www.ijarp.com ISSN 2456-9992 Page: 01-19

International Journal Advanced Research Publications

Research Article

Volume: 02

Issue: 06

AN EFFECTIVE DEEP LEARNING APPROACH FOR CROP YIELD FORECASTING IN PRECISION AGRICULTURE BASED ON CLIMATE CHANGES

Munugapati Bhavana*[1] Dr. Koppula Srinivas Rao[2]

[1]Research Scholar, [2]Professor

[1]Department of Computer Science & Engineering MLR Institute of Technology, Affiliated to JNTUH, Hyderabad.

[2]Department of Computer Science & Engineering, MLR Institute of Technology, Affiliated to JNTUH, Hyderabad.

Article Received: 10 May 2026, Article Revised: 30 May 2026, Published on: 20 June 2026

***Corresponding Author: Munugapati Bhavana**

Research Scholar, Department of Computer Science & Engineering MLR Institute of Technology, Affiliated to JNTUH, Hyderabad.

Doi: https://doi-doi.org/101555/ijarp.9722

ABSTRACT

Crop Yield Prediction (CYP) in precision agriculture is a critical application involving various data-driven techniques and advanced technologies to estimate the potential harvest output of crops. The goal is to provide farmers with valuable insights into expected yields, allowing them to make informed decisions about resource allocation, crop management practices, and harvest planning. CYP is a complex task involving challenges and uncertainties like high computational time complexity and higher accuracy. This paper proposes a novel deep-learning technique for crop yield prediction in precision agriculture. The statistical, agricultural, and weather data are initially collected from Crop Production Statistics – India Dataset. Then, perform data cleaning and normalization using the min-max normalization method on collected data. After that, the Res2Net technique was used to extract the features. Then, select the essential features by eliminating unnecessary features using the Modified Red Deer Optimization Algorithm (MRDOA) algorithm to reduce the time complexity. Finally, the Deep Spiking Convolutional Neural Network (DCSNN) technique is employed to predict the crop yield in precision agriculture. We simulate and evaluate the performance on the Python platform and analyze the effectiveness through Root Mean Squared Error (RMSE), Mean Absolute Error (MSE), and Correlation Coefficient (R) metrics to calculate the outcomes of the DL crop yield prediction. Local, regional, and global decision-making can be improved by using the proposed method to anticipate agricultural yields more accurately and efficiently.

KEYWORDS: Precision agriculture, crop yield prediction, deep learning, bi-directional convolutional long-short term memory.

1. INTRODUCTION

Due to rising living standards and a growing global population, food demand will outpace supply after the middle of the twenty-first century. By making maximum use of the limited land resources available, this agricultural planning seeks to enhance agricultural productivity. Machine learning algorithms offer innovative methods to increase agricultural output and cut losses during poor weather [1-3].Crop selection techniques are crucial for maximizing yield under varied conditions, which is advantageous for countries' economies. Since high-quality seeds significantly influence crop output, a thorough assessment of seed quality is required to ensure high yields. Hybridization methods facilitate crop selection by enabling higher crop rates under ideal circumstances.

Numerous factors, such as the kind of soil, the crop chosen, the location, the weather, and the harvesting methods, impact crop productivity [4, 5].Categorization and machine learning methods are employed to increase agricultural output to forecast the most effective outcomes. Geographical elements like hills, riverbeds, and depths, as well as meteorological elements like rainfall, humidity, cloud cover, and temperature, significantly impact crop productivity. Crop yields and crop yield predictions have a significant impact on the global and national economies each year and on the production of food. Climate data and irrigation levels significantly impact crop production [6-8]. It is crucial to optimize irrigation and use more effective irrigation systems because additional watering only sometimes increases production. One method of streamlining the procedure is forecasting yield using various watering methods.

Machine learning has a subfield known as deep learning (DL). DL algorithms are being heavily researched to create decision-making systems for the entire cycle of agricultural production and harvesting [9, 10].The concept is to feed massive artificial neural networks with growing amounts of data, automatically extract attributes from them, and then make decisions based on these features. In this context, “deep” refers to the amount of neural network's hidden layers. As the network gets more profound, the model's performance gets better.

For the country to have secure food supplies, accurate and timely crop production forecasting from several environmental data sources is essential [11, 12].In general, mathematical models and process-based crop models are popular approaches used for yield forecasting. Crop growth and yield production processes can be simulated using process-based crop models (such as APSIM, CERES, and STICS), which makes it possible to investigate the relationships between crop yield and environmental factors [13-15].However, running crop models on a wide scale takes a lot of time. It typically needs much data from field observations (such as cultivar traits, management choices, and soil variables) to calibrate appropriately. Additionally, most crop models drastically simplify the processes connected to extreme climate events (ECE), which leads to less precise yield simulations.

