Received: 8 October 2023 | *Revised:* 21 March 2024 | *Accepted:* 13 April 2024
DOI: 10.37917/ijeee.22.1.4 *Vol. 22 | Issue 1 | June 2026*

Iraqi Journal for Electrical and Electronic Engineering
Original Article

Open Access
IJEEE University of Basrah College of Engineering

Transforming Wind Nowcasting: Innovative Strategies for Next-Frame Prediction Using Conv-LSTM-3D Model

Abhay B. Upadhyay*, Saurin R. Shah, Rajesh A. Thakkar
Silver Oak University, Ahmedabad, Gujarat, India

Correspondance
*Abhay B. Upadhyay
Silver Oak University, Ahmedabad Gujarat, India
Email: abhayupadhyay.rs@silveroakuni.ac.in

Abstract
This research paper presents an innovative approach to wind nowcasting, addressing specific performance parameters through advanced machine learning techniques. The research aims to overcome inherent challenges in capturing intricate spatiotemporal relationships within wind data. Our novel methodology integrates Conv-LSTM-3D models, emphasizing the prediction of next-frame wind patterns. The Conv-LSTM-3D architecture, combining 3D convolutions and LSTM networks, is specifically tailored to effectively learn temporal dependencies and spatial features in wind data. The introduction outlines the pressing issues associated with traditional wind nowcasting methods, emphasizing the need for improved accuracy and prediction reliability. The primary objectives of this study are to explore the potential of Conv-LSTM-3D models in enhancing wind nowcasting and to assess their performance against traditional methods. Through comprehensive experiments, our approach demonstrates significant improvements in critical performance metrics, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index (SSIM). Specifically, improvements of 0.01, 19.23, 0.11, and 0.64 are observed, highlighting the enhanced accuracy and prediction reliability in the context of next-frame wind nowcasting. Notably, the system achieves these advancements within a reduced time frame, taking only 1149 seconds. This research contributes significantly to the advancement of meteorological prediction techniques, offering a refined short-term wind forecasting tool with potential applications across various fields. The improved clarity and organization of our methodology and findings pave the way for more effective utilization of Conv-LSTM-3D models in enhancing wind nowcasting capabilities.

Keywords
Convolution-Long Short-Term Memory, Machine Learning, Nowcasting, Next-Frame Prediction, Spatiotemporal Relationships.

I. Introduction

In the field of meteorological forecasting, making accurate and timely predictions about wind patterns holds paramount importance for a range of applications, from managing renewable energy resources to ensuring safety in aviation [1]. The advent of machine learning techniques has provided new opportunities to enhance the accuracy and reliability of short-term weather predictions. This research strives to introduce an innovative approach to wind nowcasting by leveraging advanced deep learning architectures. Wind nowcasting, which involves predicting wind conditions over a short time horizon, presents a formidable challenge due to the complex spatiotemporal dynamics governing atmospheric processes [2–4]. Traditional methods often struggle to capture the intricate interactions and dependencies within wind data, resulting in limitations in predictive accuracy [5–7]. To overcome these challenges, this study introduces a novel methodology that harnesses the synergy of Convolutional Long Short-Term Memory-3D (Conv-LSTM-3D) models [8]. The proposed approach capitalizes on the strengths of both 3D convolutions and LSTM networks. Using 3D convolutions facilitates the extraction of spatial features from wind data, accommodat-

ing the multidimensional nature of atmospheric phenomena. Additionally, the incorporation of LSTM networks enables the model to learn temporal dependencies, allowing it to understand evolving patterns and transitions in wind conditions over consecutive frames. This paper is organized as follows: Section 2 provides an in-depth review of nowcasting methods, emphasizing the challenges and constraints of current approaches. Section 3 delves into the theoretical foundation of Conv-LSTM-3D models, elucidating their architecture and mechanisms for capturing spatiotemporal relationships. Section 4 outlines the proposed flow for evaluating the model. Section 5 presents the results of our experiments, showcasing significant improvements in prediction accuracy and reliability. Finally, Section 6 offers a comprehensive discussion of the findings, implications, and potential applications of the novel methodology. With the introduction of this innovative methodology, our research aims to contribute to the advancement of wind nowcasting techniques, fostering more effective and precise short-term wind predictions for various practical domains.

II. Literature Review

In this literature study Table I, analyze and compare different research papers that delve into nowcasting techniques, with a specific emphasis on both radar-based and deep learning methods. The objective is to provide a comprehensive overview of the current state of the art in this field, identifying common trends and highlighting differences in approaches and outcomes. Our analysis will be guided by specific parameters, including the publication year of each paper, the methodologies employed, the datasets utilized, and the key findings derived. This approach will allow us to gain insights into the evolution of nowcasting techniques, understand the methodologies shaping the research landscape, and discern the key contributions and trends across various studies. Key Observations and Trends: Several deep learning techniques are actively utilized for precipitation nowcasting, including convolutional LSTM, deep LSTM, and multi-scale feature fusion. Many research papers employ radar images as input data for nowcasting, while others explore the fusion of radar images with additional data sources like wind forecasts and satellite data. The geographic scope of these studies varies, with some focusing on specific regions like Brazil, China, and India, while others present nationwide or global approaches. Evaluation metrics and benchmarks used for assessing nowcasting performance may differ between studies. Some papers go beyond deep learning and explore other machine learning techniques, such as optical flow and distributed deep learning. This literature study offers an overview of recent research in the field of nowcasting, showcasing the diverse approaches and datasets used. Researchers in this field can refer to these papers for insights into the latest advancements and methodologies, providing valuable guidance for enhancing the accuracy of precipitation nowcasting.

