Computing and Informatics, Vol. 45, 2026, 748–776, doi: 10.31577/cai_2026_3_748

CNN-LSTM-TWISTED POWER OPTIMIZATION SCHEME FOR LORAWAN-BASED INTELLIGENT STREETLIGHT SYSTEM

Zhenpeng Lai*

Maynooth International Engineering College, Fuzhou University
Fuzhou, Fujian, 350100, China
e-mail: zhenpeng.lai@fzu.edu.cn

Kunpeng Zeng

Concord University College, Fujian Normal University
Fuzhou, 350100, China
e-mail: 124382022009@student.fjnu.edu.cn

Abstract. With the development of smart cities, energy saving has become an important goal, and intelligent streetlight systems (ISLS) play an important role in reducing power consumption. For energy saving purposes, this paper proposes an ISLS based on LoRaWAN technology, incorporating a twisted Artificial Intelligence (AI) Model that integrates Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks intertwining their strengths to achieve automatic brightness control and power consumption optimization. The CNN in the model is responsible for accurately analysing current brightness and traffic flow levels, while the LSTM is used to predict future visibility changes. With the combination, the system can be flexible in dynamic environments and cope with changes in severe weather by adjusting the streetlights' brightness in advance. This research utilizes the traffic road image dataset from the Kaggle platform and the weather dataset from the Szeged area to simulate the real environment. During the 100-day simulation, the average daily power consumption of ISLS was reduced by 13.98 % compared to the typical energy-saving method, which shows relatively high adaptability and energy saving effect in the complex dynamic environment. Hence, this paper provides an efficient energy-saving solu-

* Corresponding author

tion for intelligent streetlights and an innovative idea for energy management in smart cities.

Keywords: LoRaWAN intelligent streetlight systems, CNN and LSTM twisted AI models, brightness power optimization

1 INTRODUCTION

To achieve sustainable urban development and a more comfortable living environment, smart cities are committed to improving the quality of life of citizens by reducing resource consumption, which has attracted increasing attention from governments and researchers in recent years [1]. Nowadays, ISLS has become one of the systems that must be built in smart cities because efficient intelligent streetlights can often help people reach the low carbon destination [2]. In the current market, the Internet of Things (IoT) technology has gained a lot of attention in terms of building ISLS [3].

Traditional IoT technologies such as BLE, ZigBee, and Wi-Fi, shown in Table 1, have many problems and challenges in Intelligent Streetlight Systems (ISLS). The data transmission distance of these technologies is very limited, and more relay devices are required to cover a larger area, which greatly increases the deployment cost and is unsuitable for ISLS with large-area coverage requirements, and the power consumption of these traditional technologies is relatively high, which fails to meet the requirements of decarbonization and prolonged operation.

Technology	**Transmission Range**	**Sleep Power Consumption (μA)**	**Transmission Speed**
LoRa	Up to 15 kilometers	0.01–10	0.3–50 kbps
WiFi	About 50 meters	1 000–5 000	150 Mbps–1 Gbps
Zigbee	10–100 meters	0.005–1	20–250 kbps
BLE	50–150 meters	0.01–15	1–2 Mbps

Table 1. Comparison among communication technologies

In contrast, LoRa technology has obvious advantages in ISLS. LoRa utilizes chirped spread spectrum technology to achieve long-distance communication, with a propagation range that far exceeds traditional IoT technologies, reaching up to 15 km and LoRa networks are designed with low power consumption, allowing devices to operate for extended periods without frequent maintenance [4]. In addition, when the device is in standby mode, the current consumption of LoRa is only a few microamps, which is ideal for battery-powered devices that require long-term operation, such as ISLS and thus greatly extends the maintenance cycle of the device and reduces the maintenance cost. Therefore, this paper identifies LoRa technology as the main point of interest for building ISLS.

For the ISLS, it should have the following key features: automation and energy efficiency [2, 5]. Automation implies that the system can perform decisions and tasks autonomously to improve efficiency and accuracy while energy efficiency requires a system that can operate optimally to maximize energy efficiency through dynamic light conditioning, especially in resource-limited environments. These features are essential to achieve an efficient ISLS.

Although LoRa technology has many advantages regarding ISLS, it also faces some challenges. LoRa technology was originally designed to support the communication needs of low power consumption and wide-area coverage with low data transmission rates (typically between 0.3 kbps and 50 kbps) [6]. The low transmission rate limits its ability to handle large amounts of data, especially in ISLS scenarios that require complex analyses and fast decision-making. Moreover, while LoRa is a low power technology, using LoRa on ISLS currently lacks sufficient energy management policies or methodologies and only provides basic communication capabilities, whereas ISLS need to adjust brightness according to environment and demand to maximize energy efficiency [7].

Hence, although LoRa technology is well suited for remote communication and monitoring due to its advantages, current LoRaWAN-based ISLS systems have difficulty in effectively applying advanced models, such as machine learning and deep learning, which are critical for achieving adaptive control and automation in dynamic environments. Therefore, by combining AI technology with LoRaWAN, the system can take full advantage of LoRa communications while utilizing the computational power of AI to enhance the responsiveness and automation of the system, resulting in smarter and adaptive management in a rapidly changing environment.

In this paper, the AI Model predicts future visibility by analyzing historical and current data on temperature, humidity, weather condition data, and visibility. It also incorporates image data captured by cameras to automatically identify environmental brightness and traffic flow to generate scenarios for dynamically adjusting streetlights' brightness to optimize power consumption. By analyzing these multidimensional data, the AI Model not only improves the adaptability of ISLS but also effectively saves energy, demonstrating significant advantages in complex environments. Furthermore, compared to traditional methods, the integration of AI empowers the system with learning capabilities that continuously optimize decisions based on changing environmental data, which ensures more accurate decision-making under diverse conditions while reducing manual intervention requirements and enhancing energy management efficiency within the system. Consequently, ISLS maintains relatively higher energy utilization across various environments, significantly optimizing overall power consumption. Therefore, this paper introduces a deep learning-based AI Model to bridge gaps in LoRa technology's data processing rate and power consumption management while further promoting its application in ISLS.

The remainder of this paper is organized as follows. Section 2 reviews the related work and discusses the limitations of existing ISLS technologies. Section 3 presents the proposed system architecture and methodology. Section 4 provides experimental results and analysis. Finally, Section 5 concludes the paper and outlines future research directions.

2 RELATED WORKS

With the popularity of current smart city concept, ISLS based on LoRaWAN and deep learning have been an active research topic.

Among them, LoRaWAN technology has attracted a lot of attention in ISLS due to its long-range and low-power communication capability. For example, Zhang et al. designed an ISLS combining NB-IoT and LoRa communication technology to realize automatic control of streetlights through real-time traffic flow information to achieve basic energy saving and intelligent management [8]. Rahman et al. developed an intelligent highway lighting system based on LoRa and 802.15.4 communication protocols that ensures visual comfort for drivers through adaptive lighting while improving energy efficiency [9]. Akindipe et al. propose a smart streetlight system that utilizes multiple sensors and the LoRaWAN protocol to collect and transmit environmental data, supports intelligent dimming and remote control, and integrates electric vehicle charging capabilities [10]. In addition, Sanchez-Sutil and Cano-Ortega's ISLS using LoRa technology dynamically adjusts the lighting level through optimization algorithms to achieve more efficient energy use [11].

While LoRa technology has played a significant role in advancing ISLS, many existing studies emphasize basic remote control and low-power communication. Optimization algorithms are often employed to regulate brightness, but these approaches typically utilize simpler environmental data, which may limit their ability to address more complex, multidimensional factors. Consequently, their energy optimization strategies might have room for improvement in terms of adaptability. Incorporating a broader range of environmental data could enhance decision-making processes and potentially lead to further improvements in energy efficiency.

Deep learning techniques are also widely used in the research of ISLS. It can maximize energy efficiency through prediction and optimization algorithms. When dealing with time series data, traditional recurrent neural networks (RNN) and their variants, although theoretically applicable, often face the problem of vanishing or exploding gradients in practical applications, especially when dealing with long series data [12]. In order to solve these problems, LSTM introduced a gating mechanism, which effectively overcame the limitations of RNN when dealing with long-time sequences [13, 14]. Therefore, many researchers prefer LSTM networks when dealing with long-time sequence data. For example, Tukymbekov et al. described an intelligent autonomous street lighting system powered entirely by solar energy and based on weather forecasting, which uses a LSTM network to predict the energy output of the solar panels, and LoRa technology to enable efficient communication and data transfer between the devices, thus automatically adjusting the power consumption of the streetlights according to the environmental changes [15]. De Paz et al. described an intelligent lighting control system incorporating techniques such as Artificial Neural Networks (ANN) and Multi-Agent Systems to achieve centralized management and optimization of public lighting [16]. Mohandas et al. investigated an ISLS based on ANN and Fuzzy Logic Controller, which was implemented through light sensors, motion sensors and Passive Infrared sensors for real-time environment detection and lighting demand analysis [17].