Most crop models have demonstrated significantly limited applicability at large scales compared to statistical models. Weather conditions play a significant role in crop yield, but climate can be highly variable and complex to predict accurately, especially in the long term. Extreme weather events like droughts, floods, and storms can significantly impact yield, making prediction more challenging. We proposed a novel DL approach to crop yield forecasting to overcome the issues. The key contribution of this research is,

- Detecting missing values, remove noise, data transformation, and outlier detection using the min-max normalization method on collected data. After that, the Res2Net technique was used to extract the features.
- Then, select the essential features by eliminating unnecessary features using the Modified Red Deer Optimization Algorithm (MRDOA) algorithm to reduce the time complexity.
- Finally, the deep spiking convolutional neural network (DCSNN) technique is employed to predict the crop yield in precision agriculture.

The other parts of the paper are arranged as follows. Section 2 discusses the related study on crop yield prediction. Section 3 discusses in extensive detail the proposed technique and its parts. Section 4 describes the experimental approach. In Section 5, the work is discussed, along with ideas for further research.

2. Literature Survey

In this section, the paper reviewed studies on crop yield prediction. A deep learning architecture called SSTNN (Spatial-Spectral-Temporal Neural Network), which mixes 3D recurrent and convolutional neural networks to use their complementary properties, was suggested by Qiao et al. [16] for predicting agricultural yield. The SSTNN specifically combines a spatial-spectral learning component and a temporal dependency capture module into a single convolutional network to detect the combined spatial-spectral-temporal representation. The unique spatial-spectral feature learning module initially extracts suitable spatial-spectral characteristics from the multi-spectral pictures. The spatial-spectral feature learning component is then concatenated with the temporal dependency capturing component to extract the temporal relationship from the lengthy time-series pictures. Finally, projections of Chinese winter wheat and corn yields validate the suggested SSTNN.

Ma et al. [17] created a BNN-based county-level corn yield forecasting system that can offer a timely estimate of the yield and the associated predictive uncertainty. Due to the availability of enough yield statistics for model building and validation, 12 states in the American Corn Belt were chosen for this research. Informational variables, such as time-series VIs, sequential climatic measurements, and soil qualities, were initially retrieved from various data sources and consolidated at the county level. Then, using the retrieved characteristics and observed yield as a basis, a BNN yield forecasting model was created. It was compared to five other popular machine-learning methods. Finally, the spatial patterns of the prediction uncertainty and any relevant contributing factors were examined.

Jeong et al. [18] presented an approach that combines a crop model and a DL algorithm for early forecasting of rice yield at the pixel size for various agricultural systems in South and North Korea. Initially, pixel-scale reference rice yields were obtained using satellite-integrated crop models. The deep learning algorithm was then utilized to leverage the benefits of crop models by using the pixel-scale reference rice yields as target labels. By forecasting, the models of five alternative Dl network frameworks were utilized to help decide the hybrid structure of the one-dimensional convolutional neural network (1D-CNN) and long-short-term memory (LSTM) layers.

The Crop Yield Prediction Algorithm (CYPA), a method employing IoT methods in PA, was presented by Talaat [19]. The understanding of the cumulative effects of field parameters, such as nutrient and water deficiencies, illnesses, and pests during the growth time, is made more accessible by crop yield simulations. Policymakers and farmers can more easily predict annual crop yields using the suggested CYPA, which integrates climate, agricultural yield, weather, and chemical data.

For effective crop yield prediction, a hybrid MLR-ANN approach has been put forth by Gopal & Bhargavi[20] in this research work. The Statistical, Meteorological, and Agricultural Departments provided 30 years of paddy crop-related data. The data set involves preprocessing for feature recognition, interpretation, treatment of missing values, and handling outliers. To make a successful forecast, it is essential to choose high-level features which enhance prediction accuracy. The most utilized statistical approach to yield forecasting is the MLR. The most utilized ML technique for predicting crop yields is the ANN.

From the literature survey, they have some research problems to solve. It obtained more computational time to complete the process, and the accuracy level is higher. Then, feature selection is needed to reduce the classification computation time and predict better accuracy. So, we propose a novel DL technique to solve these problems. A process on dataset data is done in a preprocessing step to estimate the missing values, remove noise, data transformation, and outlier detection. Then, select the essential features from extracted features to perform classification accurately with less computation time.

3. PROPOSED METHODOLOGY

Crop yield prediction uses data and models to estimate the quantity of crops that will be harvested in a specific agricultural area, typically for a given growing season. Accurate crop yield predictions are essential for farmers, agricultural policymakers, and the food supply chain to make informed decisions regarding planting, harvesting, resource allocation, and food security. However, predicting the yield has some limitations. To overcome all the issues in crop yield prediction, we proposed a new DL technique to predict crop yield accurately. Firstly weather, statistical, and agricultural data are gathered. Next, apply the min-max normalization technique to clean and normalize the gathered information. The Res2Net method was then used to extract the features. Next, to reduce the temporal complexity, use the Modified Red Deer Optimization Algorithm (MRDOA) algorithm to eliminate extraneous characteristics and choose the key features. Precision agriculture uses the deep spiking convolutional neural network (DSCNN) approach to forecast crop productivity.

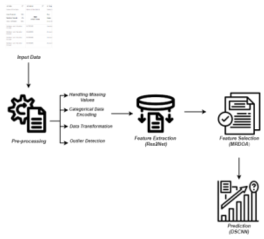


Figure 1. The framework of the proposed methodology.

