empty text input field. At the bottom center of the "Output SNR" section is a "Check" button.

**Fig. 7.** Graphical user interface to predicate SNR



The figure shows a MATLAB App window titled "MATLAB App" with a subtitle "HEAT Effect on RSSI App". The interface is divided into two main sections: "Input" and "Output". In the "Input" section, there is a "Modulation Freq." label followed by a dropdown menu currently showing "2.4GHz", and a "TEMP" label followed by an empty text input field. The "Output" section contains an "RSSI" label followed by an empty text input field. At the bottom center of the "Output" section is a "Check" button.

**Fig. 8.** Graphical user interface to predicate RSSI

This graphical user interface (GUI) contains the following elements:

- A. The second field has a drop-down list representing the type of frequency carrier (2.4 GHz, 5.8 GHz).

- B. The second field representing the temperature value.
- C. The third field represents the RSSI value.

Pressing the “Check” button displays the RSSI value in the RSSI field,

In short, we collected the information from our test in the drone lab. We collected data for each measure, including frequency carrier and temperature, in different amounts. Next, standardise the incoming data and prepare the labels to align with CNN's training data format. Finally, give each set the right SNR and RSSI. I taught CNN how to connect the input parameters with their SNR and RSSI cross-ponding. Next, we send the information to the input layer. We use seventy percent of the last set of data for training and thirty percent for testing. In the convolution layer, we evaluate the mode performance and feature extraction from the input. The last step is to lower the number of dimensions in the pooling layer, combine the features, and predict the effect in the fully connected layer. The output layer creates the SNR and RSSI predictions. The MATLAB app performs the last step.

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

## 5. The Result for the Heat Effect

The results of the Heat effect on the communications system in drone is divided into two parts, the first a measurement SNR the second a measurement RSSI.

### 5.1. The Result of One-Dimension Convolution Neural Networks For SNR

After completing the architectural model, dataset collection and loading, training, and regression phases, 1DCNN performs the regression training method for SNR [Fig. 9](#), indicates that the training and tests have yielded good accuracy. The confirmed root-mean-square-error (RMSE) is 2.3954, so we can get a full picture of how well the CNN regression model predicts how temperature will affect the system of communications by measuring the SNR. We present these numbers in the method section. This lets us see how well we're doing and make changes as needed. Make the training results better by ensuring that the collection is equal, diverse, and larger.

[Fig. 10](#) shows the visualized predictions. The x-axis represents the true values, and the y-axis represents the predicted values. We notice the regression line crossing above the diagonal line (representing the good prediction).

In [Fig. 11](#), the predicted value that can be seen matches the real value. This means that our CNN model did well on the SNR measurement. We tested the system's overall performance after fine-tuning parameters such as epochs, batch size, and learning rate to achieve optimal performance. We used the parameters shown in [Table 2](#) for the test.

**Table 2.** Evolution parameters of regression

Parameters	Value
Mean square error	5.737
Root mean square error	2.395
Mean Absolute Error	17.313
Root Mean Absolute Error	4.161
Relative Error	0.000087 %

When you use CNN, we send the data through many layers, collecting data at each level and coming up with an estimate of the SNR as the end result. To extract basic traits from the input information, training uses kernel parameters. The convolutional neural network (CNN) was trained well, as shown by the validation (RMSE) of 2.3954. It made accurate predictions of the signal-to-noise ratio (SNR) with little error. We notice that the SNR decrease when the temperature increase as shown in [Fig. 12](#).

