IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Hybrid CNN-LSTM Framework And Infrared Image Processing For Solar Irradiance

[1]Kanchan Patil, [2]Ravikiran Dalawai, [3]Akash Vaggannavar, [4]Ariyan Pujari, [5]Ramesh Taradi

[1]Assistant Professor, Maratha Mandal's Engineering College, Belagavi, Karnataka, India

[2]UG Scholar, Maratha Mandal's Engineering College, Belagavi, Karnataka, India

[3] UG Scholar, Maratha Mandal's Engineering College, Belagavi, Karnataka, India

[4]UG Scholar, Maratha Mandal's Engineering College, Belagavi, Karnataka, India

[5] UG Scholar, Maratha Mandal's Engineering College, Belagavi, Karnataka, India

Abstract:

Getting short-term solar irradiance forecasts right really matters these days—solar power keeps growing, and the grid needs to stay stable and efficient. Trouble is, the usual methods (like satellite data, clear-sky models, or regular sky cameras) just don't keep up when the weather turns fast or the light gets low. This work takes a different route: it combines infrared (IR) sky image processing with a deep learning framework that mixes a Convolutional Neural Network (CNN) and a Bidirectional Long Short-Term Memory (BiLSTM) network. The goal? Fast, accurate solar irradiance prediction—basically, nowcasting plus a short-term look ahead.

Here's how it goes. The system starts by running the raw, low-res, 16-bit greyscale IR images through a three-step process: first, it normalizes them down to 8-bit, then uses bicubic interpolation to sharpen things up, and finally, applies OpenCV's JET colormap to turn the images into high-quality pseudo-RGB. These enhanced images are perfect for transfer learning. An EfficientNet-B0 CNN, fine-tuned on these IR images, handles real-time irradiance regression. Then, the nowcast results feed straight into a BiLSTM module, which digs into the time series and predicts solar irradiance up to one minute ahead, updating every 15 seconds.

Tests on the GIRASOL IR dataset show that this preprocessing pipeline cuts cumulative error by nearly 80% compared to using raw images. The EfficientNet-B0 CNN alone brings down RMSE by 22.4% over a plain CNN. When you stack EfficientNet-B0 with BiLSTM, you get another 13.9% accuracy boost over LSTM-only setups, with RMSE scores of 19.53 W/m² for nowcasting and 28.1 W/m² for 1-minute-ahead predictions. The best part? The whole thing runs light enough for real-time use on GPU servers or even edge devices. It's ready for smart grids, PV plants, and IoT energy management.

Index Terms — Solar irradiance forecasting, infrared sky imaging, EfficientNet-B0, BiLSTM, deep learning, transfer learning, time-series prediction, smart grid.

I. INTRODUCTION

Solar power is taking off, and with it comes a new set of headaches for grid operators. PV systems are everywhere, but their output jumps around with every passing cloud, change in humidity, or shift in the air. Those quick swings in sunlight make it tough to keep the grid balanced, run inverters smoothly, or manage batteries efficiently.

Most of the time, people rely on weather models, clear-sky assumptions, or cameras that see only the visible light. They work fine when the sky behaves, but their predictions start to miss when the weather changes suddenly—at dawn, dusk, or when the clouds roll in thick. They just can't deliver the second-to-minute-level forecasts that real-time grid operations demand.

That's where infrared (IR) sky imaging comes in. IR sensors pick up thermal patterns, and they still work when visible cameras are blind—early morning, late evening, or inside thick clouds. But, honestly, the raw IR images aren't great: they're grainy, low-contrast, and stuck in 16-bit greyscale, which doesn't play well with most deep learning models.

So, here's what this study does differently:

- It builds a three-step IR image enhancement pipeline: normalization, bicubic interpolation, and JET colormap transformation.
- It uses EfficientNet-B0, a transfer learning model, to pull out spatial features and handle the nowcasting.
- It adds a BiLSTM-based module that looks ahead up to 60 seconds, updating every 15 seconds.

The result is a hybrid CNN–LSTM system that delivers accurate, real-time solar irradiance forecasts, perfect for smart grids and modern renewable energy setups.

II. LITERATURE REVIEW

Lately, solar irradiance forecasting has taken a big leap forward. Researchers are tapping into deep learning, infrared (IR) sky imaging, and hybrid spatial–temporal models to sharpen short-term predictions. The aim is simple: do better than the old-school methods, especially when the weather changes fast or light is low.

1. Nijhum et al. (2024) came up with a hybrid CNN–LSTM setup that uses infrared sky images for short-term solar irradiance forecasting. By letting CNNs handle the spatial stuff and LSTM networks tackle the time sequences, their model left standalone methods in the dust. Turns out, merging spatial and temporal learning really works.

2. Yearat et al. (2023) looked into boosting low-res IR sky images with bicubic interpolation. They resampled each pixel using its 16 nearest neighbors, which smoothed things out and sharpened the edges. This let CNNs pick up on smaller cloud shapes and cut down RMSE compared to using raw IR images.