However, these studies face certain challenges in applying deep learning models effectively. While they demonstrate the potential of deep learning in specific scenarios, many focus primarily on single environmental factors or depend on preset rules for decision-making. This approach can limit their ability to comprehensively integrate and analyze multidimensional data, which may reduce the overall level of automation and adaptability. Furthermore, the application of deep learning models in these studies is often tailored to optimize specific scenarios, which could constrain their flexibility in addressing complex environments and dynamic changes. As a result, achieving consistently high energy efficiency under varying conditions remains an area for further exploration.

In contrast, the proposed system in this research adopts a comprehensive approach to data processing and analysis. By leveraging LoRa technology for low-power, long-range communication, it integrates a deep learning-based AI model designed to analyze multidimensional environmental data. This integration aims to facilitate brightness adjustment, enhance automation, and optimize power consumption, offering a potential solution for addressing the challenges in intelligent streetlight systems.

3 METHODOLOGY

This section aims to present the design of the ISLS framework based on LoRaWAN technology. Inspired by the aforementioned, several deep learning models, specifically the CNN and LSTM-based models, are twisted to implement the automation of control on the ISLS and facilitate the optimization of power consumption. In the following sections, this paper will detail our approach and methodologies, the first part details the system design, the second part discusses the multidimensional data processing, the third part focuses on the implementation of CNN and LSTM models and the final part specifies the rule determiner for brightness adjustment.

3.1 System Design of LoRaWAN for Intelligent Streetlight System

The ISLS consists of end devices, gateways, network servers and application servers. Each streetlight is equipped with various sensors to detect the environment and transmit data to the gateway via LoRa radio technology [18]. The gateway receives data packets from the end devices and forwards them to the network server over a 4G network. The network server is responsible for managing and controlling the entire network, receiving data from the gateway, parsing, processing and forwarding it to the appropriate application server [19]. Finally, the application server then analyzes the data in depth and sends control commands to end devices according to the environmental changes to adjust the brightness of streetlights [20]. Due to the limitation of the 1 % duty cycle of the LoRa protocol, the total communication time of each streetlight must be controlled within 432 seconds per day, which limits our system from real-time control, but since the AI Model in this paper can predict the future environmental changes so as to adjust the streetlights ahead of time, this research chooses to adjust the brightness of the streetlights once every 5 minutes. Figure 1 shows the system framework.

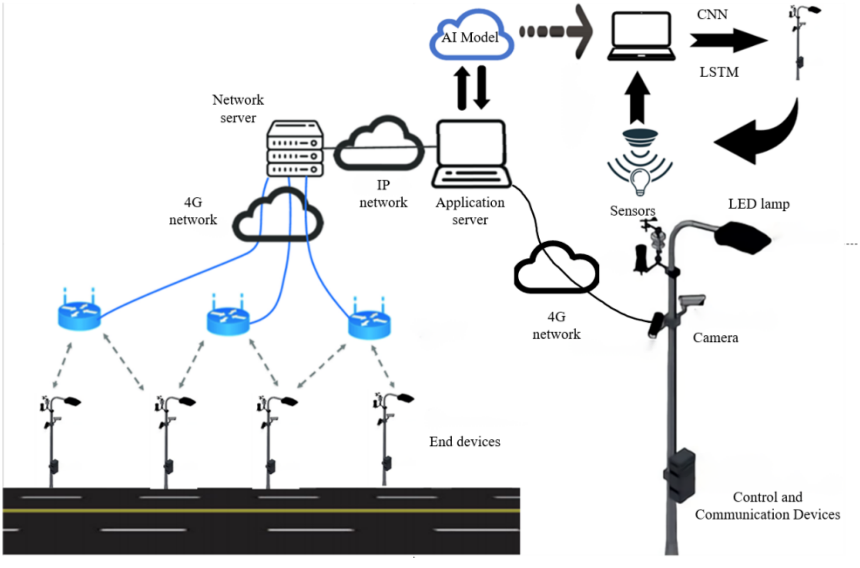


Figure 1. LoRaWAN-based streetlight system framework with AI-enabled power control

Figure 1 depicts a novel LoRaWAN-based streetlight system in which multiple streetlights are installed along a roadway. Each streetlight is an end device in the LoRaWAN framework, equipped with sensors, LEDs, and a set of control and communication devices. These devices include micro control units (MCUs) and LoRa communication modules to manage the operation of the streetlights. These end devices are responsible for collecting environmental data and sending it out over the LoRa network. The sensors in the system include temperature and humidity sensors, whose main role is to monitor the environmental conditions. As well known, lower temperatures usually lead to reduced visibility at night, while higher humidity may indicate haze or rainfall, which further reduces visibility [21]. Given the limited bandwidth of LoRa networks, the size of the packets they transmit is restricted, so only basic environmental data can be sent [22]. More complex weather condition data including cloudy, cloudy, sunny or hazy and visibility data are retrieved in the cloud via the OpenWeatherMap API. Subsequently the system combines sensor data with weather condition data and visibility, processes and analyzes them through an AI Model, utilizes learning techniques to predict changes in visibility.

In order to detect traffic flow and obtain information about the brightness of the current environment, each streetlight also integrates a camera. Brighter environments usually require relatively low power consumption, while unlit areas require brighter light [23]. However, monitoring environmental brightness often faces several challenges. Due to the low bandwidth limitations of LoRaWAN, it is not possible to transmit full HD image data [22]. Therefore, the camera is not used as a LoRaWAN end device and the image data captured by the camera will be transmitted directly to the application server via 4G network. The application server performs detailed image processing and analysis using learning-based techniques to extract key information such as traffic information and the brightness level of the environment. Based on the information, as well as the predicted visibility information, a final brightness adjustment decision is made and transmits commands to the end device to adjust the brightness of the lights accordingly.

In our system, due to the limitation of MCU computational resources, current MCU cannot support complex AI Models. Therefore, the control logic of our ISLS relies on the application server for processing. The application server receives multidimensional data from multiple sources transmitted over LoRa and 4G networks and analyzes the received data by running complex AI Models through its powerful computational capabilities, and generates corresponding control commands based on the analysis results to regulate the brightness of streetlights.

In the ISLS shown in Figure 1, we need to collect four key types of data: sensor data, visibility and weather condition data, and image data. The former three types of data are time-series-based data, while image data from cameras contain complex spatial features. In order to adequately handle these two different data types, we need to use two different deep learning models to process these data separately. We should choose a model that is sensitive to time series to process former three types of data, and a model that can extract brightness and traffic flow features from images to process image data.

Similar to previous works, for processing time-series data, this paper also selects LSTM networks to process these long time-series data [15]. For complex spatial data, LSTM may become less efficient [24]. Therefore, we propose using CNN to process image data, as CNN has a significant advantage in extracting spatial features through convolutional layers, enabling it to capture brightness information and effectively analyze traffic flow [25]. By combining the advantages of CNN and LSTM, the system can accurately analyze and predict based on the comprehensive environmental information to achieve intelligent and efficient lighting control.

Therefore, we twist two deep learning models CNN and LSTM to perform intelligent power control functionalities. If only LSTM is used to process all the data, it will result in relatively low image processing accuracy, which will cannot accurately recognize and detect pedestrians and vehicles on the road [24]. Whereas, if only CNN is used to process all the data, zero-padding of the time-series data is required, and the processing may result in the valid information in the data, thus affecting the prediction accuracy [26]. In summary, using two independent models can optimize the processing of time-series data and image data separately to ensure better results. In this way, by leveraging the advantages of CNN and LSTM, the system can effectively adjust lighting based on the current environment and the predicted visibility, reducing power consumption and preparing the streetlights for future conditions.

While previous studies have focused on basic control functions, our approach further introduces a rule determiner to improve the automation and energy efficiency of the system [8, 9, 10, 11]. Therefore, in the next sections, we will detail the data processing procedures of the AI Model, the specific implementations of the CNN- and LSTM-twisted model, as well as the specific rule determiner and demonstrate the potential of their application in ISLS.

3.2 Data Preprocessing

In this section, in order to enable the AI Model to effectively utilize different types of data, this paper preprocesses environmental data, weather condition data and image data according to the characteristics of each data type to ensure that they can be effectively integrated and used for model training. Next, the processing steps for each type of data are described in detail.

3.2.1 Environment Data Preprocessing

The environmental data include temperature, humidity, and visibility. Since all three are numerical data, they can be preprocessed in a unified way for subsequent model training.

Due to the large magnitude differences between these variables, as well as variations in their mean value (μ) and standard deviation (σ), certain features may dominate the model training process and affect performance. To balance feature contributions and reduce computational burden, we apply Z-score normalization to temperature, humidity, and visibility data [27]. This transformation standardizes the data to have a mean of 0 and a standard deviation of 1, ensuring consistency across features and improving model stability and training effectiveness.

Each column in the data matrix represents a feature, including temperature, humidity, and visibility, and each row represents a sample. The Z-score normalization is applied to each column using the formula:

$$Z = \frac{x - \mu}{\sigma}, \tag{1}$$

where x is the original data value, μ is the mean value of the data, and σ is the standard deviation of the data. With this transformation, each feature in the dataset has zero mean and unit variance, which enhances the model's ability to learn effectively and reduces the risk of overfitting.