III. Methodology

A. Dataset [5, 9]

To make array-oriented scientific data easily accessible, we adopted the NetCDF (Network Common Data Form) format, specifically designed for this purpose. The NetCDF format provides portability and self-descriptiveness, with a detailed header outlining the file's structure, including information from data arrays to name/value attributes. Metadata, which explains the properties of a file or variable, is an essential component of NetCDF, establishing it as the standard for climate model output data.

Meteorological processes, frequently shaped by the horizontal movement of air masses, play a crucial role in comprehending phenomena like the convergence and divergence of air flows. The U and V components of wind play a pivotal role in comprehending these processes, aiding meteorologists in short-term weather system predictions. In our study, U and V are classified as variables, while time, latitude, longitude, and data are designated as coordinate variables. The dataset

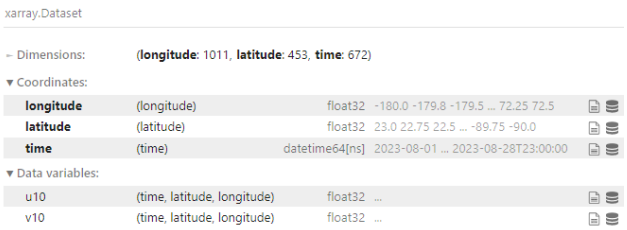

Fig. 1. Dataset dimension.

comprises data Fig 1 extracted from the months of July-23 and Aug-23, with a frequency of 60 data points per day, resulting in a total of 60x24=1440 data points. For training purposes, 720 data points from the month of July-23 were utilized, while the remaining 720 data points from Aug-23 were reserved for testing. This partitioning ensures a comprehensive assessment of our Conv-LSTM-3D model's performance on distinct datasets, contributing to the robustness and generalization of the proposed methodology.

B. Data Preprocessing [10–12]

In the pre-process of preparing NetCDF data for modeling or analysis, we encountered several challenges that required strategic solutions. One significant challenge was the potential for an imbalanced dataset, which can introduce bias into model predictions. To address this, we ensured the initial balancing of the dataset, eliminating the need for additional

TABLE I. SUMMARY OF PAPERS AND METHODOLOGIES

Paper	Publication Year	Methodology	Datasets	Key Findings
R. Reinoso et al.	2022	Localization Filtering	Radar data	Nowcasting with localisation filtering
M. Marrocu et al.	2020	DL	Radar images	DL and Optical Flow
V. Bouget et al.	2021	DL Fusion	Radar images	Employing DL to combine radar images
S.M. Bonnet et al.	2020	Deep Learning (Brazil)	Radar images	Brazilian city of São Paulo nowcasting using DL.
G. Yao et al.	2020	Deep LSTM	Radar images	Weather radar image prediction using Deep LSTM.
A. Kumar et al.	2020	Convolutional LSTM	Satellite data	ConvCast: Using satellite data, Conv-LSTM is used for precipitation nowcasting.
Y. Zhou et al.	2020	DL Benchmark	Not specified	Benchmark review of DL in next-frame prediction.
L. Berthomier et al.	2020	DL (Cloud Cover)	Not specified	Cloud cover nowcasting using DL.
G. Jianhong et al.	2020	Radar Nowcasting Extrapolation	Not specified	Research on weather radar nowcasting extrapolation.
S. Chkeir et al.	2023	ML (Extreme Weather)	Not specified	Nowcasting extreme rain and wind speed with ML.
D. Kong et al.	2023	DL	Not specified	DL-based precipitation nowcasting.
J. Tan et al.	2023	Multi-scale Feature Fusion	Not specified	Precipitation nowcasting using a DL.
S. Yao et al.	2022	Improved DL	Not specified	More effective DL nowcasting model for severe weather.
D.N. Tuyen et al.	2022	RainPredRNN	Not specified	Precipitation forecasting with RainPredRNN.

efforts in this regard. Another critical aspect was the presence of outliers, missing values, and noise within the data, which could compromise the accuracy of our model. Rigorous data cleaning was performed, identifying, and removing outliers, filling in missing values, and filtering out noise to enhance the overall data quality. Furthermore, the raw format of NetCDF data may not be optimal for training and validation, especially when dealing with time-varying phenomena. To overcome this, we transformed the data into time series sequences, enabling the model to capture temporal dependencies and trends crucial for understanding time-varying environmental or scientific phenomena. Lastly, ensuring a well-organized data structure was paramount for meaningful analysis. Through thorough cleaning and structuring of the NetCDF data, we provided researchers and analysts with a clear and navigable dataset, facilitating the extraction of valuable insights. This comprehensive approach has not only addressed the challenges of data preprocessing but has also empowered us to unlock accurate and robust modeling results.

C. Data Splitting [13, 14]

In addressing this time series challenge using NetCDF data, it's crucial to meticulously partition the dataset into two sets: training and testing. Since the data follows a time sequence, preserving the order of observations is paramount during this division. Specifically, for the months of July-23 and Aug-23, where we have data spanning 60 days, each day consisting of 24 hours, we accumulate a total of 60 x 24 = 1440 data points. To uphold this chronological order, we divide the data as follows: 720 data points from July-23 are reserved for training, ensuring the preservation of the time order. Likewise, the remaining 720 data points from Aug-23 are selected for the testing dataset. This approach to data division, where the model learns from historical data and is then tested on future observations, closely mirrors the conventional methodology employed in time series forecasting.