3. Batchuluun et al. (2022) introduced a two-step process: Deblur-SRRGAN first cleans up thermal IR images, then Mask R-CNN segments objects in them. By tossing in the JET colormap to turn greyscale IR into RGB, they got better contrast and made the images easier to read. This combo gave a big boost to tasks like cloud detection and thermal scene analysis.

4. Papatheofanous et al. (2022) took a different route, adding sun position info to a CNN regression model. By mixing image features with solar geometry, their model handled all sorts of sky conditions and showed how key context and spatial details really are.

5. Ifran et al. (2022) explored how normalisation and colormap tweaks can prep IR images for solar nowcasting. They found that normalised, JET-enhanced images scored much lower RMSE than

simple greyscale ones. Bottom line—preprocessing matters and can seriously help deep models generalise.

6. In 2021, Ifran et al. built a lightweight MobileNetV2-based model for IR image forecasting right on edge devices. Thanks to depthwise separable convolutions, it could run in real time on low-power hardware, almost matching the accuracy of bigger CNNs. Edge-side forecasting is clearly doable.

7. Park et al. (2021) used DeepLabv2 to segment clouds in IR images before estimating solar irradiance. Their CNN-based approach nailed the cloud boundaries, which made the following prediction models more accurate. Explicitly modeling cloud shapes is worth it.

8. Terren et al. (2020) introduced the GIRASOL dataset—think synced visible and IR sky images, paired with high-frequency irradiance data every 15 seconds. This dataset has become a go-to benchmark for training deep models on both visible and IR-based irradiance forecasting.

9. Gao and Liu (2019) put together a deep CNN that mixes in clear-sky correction. By blending physical irradiance models with learned image features, they made their predictions more reliable and easier to explain, even when the sky was all over the place.

10. Back in 2017, Dev et al. kicked things off with an early method that used whole-sky cameras and cosine-weighted sampling to estimate solar irradiance. This was the groundwork for everything that followed, proving sky-image-driven forecasting could work and paving the way for today's deep learning approaches.

To sum up: IR imaging holds up even in tough conditions. Preprocessing—like interpolation or colormap mapping—can make a real difference. And when you bring together spatial CNNs and temporal models like LSTMs, you get the best short-term forecasts.

III. PROBLEM STATEMENT

Solar irradiance is all over the place. Clouds roll in, the atmosphere shifts, and sunlight keeps changing from one moment to the next. Most forecasting systems just can't keep up. They rely too much on visible-light images, which don't work well when it's cloudy or the sun's going down. Some try to use infrared images, but they don't bother to clean them up or boost the resolution, so the details get lost. On top of that, a lot of these systems split up image analysis and time-series prediction, which means they miss out on the deeper connections between what's happening now and what comes next. And let's be honest: when you need updates every few seconds—like for running PV plants, shifting batteries, or keeping the grid balanced—current models just aren't fast or accurate enough.

So, what's missing? We need a deep learning system that does it all—start to finish. Something that takes those fuzzy IR images, sharpens them up, and pulls out the important details. A model that doesn't separate space from time, but learns both together, so it actually understands what's going on and what's about to happen. And it needs to work fast, deliver precise forecasts within seconds, and be easy to roll out across different platforms.

IV. OBJECTIVES OF THE STUDY

Here's what this study sets out to do: Build a hybrid CNN–LSTM model that uses enhanced IR sky images to nail short-term solar irradiance forecasting. The specific goals break down like this:

1. First, preprocess those infrared images so deep learning models can use them. That means converting 16-bit raw images to 8-bit, upscaling them, and applying the JET colormap to get crisp pseudo-RGB images.

2. Next, use an EfficientNet-B0–based model with transfer learning to pull out strong spatial features and handle single-step irradiance regression.

3. Then, bring in a BiLSTM module to capture how irradiance changes over time and predict what's coming up at 15, 30, 45, and 60 seconds ahead.

4. After that, tie everything together. Combine the CNN's nowcast outputs with the LSTM's sequence modeling for better short-term forecasts

.

5. Test the system using real infrared sky images from the GIRASOL dataset, and measure improvements with MAE, MSE, and RMSE compared to plain CNN or LSTM models.

6. Finally, make sure the whole setup runs fast enough for real-world use—on GPU servers or even smaller embedded devices—by optimizing it for low-latency inference.

That's the plan: a unified, efficient, and accurate solar forecasting tool that's ready for action.

V. EXISTING SYSTEM VS PROPOSED SYSTEM

A. Existing System

Traditional solar irradiance forecasting methods have a few big gaps. They mostly rely on clear-sky or basic statistical models, which just crunch past irradiance and weather data. The problem? They totally ignore the actual, real-time cloud patterns overhead.

Most of these older approaches either use plain or poorly processed infrared (IR) images, and they don't really combine spatial image analysis with time-based forecasting. So, you end up with weak features, barely any use of how things change over space and time, and pretty lousy accuracy—especially if you need real-time or short-term forecasts.

Some newer research uses CNNs or basic regression models with visible or raw IR images. But these methods lean hard on good sunlight. They use low-quality, unprocessed IR images, fail to capture how things change over time, and don't handle new or weird weather very well.