3.2.2 Weather Condition Data Preprocessing

Since weather conditions are categorical data, using them directly may result in the model failing to understand the relationship between these categories. To address this problem, we perform one-hot encoding on the weather condition data. This transformation converts categorical data into binary vectors, making it easier for the model to capture the relationships between different weather conditions and their impact on visibility and other factors [28].

After the environmental data and weather condition data are preprocessed, these data are combined into a unified input vector that integrates temperature, humidity, weather conditions, and visibility to provide comprehensive inputs to the training model so that it can better understand the impact of these factors on visibility.

3.2.3 Image Data Preprocessing

The image data are used to detect traffic flow and extract brightness information so that the system can more effectively optimize the power consumption of streetlights. To ensure that the model can be trained effectively, each image is labeled in a consistent way.

Each image is divided into 16×16 pixel regions, and the overall brightness is calculated by averaging these local values, providing a more precise measure than relying on a single global brightness value. To ensure that the classification reflects meaningful differences in visual perception, this research categorizes image brightness into four levels based on the average brightness value: below 50 as "low brightness", between 50 and 85 as "medium brightness", between 85 and 115 as "high brightness", and above 115 as "ultra-high brightness".

The classification of traffic flow in this research builds upon prior studies, which highlight the effectiveness of vehicle and pedestrian counts as quantitative measures for analyzing traffic patterns and estimating flow density [29, 30, 31]. Drawing inspiration from these studies, this research establishes classification criteria by defining "high" traffic flow as instances where an image contains at least 5 vehicles or 8 pedestrians, or when the combined total reaches 10 or more; otherwise, the flow is categorized as "low". These classifications provide reliable labels for both brightness and traffic flow, thereby ensuring high-quality data to support subsequent model training.

After preprocessing all the data, this research uses a CNN-LSTM-twisted model to optimize power consumption by comprehensively analyzing the preprocessed multidimensional data.

3.3 Optimized Control Strategies Using CNN and LSTM

This section elaborates on the design details of the CNN and LSTM models, covering the structural configurations of the input, hidden, and output layers, as well as the implementation of the optimization strategy. It also summarizes the working mechanism of the integrated AI model.

3.3.1 Application and Optimization of CNN Model in Intelligent Streetlight System

In order to effectively detect traffic flow and environmental brightness, we select the VGG16 model as the CNN part. VGG16 contains 13 convolutional layers and 3 fully connected layers, which can effectively extract complex spatial features, and it performs well in image classification and feature extraction tasks with relatively high accuracy and stability, providing reliable environmental information support for ISLS [32]. For better adaptation to the system requirements, we customize the VGG16 model by modifying the output dimensions of the original three fully connected layers to 1024 and 256, and then combine them with the Softmax function to output the results of eight categories formed by different environmental brightness and traffic flow levels. In addition, between each fully connected layer, we add the ReLU activation function and a Dropout layer to enhance the nonlinear representation of the model and reduce the risk of overfitting [33, 34]. The specific VGG16 architecture is shown in Figure 2.

In input image processing, we adjust the resolution to 224×224 to reduce the computational burden while still ensuring classification accuracy [35]. The first two layers of the model use convolutional layers with 3×3 kernels, and each convolutional layer is followed by a 2×2 max-pooling layer to extract essential image features and reduce computational complexity. The next layers, from 3 to 10, continue to extract complex features through convolution and pooling, while layers 11 to 13 are used to identify subtle visual elements, enabling the model to more accurately determine environmental brightness and traffic flow levels. The convolution operation is expressed as follows:

$$\mathrm{conv}(x) = \mathrm{ReLU}(w * x + b), \tag{2}$$

where w is the convolution kernel, x is the input image, and b is the bias. The operation of each pooling layer can be represented as:

$$\mathrm{Pool}(x) = \max(x_i, x_j), \tag{3}$$

where x_i and x_j denote the elements in the pooling window. Next, the image features after convolution and pooling are flattened and fed into the fully connected layer, which can be written as:

$$\mathrm{fc}(x) = \mathrm{ReLU}(W_f x + b_f), \tag{4}$$

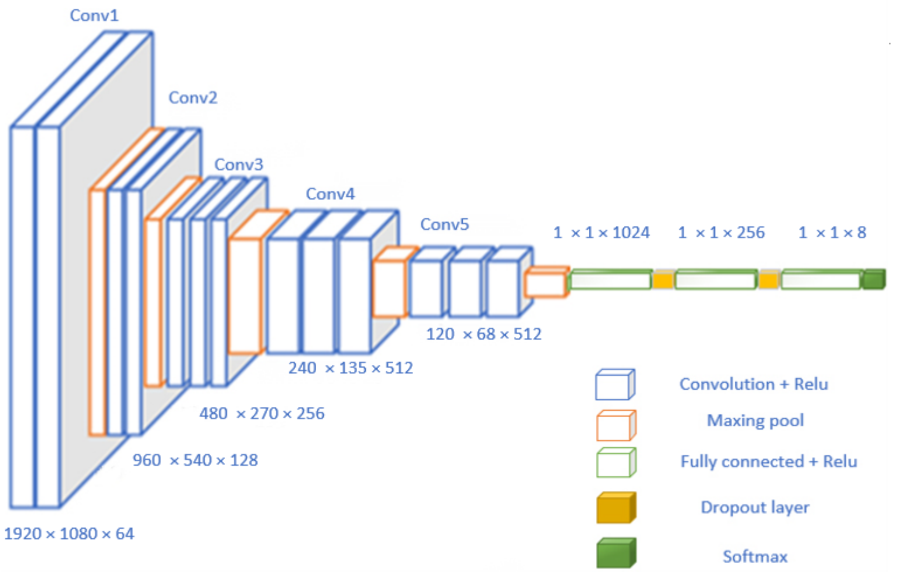


Figure 2. Detailed architecture of the VGG16 network

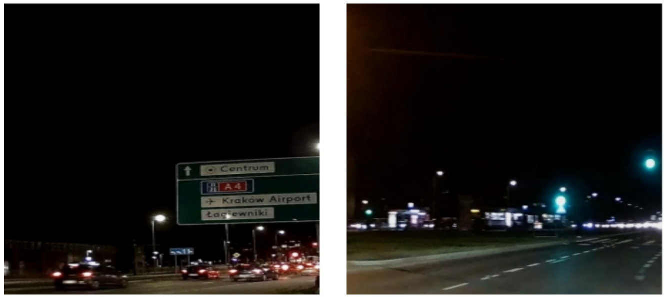
where W_f is the weight matrix, x is the input vector, and b_f is the bias.

To accurately determine brightness category and traffic flow, we use the Softmax activation function in the classification task because it converts the output into a probability distribution [36]. The mathematical expression of the Softmax function is as follows:

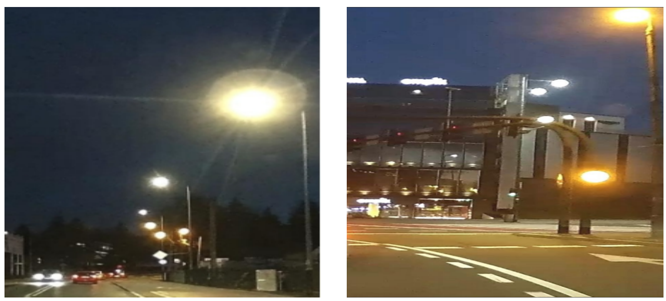
$$\mathrm{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}, \tag{5}$$

where z_i is the raw score of the i^{th} category and N denotes the total number of categories. In our model, $N = 8$, which corresponds to the combination of four brightness levels (ultra-high, high, medium, and low) and two traffic flow levels (high and low). Figure 3 shows examples of different brightness levels and traffic flow levels.

In our system, the task of the model is to make multi-category predictions for the combined categories of brightness level and traffic flow. For this purpose, we employ a cross-entropy loss function (CELF) to measure the discrepancy between predicted and actual labels [37]. Since category imbalance may cause the model to be biased toward categories with more samples, we use weighted cross-entropy to enhance the recognition of minority categories and combine it with L2 regularization to prevent overfitting [38]. The formulas for weighted cross-entropy and L2 regularization are



a) Low brightness with high and low traffic flow



b) Medium brightness with high and low traffic flow

given below:

$$\text{Loss} = -\sum_{i=1}^{N} \omega_i y_i \log(\hat{y}_i) + \lambda \sum_{j=1}^{M} \theta_j^2, \tag{6}$$

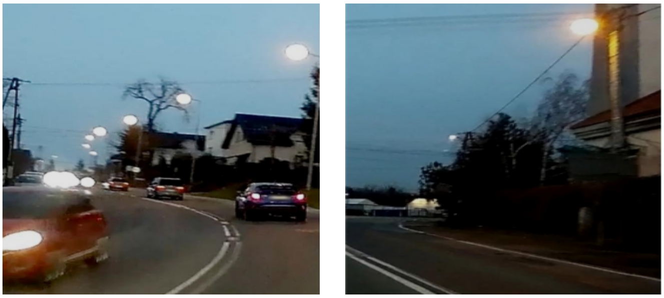
where N is the number of classes, y_i is the actual label, $\hat{y}_i$ is the predicted probability, M is the total number of model parameters, θ_j is the j^{th} weight, and λ is the regularization strength. By assigning different weights ω_i, the loss function can improve the model's ability to recognize categories with fewer samples.