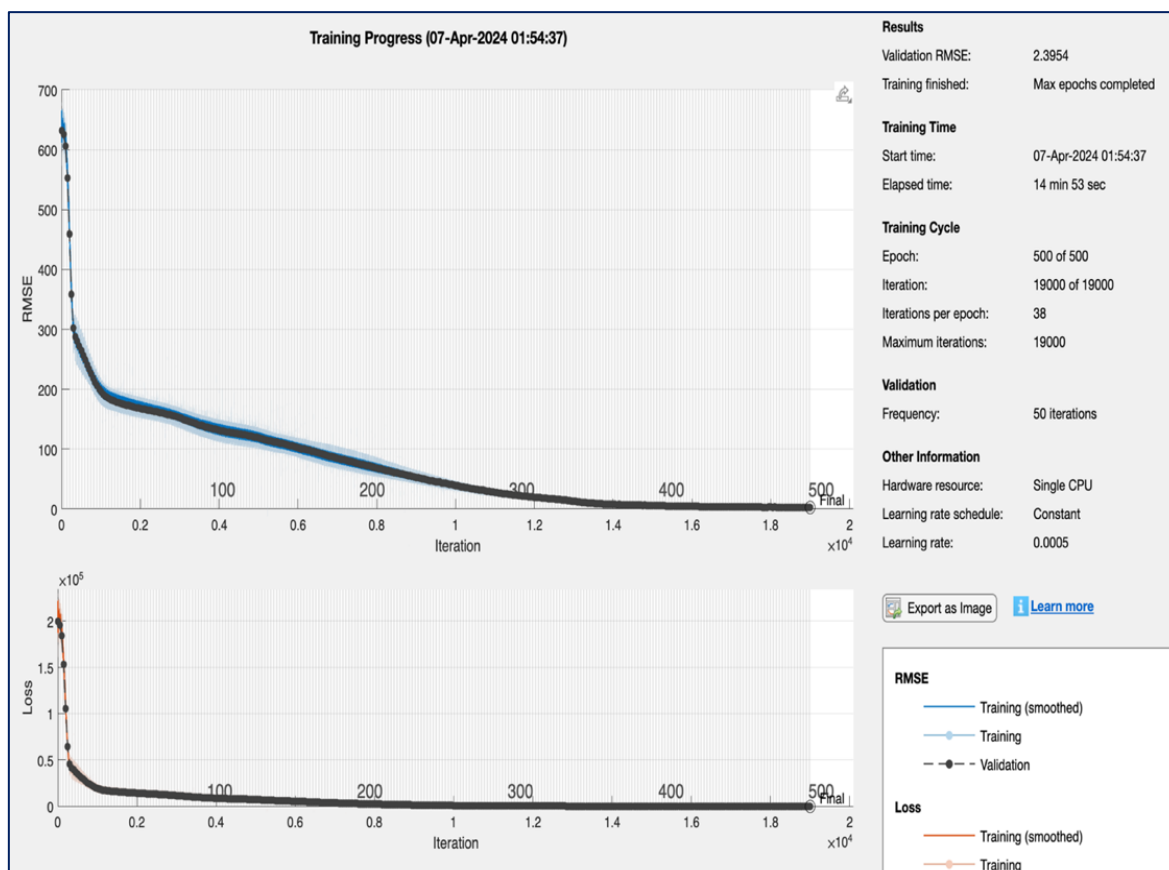


Fig. 9. Regression training process for SNR

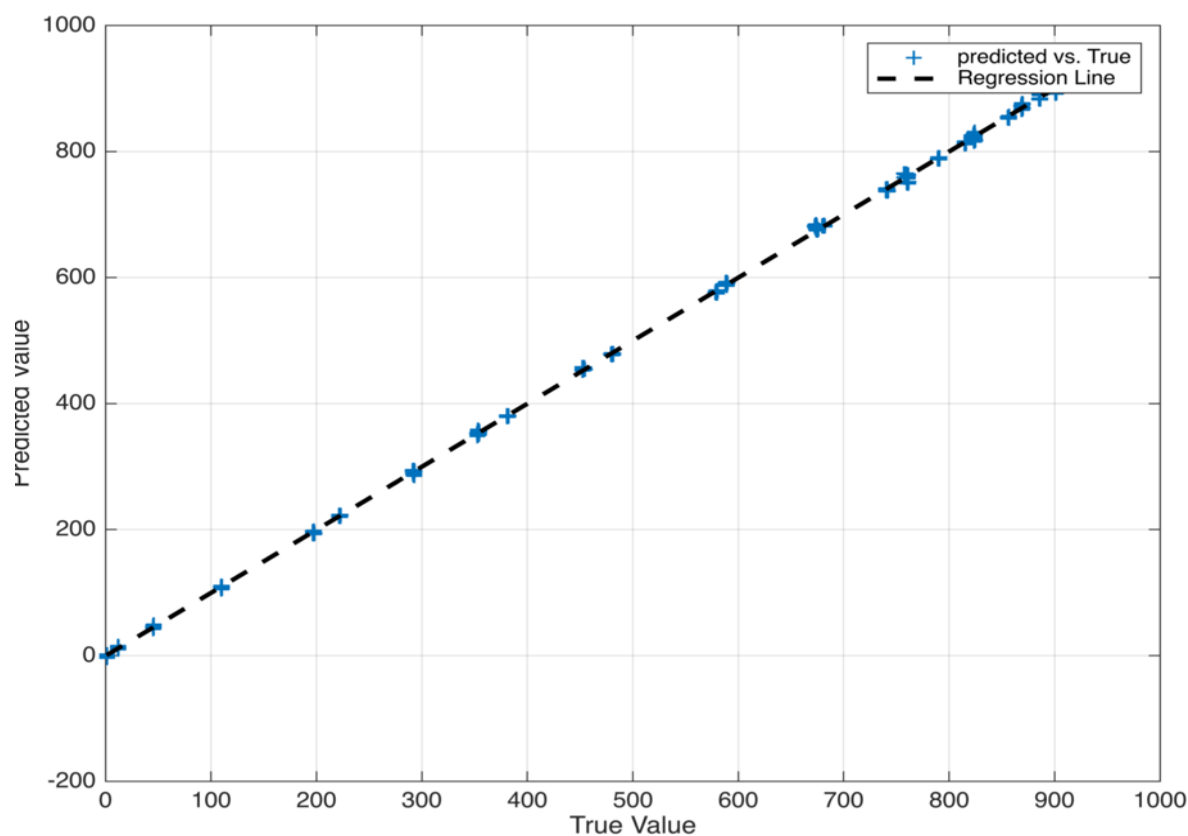


Fig. 10. Visualize prediction values vs. true values for SNR



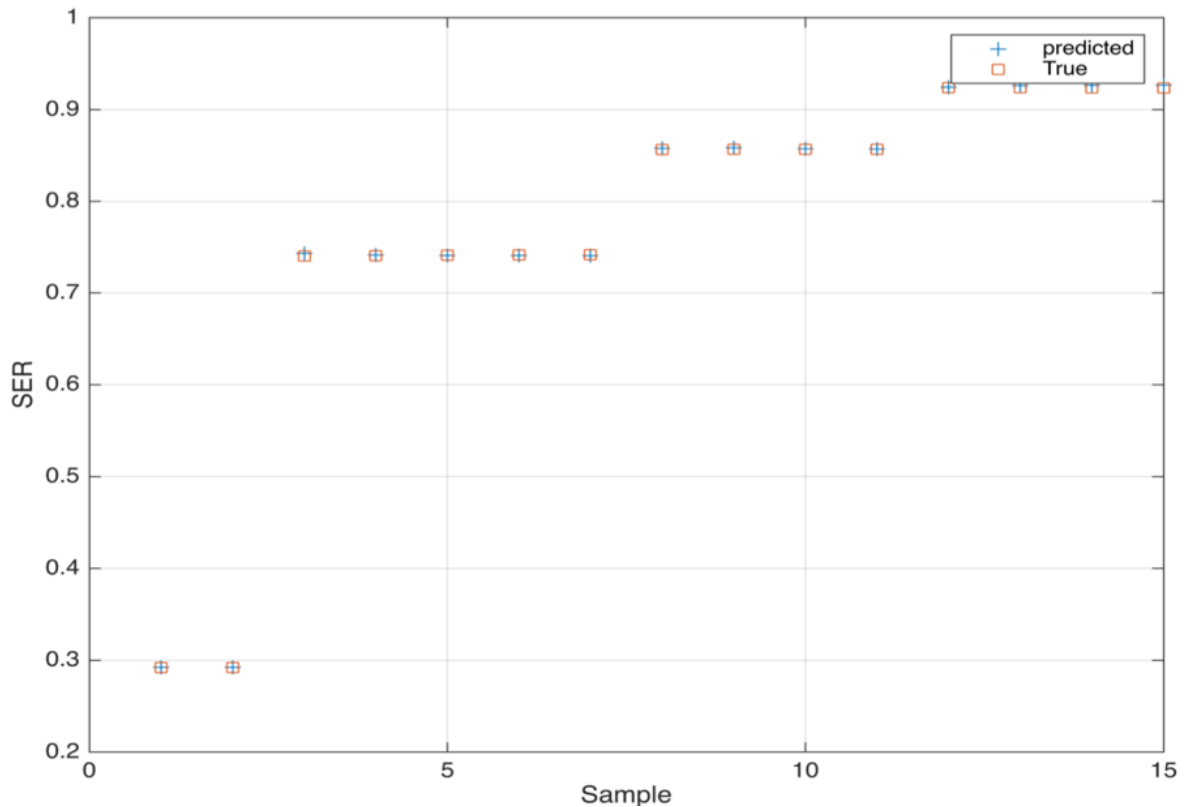


Fig. 11. Pattern prediction values vs. true values for SNR

**MATLAB App**

**HEAT Effect on SNR App**

**Input**

Modulation Freq.

TEMP

**Output SNR**

SNR

Fig. 12. Graphical user interface for SNR result

## 5.2. The Result of One-Dimension Convolution Neural Networks For RSSI

Fig. 4 depicts the steps involved in training and testing the RSSI model. These steps include creating the structure of the model, gathering and uploading the dataset, and then training with regression stages.

Fig. 13 demonstrates that the training and tests have yielded good accuracy. The validated RMSE is 3.591, and measuring the RSSI gives us a full picture of how well the CNN regression model predicts how temperature will affect the communication system. This lets us see how well we're doing and make changes as needed. Enhance the training results by ensuring the collection is equitable, diverse, and expanded. Fig. 14 shows the visualized predictions.