Figure 1 shows the framework of the proposed deep spiking convolutional neural network DSCNN method.

3.1 Preprocessing

Preprocessing included with the estimation of values that are missing, the elimination of noise such as outliers, normalization and verification of uneven data, and more.

3.1.1 Handling Missing Values

The easiest way to deal with missing values is to ignore records, although this is only useful for some data sets. The data set is reviewed during the information preparation process to determine if any characteristic values are missing. The statistical method of mean imputation was used to approximate the missing information for numerical attributes. The nominal feature values that were absent were substituted using the mode approach.

3.1.2 Categorical Data Encoding

Category values must be converted into numerical information since most DL methods only accept numerical information as input. The binary numbers "0" and "1" are used to symbolize the properties of categories like "yes" and "no."

3.1.3 Data Transformation

Data transformation involves putting numbers on the same scale to prevent one variable from dominating the others. If not, learning algorithms, independent of weight units, interpret greater values as more significant and fewer values as lower.Data transformations modify a data set's values to enable additional processing. This study uses a data normalization strategy to increase the accuracy of the DL technique. Data is transformed from the −1 to the +1 range. The transformed data has a mean of 0 and a standard deviation 1.

3.1.4 Outlier Detection

An isolated observation point from the rest of the data is called an outlier. Measuring variability may be the source of an outlier or indicate an experiment error. The DL algorithm's learning process can be distorted and misled by an outlier.The end consequences are more extended training periods, decreased model accuracy, and worse outcomes. Before transmitting data to the learning algorithm, this research applies a technique based on the Interquartile Range (IQR) to remove outliers.

3.2 Feature Extraction

After preprocessing is done, the features in the Dataset should be extracted. To extract the features, use Res2Net from a set of multi-view $V = \{V_1, \quad V2, V_3, \ldots, V_n\}$ of 3D models. This method increases the amount of receptive fields, increases the power of feature extraction, and minimizes information loss. The Res2Net module swaps out the underlying block in the ResNet structure [21], as seen in Figure 2. First, the 3×3 convolution layer is uniformly split into p subsets, denoted by the expression $x = \{x_1, x_2, x_3, \ldots, x_p\}$.Then, each subgroup (apart fromx_1) is sent into a 3×3 convolution called$Conv_p$. The outcome of $Conv_p$ is inserted before the input of $Conv_{p-1}$starting at $x_3$and increases the number of possible perception regions in a single layer. The Res2Net module's processing formula can be expressed as follows:

$$y_p = \begin{cases} x_p \\ Conv_p(x_p) \\ Conv_p(x_p + y_{p-1}), \end{cases} \Bigg| \begin{array}{l} p = 1; \\ p = 2; \\ p \geq 3, p \text{ is an integer} \end{array} \qquad (1)$$

In this case, the outcome of the Res2Net component is $y = \{y_1, y_2, y_3, \ldots, y_p\}$. The channel size for the Res2Net residual component is then confirmed by connecting it and sending it to a 1×1convolution layer.

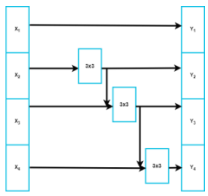


Figure 2. Res2Net block structure.

The Res2Net block attempts to increase the available appropriate fields, enhancing the framework's multi-scale representation capability. It extracts multi-scale channel dimension information within every residual block using hierarchical residual-like connectivity. This method performs better in speaker confirmation, particularly for brief utterances. Feature maps were divided into s feature map subsets, identified by the letters x_i, where $i \in \{1, 2, \ldots, s\}$. The spatial size of every feature subset x_i is the same. However, there are only $1/s$channels.$K_i()$(a 3×3 convolution filter) is applied to all x_i except for x_1. You may express the outcome of y as follows:

$$y_i = \begin{cases} x_i, & i = 1 \\ K_i(x_i), & i > 2 \\ K_i(x_i + y_{i-1}), & i > 2 \end{cases} \qquad (2)$$

A single Res2Net block, which serves as the approach's fundamental building component, is shown in Figure 3. Let the input for every Res2Net block be a cube with the dimensions $I \in \mathbb{R}^{W \times H \times C}$, where W, H, and C stand for width, height, and depth, respectively. This cube goes across a block of 1×1convolution. The procedure can be expressed as

$$y = Conv_{1\times1}(I) \quad (3)$$

where the outcome, $y \in \mathbb{R}^{W \times H \times N}$, has N feature maps. The notation ni identifies every of these s equal subsets of these N feature maps with w channels , where$i \in 1,2,\ldots,s$. Scale and depth are the names given to the variables s and w, respectively.The attributes are retrieved using a multi-scale method, and all the subsets except n_1 go through a 3×3 convolutional block. The procedure can be divided into three steps:

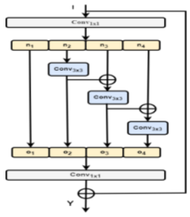


Figure 3. A single Res2Net block (scale s = 4).