In addition, we use the Adam optimizer for parameter updating because it combines the advantages of momentum and adaptive learning rate to accelerate model convergence and stabilize the training process [39]. It can be described by the following formulas:

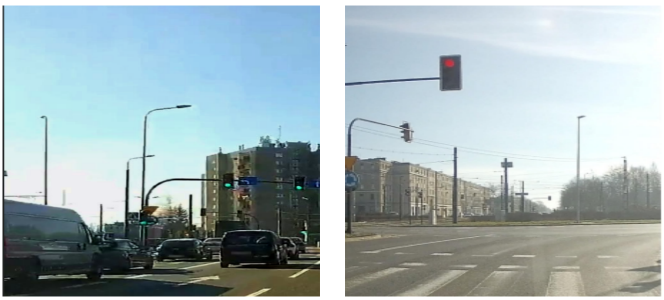
$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \tag{7}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \tag{8}$$

$$\theta_t = \theta_{t-1} - \frac{a m_t}{\sqrt{v_t} + \epsilon}, \tag{9}$$



c) High brightness with high and low traffic flow



d) Ultra-high brightness with high and low traffic flow

Figure 3. Examples of different brightness levels and traffic flow levels

where m_t and v_t are the first- and second-order momentum estimates of the gradient, respectively, a is the learning rate, and ϵ is a tiny value that prevents division by zero.

To further optimize the training process, we also employ the ReduceLROnPlateau learning rate scheduler, which dynamically adjusts the learning rate by multiplying it by a decay factor γ when the validation loss does not improve. This strategy helps quickly optimize the model parameters in the early stages of training while refining model performance in the later stages [40].

With these optimization strategies, the VGG16 model can effectively extract the features of environmental brightness and traffic flow to provide accurate data support for the ISLS. Next, we explore how the LSTM model handles time-series data to further enhance the prediction capability of the system.

3.3.2 Application and Optimization of LSTM Model in Intelligent Streetlight System

The LSTM model is used to process time-series data from the system, including temperature and humidity, visibility, and weather conditions. As shown in Figure 4, through forget gates, input gates and output gates, LSTM can effectively capture long-term dependencies, comprehensively analyze environmental changes and predict future visibility [41].

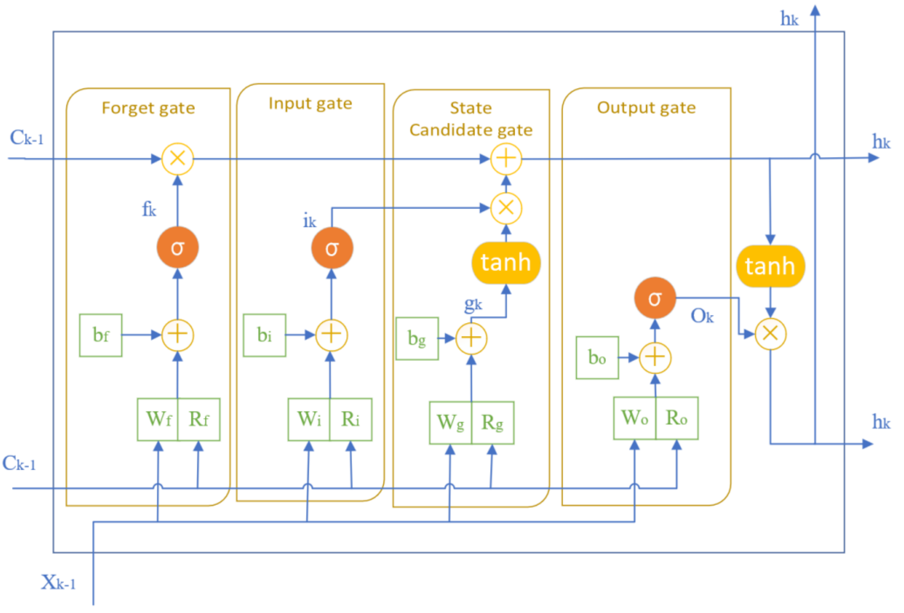


Figure 4. LSTM cell internal mechanisms

In this paper, the LSTM hidden layer consists of three layers of LSTM cells, containing 256 hidden cells in each layer. Each LSTM cell consists of a forget gate, an input gate, a cell state update and an output gate. The forget gate decides the information to be kept or discarded based on the current and past data with the following formula:

$$f_k = \sigma(W_f h_{k-1} + R_f x_k + b_f), \tag{10}$$

where σ is the Sigmoid function, w_f is the weighting matrix for the forget gate, R_f is the input weight matrix, h_{k-1} is the hidden state at the previous moment, x_k is the current input, and b_f is the bias. The output f_k of the forget gate controls which information in the cell state c_{k-1} needs to be retained and which needs to be discarded.

Next, the input gate is responsible for updating the state of the unit in conjunction with new information with the formula:

$$i_k = \sigma(W_i h_{k-1} + R_i x_k + b_i), \tag{11}$$

$$g_k = \tanh(W_g h_{k-1} + R_g x_k + b_g), \tag{12}$$

where i_k is the output of the input gate, g_k is the new candidate state, W_i and W_g are the weight matrices of the input gate and the candidate state, respectively, R_i and R_g are the weight matrices of the inputs.

The cell state update is done by both the forget gate and the input gate with the following equation:

$$c_k = f_k * c_{k-1} + i_k * g_k, \tag{13}$$

where c_{k-1} is the cell state at the previous moment and c_k is the updated cell state.

The output gate generates the hidden state at the current moment, which is given by the equation:

$$o_k = \sigma(W_o h_{k-1} + R_o x_k + b_o), \tag{14}$$

$$h_k = o_k * \tanh(c_k), \tag{15}$$

where o_k is the output of the output gate, h_k is the hidden state at the current moment, W_o is the weight matrix of the output gate, and R_o is the weight matrix of the input.

Finally, the hidden states of the LSTM are processed through two fully connected layers, the first layer reduces the output dimension from 256 to 64 to simplify the computation, and the second layer outputs the final prediction results, which categorize the visibility into three classes: low (less than 1 km), medium (1–5 km), and high (more than 5 km), in order to facilitate the subsequent decision-making on the brightness of the streetlights.

Similar to the VGG16 model, we also adopt CELF as the optimization objective and use the Adam optimizer for parameter updating. Additionally, to prevent overfitting, we introduce the Dropout mechanism in the first two layers of LSTM cells and add an additional Dropout layer between the LSTM layer and the fully connected layer.

With the introduction of the LSTM model, the AI model is capable of capturing the long-term dependency of environmental changes and accurately predict future visibility. By twisting CNN and LSTM models, the ISLS comprehensively analyzes the current environmental state and predicts the future visibility trend, so as to control the streetlights in advance to cope with severe weather. Figure 5 illustrates the proposed AI model architecture.

The VGG16 model is used to extract image data to classify environmental brightness and traffic flow. Meanwhile, the LSTM model predicts future visibility levels by analyzing historical and current sensor data, weather condition data and visibility. Ultimately, the rule determiner adjusts the streetlights brightness settings in advance based on these classifications and predictions to respond to upcoming environmental changes, thus ensuring road safety and optimizing power consumption.

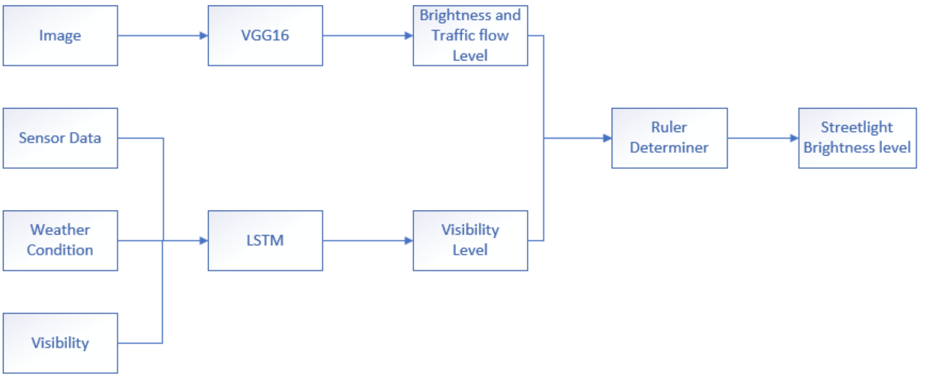


Figure 5. AI model architecture in ISLS

3.4 Brightness Control Strategy Based on the Rule Determiner

This section will specify the control strategy of the rule determiner, the rule determiner adjusts the brightness of streetlights by three factors: traffic flow, environmental brightness and predicted visibility to balance energy saving and safety. The traffic flow level is categorized as "High" and "Low" based on the intensity of road use, and the environmental brightness level is categorized as defined by the external lighting conditions, "Ultra-high", "High", "Medium" and "Low"; predicted visibility level is categorized into "High", "Medium" and "Low" depending on the distance. And the streetlight' brightness levels include "Low", "Moderate", "Medium", "Enhanced", "High" and "OFF". Figure 6 shows in detail the rule determiner's decision-making process for brightness under different conditions.