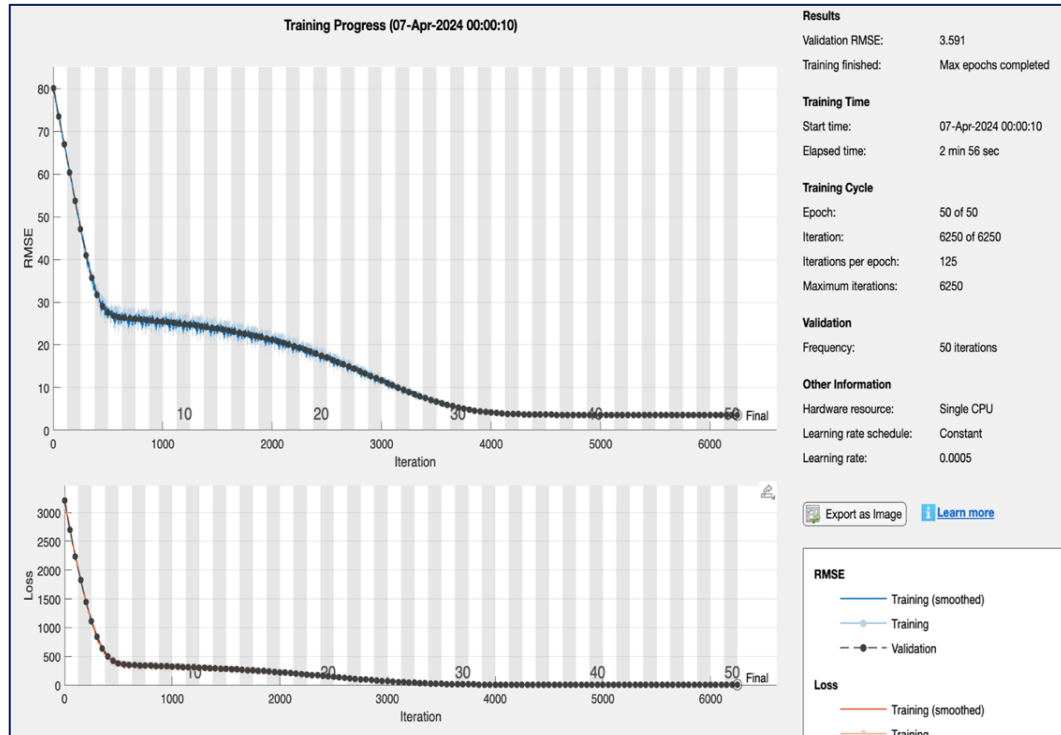


Fig. 13. Regression training process for RSSI

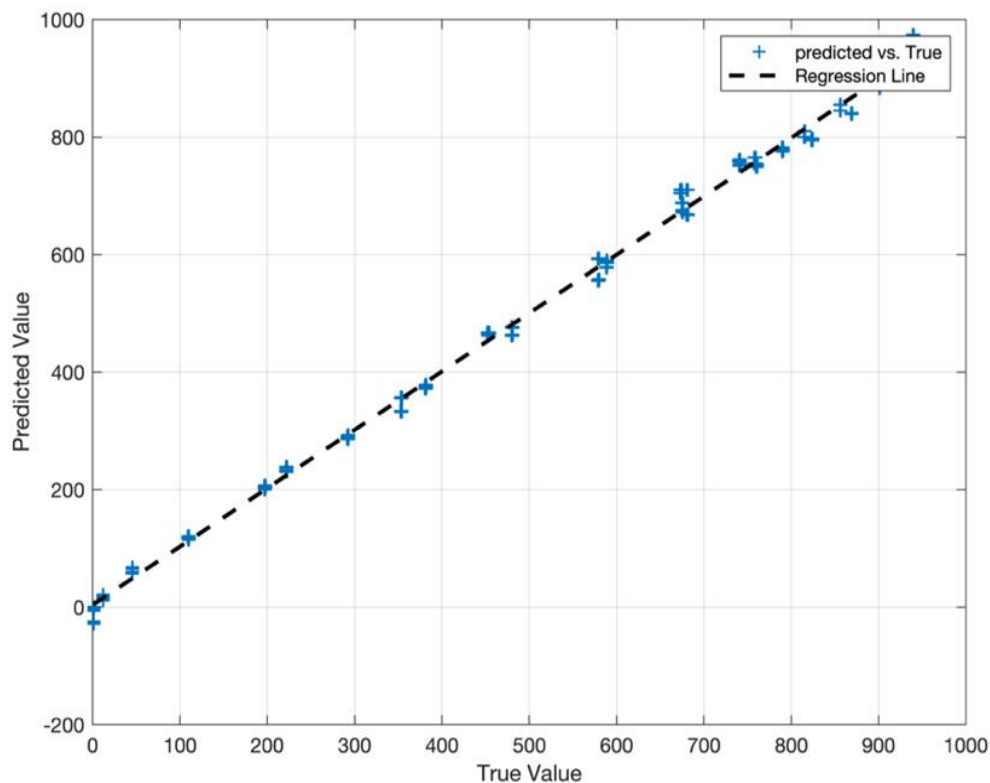


Fig. 14. Visualize prediction values vs. true values for RSSI

The x-axis represents the true values, and the y-axis represents the predicted values. We notice the regression line crossing above the diagonal line (representing the good prediction). In Fig. 15 the observed predicted value aligns with the actual value. This indicates that our CNN model performed well in its evaluation. To get the best performance, we tested the system's total performance after fine-tuning parameters like epochs, batch size, learning rate, and so on. Table 3 displays the parameters used during the test.

The 1DCNN method works by running data through many layers and learning new representations at each stage. The output is an RSSI prediction. We use kernel parameters during training to extract significant features from the original data. We obtain a validated RMSE of 3.5909, indicating that CNN training went well and that the error rate in predicting RSSI was low. We notice that the RSSI decrease when the temperature increase as shown in Fig. 16.

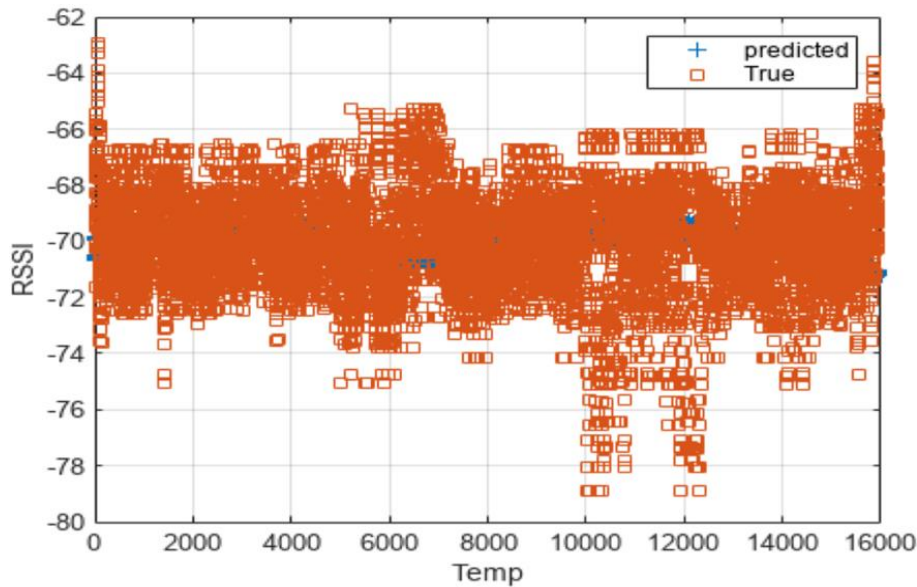


Fig. 15. Pattern prediction values vs. true values for RSSI

A screenshot of a MATLAB App titled 'HEAT Effect on RSSI App'. The interface is divided into 'Input' and 'Output' sections. In the 'Input' section, there is a 'Modulation Freq.' dropdown menu set to '2.4GHz' and a 'TEMP' text input field containing the value '60'. In the 'Output' section, there is an 'RSSI' text output field displaying the value '-82.4991'. At the bottom center of the app, there is a 'Check' button.