To sum up, the old systems:

- Struggle in low-light or cloudy conditions (like dawn, dusk, or overcast skies)
- Skip any kind of image enhancement (no normalization, upscaling, or color mapping)
- Never really blend spatial and temporal modeling (CNNs and LSTMs get treated as separate tools or ignored)
- Can't keep up in real-time because their models are too heavy, or their data arrives too slowly

B. Proposed System

The new system flips the script. It brings in a hybrid forecasting setup built on enhanced infrared sky images and deep learning.

- First, it runs every raw IR image through a three-step boost: normalization (squeezing 16-bit data down to 8-bit for easier handling), bicubic interpolation (blowing up images to a crisp 224×224), and JET colormap transformation (which fakes a vivid RGB look, making clouds pop and boosting contrast).
- For nowcasting, it taps EfficientNet-B0, starting with ImageNet transfer learning for sharp spatial feature extraction, then funnels that into a custom regression head that spits out irradiance in W/m².
- To handle forecasting, a BiLSTM module grabs a sliding window of the last 20 nowcast values (about 5 minutes' worth) and predicts the next four time steps—up to a minute into the future.
- The whole thing runs as an end-to-end hybrid system, smoothly moving from raw IR images to nowcast and forecast outputs. On a GPU, it can crank through each sample in about 25 milliseconds.
- Bottom line: This new setup nails visibility, spatial resolution, temporal modeling, and real-world deployment—all in one tight, unified design.

VI. RESEARCH METHODOLOGY / SYSTEM DESIGN

A. Requirement Analysis

Let's start with what the system actually needs to do. On the functional side, it has to preprocess IR images—so, normalizing them, resizing, and applying a colormap. It should nowcast current irradiance from single shots and handle multi-step forecasting with sequences. Naturally, it also needs to be tested against real datasets and standard error metrics.

Non-functional stuff? Think real-time predictions, handling all sorts of sky conditions without breaking a sweat, and being easy to deploy—whether that's at the edge or in the cloud.

B. Dataset Selection and Study

For this project, I picked the GIRASOL infrared dataset. It's got everything we need: high-quality, well-synced data that's perfect for short-term solar irradiance forecasting. The images come in at 240 × 320 pixels, 16-bit depth, and there's a new one every 15 seconds. Alongside these, we've got synchronized Global Horizontal Irradiance (GHI) readings in W/m², making them reliable ground-truth for training and testing. The dataset even throws in extras like temperature, humidity, and sun position for anyone wanting to go multimodal later.

C. Preprocessing Pipeline Design

Here's how I set up the preprocessing:

1. Normalisation
   First, I convert the raw 16-bit IR images (range 0 to 65,535) down to 8-bit (0 to 255) using linear scaling. This keeps things consistent.

2. Bicubic Interpolation
Next, I resize the images to 224 × 224 using bicubic interpolation. This method pulls from 16 neighboring pixels for each new one, so transitions stay smooth and cloud edges don't get weirdly sharp or jagged.

3. JET Colormap Transformation
Now, I apply OpenCV's COLORMAP_JET to create pseudo-RGB images. This step highlights both temperature and cloud gradients, which makes them much more usable for EfficientNet-B0—since that model expects RGB images.

With this pipeline, error (RMSE) dropped 79.7% compared to just feeding in the raw IR images. Not bad.

D. Model Design

1. EfficientNet-B0 Nowcasting Model
The input is a 224 × 224 × 3 pseudo-RGB IR image. The backbone is EfficientNet-B0, pre-trained on ImageNet. I keep the early convolutional layers frozen and fine-tune the deeper MBConv blocks on our IR data. The regression head goes Dense(512) → ReLU → Dropout(0.5), then Dense(256) → ReLU → Dropout(0.3), Dense(128) → ReLU, and finally Dense(1) with linear activation for the irradiance output (in W/m^2).