D. Model Selection [15–17]

Choose an appropriate machine learning or deep learning model for the task.

1) Machine Learning Model [18–20]

In the realm of predictive modeling techniques, linear regression stands out as a popular and straightforward model. It utilizes a linear equation to link input values (x) for projecting output values (y), with both x and y being numeric. Coefficients (Beta, denoted as B) are assigned to each input column, and an intercept or bias coefficient further enhances the line's orientation. Another notable approach, the Support Vector Machine (SVM), excels in addressing classification and regression tasks through supervised learning. SVM aims to find the optimal decision boundary for categorizing subsequent data points after an initial point, identifying support vectors that define the extreme points of hyperplanes. Additionally, Decision Trees shine in classifying multinomial attributes by making sequential decisions based on various questions at each level, prioritizing interpretability.

2) Deep Learning Model [21, 22]

This deep learning technique ensures that the model is trained on historical data and evaluated on future observations, closely simulating the actual application of time series forecasting. It is frequently employed for tasks involving sequence prediction, such as predicting the next frame in video sequences. In a Convolutional Long Short-Term Memory (Conv-LSTM) model, the convolutional layers are responsible for capturing spatial features from the input data, while the LSTM cells excel at capturing temporal dependencies. Let's delve into the architecture of Conv-LSTM and offer a simplified fig 2 for better comprehension. Convolutional Layers: First, a series

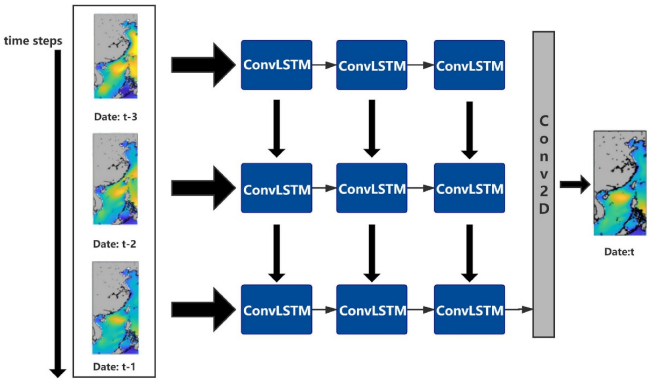

Fig. 2. Conv-LSTM architecture.

of convolutional layers processes the incoming data, which is often a sequence of pictures or frames. Extracting spatial characteristics from each frame in the sequence is the job of these layers. Applying filters to the input data is what convolutional operations are all about when trying to identify patterns and features. LSTM Cells: The LSTM cells get the data after the convolutional layers. It is possible to capture long-range relationships in sequential data using LSTM (Long Short-Term Memory) cells, which are used in recurrent neural networks. They can do this because they keep track of both visible and invisible states, or "states," in their cells. To next-frame prediction, LSTM cells retain memories of past frames and apply them to the prediction of the subsequent frame. Output Layer: To make the final prediction, the output of the LSTM cells is often fed into one or more fully linked layers. The output layer is responsible for generating the projected next frame at each time step.

E. Training/Testing

Implement appropriate optimization strategies when training the model on the training dataset (e.g., Adam, RMSProp). Keep an eye on how the model does on the validation set to catch any signs of overfitting so you can tweak the hyperparameters to fix it. The model's hyperparameters should be fine-tuned depending on how well it performs on the validation set. Methods for doing so include playing around with things like regularization settings, batch sizes, and learning rates. Make predictions for the next time step in the sequence using the trained and verified model. Create a series of potential weather frames using the projected values as inputs for the subsequent prediction process.

F. Evaluation Parameters

1) Quantitative Parameter

R2-Score (Coefficient of Determination): The R2-Score (or coefficient of determination) is a statistical metric for determining how much of the variation in the dependent variable (target) can be attributed to the features (independent variables) in a regression model. It provides light on how well the model's predictions match the observed data. Higher R2-Scores indicate a more precise fit within the given data. For flawless prediction of the target values, the R2-Score should be 1, whereas for no improvement over using the mean of the target values, the score should be 0. The sum of squared residuals (SSR) is denoted by the symbol in Equation (1), whereas the sum of squares (SST) is denoted by the symbol.

$$R^2 = 1 - \frac{\text{SSR}}{\text{SST}} \tag{1}$$

EVS (Explained Variance Score): The percentage of the dependent variable's volatility that is accounted for by the model's predictions is what the Explained Variance Score measures. Like the R2-Score, it assesses how well a model fits the data. Higher EVS values indicate a better match, and the scale runs from 0 to 1. An EVS of 1 indicates that the

model's predictions are indistinguishable from the observed data, while an EVS of 0 indicates that the model fails to account for any observed variation. Var(x) in equation (2) stands for the variance of x, y for the actual values, and for the anticipated values.

$$1 - \frac{\text{Var}(y - \hat{y})}{\text{Var}(y)} \tag{2}$$

MSE (Mean Squared Error): Significance: MSE equation (3) measures the average squared deviation between predicted and actual values in a regression model. In wind nowcasting, it helps quantify the accuracy of predicted wind conditions compared to the observed values. Relevance: MSE is sensitive to outliers, making it valuable in identifying significant errors that might have a pronounced impact on wind predictions. A lower MSE indicates better model performance and a closer match between predicted and observed wind data.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3}$$

MAE (Mean Absolute Error): Mean Absolute Error measures equation (3)(4) how off your predictions are, on average, from reality. When compared to MSE, MAE is less sensitive to outliers and treats all mistakes the same. Lower MAE values indicate higher model performance, much as MSE values do.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4}$$