When the traffic flow level is "High", the system prioritizes traffic and pedestrian safety. In this case, the system adjusts the streetlights' brightness settings in advance based on the current environmental brightness and predicted visibility levels. If the environmental brightness level is "High" and the predicted visibility level is also "High", the system will set the streetlights' brightness level to "Low" to avoid over-illumination. When visibility is predicted to drop to "Medium" level, the system will adjust the brightness to "Moderate" mode in advance to cope with the future lighting shortage. If the predicted visibility decreases to "Low", the system will preemptively increase the streetlights' brightness to "Medium" mode to ensure traffic safety in low visibility conditions.

For the case of "Medium" environmental brightness level, when the predicted visibility level is "High", the system will maintain the streetlights at "Medium" level to provide basic lighting and save energy. When predicted visibility level is "Medium", the system will adjust the brightness in advance to the "Enhanced" mode in order to address the potential risk of reduced visibility; when the predicted visibility level

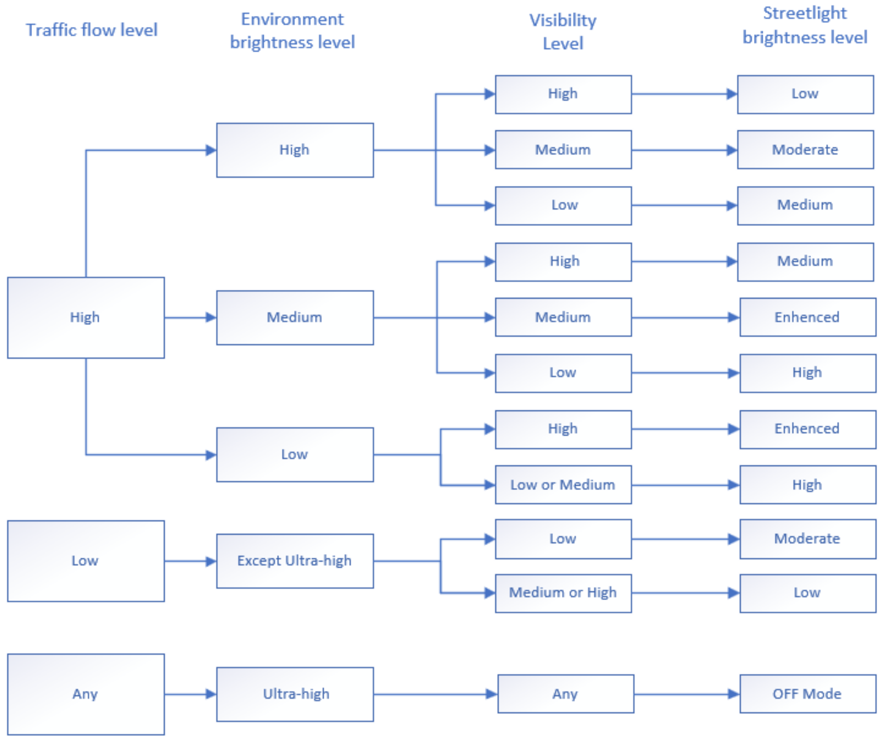


Figure 6. Brightness adjustment rules based on brightness, visibility, and traffic flow level

is "Low", the system will increase the streetlights' brightness to "High" mode to ensure road safety under complicated weather conditions.

When the environmental light level is low, streetlights become the main source of light. If future visibility level is predicted to be "High", the system sets the streetlight brightness to "Enhanced" mode to provide sufficient illumination; while when the predicted visibility level is "Medium" or "Low", the system will raise the streetlights' brightness to "High" mode in advance to cope with the lower visibility under complicated weather conditions to protect the safety of pedestrians and vehicles.

When traffic flow level is low, the main objective of the system is to conserve energy. If the predicted visibility level is "Medium" or "High", the system will keep the streetlights at "Low" level to minimize power consumption; while when the visibility level is predicted to be "Low", the system will increase the streetlights' brightness to "Moderate" mode to cope with the potential risks of future low visibility conditions and to ensure the basic safety of the road.

In addition, when the environmental brightness reaches the "Ultra-high" level, it indicates that there is enough light from outside and it is unnecessary to illuminate the roadway additionally. As a result, the system automatically switches off streetlights regardless of traffic flow or predicted visibility level, further optimizing power consumption.

4 EXPERIMENT AND RESULTS ANALYSIS

This section will introduce the experimental process of the ISLS, covering the data preparation, training process and parameter settings of the CNN and LSTM models, as well as conducting power consumption comparison experiments between the system and the baseline model to comprehensively evaluate its performance at saving energy.

4.1 Data Preparation

4.1.1 CNN Data Preparation

In order to train the VGG16 model, this research uses the Polish road image dataset published by Mikolaj Kolek on the Kaggle platform [42]. The dataset includes a variety of lighting conditions, ranging from low to high light levels, as well as diverse traffic scenarios, from low to high traffic density. These varied conditions enable the model to learn from a wide range of real-world situations, effectively supporting the classification tasks for brightness and traffic flow.

However, the dataset suffers from class imbalance, with some categories having fewer samples, which may affect the model's ability to learn from these categories during training. To address this issue, this research balanced the dataset by collecting additional samples from underrepresented categories and removing samples from overrepresented categories. As a result, the dataset contains a total of 17 000 samples, providing more comprehensive training data for the model under different traffic flow and lighting conditions, thereby enhancing its classification ability in real-world scenarios.

4.1.2 LSTM Data Preparation

For the LSTM model, this paper chose the dataset published by Norbert Budincsevity in Kaggle for the Szeged region of Hungary for the years 2006 to 2016, which contains 96 454 records on temperature, humidity, weather conditions and visibility, with a range of features that are highly consistent with the environmental variables of ISLS [43].

Due to the long data sending interval and duty cycle limitations of LoRa technology, it is important to ensure that the data features match those of the actual deployment environment [44]. Although the dataset is different from the actual data collection scenarios of our system, it still provides rich environment features for effective learning and training of our model.

4.2 Model Training

In the training of VGG16 and LSTM, in order to avoid overfitting and improve the accuracy of performance evaluation, we used ten-fold cross-validation, where 90 % of the data is used for training and 10 % for testing and iterative updating through forward and back propagation ensured that every data point was involved in training and validation, thus comprehensively evaluating the overall performance of the model [45].

4.2.1 VGG16 Model Training Process and Parameter Settings

We empirically set the initial learning rate $a = 0.00010$, L2 regularization coefficient $\lambda = 0.005$, learning rate decay coefficient $\gamma = 0.5$, Dropout rate 0.5, and the training period is 70 epochs. In order to effectively evaluate the stability and generalization ability of the model, we used ten-fold cross-validation. By averaging the results of each training and validation, the average loss curve shown in Figure 7 was generated.

During the training process, considerable fluctuations in the validation loss were observed, primarily due to the large updates of model parameters at high learning rates. To address the issue, this paper implemented the ReduceLROnPlateau learning rate adjuster, which automatically lowers the learning rate when the model performance plateaus, thereby optimizing the loss further. As shown in Figures 7 and 8, the fluctuations in the loss curve gradually decrease and stabilize after each learning rate adjustment.

Additionally, it was noted that the validation loss is generally lower than the training loss. The phenomenon can be attributed to the use of Dropout and L2 regularization during training, which mitigate overfitting by randomly deactivating certain neurons and imposing weight penalties, resulting in an elevated training loss. In contrast, these regularization techniques were disabled during the validation phase, allowing the model to fully utilize all neurons, thus yielding a lower validation loss.

Overall, although the validation loss fluctuates considerably in the early stages, it gradually stabilizes as training progresses, indicating an improvement in the model's generalization ability. After 70 training epochs, the model demonstrates promising convergence in both training and validation loss, validating the effectiveness of the learning rate scheduling and regularization strategies in optimizing model performance.

Additionally, to comprehensively evaluate the model, this research calculated multiple metrics including Accuracy, F1 Score, True Positive Rate (TPR), and False Positive Rate (FPR) for each fold. Accuracy is a valid measure of the model's overall classification performance in balanced datasets, but it may not fully reflect the model's capability in the presence of class imbalance. Therefore, we introduced the F1 score, which balances Precision and Recall. Moreover, TPR and FPR are also considered to assess the model's ability to correctly identify positive and negative categories, respectively [46].