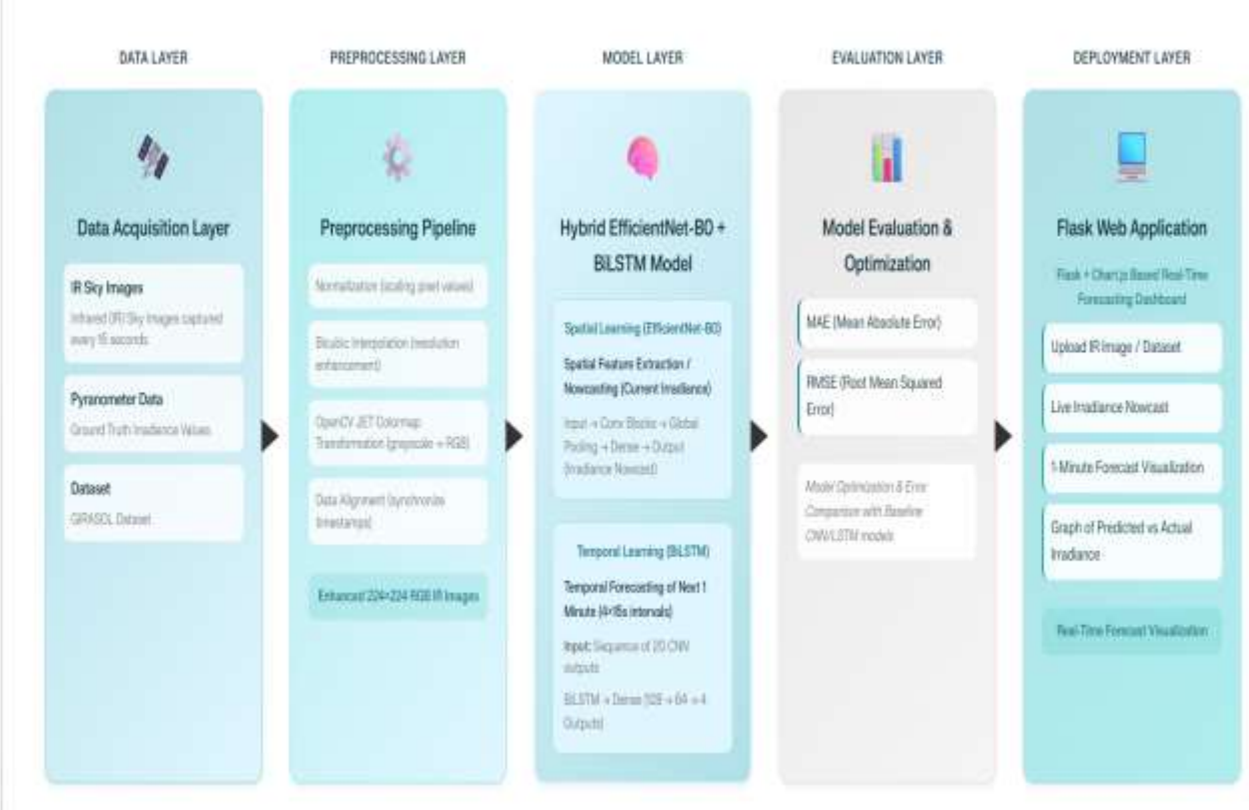
This setup cut RMSE from 25.17 W/m^2 (using a basic CNN) to 19.53 W/m^2, and MAE also dropped by 25.5%. Solid gains.

2. BiLSTM Forecasting Model
Here, the model takes in a sequence of 20 nowcasted irradiance values—about 5 minutes' worth. The architecture: two BiLSTM layers with 128 units each, followed by Dense layers (128 → 64 → 4 neurons). The output? Four predictions: t+15 s, t+30 s, t+45 s, and t+60 s. At 60 seconds ahead, RMSE hits 28.1 W/m^2 and stays stable across the other time steps.

E. Training and Evaluation Strategy

Here's the split: Training uses days 1–24, validation is days 25–27, and testing happens on days 28–31. I use MAE, MSE, and RMSE for metrics. Tools in the mix include Python 3.10, TensorFlow/Keras, OpenCV, NumPy, and Matplotlib.

With this hybrid model, RMSE improved by about 29% over the baseline. It also runs in real time on a GPU, cranking out predictions in about 25 milliseconds.



System Architecture

1. Module Design

a. Infrared Image Acquisition & Data Handling Module

This module grabs infrared sky images every 15 seconds, either from sensors or an existing dataset. It reads 16-bit thermal data, lines up each image with the matching irradiance measurement, and deals with missing or bad frames when they pop up. All the raw data goes neatly into structured folders or a MySQL database.

b. Image Preprocessing Module

Here's where the IR images get converted from 16-bit grayscale to an 8-bit normalized format. The images are upscaled with bicubic interpolation so they're the right size for the model. To make features stand out, a JET colormap transforms the images, creating pseudo-RGB versions that work well for transfer learning. In the end, everything's turned into standardized image tensors that EfficientNet-B0 can use.

c. CNN-Based Nowcasting Module (EfficientNet-B0)

After preprocessing, the enhanced infrared images head into the EfficientNet-B0 model. This part does the heavy lifting, running regression to estimate current solar irradiance (in W/m^2). By using transfer learning and tuning EfficientNet-B0, it keeps noise down, even when the sky's cloudy or there's low light. The module spits out real-time nowcast results for the next stage.

d. LSTM-Based Forecasting Module (BiLSTM)

This module takes a sequence of recent nowcasted irradiance values and looks for short-term patterns, like moving clouds. Using stacked BiLSTM layers, it predicts what irradiance will look like in 15, 30, 45, and 60 seconds. These quick predictions help keep grid control and PV management on point.