The convolutional operation is dropped for $i = 1$to reuse attributes and trim variables. The outcome from o_1 is displayed as

$$o_1 = n_1 \quad (4)$$

The subset n_2 goes through a 3×3 convolutional block when $i = 2$. As shown in the result o_2,

$$o_2 = Conv_{3\times3}(n_2) \quad (5)$$

The outcome o_{i-1} is added to the subset n_i for $s \geq i > 2$and then supplied to the convolutional block. The procedure is expressed as

$$o_i = Conv_{3\times3}(n_i + o_{i-1}) \quad (6)$$

The outcome maps are joined together to create a single block ( ) to use all the data taken from different splits. This block is then sent through a1×1 convolutional block and combined with the input I via a leftover link to produce the outcome Y.

3.3 Feature Selection

The Modified Red Deer Optimization Algorithm (MRDOA) is utilized to select the necessary features by eliminating the unnecessary features to reduce the computational time. The proposed method is compared to its original notion and several well-known algorithms from research, including MA, GA, and PSO. The originality of MRDOA is that it maintains the exploration and exploitation phases on an equal footing, assisting in selecting the best characteristics with the most minor complexity [22]. Furthermore, an adequate number of hinds and male red deer in the population produced an ample convergence space for selecting features. Modern optimization techniques should have taken into account these features.

The primary source of inspiration for this part was the behaviors of red deer during the breeding season, such as mating, shouting, and fighting. It is essential to create a selection strategy when a met heuristic technique is utilized to solve a challenging optimization model.Algorithm calculation times are slashed with an effective solution representation, making the model more accessible for computing and memory needs. Because of this, this research used the two-stage Random-Key selection method. Uniform functions generate random numbers, which can be modified using met heuristic techniques. In the second stage of this method, uniform functions are used to generate random integers. The variables are then calculated using the amounts to create a feasible solution. This approach's key advantage is that it only requires a fixing step and processes more quickly than other approaches to creating mathematical models.

Red deer solutions, including N_{pop}, are populations found in the MRDOA. Hinds (female red deer) and MRDs (male red deer) are two subgroups of the original population of the MRDOA.N_{male} is thought to be the most effective non-dominated choice, whereas $(N_{hind} = N_{pop} - N_{male})$is thought to represent the remaining population.Every MDR then lets out a roar to start the local search phase of the first stage. The most recent modification to how men are positioned is as follows:

$$male_{new} = \begin{cases} male_{old} + a_1 \times \left(\left((U_B - L_B) \times a_2\right) + L_B\right), & if\ a_3 \geq 0.5 \\ male_{old} - a_1 \times \left(\left((U_B - L_B) \times a_2\right) + L_B\right), & if\ a_3 < 0.5 \end{cases} \quad (7)$$

U_B and $L_B$are employed in this situation to reduce the search space. Man old represents the current situation, whereas man new represents the new situation.The range of random values was produced using uniform distribution-based random numbers because the roar operator is a random amount generator. The MDRs are then divided into the commanders and stags groups. There are as many stags as follows:

$$N_{stag} = N_{male} - N_{com} \quad (8)$$

N_{stag}, which refers to the population's total amount of MDRs, stands for the amount of stags in this formula. The combat strategy carries out another crucial step in the algorithm. The combat strategy carries out another crucial step in the algorithm. The exploiting stage of the MRDOA is the fighting phase.The result of this combat depends on the number of commanders and stags involved. This operator creates two new solutions, which are denoted by the equations below:

$$New_1 = \frac{Com + Stag}{2} + b_1 \times \left(\left((U_B - L_B) \times b_2\right) + L_B\right) \quad (9)$$

$$New_2 = \frac{Com + Stag}{2} - b_1 \times \left(\left((U_B - L_B) \times b_2\right) + L_B\right) \quad (10)$$

New_1 And New_2 are two instances of novel solutions in this situation. Com and Stag, respectively, are the names given to the commander and Stag. The uniform distribution-based random values c, b_1, and b_2 are in the range [0,1] because the combat is random.The new commander is the most promising option. In the following phase, the commanders create the harems. Based on each commander's ability to attract hinds to him, the hinds are randomly assigned to different harems. The mating nature, inside and outside harems, is logically organized to maintain the exploring phase. Stags also mate with hidden hinds to repeat the exploitation stage once. This mating procedure is carried out according to the formula below:

$$d_{offspring} = \frac{Commander + Hind}{2} + (U_B - L_B) \times c \quad (11)$$

Here, the offspring distance ($_{offspring}$)between Commander and Hind connects them.

3.4 Prediction

The proposed DCSNN framework uses historical data to predict future yield and production. Both phases comprise some layers, including three fully connected layers, two convolutional layers, two max-pooling levels, two dropout layers, one spike encoding layer, and two max-pooling layers.

Convolution layer

The feature extraction provides a spatial and local characteristic to the convolution layer. This layer operates a filter with a size of k and an input matrix of i ×j. As a result, this layer provides a collection of k feature maps with a dimension of i > j. Due to the inclusion of various disseminators, including Gaussian and Normal, each filter can learn a variety of attributes.