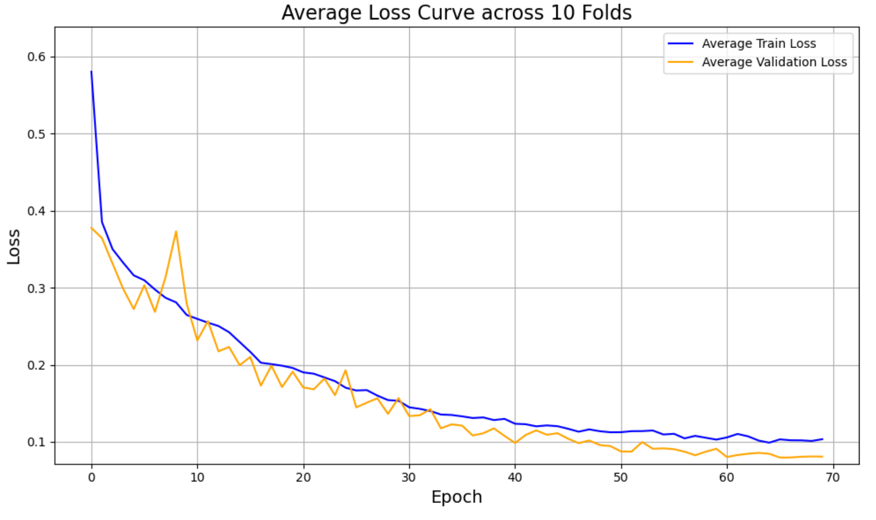


Figure 7. Model training and validation loss over epochs

In ten-fold cross-validation, the model achieves an average F1 score and accuracy of 0.974, indicating that it performs well in image brightness and traffic flow classification, and shows promising ability in handling category imbalance. In addition, the average TPR and FPR of the model are 0.975 and 0.004, respectively, indicating that the model can accurately identify most positive samples with a low misclassification rate, which further proves its strong classification and generalization ability.

In summary, the model exhibits balanced and stable performance across various metrics, highlighting its potential for reliable application in ISLS, where it can achieve efficient and stable image classification even in complex environments.

4.2.2 LSTM Model Training Process and Parameter Settings

For LSTM model, we empirically set the learning rate to $a = 0.0005$, the L2 regularization coefficient to $\lambda = 0.0001$, the Dropout rate to 0.5, and chose 30 epochs of training. These parameters are set to ensure that the model converges quickly and suppresses overfitting. Similar to the previous section, this research also used ten-fold cross-validation to effectively assess the stability and generalization ability of the model, and averaged the results of each training and validation to generate the average loss curve.

As shown in Figure 9, the high initial learning rate makes the training loss decrease rapidly in the first 5 epochs, which enables the model to converge quickly in the early stage. The L2 regularization and Dropout successfully suppress overfitting by reducing the gap between the training and validation loss curves, and keep the

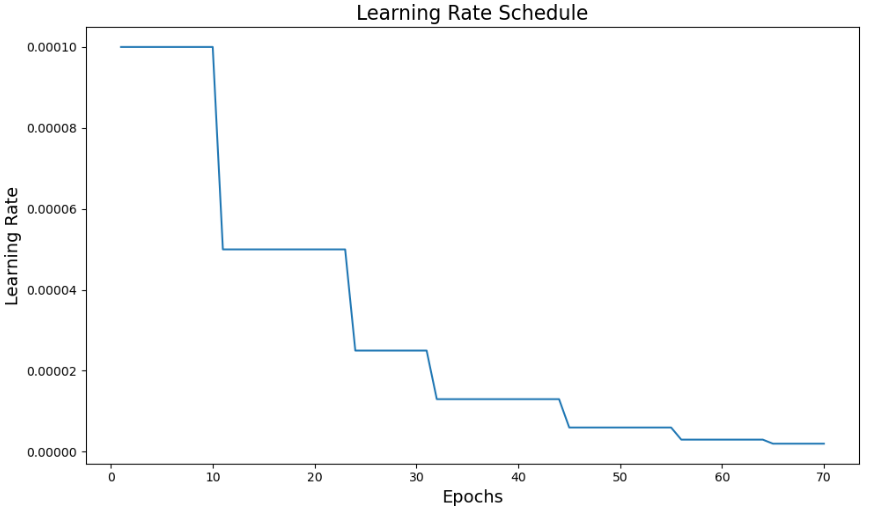


Figure 8. Model learning rate over epochs

validation loss smooth. In addition, the training and validation losses stabilize around the 10^{th} epoch, which indicates that the model has basically completed the feature learning, and satisfying the training or validation requirements. Therefore, 30 epochs of training ensures that the model achieves the desired results and avoids overtraining and overfitting.

Consistent with the results of VGG16, the regularization techniques used in the training phase, including Dropout and L2 regularization, result in the validation loss being lower than the training loss. The model effectively suppresses overfitting through these strategies during training, enhancing its generalization ability on the validation set and allowing it to maintain higher performance when faced with unseen data.

To comprehensively evaluate the model performance, this research still adopted the evaluation metrics of F1 score, accuracy, TPR and FPR to gain insights into the performance of the model in each category. In the 10-fold cross-validation, the LSTM model in this paper has an average F1 score of 0.9930 and an average accuracy of 0.9929, which indicates that it has excellent accuracy in the visibility classification task, and is capable of effectively capturing the effects of temperature, humidity and weather conditions on visibility. The average TPR and FPR of the model are 0.9927 and 0.0026, respectively, showing that the model accurately recognizes the majority of positive samples while having a low misclassification rate.

Overall, the model demonstrates stable and accurate visibility classification ability under different environmental conditions, providing reliable environmental information for ISLS.

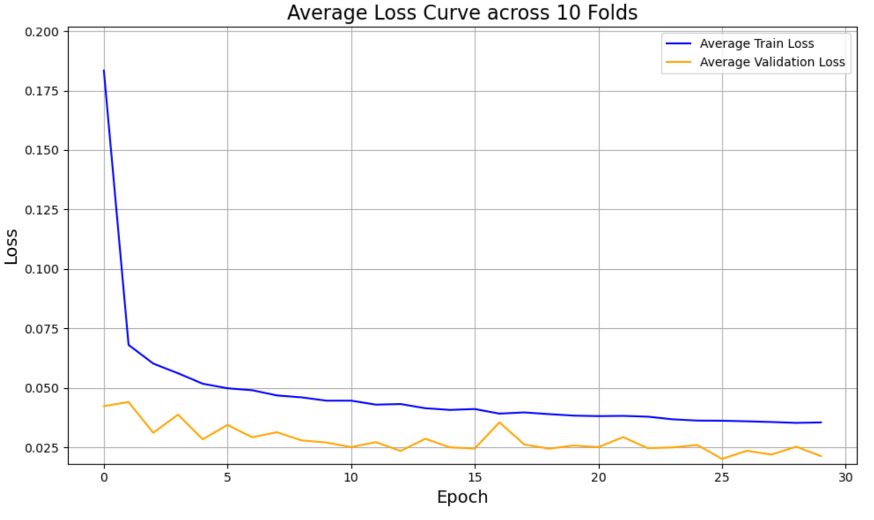


Figure 9. Model training and validation loss over epochs

4.3 Power Consumption Comparison Experiment Between AI Model and Baseline Model

To evaluate the energy-saving performance of a LoRaWAN-based ISLS, this research simulated a system comprising 100 streetlights with five adjustable power levels: high (150 W), enhanced (125 W), medium (100 W), lower (75 W), and low (50 W). The total power consumption of the system was calculated by integrating time-series data and corresponding image information into an AI-based model.

For comparison, the baseline model proposed by Kanthi and Dilli was employed [47]. This model adjusts streetlight brightness based on a predefined schedule that divides a day into active and idle periods, combined with environmental brightness. While this approach achieves a certain level of energy savings, its over-reliance on fixed rules reduces its adaptability to dynamic environmental changes. In contrast, the AI model leverages multidimensional information to perform intelligent analysis and prediction, enabling it to balance road safety and energy efficiency more effectively.

To assess the optimization effects, both models were evaluated under identical conditions for a period of 100 days, with power consumption data recorded and analyzed.

Figure 10 highlights notable differences in power consumption between the baseline model and the AI model. The baseline model exhibits a stable yet inflexible energy consumption pattern due to its reliance on fixed-time rules. This rigidity limits its ability to respond to varying environmental and traffic conditions, resulting in less dynamic adjustments and consistent energy use. In contrast, the AI model

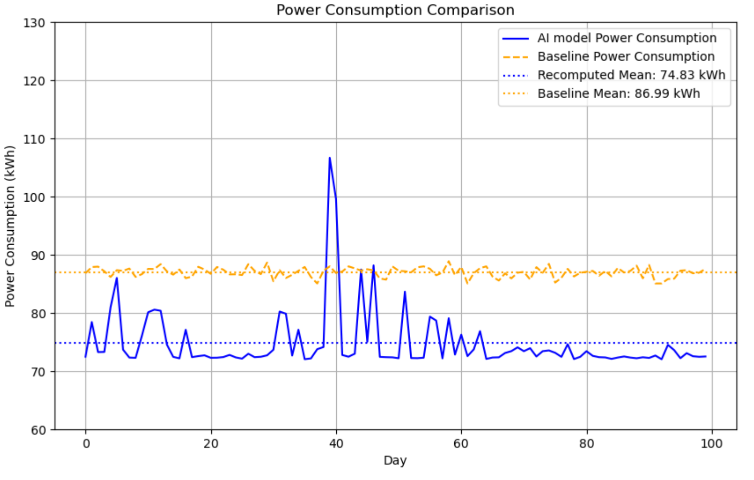


Figure 10. Power consumption comparison

dynamically adjusts streetlight brightness by predicting environmental changes or responding to traffic data. While this adaptation introduces some fluctuations in power consumption, it can significantly improve energy efficiency.