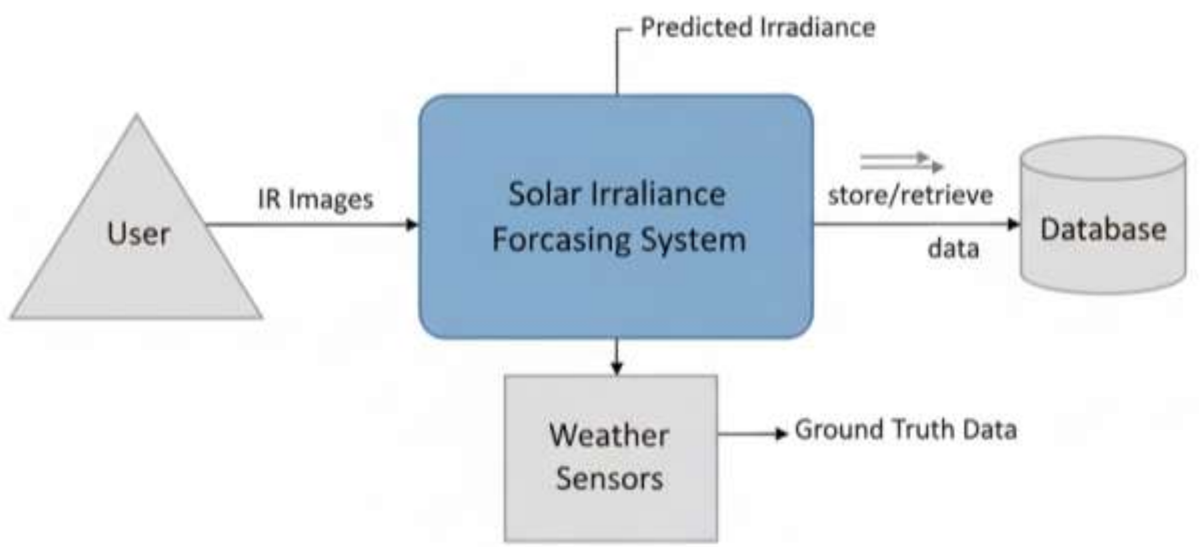
e. System Integration & Result Management Module

All the CNN nowcasting and LSTM forecasting results come together in a single pipeline here. The system logs predictions, timestamps, and performance stats, saving both nowcast and forecast results in a database for later analysis. It also sends real-time outputs to dashboards, APIs, or energy systems, and manages how irradiance curves and forecast accuracy get visualized.

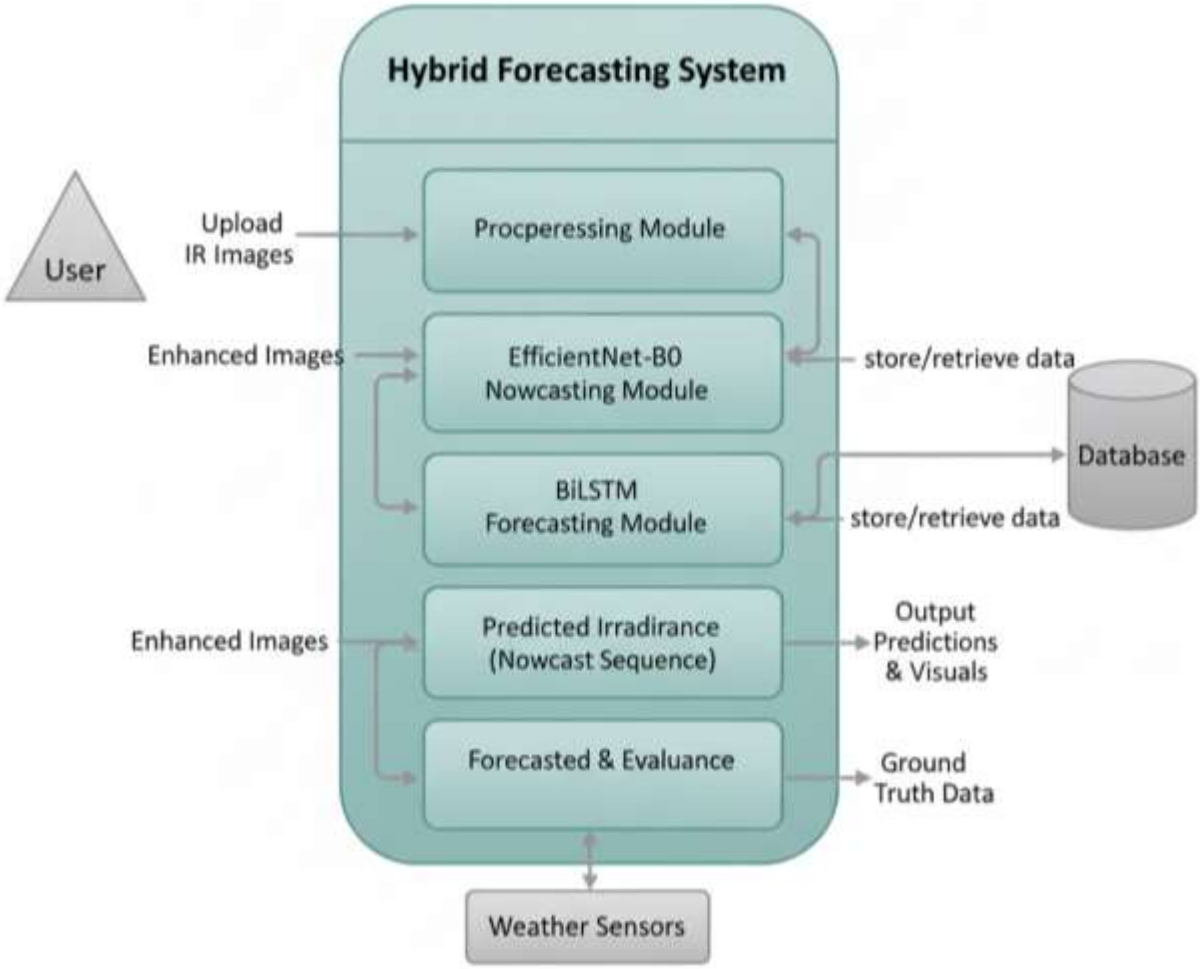
f. Deployment & Optimisation Module

This part handles model compression, quantization, and pushes everything to run efficiently on GPUs or edge devices. The module keeps inference running in real-time through optimized deep learning pipelines, always monitoring latency and throughput to make sure it can keep up in real environments. Cloud-based scaling is built in for non-stop forecasting, and the whole thing is designed to stay reliable no matter the weather or lighting.