RMSE (Root Mean Squared Error): Significance: RMSE, like MSE, measures the average squared deviation, but it is expressed in the same units as the target variable. In wind nowcasting, it provides a clear understanding of the magnitude of errors in the predicted wind speeds or directions. Relevance: RMSE equation (5) is particularly useful in regression tasks, offering a straightforward interpretation of the errors in the predicted wind values. A lower RMSE implies a better match between predicted and observed wind conditions.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5}$$

2) Qualitative Parameters

PSNR (Peak Signal-to-Noise Ratio): Significance: PSNR is commonly used to evaluate the overall quality of images and videos. In wind nowcasting, it assesses the quality of predicted wind patterns, considering them as a signal against the noise introduced by inaccuracies. Relevance: A higher PSNR equation (6) value signifies a better signal-to-noise ratio, indicating more accurate and clearer representations of wind patterns. This metric is beneficial for ensuring the fidelity of the predicted wind fields.

$$\text{PSNR} = 10\log_{10}\left(\frac{\text{MAX}^2}{\text{MSE}}\right) \tag{6}$$

SSIM (Structural Similarity Index): Significance: SSIM equation (7) focuses on assessing the structural similarity between original and distorted images. In wind nowcasting, it evaluates how well the predicted wind patterns maintain similarity with the actual wind fields in terms of brightness, contrast, and structure. Relevance: SSIM provides a comprehensive measure of the visual quality of predicted wind fields. A higher SSIM indicates a closer resemblance to the actual wind patterns, enhancing the confidence in the model's predictions.

$$\text{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{7}$$

Metrics like these are essential for determining how well a machine learning model performs and how accurate a prediction, picture, or other data representation is.

IV. Proposed Flow

Fig. 3 demonstrates the concept. Convolutional LSTM designs, as depicted, integrate a convolutional recurrent cell into an LSTM layer, effectively bridging the gap between time series processing and computer vision. This hybrid architecture is particularly relevant for tasks like next-frame prediction, where the goal is to forecast upcoming frames in a video based on information from previously displayed frames. With our model constructed and trained, it can independently make video-based frame predictions. To validate its performance, we select the initial 10 frames from the validation set using a random example and then generate an additional 20 frame predictions for comparison with the actual outcomes. The rationale behind selecting Conv-LSTM-3D lies in its ability to effectively capture both spatial and temporal dependencies in sequential data, making it well-suited for tasks like wind nowcasting. By incorporating convolutional and recurrent elements, this model is adept at recognizing patterns in time series data, such as satellite data, radar data, temperature records, and weather patterns. The training phase for deep learning models like Conv-LSTM-3D involves a substantial amount of data, including satellite data, radar data, and weather patterns. This extended training period is essential for the algorithm to discern intricate patterns in the data effectively. However, once trained, the deep neural network can swiftly generate forecasts based on real-time atmospheric measurements. This rapidity is particularly crucial for anticipating events in the next few hours, as traditional models are

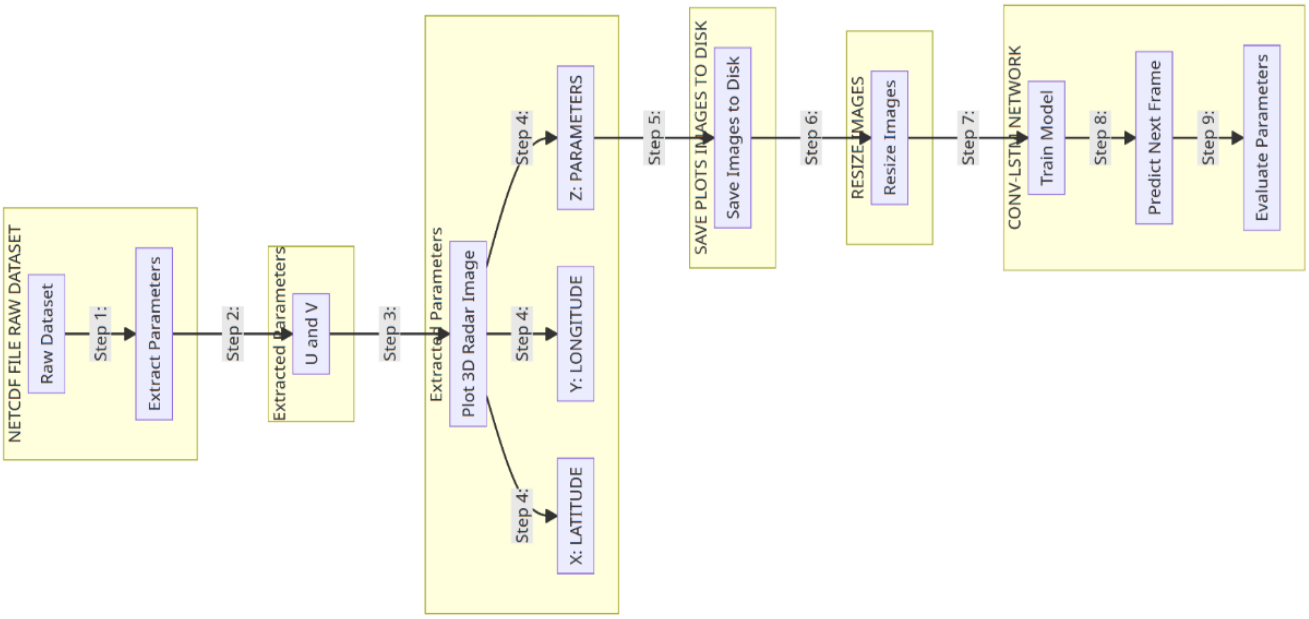

Fig. 3. Proposed flow diagram.

computationally intensive and time-consuming in providing results. Conv-LSTM-3D, with its unique architecture, enables quicker and more efficient real-time forecasting, enhancing its applicability in the context of wind nowcasting.