Fig. 16. Graphical User Interface for RSSI Result

**Table 3.** The Factors used to evaluate regression for RSSI

Parameters	Value
Mean square error	12.895
Root mean square error	3.5909
Mean Absolute Error	2.9414
Root Mean Absolute Error	1.7150
Relative Error	0.000 735 %

## 6. Discussion

As mentioned in the introduction section, the majority of earlier research focused on using simulations or experiments to demonstrate how temperature affects drone communication systems. In order to assess how temperature affects a drone's communication system, this study uses a one-dimensional neural network (1D CNN) to measure the signal-to-noise ratio (SNR) and received signal strength indicator (RSSI). Next, we use a drone laboratory to generate the dataset, which we then process and upload to our Convolutional Neural Network (CNN) model. To maximise the CNN model's performance, we adjust its parameters (epoch, number of iterations, batch size, etc.). The outcome implies that the temperature increase is being caused by the usage of high-frequency carriers and amplitude frequencies. There is limited capacity of 1DCNN to study the impact of temperature on communication networks. One dataset that 1DCNN trains on has performance communication system and temperature data. Acquiring this data in excellent quality and a wide range of variances is a difficult task. Unmanned aerial vehicles (UAVs) have special challenges in real-world operations, such unpredictable weather patterns. Restrict the model's ability to adapt to this problem by carrying out extensive training.

## 7. Conclusion

The aim of this work is to investigate the relationship between temperature and drone communication systems since it can impair the operation of wireless communication systems. We assess the communication system efficacy of the drone using measurements of the received signal strength indicator (RSSI) and signal-to-noise ratio (SNR). We forecast the RSSI and SNR by a convolutional neural network (CNN). The MATLAB App Designer produces an understandable graphical user interface after training and testing our model on the dataset of the drone laboratory.

Temperature noise often increases with higher bandwidth frequencies. In other words, as the frequency travels farther from the carrier, the power spectral density of thermal noise rises. With thermal noise, distortion increases. Use a variety of strategies to enhance flight control algorithms, like PID (proportional integral derivative), mechanical isolation between the drone's frame and communication components, digital signal processing technology to reduce noise in received signals, and other strategies to lessen the impact of temperature on the drone's communication system. It used materials like polyimide aerogel and glass fibres, which are lightweight and extremely insulating against heat, to build a protective structural enclosure. They coated it with a layer of highly reflective aluminum to keep heat from escaping. When exposed to high temperatures, the excellent insulation prevents materials from contracting and pore structures from deteriorating.

Studies on the effects of temperature on communication systems with 1DCNN are not without limits. We use data from performance communication systems and temperatures to train 1DCNN. It is hard to find this kind of diverse and high-quality data. Changing conditions present unique challenges for UAVs operating in the real world. Hard training will reduce the model's capacity to adjust to this difficulty. Unlike other studies that gauge the UAV communication system's efficacy only through simulation, our contribution to this project is the application of 1DCNN. To improve performance, we propose optimizing system characteristics in this study, such as power transfer, by implementing a feedback loop between the CNN conclusion and the communication system. Furthermore, we investigate how different environmental factors, including wind and rain, affect UAV communication systems.

**Author Contribution:** Each author made an equal contribution to the substantial contribution of this paper. Each author evaluated and gave their approval to the work.

**Funding:** The study was entirely self-funded.

**Conflicts of Interest:** Conflicts of interest are not disclosed by any writers.

## References

- [1] K. Telli *et al.*, “A comprehensive review of recent research trends on unmanned aerial vehicles (uavs),” *Systems*, vol. 11, no. 8, p. 400, 2023, <https://doi.org/10.3390/systems11080400>.
- [2] P. Chu, Y. T. Huang, C. Pi, and S. Cheng, “Autonomous Landing System of a VTOL UAV on an Upward Docking Station Using Visual Servoing,” *IFAC-PapersOnLine*, vol. 55, no. 27, pp. 108-113, 2022, <https://doi.org/10.1016/j.ifacol.2022.10.496>.
- [3] N. Sethi and S. Ahlawat, “Low-fidelity design optimization and development of a VTOL swarm UAV with an open-source framework,” *Array*, vol. 14, p. 100183, 2022, <https://doi.org/10.1016/j.array.2022.100183>.
- [4] T. Patel, M. Kumar, and S. Abdallah, “Control of Hybrid Transitioning Morphing-wing VTOL UAV,” *IFAC-PapersOnLine*, vol. 55, no. 37, pp. 554-559, 2022, <https://doi.org/10.1016/j.ifacol.2022.11.241>.
- [5] M. Bahari, M. Rostami, A. Entezari, S. Ghahremani, and M. Etminan, “A comparative analysis and optimization of two supersonic hybrid SOFC and turbine-less jet engine propulsion system for UAV,” *Fuel*, vol. 319, p. 123796, 2022, <https://doi.org/10.1016/j.fuel.2022.123796>.
- [6] M. Bahari, M. Rostami, A. Entezari, S. Ghahremani, and M. Etminan, “Performance evaluation and multi-objective optimization of a novel UAV propulsion system based on PEM fuel cell,” *Fuel*, vol. 311, p. 122554, 2022, <https://doi.org/10.1016/j.fuel.2021.122554>.
- [7] K. Zhou *et al.*, “A kW-level integrated propulsion system for UAV powered by PEMFC with inclined cathode flow structure design,” *Applied Energy*, vol. 328, p. 120222, 2022, <https://doi.org/10.1016/j.apenergy.2022.120222>.
- [8] M. Stamate, C. Pupăză, F. Nicolescu, and C. Moldoveanu, “Improvement of Hexacopter UAVs Attitude Parameters Employing Control and Decision Support Systems,” *Sensors*, vol. 23, no. 3, p. 1446, 2023, <https://doi.org/10.3390/s23031446>.
- [9] N. Li, X. Liu, B. Yu, L. Li, J. Xu, and Q. Tan, “Study on the environmental adaptability of lithium-ion battery powered UAV under extreme temperature conditions,” *Energy*, vol. 219, p. 119481, 2021, <https://doi.org/10.1016/j.energy.2020.119481>.
- [10] J. Jin, S. Kim, and J. Moon, “Development of a Firefighting Drone for Constructing Fire-breaks to Suppress Nascent Low-Intensity Fires,” *Applied Sciences*, vol. 14, no. 4, p. 1652, 2024, <https://doi.org/10.3390/app14041652>.
- [11] D. Häusermann *et al.*, “FireDrone: Multi-Environment Thermally Agnostic Aerial Robot,” *Advanced Intelligent Systems*, vol. 5, no. 9, p. 2300101, 2023, <https://doi.org/10.1002/aisy.202370038>.
- [12] K. Wang, Y. Yuan, M. Chen, Z. Lou, Z. Zhu, and R. Li, “A study of fire drone extinguishing system in high-rise buildings,” *Fire*, vol. 5, no. 3, p. 75, 2022, <https://doi.org/10.3390/fire5030075>.
- [13] S. Zheng, W. Wang, and Z. Liu, “Design and research of forest farm fire drone monitoring system based on deep learning,” *6GN for Future Wireless Networks*, pp. 215-229, 2021, [https://doi.org/10.1007/978-3-031-04245-4\\_19](https://doi.org/10.1007/978-3-031-04245-4_19).
- [14] R. A. Jaber, M. S. Sikder, R. A. Hossain, K. F. N. Malia and M. A. Rahman, “Unmanned Aerial Vehicle for Cleaning and Firefighting Purposes,” *2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, pp. 673-677, 2021, <https://doi.org/10.1109/ICREST51555.2021.9331147>.