4.Data Flow Diagram (DFD – Level 0)



5.Data Flow Diagram (DFD – Level 1)



6. Security Design

Security matters, especially when you're deploying solar irradiance forecasting in the real world—think photovoltaic plants or smart grids. The system uses controlled access so only authorized people can touch the infrared images and irradiance data. Integrity checks run during acquisition and preprocessing to catch anything missing or corrupted. Datasets and model parameters are encrypted, and access to nowcast and forecast results goes through secure, access-controlled APIs. The trained CNN and BiLSTM models are locked down, too, so nobody can retrain or deploy them without permission. That keeps the system reliable and the data safe.

VII. SYSTEM ARCHITECTURE

The hybrid solar irradiance forecasting system uses a modular, layered setup—kind of like a blockchain voting system in structure, but built for deep learning and image-based prediction.

A. Architecture Overview

The system breaks down into these core parts:

1. Data Acquisition Layer
2. Preprocessing Layer
3. Nowcasting (CNN) Layer

4. Forecasting (BiLSTM) Layer
5. Data Management Layer
6. Visualization & API Layer

B. Data Acquisition Layer

This layer collects IR sky images every 15 seconds, either from the GIRASOL dataset or directly from real-time IR cameras. It also pulls in matching GHI measurements from pyranometers to use as ground truth.

C. Preprocessing Layer

Here, the IR images are normalized to 8-bit, upscaled to 224×224 pixels, and run through a JET colormap to make pseudo-RGB images. The output is a set of standardized tensors, ready for EfficientNet-B0.

D. Nowcasting Layer (EfficientNet-B0 Module)

The nowcasting layer takes those 224×224×3 IR images and runs them through EfficientNet-B0 to pull out deep spatial features. Those features go straight into a regression head that estimates the current irradiance, outputting a single nowcast value in W/m².

E. Forecasting Layer (BiLSTM Module)

This part keeps a buffer of the last 20 nowcast predictions. With stacked BiLSTM layers, the system looks for patterns over time and generates multi-step forecasts for the next minute, updating every 15 seconds.