V. Results and Analysis

```
import matplotlib.pyplot as plt
import xarray as xr
data = xr.open_dataset('/kaggle/working/adaptor.mars.internal-1694591467.0631962-29019-6-8e3688ad-38a6-461f-91c4-f372e9332c38.nc')
# dates range
print("The earliest date in the data is:", data["u10"]["time"].values.min())
print("The latest date in the data is:", data["u10"]["time"].values.max())
```

```
The earliest date in the data is: 2023-07-01T00:00:00.000000000
The latest date in the data is: 2023-08-31T23:00:00.000000000
```

Fig. 4. Reading dataset.

As illustrated in Fig. 4, the data file ".nc" is read using the xarray library, and data for two months, namely July-23 and Aug-23, is extracted for the variables u10 and v10 with respect to time. As depicted in Fig. 5, with 24 hours in a

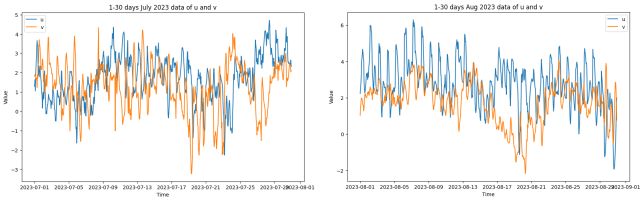

Fig. 5. Separating dataset.

day, the total data amounts to 60 x 24 = 1440 entries. Out of these, 720 data points from the month of July-23 are utilized for training, while the remaining 720 data points from Aug-23 serve as the testing or validation dataset. The results depicted

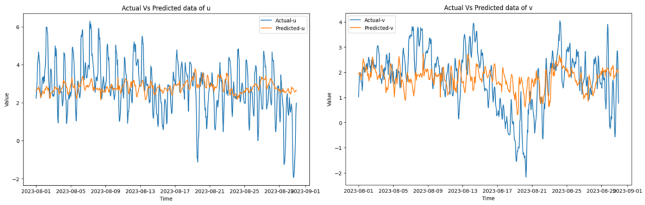

Fig. 6. Linear regression.

in Fig. 6 illustrate the application of a linear regression model trained on u and v data for July-23, validated by predicting Aug-23 data (depicted in orange) and compared with the actual Aug-23 data (depicted in blue). The interpretability of the linear regression model is high, offering insights into the linear relationship between wind components and meteorological patterns. The alignment of the predicted and actual data provides a straightforward understanding of the model's performance in capturing trends. However, it's important to note the limitations of linear regression in handling the complexity of meteorological processes, urging consideration of alternative models for a more nuanced analysis of the data. Exploring non-linear relationships, as portrayed in Fig. 7, involves the application of a Support Vector Machine (SVM) regression model trained on u and v data for July-23. This model is designed to capture more intricate patterns and non-

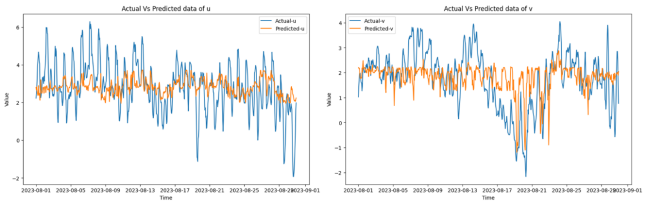

Fig. 7. Support vector machine.

linear dependencies within the meteorological data. During validation, the SVM model predicts the Aug-23 data (depicted in orange), providing insights into its ability to discern complex relationships. The comparison with the actual Aug-23 data (shown in blue) allows for an assessment of how well the SVM model performs in capturing non-linear trends when contrasted with the linear regression model. This comparison will help evaluate whether the SVM model offers improved accuracy in representing the underlying non-linear dynamics of the meteorological phenomena in comparison to the linear model results depicted in Fig. 7. In Fig. 8, a KNN regression

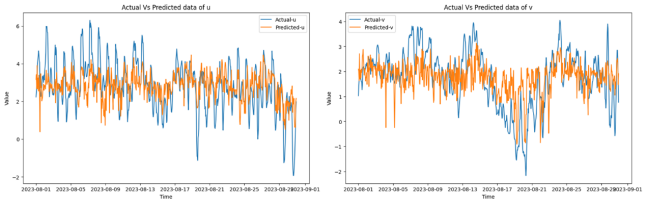

Fig. 8. K-Nearest neighbor.

model is trained with u and v data for July-23, leveraging "neighborhood patterns" for wind nowcasting. During training, the model learns from nearby historical wind data to predict future conditions. In the validation phase, the model predicts Aug-23 data (depicted in orange) and is compared with the actual Aug-23 data (shown in blue). The performance evaluation assesses how well the model captures neighborhood patterns and non-linear dependencies in wind behavior. KNN, known for its ability to capture local relationships, is expected to demonstrate effectiveness in wind nowcasting, especially in scenarios where spatial dependencies are critical, as is common in meteorological phenomena. This evaluation will provide insights into the model's capability compared to linear approaches. As illustrated in Fig. 9, a Polynomial regression model is trained using the u and v data for the month of July-23. It is then validated by predicting (depicted in orange) the data for the month of Aug-23 and compared with the actual (shown in blue) Aug-23 month data. As seen in Fig. 10, we need to consider 20 frames in one block for training in the Conv-LSTM Layer. Therefore, we have extracted 660 frames from the Aug-23 month, reshaped them into 33 blocks of 20 frames each, with dimensions 64x64x3.