- 
- [15] S. Zheng, W. Wang, Z. Liu, and Z. Wu, "Forest farm fire drone monitoring system based on deep learning and unmanned aerial vehicle imagery," *Mathematical Problems in Engineering*, vol. 2021, no. 1, pp. 1-13, 2021, <https://doi.org/10.1155/2021/3224164>.
- [16] M. O. Alade, "Investigation of the effect of ground and air temperature on very high frequency radio signals," *Journal of Information Engineering and Applications*, vol. 3, no. 9, pp. 16-21, 2013, <https://www.iiste.org/Journals/index.php/JIEA/article/view/7207>.
- [17] C. A. Boano, J. Brown, Z. He, U. Roedig, and T. Voigt, "Low-power radio communication in industrial outdoor deployments: The impact of weather conditions and ATEX-compliance," *Sensor Applications, Experimentation, and Logistics*, vol. 29, pp. 159-176, 2010, [https://doi.org/10.1007/978-3-642-11870-8\\_11](https://doi.org/10.1007/978-3-642-11870-8_11).
- [18] R. Umar *et al.*, "Preliminary study of radio astronomical lines effect of rain below 2.9 GHz," *Jurnal Teknologi*, vol. 75, no. 1, pp. 7-11, 2015, <https://doi.org/10.11113/jt.v75.3984>.
- [19] L. Zhenzhong, M. Nezih, X. George, O. Yuu, L. Guocheng, and B. Dayan, "Effects of temperature and humidity on UHF RFID performance," *International Workshop Smart Materials, Structures & NDT in Aerospace*, 2011, [https://www.ndt.net/article/ndtcanada2011/papers/102\\_Li.pdf](https://www.ndt.net/article/ndtcanada2011/papers/102_Li.pdf).
- [20] A. A. Segun, A. M. Olusope, and A. H. Kofoworola, "Influence of air temperature, relative humidity and atmospheric moisture on UHF radio propagation in South Western Nigeria," *International Journal of Science and Research*, vol. 4, no. 8, pp. 588-592, 2015, <https://www.ijsr.net/archive/v4i8/SUB157273.pdf>.
- [21] Y. S. Meng, Y. H. Lee and B. C. Ng, "The Effects of Tropical Weather on Radio-Wave Propagation Over Foliage Channel," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4023-4030, 2009, <https://doi.org/10.1109/TVT.2009.2021480>.
- [22] J. Luomala and I. Hakala, "Effects of temperature and humidity on radio signal strength in outdoor wireless sensor networks," *2015 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 1247-1255, 2015, <https://doi.org/10.15439/2015F241>.
- [23] C. A. Boano, M. Zúñiga, J. Brown, U. Roedig, C. Keppitiyagama and K. Römer, "TempLab: A testbed infrastructure to study the impact of temperature on wireless sensor networks," *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, pp. 95-106, 2014, <https://doi.org/10.1109/IPSN.2014.6846744>.
- [24] M. Numan *et al.*, "A Systematic Review on Clone Node Detection in Static Wireless Sensor Networks," *IEEE Access*, vol. 8, pp. 65450-65461, 2020, <https://doi.org/10.1109/ACCESS.2020.2983091>.
- [25] H. Wennerström, F. Hermans, O. Rensfelt, C. Rohner and L. -Å. Nordén, "A long-term study of correlations between meteorological conditions and 802.15.4 link performance," *2013 IEEE International Conference on Sensing, Communications and Networking (SECON)*, pp. 221-229, 2013, <https://doi.org/10.1109/SAHCN.2013.6644981>.
- [26] C. Huang *et al.*, "Artificial Intelligence Enabled Radio Propagation for Communications—Part II: Scenario Identification and Channel Modeling," *IEEE Transactions on Antennas and Propagation*, vol. 70, no. 6, pp. 3955-3969, 2022, <https://doi.org/10.1109/TAP.2022.3149665>.
- [27] A. Mishra, S. Bagui and J. Compo, "Design and Testing of a Real-Time Audio-Quality Feedback System for Weather Broadcasts and a Framework for a Weather Broadcast Transmission Technology Switch," *IEEE Access*, vol. 7, pp. 158782-158797, 2019, <https://doi.org/10.1109/ACCESS.2019.2950306>.
- [28] M. Tamura *et al.*, "A 0.5-V BLE Transceiver With a 1.9-mW RX Achieving -96.4-dBm Sensitivity and -27-dBm Tolerance for Intermodulation From Interferers at 6- and 12-MHz Offsets," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 12, pp. 3376-3386, 2020, <https://doi.org/10.1109/JSSC.2020.3025225>.
- [29] A. J. Onumanyi, A. M. Abu-Mahfouz and G. P. Hancke, "Cognitive Radio in Low Power Wide Area Network for IoT Applications: Recent Approaches, Benefits and Challenges," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7489-7498, 2020, <https://doi.org/10.1109/TII.2019.2956507>.
- [30] M. Massaoudi, I. Chihi, L. Sidhom, M. Trabelsi and F. S. Oueslati, "Medium and Long-Term Parametric Temperature Forecasting using Real Meteorological Data," *IECON 2019 - 45th Annual Conference of the*
-