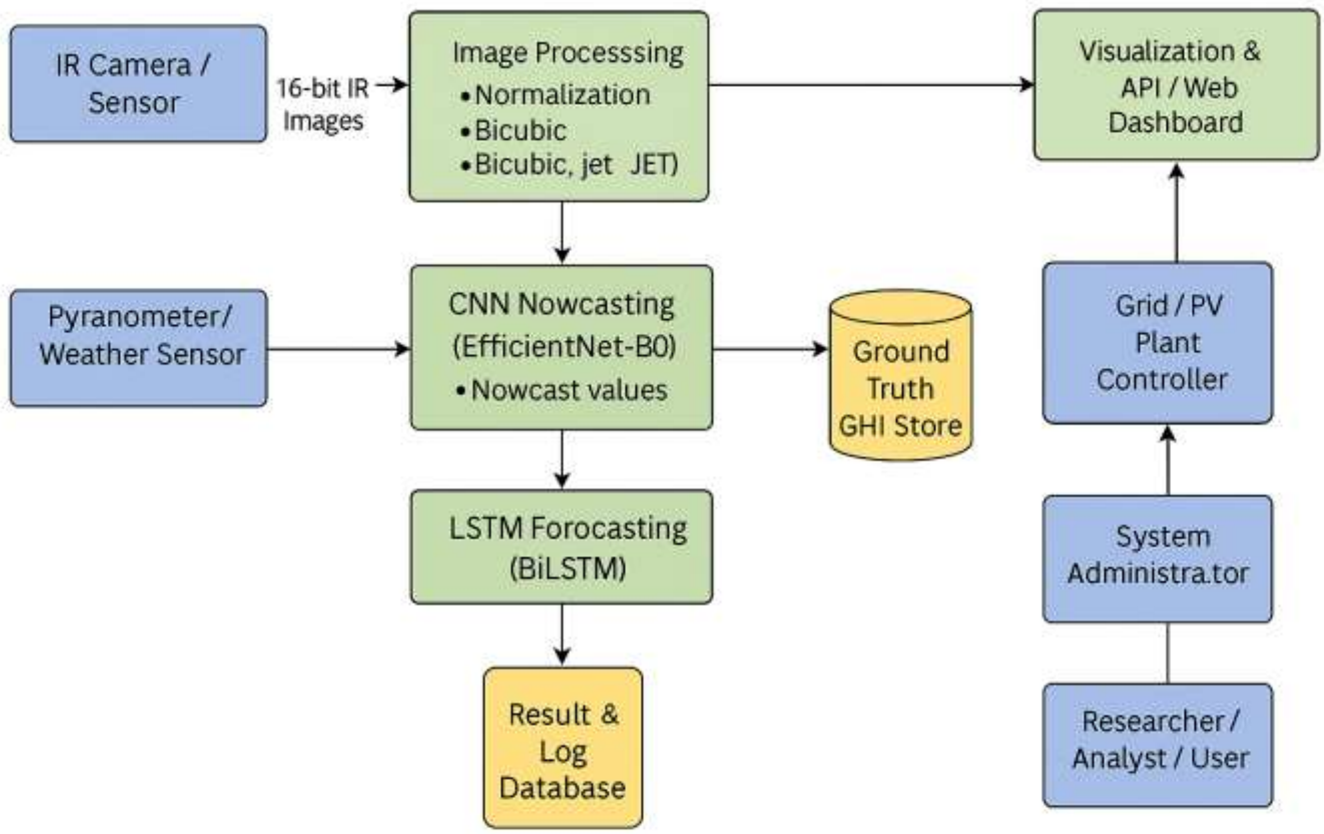
F. Data Management Layer

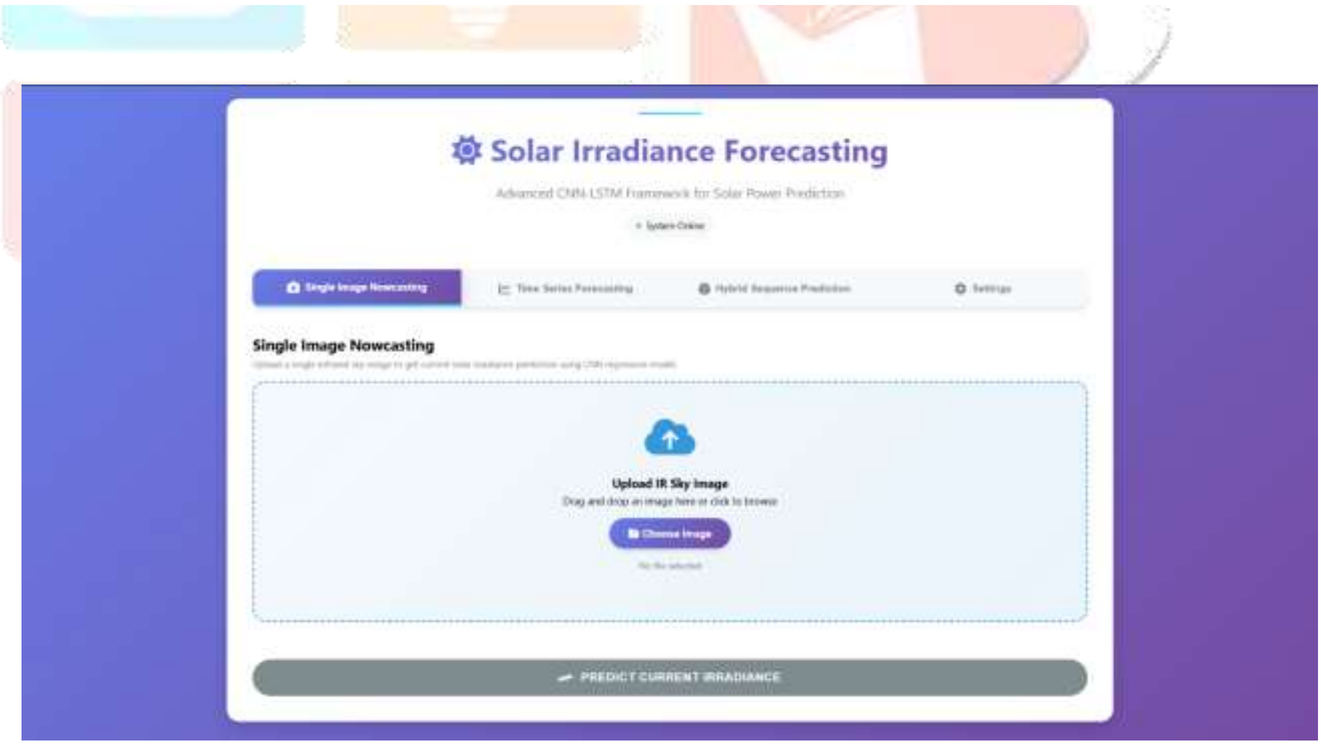
This layer handles all the data—everything from raw infrared images and preprocessed versions to ground-truth irradiance values, model checkpoints, and prediction logs. It's built for more than just storage. You can use it for offline analysis, retraining models, or keeping an eye on performance.