Max-Pooling

The non-linear input down-sampling is completed by the Max-Pooling technique [23]. The characteristics are then split up into several non-overlapping rectangular sections using this technique. Each rectangular area generates an output that is as high in value as possible, which helps to make features more diminutive and more necessary. Eq. (12), therefore, can be used to formulate max-pooling.

$$z_{ijk} = max\{c_{i'j'k} : i \le i' < i + p, j \le j' < j + p\} \quad (12)$$

Here, z stands for the output matrix (i × j size k-feature map from the convolutional layer), while c stands for the input matrix. The letter p indicates padding. The pooling layer used maximum operations to function the full feature map and resize it geographically. The downside of convolutional layers, which is that they record accurate location attributes, is also dealt with by pooling layers.

Dropout

To prevent overfitting during the training phase, it is described as the unit of visible and hidden layers that are randomly observed. The dropout process reduces the intricate neuronal co-adaptation.

Spike Features encoding

This defines the transformation of the retrieved convolution attributes for the spike train. The feature extraction of pushing down is required to convert feature maps into feature vectors. These vectors are considered to be the classifier's input. As a result, attribute vectors in various versions for leaky integration with fire models are obtained by spike encoding. The refractory period is called the "Soft-leaky integrates and fire model." It is divided into two phases: spike reset mechanism and membrane potential behavior. Eq.(13) describe the dynamics of leaky integration with fire models neuron membrane potential $v_{lif}(t)$ for input data samples.

$$\frac{dv_{lif}(t)}{dt} = -\frac{1}{R}v_{lif}(t) + \frac{t_s(t)}{C}, \; t \geq 0, \tag{13}$$

Here, C and R are the capacitance membrane and the time constant.

$$r = \frac{1}{t_{ref} - \tau_{RC}\log\left(1 - \frac{1}{t_s}\right)} \tag{14}$$

The equation above can be written as Eq. (14), assuming r = 0 and ts = 1.

$$r(t_s) = \left[t_{ref} + \tau_{RC}\log\left(1 + \frac{1}{\rho(t_s - 1)}\right)\right]^{-1} \tag{15}$$

TRC represents the membrane constant of a spiking neuron i n the equation above, and the refractory period is represented by t_{ref}. In addition, a softer maximal is used in its stead, for as ρ1(x) = λlog[1 +xλ].This softer maximal can be substituted in the previous Eq. (16), and the feature encoding procedure can be obtained. The procedure for encoding local and geographic feature data is summarized below. Assuming constant input, ts(t) = ts, Eq. (15) is used to express the steady-state firing rate as

$$r(t_s) = \begin{cases} \left[t_{ref} + \tau_{RC}\log\left(1 + \frac{1}{\rho(t_s - 1)}\right)\right]^{-1}, & if\ t_s \geq 1 \\ 0, & otherwise \end{cases} \tag{16}$$

4. RESULTS AND DISCUSSION

The outcomes of the proposed DSCNN approach were shown in this section. The proposed DSCNN models' hyper-parameter settings are shown in Table 1.

Table 1. Hyper-parameter settings.

Hyper-Parameter	Setting
Batch size	15
Epochs	850
Activation Function	Relu
Optimizer	Adam
Activation output layer	Sigmoid
Dropout rate	0.5 to 0.1
Loss	Binary Cross entropy

4.1 Dataset Description

The collection includes thorough statistics on India's crop production [24], separated by state and district. The Dataset spans 1997 to 2023 and includes data for the four principal crop seasons of rabbi, kharif, summer, and fall.The report details the annual yield and production of crops farmed nationwide. The Area Output Statistics (APS) database of the Indian government is the source of the extensive information in this Dataset about agricultural output statistics in India.The Ministry of Agriculture and Farmers Welfare maintains the APS, which offers comprehensive information on yield, crop production, and area under cultivation in various Indian states and districts.

4.2 Evaluation Metrics

The DSCNN approach's predicting efficacy was evaluated using the following metrics: RMSE, MSE, R, and NRMSE.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_{i,\exp} - y_{i,pred}\right)^2 \quad (17)$$

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{\left(y_{i,\exp} - y_{i,pred}\right)^2}{n}} \quad (18)$$

$$R\% = \frac{n\left(\sum_{i=1}^{n} y_{i,\exp} \times y_{i,pred}\right) - \left(\sum_{i=1}^{n} y_{i,\exp}\right)\left(\sum_{i=1}^{n} y_{i,pred}\right)}{\sqrt{\left[n\left(\sum_{i=1}^{n} y_{i,\exp}\right)^2 - \left(\sum_{i=1}^{n} y_{i,\exp}\right)^2\right]\left[n\left(\sum_{i=1}^{n} y_{i,pred}\right)^2 - \left(\sum_{i=1}^{n} y_{i,pred}\right)^2\right]}} \times 100 \quad (19)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_{i,\exp} - y_{i,pred}\right)^2}{\sum_{i=1}^{n}\left(y_{i,\exp} - y_{avg,\exp}\right)^2} \quad (20)$$

$$NRMSE = \frac{\sqrt{\frac{1}{n}\sum_{k=1}^{n}\left(y_{i,\exp} - y_{i,pred}\right)^2}}{y_{i,pred}} \quad (21)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (22)$$

Where $y_{i,\exp}$ and $y_{i,pred}$ are the experimental and prediction values, respectively, and n is the number of samples.