On average, the daily power consumption of the AI model was recorded at 74.83 kWh, compared to 86.99 kWh for the baseline model, achieving a reduction of 12.16 kWh per day. This equates to a 13.98 % decrease in power consumption, underscoring the AI model's capability to minimize unnecessary energy use through intelligent predictions and analyses.

In conclusion, the AI model demonstrates clear advantages over the baseline model by proactively adapting to dynamic environmental conditions, enhancing energy efficiency while contributing to road safety, and offering a potentially more sustainable approach to streetlight management.

5 CONCLUSION

This paper proposes an ISLS that combines LoRaWAN technology with a CNN-LSTM-twisted model to achieve power optimization through automatic brightness adjustment. The main contributions of this research are summarized as follows:

1. A novel LoRaWAN-based ISLS is introduced to construct controlling topology, which facilitates the automatic control of streetlight system.
2. Based on the novel LoRaWAN framework for ISLS, this research combines multidimensional data, processing using CNN and LSTM.

3. A rule determiner is developed to integrate the outputs of the CNN and LSTM models, enabling intelligent brightness adjustment of streetlights. This mechanism optimizes power consumption by dynamically balancing energy utilization and automation control, ensuring efficient and adaptive operation of the ISLS.

Experimental results show that the CNN part of the model can accurately analyse the current brightness and traffic flow levels, while the LSTM part efficiently predicts future visibility changes. Combining both models enables the system to adjust the streetlights' brightness in advance to cope with adverse weather conditions, thus significantly improving the system's responsiveness and reliability for ISLS. Using the CNN-LSTM-twisted model reduces the average daily power consumption by 13.98 % compared to the typical energy-saving method, showing the promising potential of the system in smart city energy management.

However, there are still some shortcomings in this paper, the high complexity of the model and the requirement of computational resources may affect the cost and efficiency of the actual deployment; in addition, the dataset used in this research is mainly from the public data, which may have some differences from the actual application environment, and the future research can focus on optimizing the model structure, reducing the requirement of computational resources, and expanding the system application scenarios to integrate with other urban management systems, as well as strengthening the understanding of the actual application scenarios to enhance the applicability and reliability of the model in real environments, so as to better promote the construction and development of the smart city.

Declaration of Conflicting Interests. The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Data Sharing Agreement. The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Funding. The author(s) received no financial support for the research, authorship, and/or publication of this article.

REFERENCES

[1] Jiang, Y.—Shuai, Y.—He, X.—Wen, X.—Lou, L.: An Energy-Efficient Street Lighting Approach Based on Traffic Parameters Measured by Wireless Sensing Technology. IEEE Sensors Journal, Vol. 21, 2021, No. 17, pp. 19134–19143, doi: 10.1109/JSEN.2021.3089208.

[2] Gagliardi, G.—Lupia, M.—Cario, G.—Tedesco, F.—Cicchello Gaccio, F.—Lo Scudo, F.—Casavola, A.: Advanced Adaptive Street Lighting Systems for Smart Cities. Smart Cities, Vol. 3, 2020, No. 4, pp. 1495–1512, doi: 10.3390/smartcities3040071.

[3] Chen, Z.—Sivaparthipan, C. B.—Muthu, B. A.: IoT Based Smart and Intelligent Smart City Energy Optimization. Sustainable Energy Technologies and Assessments, Vol. 49, 2022, Art. No. 101724, doi: 10.1016/j.seta.2021.101724.

[4] Shanmuga Sundaram, J. P.—Du, W.—Zhao, Z.: A Survey on LoRa Networking: Research Problems, Current Solutions, and Open Issues. IEEE Communications Surveys & Tutorials, Vol. 22, 2020, No. 1, pp. 371–388, doi: 10.1109/COMST.2019.2949598.

[5] Shahzad, G.—Yang, H.—Ahmad, A. W.—Lee, C.: Energy-Efficient Intelligent Street Lighting System Using Traffic-Adaptive Control. IEEE Sensors Journal, Vol. 16, 2016, No. 13, pp. 5397–5405, doi: 10.1109/JSEN.2016.2557345.

[6] Gkotsiopoulos, P.—Zorbas, D.—Douligeris, C.: Performance Determinants in LoRa Networks: A Literature Review. IEEE Communications Surveys & Tutorials, Vol. 23, 2021, No. 3, pp. 1721–1758, doi: 10.1109/COMST.2021.3090409.

[7] Pardo-Bosch, F.—Blanco, A.—Sesé, E.—Ezcurra, F.—Pujadas, P.: Sustainable Strategy for the Implementation of Energy Efficient Smart Public Lighting in Urban Areas: Case Study in San Sebastian. Sustainable Cities and Society, Vol. 76, 2022, Art. No. 103454, doi: 10.1016/j.scs.2021.103454.

[8] Zhang, J.—Zeng, W.—Hou, S.—Chen, Y.—Guo, L.—Li, Y.: A Low-Power and Low Cost Smart Streetlight System Based on Internet of Things Technology. Telecommunication Systems, Vol. 79, 2022, No. 1, pp. 83–93, doi: 10.1007/s11235-021-00847-1.

[9] Rahman, M. A.—Asyhari, A. T.—Obaidat, M. S.—Kurniawan, I. F.—Mukta, M. Y.—Vijayakumar, P.: IoT-Enabled Light Intensity-Controlled Seamless Highway Lighting System. IEEE Systems Journal, Vol. 15, 2021, No. 1, pp. 46–55, doi: 10.1109/JSYST.2020.2975592.

[10] Akindipe, D.—Olawale, O. W.—Bujko, R.: Techno-Economic and Social Aspects of Smart Street Lighting for Small Cities – A Case Study. Sustainable Cities and Society, Vol. 84, 2022, Art. No. 103989, doi: 10.1016/j.scs.2022.103989.

[11] Sanchez-Sutil, F.—Cano-Ortega, A.: Smart Regulation and Efficiency Energy System for Street Lighting with LoRa LPWAN. Sustainable Cities and Society, Vol. 70, 2021, Art. No. 102912, doi: 10.1016/j.scs.2021.102912.

[12] Rithani, M.—Prasanna Kumar, R.—Doss, S.: A Review on Big Data Based on Deep Neural Network Approaches. Artificial Intelligence Review, Vol. 56, 2023, No. 12, pp. 14765–14801, doi: 10.1007/s10462-023-10512-5.

[13] Papastefanopoulos, V.—Linardatos, P.—Panagiotakopoulos, T.—Kotsiantis, S.: Multivariate Time-Series Forecasting: A Review of Deep Learning Methods in Internet of Things Applications to Smart Cities. Smart Cities, Vol. 6, 2023, No. 5, pp. 2519–2552, doi: 10.3390/smartcities6050114.

[14] Cortez, B.—Carrera, B.—Kim, Y. J.—Jung, J.: An Architecture for Emergency Event Prediction Using LSTM Recurrent Neural Networks. Expert Systems with Applications, Vol. 97, 2018, pp. 315–324, doi: 10.1016/j.eswa.2017.12.037.

[15] Tukymbekov, D.—Saymbetov, A.—Nurgaliyev, M.—Kuttybay, N.—Dosymbetova, G.—Svanbayev, Y.: Intelligent Autonomous Street Lighting System Based on Weather Forecast Using LSTM. Energy, Vol. 231, 2021, Art. No. 120902,

doi: 10.1016/j.energy.2021.120902.

[16] De Paz, J. F.—Bajo, J.—Rodríguez, S.—Villarrubia, G.—Corchado, J. M.: Intelligent System for Lighting Control in Smart Cities. Information Sciences, Vol. 372, 2016, pp. 241–255, doi: 10.1016/j.ins.2016.08.045.

[17] Mohandas, P.—Dhanaraj, J. S. A.—Gao, X. Z.: Artificial Neural Network Based Smart and Energy Efficient Street Lighting System: A Case Study for Residential Area in Hosur. Sustainable Cities and Society, Vol. 48, 2019, Art. No. 101499, doi: 10.1016/j.scs.2019.101499.

[18] Osorio, A.—Calle, M.—Soto, J. D.—Candelo-Becerra, J. E.: Routing in LoRaWAN: Overview and Challenges. IEEE Communications Magazine, Vol. 58, 2020, No. 6, pp. 72–76, doi: 10.1109/MCOM.001.2000053.

[19] Milarokostas, C.—Tsolkas, D.—Passas, N.—Merakos, L.: A Comprehensive Study on LPWANs with a Focus on the Potential of LoRa/LoRaWAN Systems. IEEE Communications Surveys & Tutorials, Vol. 25, 2023, No. 1, pp. 825–867, doi: 10.1109/COMST.2022.3229846.