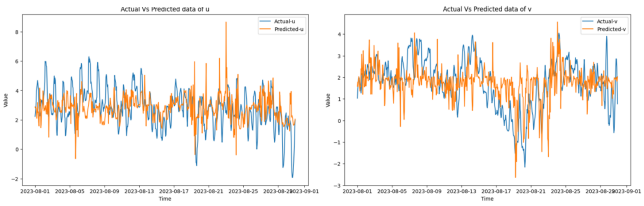

Fig. 9. Polynomial regression.

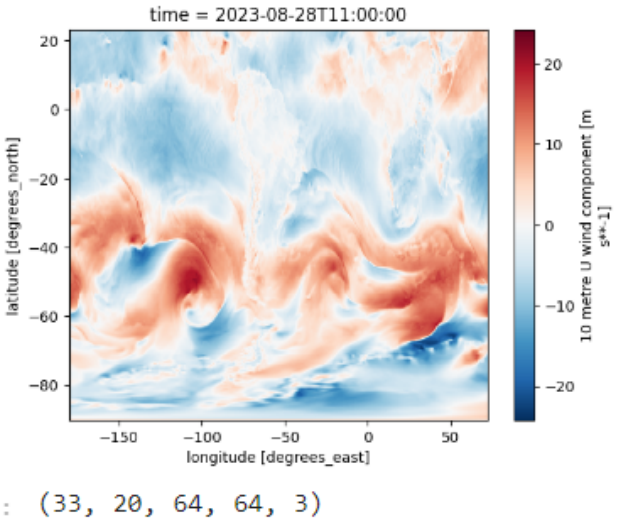

Fig. 10. Convert Data to 20 frame for deep learning.

In Fig. 11, the training dataset is illustrated, consisting of 29 blocks, each with 19 frames of dimensions 64x64x3, and one frame is predicted. Similarly, 4 blocks, each with 19 frames of dimensions 64x64x3, are used for prediction. The introduction of the third dimension in Conv-LSTM-3D, as seen in Fig. 12, adds complexity compared to Conv-LSTM-2D. While Conv-LSTM-2D captures spatial dependencies over time in two dimensions, Conv-LSTM-3D extends this to incorporate a third dimension, offering a more comprehensive understanding of spatiotemporal features. This complexity enhances the model's ability to grasp how patterns evolve not only spatially but also over time, beneficial for tasks involving dynamic changes in consecutive frames, such as time series data. The choice between the two models depends on factors like data nature and computational resources, with Conv-LSTM-3D providing nuanced insights at the cost of increased computational demands, while Conv-LSTM-2D may be preferred for computational efficiency in certain scenarios. The decision should weigh the trade-off between complexity and the specific requirements of the forecasting task. In Fig. 13, the left side displays the Conv-LSTM-2D Accuracy/Loss Plot, and on the right side, the Conv-LSTM-3D Accuracy/Loss Plot is depicted. On the x-axis, the number of epochs is represented,

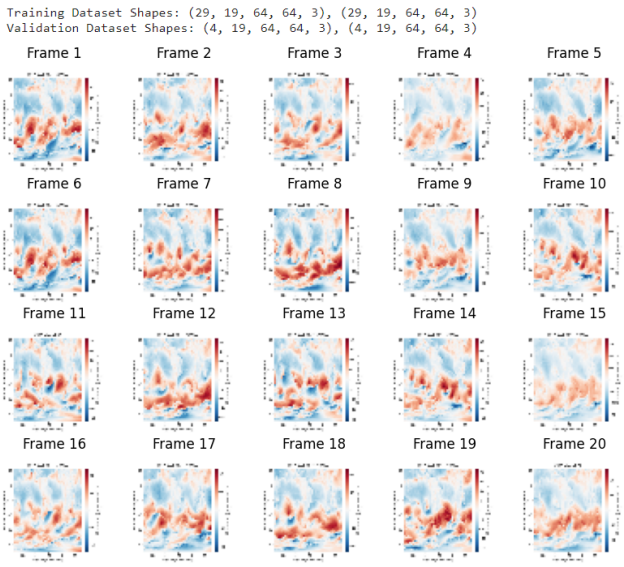

Fig. 11. Split train/test frame.