4.3 Implementation of the proposed method

This paper proposes a new technique to predict Crop Yield Production. Here, the paper show results as GUI for whether the person has Crop Yield Production. In Figure 4, display the actual front page for input information. Used Python Tkinter for the generation of this GUI.

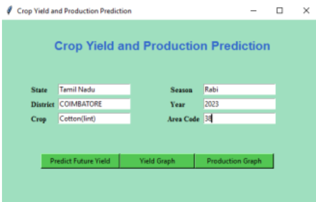


Figure 4. Front page for filling in the information.

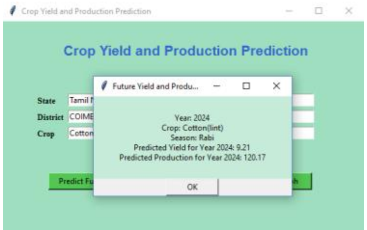


Figure 5. The predicted crop yield and production values.

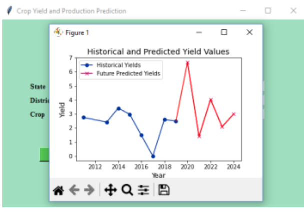


Figure 6. The predicted future yield values.

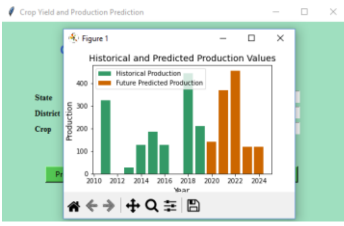


Figure 7. The predicted future production values.

Figures 5-7 depicted prediction outputs using the proposed DSCNN approach.

4.4 Performance Evaluation

In India, the creation and validation of crop production forecast models depend heavily on performance evaluation. Precise forecasts of agricultural productivity are essential for maintaining food security, sustainable farming practices, and well-informed choices for farmers. When predicting agricultural yield from an Indian crop yield dataset, a correlation matrix is a valuable tool. It facilitates comprehension of the connections between various variables and can reveal which ones significantly affect crop productivity. Examine the correlation matrix, paying particular attention to the connections between the variables that affect crop yield forecast. Weather information, past crop yields, and agricultural techniques are a few examples of variables. Determining which associations strongly positive and negative then discuss the ramifications.

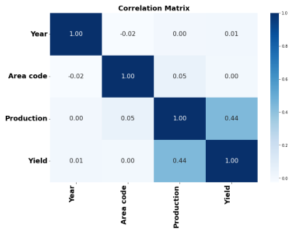


Figure 9. Correlation matrix for Dataset.

Figure 9 demonstrated the correlation matrix for the Dataset.

Compared to existing methods used to predict the Indian crop yield production with the proposed DSCNN method, the proposed method achieved lower error values and higher accuracy. Table 3 compares the proposed DSCNN and some existing methods using the same Dataset. In comparing the existing method with the proposed method, the proposed DCSNN technique achieved a better prediction accuracy of 98.54% and error rates of 0.024 for RMSE and 0.0843 for MAE.

Table 2. Comparison of proposed DSCNN and existing methods for the Dataset.

Technique	RMSE	MAE	Accuracy
J48	0.2773	0.1101	78.145
LAD Tree	0.4127	0.1997	62.251
LWL	0.3209	0.1997	66.225
IBK	0.3057	0.104	80.794
Proposed Method	0.024	0.0843	98.54

Different prediction methods give different results on the same data set. The result of errors and accuracy for all the methods is presented in Table 2.

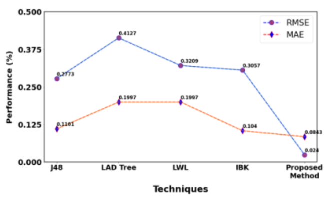


Figure 10. Comparison of proposed and existing error rates.

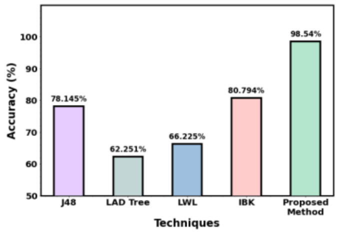


Figure 11. Comparison of proposed and existing accuracy.

.

Figures 10 & 11 about the comparison outcome of proposed DSCNN and existing methods on the same Dataset.

Overall Comparison

An overall comparison result for crop yield prediction is crucial for guiding decision-makers, farmers, and researchers in selecting the most effective methods for predicting crop yields in India. Compared to the proposed DSCNN technique with existing approaches, the proposed DSCNN method achieved a lower RMSE error rate of 0.024 than 0.84 for SSTNN and 0.051 for MLR-ANN. In comparison, not every research used the same metrics. The R2 value for the proposed DSCNN technique is 0.98 compared to 0.68 for SSTNN, 0.75 for BNN, 0.859 for LSTM-1D-CNN, and 0.92 for MLR-ANN. Table 3 demonstrates the overall comparison of proposed DSCNN and existing techniques.