- 
- IEEE Industrial Electronics Society*, pp. 2402-2407, 2019, <https://doi.org/10.1109/IECON.2019.8927778>.
- [31] M. Mafuta, M. Zennaro, A. Bagula, G. Ault, H. Gombachika, and T. Chadza, "Successful deployment of a wireless sensor network for precision agriculture in Malawi," *International Journal of Distributed Sensor Networks*, vol. 9, no. 5, p. 150703, 2013, <https://doi.org/10.1155/2013/150703>.
- [32] R. Marfievici, A. L. Murphy, G. P. Picco, F. Ossi and F. Cagnacci, "How Environmental Factors Impact Outdoor Wireless Sensor Networks: A Case Study," *2013 IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems*, pp. 565-573, 2013, <https://doi.org/10.1109/MASS.2013.13>.
- [33] F. Yuan, Y. H. Lee, Y. S. Meng, S. Manandhar and J. T. Ong, "High-Resolution ITU-R Cloud Attenuation Model for Satellite Communications in Tropical Region," *IEEE Transactions on Antennas and Propagation*, vol. 67, no. 9, pp. 6115-6122, 2019, <https://doi.org/10.1109/TAP.2019.2916746>.
- [34] I. S. Lee, J. H. Noh, S. J. Oh, "A Survey and analysis on a troposcatter propagation model based on ITU-R recommendations," *ICT Express*, vol. 9, no. 3, pp. 507-516, 2023, <https://doi.org/10.1016/j.icte.2022.09.009>.
- [35] F. A. Semire, R. Mohd-Mokhtar, I. A. Akanbi, "Validation of new ITU-R rain attenuation prediction model over Malaysia equatorial region," *MAPAN*, vol. 34, pp. 71-77, 2019, <https://doi.org/10.1007/s12647-018-0295-z>.
- [36] C. A. Boano, N. Tsiftes, T. Voigt, J. Brown and U. Roedig, "The Impact of Temperature on Outdoor Industrial Sensor Applications," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 3, pp. 451-459, 2010, <https://doi.org/10.1109/TII.2009.2035111>.
- [37] C. Hsieh, J. Chen and B. Nien, "Deep Learning-Based Indoor Localization Using Received Signal Strength and Channel State Information," *IEEE Access*, vol. 7, pp. 33256-33267, 2019, <https://doi.org/10.1109/ACCESS.2019.2903487>.
- [38] K. Ohshima, H. Hara, Y. Hagiwara and M. Terada, "Field investigation of the radio transmission performance and distance in a environmental wireless sensor network," *The International Conference on Information Network 2012*, pp. 132-137, 2012, <https://doi.org/10.1109/ICOIN.2012.6164364>.
- [39] M. Shahid *et al.*, "Link-Quality-Based Energy-Efficient Routing Protocol for WSN in IoT," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 4645-4653, 2024, <https://doi.org/10.1109/TCE.2024.3356195>.
- [40] D. Gotterbarn, K. Miller and S. Rogerson, "Computer society and ACM approve software engineering code of ethics," *Computer*, vol. 32, no. 10, pp. 84-88, 1999, <https://doi.org/10.1109/MC.1999.796142>.
- [41] V. A. Mardiana, T. Adiono, S. Harimurti, M. M. M. Dinata, A. Mitayani and G. N. Nurkahfi, "APSoC Architecture Design of 2.4 GHz ZigBee Baseband Transceiver for IoT Application," *2019 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)*, pp. 74-78, 2019, <https://doi.org/10.1109/ICRAMET47453.2019.8980407>.
- [42] D. Zhou, "Theory of deep convolutional neural networks: Downsampling," *Neural Networks*, vol. 124, pp. 319-327, 2020, <https://doi.org/10.1016/j.neunet.2020.01.018>.
- [43] S. Jhong *et al.*, "An Automated Biometric Identification System Using CNN-Based Palm Vein Recognition," *2020 International Conference on Advanced Robotics and Intelligent Systems (ARIS)*, pp. 1-6, 2020, <https://doi.org/10.1109/ARIS50834.2020.9205778>.
- [44] A. Al-Azzawi, A. Ouadou, H. Max, Y. Duan, J. J. Tanner, and J. Cheng, "DeepCryoPicker: fully automated deep neural network for single protein particle picking in cryo-EM," *BMC Bioinformatics*, vol. 21, no. 59, pp. 1-38, 2020, <https://doi.org/10.1186/s12859-020-03809-7>.
- [45] X. Xu and H. Liu, "ECG Heartbeat Classification Using Convolutional Neural Networks," *IEEE Access*, vol. 8, pp. 8614-8619, 2020, <https://doi.org/10.1109/ACCESS.2020.2964749>.
- [46] G. Li, M. Zhang, J. Li, F. Lv, and G. Tong, "Efficient densely connected convolutional neural networks," *Pattern Recognition*, vol. 109, p. 107610, 2021, <https://doi.org/10.1016/j.patcog.2020.107610>.
-