ConvLSTM2D

Model: "model"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, None, 64, 64, 3)]	0
conv_lstm2d (ConvLSTM2D)	(None, None, 64, 64, 64)	429056
batch_normalization (BatchNormalization)	(None, None, 64, 64, 64)	256
conv_lstm2d_1 (ConvLSTM2D)	(None, None, 64, 64, 64)	295168
batch_normalization_1 (BatchNormalization)	(None, None, 64, 64, 64)	256
conv_lstm2d_2 (ConvLSTM2D)	(None, None, 64, 64, 64)	33024
conv3d (Conv3D)	(None, None, 64, 64, 3)	5187

Total params: 762947 (2.91 MB)
Trainable params: 762691 (2.91 MB)
Non-trainable params: 256 (1.00 KB)

ConvLSTM3D

Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv_lstm2d_3 (ConvLSTM2D)	(None, None, 64, 64, 64)	840704
batch_normalization_3 (BatchNormalization)	(None, None, 64, 64, 64)	256
conv_lstm2d_4 (ConvLSTM2D)	(None, None, 64, 32, 64)	602240
batch_normalization_4 (BatchNormalization)	(None, None, 64, 32, 64)	128
conv_lstm2d_5 (ConvLSTM2D)	(None, None, 64, 32, 64)	401536
batch_normalization_5 (BatchNormalization)	(None, None, 64, 32, 64)	128
conv3d_1 (Conv3D)	(None, None, 64, 64, 3)	2595

Total params: 1847587 (7.05 MB)
Trainable params: 1847331 (7.05 MB)
Non-trainable params: 256 (1.00 KB)

Fig. 12. Two model architecture 2D and 3D Conv-LSTM.

while the y-axis shows accuracy or loss values. It's evident that the accuracy of the 3D model gradually increases, and the loss decreases. Fig. 14 shows the Conv-LSTM-2D and Conv-LSTM-3D models, with the actual frame above and the predicted frame below. The results indicate that the 3D model performs better than the 2D model. In Table II, ML algorithms exhibit very low R2-Score and EVS values, and very high MSE, MAE, and RMSE values, indicating their failure to predict the data using regression models. As presented in Table III, DL algorithms, specifically the proposed Conv-LSTM-3D model, demonstrate much lower MSE and RMSE values compared to existing approaches.

VI. Conclusion and Future Work

In this research, conducted a thorough analysis of innovative strategies for next-frame prediction, with a special focus on

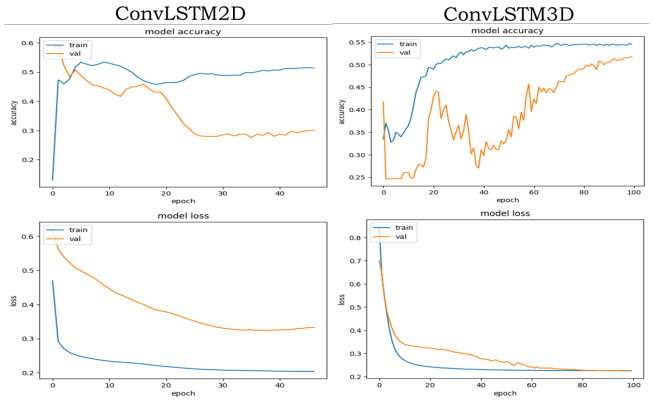

Fig. 13. Analysis of models accuracy and loss.

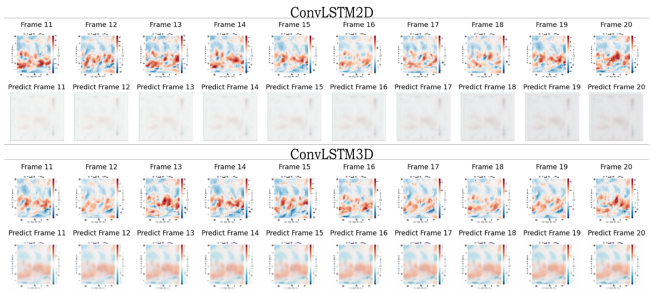

Fig. 14. Actual Vs predicted frame.

integrating Convolution with LSTM-3D models. Our main goal was to assess the performance of different predictive techniques in forecasting future frames. Research initiated our study by exploring various models, including Linear Regression (LR), k-Nearest Neighbors (KNN), Support Vector Regression (SVR), and Polynomial Regression. Through evaluation using various measures, we observed that these models demonstrated improved prediction ability as their complexity increased. Notably, Polynomial Regression stood out by outperforming simpler models, achieving the highest R2 Score and EVS values of 0.34. This indicates its superior ability to explain variance and make more accurate predictions for future frames. Transitioning to image quality prediction, we investigated Conv-LSTM-2D and Conv-LSTM-3D models. Results from our comprehensive experiments illustrated significant improvements in critical performance metrics such as MSE, RMSE, PSNR, and SSIM when compared to traditional methods. These enhancements, particularly with values like 0.01, 19.23, 0.11, and 0.64, were evident in accuracy and prediction reliability, specifically in the context of next-frame wind nowcasting. The system's processing time was measured at 1149 seconds. While these results contribute significantly to advancing predictive modeling techniques and showcase promising applications across various domains, it's essential

TABLE II. MACHINE LEARNING COMPARATIVE ANALYSIS

Mode	R^2 Score	EVS	MSE	MAE	RMSE
LR	0.073	0.073	2.41	1.23	1.55
SVR-Linear	0.21	0.22	2.03	1.09	1.42
SVR-Kernel	0.12	0.11	4.54	4.32	3.23
SVR-RBF	0.15	0.14	3.22	3.13	2.15
KNN (n=2)	0.30	0.31	1.80	1.06	1.34
KNN (n=4)	0.29	0.29	1.77	1.01	1.21
KNN (n=6)	0.28	0.29	1.76	1.02	1.23
Polynomial	0.34	0.34	1.70	1.02	1.30

TABLE III. DEEP LEARNING COMPARATIVE ANALYSIS WITH EXISTING METHODS

Model	MSE	PSNR	RMSE	SSIM
ConvLSTM2D	0.02	16.64	0.15	0.48
ConvLSTM3D	0.01	19.23	0.11	0.64
LSTM E/D [18]	0.129	-	0.14	-
ConvLSTM [19]	0.012	-	0.14	-
MFF [20]	0.45	-	0.16	0.93
Sa-GRU [21]	0.58	-	-	-
RNN [22]	0.43	-	0.17	0.93

to acknowledge certain limitations in our study. Potential biases within the dataset may influence the generalizability of our findings. Future research directions should aim to address these biases, explore additional predictive models, and extend the application of these techniques to other weather parameters such as rain and temperature. This will further enhance the robustness and applicability of our predictive modeling approach.