Table 3. Overall comparison of proposed and existing techniques.

Techniques	RMSE	R2	MAP	MAPE	NMSE
SSTNN [16]	0.84	0.68	-	12.23	-
BNN [17]	-	0.75	-	-	-
LSTM+1D-CNN [18]	-	0.859	0.605	-	0.858
MLR-ANN [20]	0.051	0.92	-	0.041	-
Proposed DSCNN Method	0.024	0.98	0.341	0.027	0.512

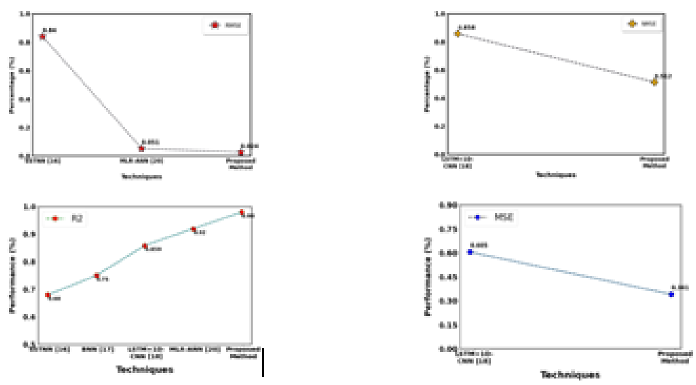
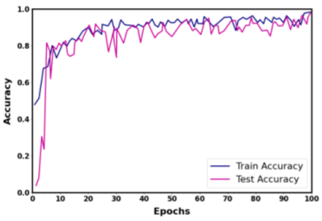


Figure 12. Overall comparison of proposed DSCNN and existing methods in crop yield prediction.

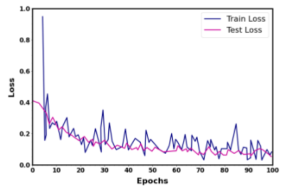
Figure 12 shows the overall comparison of proposed and existing methods for crop yield prediction.

4.5 Evaluation of training and testing

Figure 13 displays a prediction accuracy and loss value graph as the number of iteration steps grew. The graph shows that convergence is positively impacted by the strategy covered in this research. There were training and testing steps for the Dataset. The proposed DSCNN approaches are trained for 100 iterations using the processed training set during the training phase. Right now, the learning rate is 0.1.



(a)



(b)

Figure 13. (a) Accuracy curve for training and testing, (b) Loss curve for training and testing loss for the Dataset.

Figure 13 shows the training, testing accuracy, and training loss functions.

4.6 Computation Time

The prediction method's computing time and error are used as performance metrics. The accuracy of the forecast will be taken into account. Every prediction method accounts for the computation time. The proposed DSCNN technique has a short calculation time and good accuracy. The computation time of the proposed DSCNN system is displayed in Table 4.

Table 4. Utilization of the proposed DSCNN and existing methods.

Methods	Computation Time
SSTNN [16]	0.23
BNN [17]	0.19
LSTM+1D-CNN [18]	0.21
MLR-ANN [20]	0.25
Proposed DSCNN system	0.16

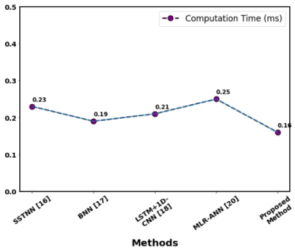


Figure 14. The computation time for the proposed & existing.

.

The computation time using the proposed DSCNN framework and the most recent methods is shown in Figure 14. Compared to earlier methods, our proposed DSCNN approach produced predictions with greater accuracy and required less calculation time.

5. CONCLUSION AND FUTURE SCOPE

Crop yield prediction is crucial for maximizing agricultural productivity, minimizing waste, and ensuring food security. This paper used the initial source to compile weather, statistical, and agricultural data from the India crop yield production dataset. Next, tidy up and standardize the collected data using the min-max normalization technique. The features were then extracted using the Res2Net technique. Next, employ the Modified Red Deer Optimization Algorithm to choose the essential aspects and remove superfluous characteristics to decrease the temporal complexity. Finally, the deep spiking convolutional neural network technique is used in precision agriculture to predict crop productivity. The experimental investigations demonstrate that the proposed DSCNN approach achieved higher performances when compared with existing "state-of-the-art" models. Our proposed DSCNN approach obtained 0.024 for RMSE, the least among current approaches. In the future, we will optimize the hyper-parameters in the proposed DSCNN approach utilizing a novel approach to improve crop yield prediction's performance even more and reduce computational complexity.

REFERENCES

1. Khaki, S., Wang, L., & Archontoulis, S. V. (2020). A cnn-rnn framework for crop yield prediction. Frontiers in Plant Science, 10, 1750.
2. Elavarasan, D., & Vincent, P. D. (2020). Crop yield prediction using deep reinforcement learning model for sustainable agrarian applications. IEEE access, 8, 86886-86901.