- 
- [47] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354-377, 2018, <https://doi.org/10.1016/j.patcog.2017.10.013>.
- [48] W. Fang, P. E. D. Love, H. Luo, and L. Ding, "Computer vision for behaviour-based safety in construction: A review and future directions," *Advanced Engineering Informatics*, vol. 43, p. 100980, 2020, <https://doi.org/10.1016/j.aei.2019.100980>.
- [49] D. Palaz, M. Magimai-Doss, and R. Collobert, "End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition," *Speech Communication*, vol. 108, pp. 15-32, 2019, <https://doi.org/10.1016/j.specom.2019.01.004>.
- [50] H. Li, Z. Deng, and H. Chiang, "Lightweight and resource-constrained learning network for face recognition with performance optimization," *Sensors*, vol. 20, no. 21, p. 6114, 2020, <https://doi.org/10.3390/s20216114>.
- [51] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *The Journal of Physiology*, vol. 160, no. 1, p. 106-154, 1962, <https://doi.org/10.1113/jphysiol.1962.sp006837>.
- [52] L. Jiao and J. Zhao, "A Survey on the New Generation of Deep Learning in Image Processing," *IEEE Access*, vol. 7, pp. 172231-172263, 2019, <https://doi.org/10.1109/ACCESS.2019.2956508>.
- [53] S. Kim, Z. W. Geem, and G. Han, "Hyperparameter optimization method based on harmony search algorithm to improve performance of 1D CNN human respiration pattern recognition system," *Sensors*, vol. 20, no. 13, p. 3697, 2020, <https://doi.org/10.3390/s20133697>.
- [54] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2017, <https://doi.org/10.1145/3065386>.
- [55] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv*, 2014, <https://doi.org/10.48550/arXiv.1409.1556>.
- [56] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, pp. 1929-1958, 2014, <https://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf>.
- [57] D. Shi, Y. Ye, M. Gillwald, and M. Hecht, "Designing a lightweight 1D convolutional neural network with Bayesian optimization for wheel flat detection using carbody accelerations," *International Journal of Rail Transportation*, vol. 9, no. 4, pp. 311-341, 2021, <https://doi.org/10.1080/23248378.2020.1795942>.
- [58] O. Avci, O. Abdeljaber, S. Kiranyaz, and D. Inman, "Structural damage detection in real time: implementation of 1D convolutional neural networks for SHM applications," *Structural Health Monitoring & Damage Detection, Volume 7*, pp. 49-54, 2017, [https://doi.org/10.1007/978-3-319-54109-9\\_6](https://doi.org/10.1007/978-3-319-54109-9_6).
- [59] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," *Journal of Sound and Vibration*, vol. 388, pp. 154-170, 2017, <https://doi.org/10.1016/j.jsv.2016.10.043>.
- [60] O. Avci, O. Abdeljaber, S. Kiranyaz, B. Boashash, H. Sodano, and D. J. Inman, "Efficiency validation of one dimensional convolutional neural networks for structural damage detection using a SHM benchmark data," *Qspace Institutional Repository*, pp. 4600-4607, 2018, <http://hdl.handle.net/10576/30622>.
- [61] O. Abdeljaber, O. Avci, M. S. Kiranyaz, B. Boashash, H. Sodano, and D. J. Inman, "1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data," *Neurocomputing*, vol. 275, pp. 1308-1317, 2018, <https://doi.org/10.1016/j.neucom.2017.09.069>.
- [62] T. Ince, S. Kiranyaz, L. Eren, M. Askar and M. Gabbouj, "Real-Time Motor Fault Detection by 1-D Convolutional Neural Networks," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7067-7075, 2016, <https://doi.org/10.1109/TIE.2016.2582729>.
- [63] S. Kiranyaz, A. Gastli, L. Ben-Brahim, N. Al-Emadi and M. Gabbouj, "Real-Time Fault Detection and Identification for MMC Using 1-D Convolutional Neural Networks," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 11, pp. 8760-8771, 2019, <https://doi.org/10.1109/TIE.2018.2833045>.
-

- [64] O. Abdeljaber, S. Sassi, O. Avci, S. Kiranyaz, A. A. Ibrahim and M. Gabbouj, "Fault Detection and Severity Identification of Ball Bearings by Online Condition Monitoring," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 10, pp. 8136-8147, 2019, <https://doi.org/10.1109/TIE.2018.2886789>.
- [65] L. A. Al-Haddad, A. A. Jaber, P. Neranon, and S. A. Al-Haddad, "Investigation of Frequency-Domain-Based Vibration Signal Analysis for UAV Unbalance Fault Classification," *Engineering and Technology Journal*, vol. 41, no. 7, pp. 915-923, 2023, <https://doi.org/10.30684/etj.2023.137412.1348>.
- [66] Z. Zhang, P. Cui and W. Zhu, "Deep Learning on Graphs: A Survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 249-270, 2022, <https://doi.org/10.1109/TKDE.2020.2981333>.