CONFLICT OF INTEREST

We hereby certify that there is not any actual or potential conflict of interest or unfair advantage at this time, in us providing the Offer Submission or performing the Services required.

REFERENCES

[1] R. Reinoso-Rondinel, M. Rempel, M. Schultze, and S. Tromel, "Nationwide radar-based precipitation nowcasting - a localization filtering approach and its application for germany," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 15, pp. 1670–1691, 2022.

[2] A. Kumar, T. Islam, Y. Sekimoto, C. Mattmann, and B. Wilson, "Convcast: An embedded convolutional lstm based architecture for precipitation nowcasting using satellite data," *PLoS One*, vol. 15, no. 3, pp. 1–18, 2020.

[3] Y. Zhou, H. Dong, and A. E. Saddik, "Deep learning in next-frame prediction: A benchmark review," *IEEE Access*, vol. 8, pp. 69273–69283, 2020.

[4] L. Berthomier, B. Pradel, and L. Perez, "Cloud cover nowcasting with deep learning," in *2020 10th Int. Conf. Image Process. Theory, Tools Appl. IPTA 2020*, 2020.

[5] "Climate data store." https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=form. Retrieved March 10, 2022.

[6] S. S. Roy, S. S. Saha, and D. R. Paul, "A new paradigm for short-range forecasting of severe weather over the indian region," *Meteorol. Atmos. Phys.*, vol. 133, no. 4, pp. 989–1008, 2021.

[7] S. Agrawal, L. Barrington, C. Bromberg, J. Burge, C. Gazen, and J. Hickey, "Machine learning for precipitation nowcasting from radar images," 2019. no. NeurIPS.

[8] S. Goyal, A. S. K. Dey, S. K. Sharma, and S. B. Roy, "Satellite-based technique for nowcasting of thunderstorms over indian region," *J. Earth Syst. Sci.*, vol. 126, no. 6, pp. 1–13, 2017.

[9] S. Yao, H. Chen, E. J. Thompson, and R. Cifelli, "An improved deep learning model for high-impact weather nowcasting," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 15, pp. 7400–7413, 2022.

[10] M. Marrocu and L. Massidda, "Performance comparison between deep learning and optical flow-based techniques for nowcast precipitation from radar images," *Forecasting*, vol. 2, no. 2, pp. 194–210, 2020.

[11] V. Bouget, D. Béréziat, J. Brajard, A. Charantonis, and A. Filoche, "Fusion of rain radar images and wind forecasts in a deep learning model applied to rain nowcasting," *Remote Sens.*, vol. 13, no. 2, pp. 1–21, 2021.

[12] S. M. Bonnet, A. Evsukoff, and C. A. M. Rodriguez, "Precipitation nowcasting with weather radar images and deep learning in são paulo, brazil," *Atmosphere (Basel)*, vol. 11, no. 11, pp. 1–16, 2020.

[13] G. Yao, Z. Liu, X. Guo, C. Wei, X. Li, and Z. Chen, "Prediction of weather radar images via a deep lstm for nowcasting," in *Proc. Int. Jt. Conf. Neural Networks*, 2020.

[14] S. Samsi, C. J. Mattioli, and M. S. Veillette, "Distributed deep learning for precipitation nowcasting," in *2019*

IEEE High Perform. Extrem. Comput. Conf. HPEC 2019, 2019.

[15] G. Jianhong, Q. Hui, and H. Wendong, "Research on weather radar nowcasting extrapolation," in *Proc. - 2020 Int. Conf. Comput. Vision, Image Deep Learn. CVIDL 2020*, no. Cvidl, pp. 84–89, 2020.

[16] S. Hoyer and J. J. Hamman, "xarray: N-d labeled arrays and datasets in python," 2017.

[17] S. S. Roy, M. Mohapatra, A. Tyagi, and S. K. R. Bhowmik, "A review of nowcasting of convective weather over the indian region," *Mausam*, vol. 70, no. 3, pp. 465–484, 2019.

[18] R. Suresh, "Forecasting and nowcasting convective weather phenomena over southern peninsular india - part ii: Severe local storms," *Indian J. Radio Sp. Phys.*, vol. 41, no. 4, pp. 435–447, 2012.

[19] S. Chkeir, A. Anesiadou, A. Mascitelli, and R. Biondi, "Nowcasting extreme rain and extreme wind speed with machine learning techniques applied to different input datasets," *Atmospheric Research*, vol. 282, no. December 2022, p. 106548, 2023.

[20] D. Kong, H. Zhang, and Y. Jiang, "Precipitation nowcasting based on deep learning over guizhou, china," *Atmosphere*, vol. 14, no. 5, 2023.

[21] J. Tan, Q. Huang, and S. Chen, "Deep learning model based on multi-scale feature fusion for precipitation nowcasting," 2023. no. July.

[22] D. N. Tuyen, T. M. Tuan, X. Le, N. T. Tung, and T. K. Chau, "Rainpredrnn: A new approach for precipitation nowcasting," *no.*, pp. 1–14, 2022.