3. Shahhosseini, M., Hu, G., Huber, I., & Archontoulis, S. V. (2021). Coupling machine learning and crop modeling improves crop yield prediction in the US Corn Belt. Scientific reports, 11(1), 1606.
4. Nishant, P. S., Venkat, P. S., Avinash, B. L., & Jabber, B. (2020, June). Crop yield prediction based on Indian agriculture using machine learning. In 2020 International Conference for Emerging Technology (INCET) (pp. 1-4). IEEE.
5. Abbas, F., Afzaal, H., Farooque, A. A., & Tang, S. (2020). Crop yield prediction through proximal sensing and machine learning algorithms. Agronomy, 10(7), 1046.
6. Hara, P., Piekutowska, M., & Niedbała, G. (2021). Selection of independent variables for crop yield prediction using artificial neural network models with remote sensing data. Land, 10(6), 609.
7. Oikonomidis, A., Catal, C., & Kassahun, A. (2022). Hybrid deep learning-based models for crop yield prediction. Applied artificial intelligence, 36(1), 2031822.
8. Fan, J., Bai, J., Li, Z., Ortiz-Bobea, A., & Gomes, C. P. (2022, June). A GNN-RNN approach for harnessing geospatial and temporal information: application to crop yield prediction. In Proceedings of the AAAI conference on artificial intelligence (Vol. 36, No. 11, pp. 11873-11881).
9. Agarwal, S., & Tarar, S. (2021). A hybrid approach for crop yield prediction using machine learning and deep learning algorithms. In Journal of Physics: Conference Series (Vol. 1714, No. 1, p. 012012). IOP Publishing.
10. Bali, N., & Singla, A. (2021). Deep learning based wheat crop yield prediction model in Punjab region of North India. Applied Artificial Intelligence, 35(15), 1304-1328.
11. Srivastava, A. K., Safaei, N., Khaki, S., Lopez, G., Zeng, W., Ewert, F., ... & Rahimi, J. (2022). Winter wheat yield prediction using convolutional neural networks from environmental and phenological data. Scientific reports, 12(1), 3215.
12. Shetty, S. A., Padmashree, T., Sagar, B. M., & Cauvery, N. K. (2021). Performance analysis on machine learning algorithms with deep learning model for crop yield prediction. In Data Intelligence and Cognitive Informatics: Proceedings of ICDICI 2020 (pp. 739-750). Springer Singapore.
13. Gong, L., Yu, M., Jiang, S., Cutsuridis, V., & Pearson, S. (2021). Deep learning based prediction on greenhouse crop yield combined TCN and RNN. Sensors, 21(13), 4537.
14. Vignesh, K., Askarunisa, A., & Abirami, A. M. (2023). Optimized Deep Learning Methods for Crop Yield Prediction. Comput. Syst. Sci. Eng., 44(2), 1051-1067.

15. Ji, Z., Pan, Y., Zhu, X., Wang, J., & Li, Q. (2021). Prediction of crop yield using phenological information extracted from remote sensing vegetation index. Sensors, 21(4), 1406.
16. Qiao, M., He, X., Cheng, X., Li, P., Luo, H., Zhang, L., & Tian, Z. (2021). Crop yield prediction from multi-spectral, multi-temporal remotely sensed imagery using recurrent 3D convolutional neural networks. International Journal of Applied Earth Observation and Geoinformation, 102, 102436.
17. Ma, Y., Zhang, Z., Kang, Y., & Özdoğan, M. (2021). Corn yield prediction and uncertainty analysis based on remotely sensed variables using a Bayesian neural network approach. Remote Sensing of Environment, 259, 112408.
18. Jeong, S., Ko, J., & Yeom, J. M. (2022). Predicting rice yield at pixel scale through synthetic use of crop and deep learning models with satellite data in South and North Korea. Science of The Total Environment, 802, 149726.
19. Talaat, F. M. (2023). Crop yield prediction algorithm (CYPA) in precision agriculture based on IoT techniques and climate changes. Neural Computing and Applications, 1-12.
20. Gopal, P. M., & Bhargavi, R. (2019). A novel approach for efficient crop yield prediction. Computers and Electronics in Agriculture, 165, 104968.
21. Zhou, W., Chen, Y., Liu, C., & Yu, L. (2020). GFNet: Gate fusion network with Res2Net for detecting salient objects in RGB-D images. IEEE Signal Processing Letters, 27, 800-804.
22. Asghari, A., Vahdani, A., Azgomi, H., & Forestiero, A. (2023). Dynamic edge server placement in mobile edge computing using modified red deer optimization algorithm and Markov game theory. Journal of Ambient Intelligence and Humanized Computing, 14(9), 12297-12315.
23. Chen, Y., Mai, Y., Feng, R., & Xiao, J. (2022). An adaptive threshold mechanism for accurate and efficient deep spiking convolutional neural networks. Neuro computing, 469, 189-197.
24. Dataset source: https://www.kaggle.com/datasets/nikhilmahajan29/crop-production-statistics-india/code.