[20] Bouras, C.—Gkamas, A.—Salgado, S. A. K.: Energy Efficient Mechanism for LoRa Networks. Internet of Things, Vol. 13, 2021, Art. No. 100360, doi: 10.1016/j.iot.2021.100360.

[21] Shen, X. J.—Sun, J. Y.—Zhang, X. Y.—Zhang, Y. M.—Zhang, L.—Che, H. C.—Ma, Q. L.—Yu, X. M.—Yue, Y.—Zhang, Y. W.: Characterization of Submicron Aerosols and Effect on Visibility During a Severe Haze-Fog Episode in Yangtze River Delta, China. Atmospheric Environment, Vol. 120, 2015, pp. 307–316, doi: 10.1016/j.atmosenv.2015.09.011.

[22] Yang, W.—Wang, M.—Zhang, J.—Zou, J.—Hua, M.—Xia, T.—You, X.: Narrowband Wireless Access for Low-Power Massive Internet of Things: A Bandwidth Perspective. IEEE Wireless Communications, Vol. 24, 2017, No. 3, pp. 138–145, doi: 10.1109/MWC.2017.1600298.

[23] Bedrosian, T. A.—Nelson, R. J.: Influence of the Modern Light Environment on Mood. Molecular Psychiatry, Vol. 18, 2013, No. 7, pp. 751–757, doi: 10.1038/mp.2013.70.

[24] Cao, Z.—Zhu, Y.—Sun, Z.—Wang, M.—Zheng, Y.—Xiong, P.—Hou, J.—Tian, L.: Improving Prediction Accuracy in LSTM Network Model for Aircraft Testing Flight Data. 2018 IEEE International Conference on Smart Cloud (SmartCloud), 2018, pp. 7–12, doi: 10.1109/SmartCloud.2018.00010.

[25] Liu, Y.—Pu, H.—Sun, D. W.: Efficient Extraction of Deep Image Features Using Convolutional Neural Network (CNN) for Applications in Detecting and Analysing Complex Food Matrices. Trends in Food Science & Technology, Vol. 113, 2021, pp. 193–204, doi: 10.1016/j.tifs.2021.04.042.

[26] Tong, J.—Xie, L.—Fang, S.—Yang, W.—Zhang, K.: Hourly Solar Irradiance Forecasting Based on Encoder–Decoder Model Using Series Decomposition and Dynamic Error Compensation. Energy Conversion and Management, Vol. 270, 2022, Art. No. 116049, doi: 10.1016/j.enconman.2022.116049.

[27] Jain, A.—Nandakumar, K.—Ross, A.: Score Normalization in Multimodal Biometric Systems. Pattern Recognition, Vol. 38, 2005, No. 12, pp. 2270–2285, doi:

10.1016/j.patcog.2005.01.012.

[28] Hancock, J. T.—Khoshgoftaar, T. M.: Survey on Categorical Data for Neural Networks. Journal of Big Data, Vol. 7, 2020, No. 1, Art. No. 28, doi: 10.1186/s40537-020-00305-w.

[29] Fedorov, A.—Nikolskaia, K.—Ivanov, S.—Shepelev, V.—Minbaleev, A.: Traffic Flow Estimation with Data from a Video Surveillance Camera. Journal of Big Data, Vol. 6, 2019, Art. No. 79, doi: 10.1186/s40537-019-0234-z.

[30] Chen, C.—Liu, B.—Wan, S.—Qiao, P.—Pei, Q.: An Edge Traffic Flow Detection Scheme Based on Deep Learning in an Intelligent Transportation System. IEEE Transactions on Intelligent Transportation Systems, Vol. 22, 2020, No. 3, pp. 1840–1852, doi: 10.1109/TITS.2020.3025687.

[31] Li, C.—Xu, P.: Application on Traffic Flow Prediction of Machine Learning in Intelligent Transportation. Neural Computing and Applications, Vol. 33, 2021, No. 2, pp. 613–624, doi: 10.1007/s00521-020-05002-6.

[32] Sitaula, C.—Hossain, M. B.: Attention-Based VGG-16 Model for COVID-19 Chest X-Ray Image Classification. Applied Intelligence, Vol. 51, 2021, No. 5, pp. 2850–2863, doi: 10.1007/s10489-020-02055-x.

[33] Shen, S. L.—Zhang, N.—Zhou, A.—Yin, Z. Y.: Enhancement of Neural Networks with an Alternative Activation Function TanhLU. Expert Systems with Applications, Vol. 199, 2022, Art. No. 117181, doi: 10.1016/j.eswa.2022.117181.

[34] Yao, J.—Xing, W.—Wang, D.—Xing, J.—Wang, L.: Active Dropblock: Method to Enhance Deep Model Accuracy and Robustness. Neurocomputing, Vol. 454, 2021, pp. 189–200, doi: 10.1016/j.neucom.2021.04.101.

[35] Sikha, O. K.—Bharath, B.: VGG16-Random Fourier Hybrid Model for Masked Face Recognition. Soft Computing, Vol. 26, 2022, No. 22, pp. 12795–12810, doi: 10.1007/s00500-022-07289-0.

[36] Arnab, A.—Zheng, S.—Jayasumana, S.—Romera-Paredes, B.—Larsson, M.—Kirillov, A.—Savchynskyy, B.—Rother, C.—Kahl, F.—Torr, P. H. S.: Conditional Random Fields Meet Deep Neural Networks for Semantic Segmentation: Combining Probabilistic Graphical Models with Deep Learning for Structured Prediction. IEEE Signal Processing Magazine, Vol. 35, 2018, No. 1, pp. 37–52, doi: 10.1109/MSP.2017.2762355.

[37] Li, X.—Yu, L.—Chang, D.—Ma, Z.—Cao, J.: Dual Cross-Entropy Loss for Small-Sample Fine-Grained Vehicle Classification. IEEE Transactions on Vehicular Technology, Vol. 68, 2019, No. 5, pp. 4204–4212, doi: 10.1109/TVT.2019.2895651.

[38] Peng, Y. L.—Lee, W. P.: Data Selection to Avoid Overfitting for Foreign Exchange Intraday Trading with Machine Learning. Applied Soft Computing, Vol. 108, 2021, Art. No. 107461, doi: 10.1016/j.asoc.2021.107461.

[39] Reyad, M.—Sarhan, A. M.—Arafa, M.: A Modified Adam Algorithm for Deep Neural Network Optimization. Neural Computing and Applications, Vol. 35, 2023, No. 23, pp. 17095–17112, doi: 10.1007/s00521-023-08568-z.

[40] Pratik, K.—Rao, B. D.—Welling, M.: RE-MIMO: Recurrent and Permutation Equivariant Neural MIMO Detection. IEEE Transactions on Signal Processing, Vol. 69, 2021, pp. 459–473, doi: 10.1109/TSP.2020.3045199.

[41] Yunpeng, L.—Di, H.—Junpeng, B.—Yong, Q.: Multi-Step Ahead Time Series Forecasting for Different Data Patterns Based on LSTM Recurrent Neural Network. 2017 14$^{\text{th}}$ Web Information Systems and Applications Conference (WISA), 2017, pp. 305–310, doi: 10.1109/WISA.2017.25.

[42] Kołek, M.: Traffic Road Object Detection Polish 12k. Kaggle Dataset. `https://www.kaggle.com/datasets/mikoajkoek/traffic-road-object-detection-polish-12k`.

[43] Budincsevity, N.: Weather in Szeged 2006–2016. Kaggle Dataset. `https://www.kaggle.com/datasets/budincsevity/szeged-weather`.

[44] Jouhari, M.—Saeed, N.—Alouini, M. S.—Amhoud, E. M.: A Survey on Scalable LoRaWAN for Massive IoT: Recent Advances, Potentials, and Challenges. IEEE Communications Surveys & Tutorials, Vol. 25, 2023, No. 3, pp. 1841–1876, doi: 10.1109/COMST.2023.3274934.

[45] Stock, A.: Spatiotemporal Distribution of Labeled Data Can Bias the Validation and Selection of Supervised Learning Algorithms: A Marine Remote Sensing Example. ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 187, 2022, pp. 46–60, doi: 10.1016/j.isprsjprs.2022.02.023.

[46] Zhao, D.—Liu, Y.—Yin, H.—Wang, Z.: A Novel Multi-Scale CNNs for False Positive Reduction in Pulmonary Nodule Detection. Expert Systems with Applications, Vol. 207, 2022, Art. No. 117652, doi: 10.1016/j.eswa.2022.117652.

[47] Kanthi, M.—Dilli, R.: Smart Streetlight System Using Mobile Applications: Secured Fault Detection and Diagnosis with Optimal Powers. Wireless Networks, Vol. 29, 2023, No. 5, pp. 2015–2028, doi: 10.1007/s11276-023-03278-9.

Zhenpeng Lai is an undergraduate student majoring in electrical engineering at the Fuzhou University, Maynooth International Engineering College. He is interested in artificial intelligence and the Internet of Things (IoT) and is actively exploring innovative applications in these fields.

Kunpeng Zeng is currently studying at the Union College of Fujian Normal University, majoring in intelligent science and technology. He is interested in the Internet of Things (IoT) and engineering.