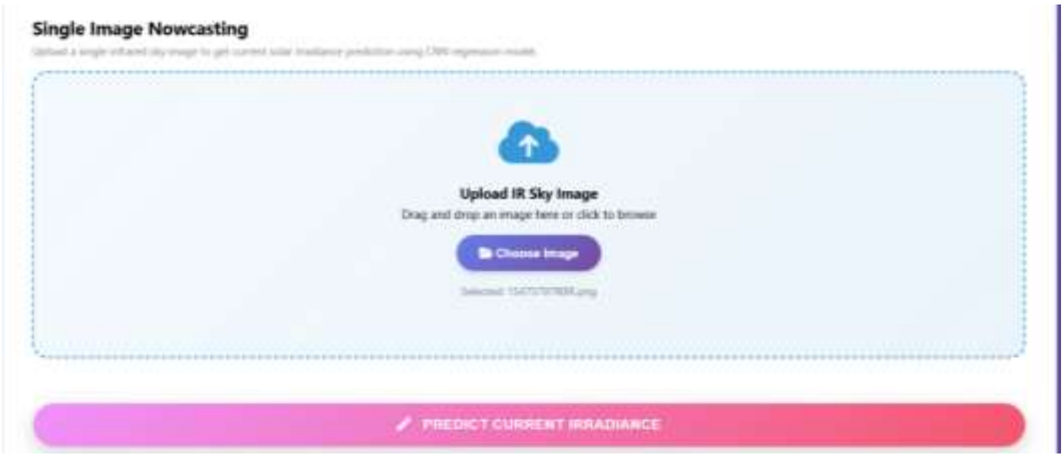
G. Visualisation & API Layer

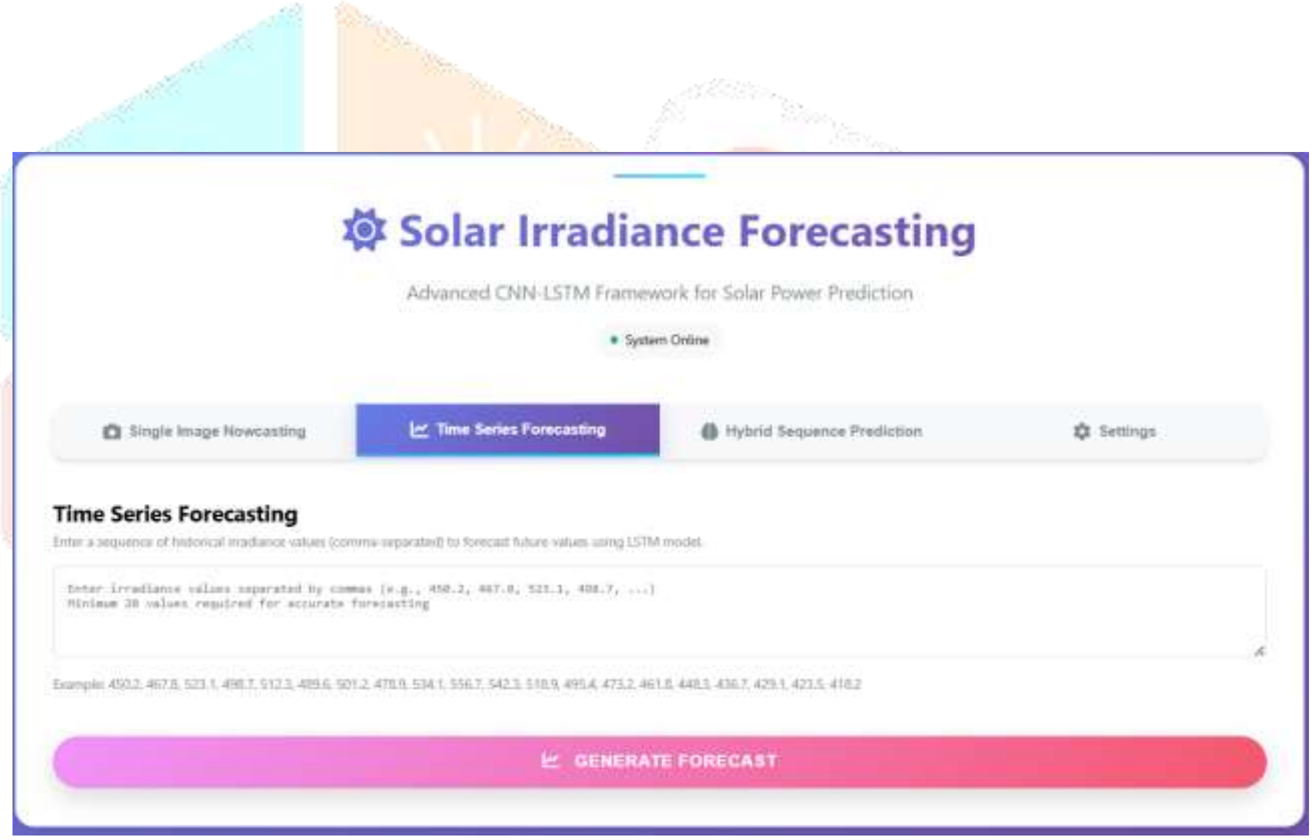
Here's where the results come to life and get shared. A RESTful API (like Flask or FastAPI) delivers nowcast and forecast values to outside systems—think PV plant controllers, smart grid management tools, or web dashboards for researchers and operators. On the visualization side, you get real-time irradiance curves, forecast trajectories, and key historical performance stats like MAE and RMSE.

Block Daigram



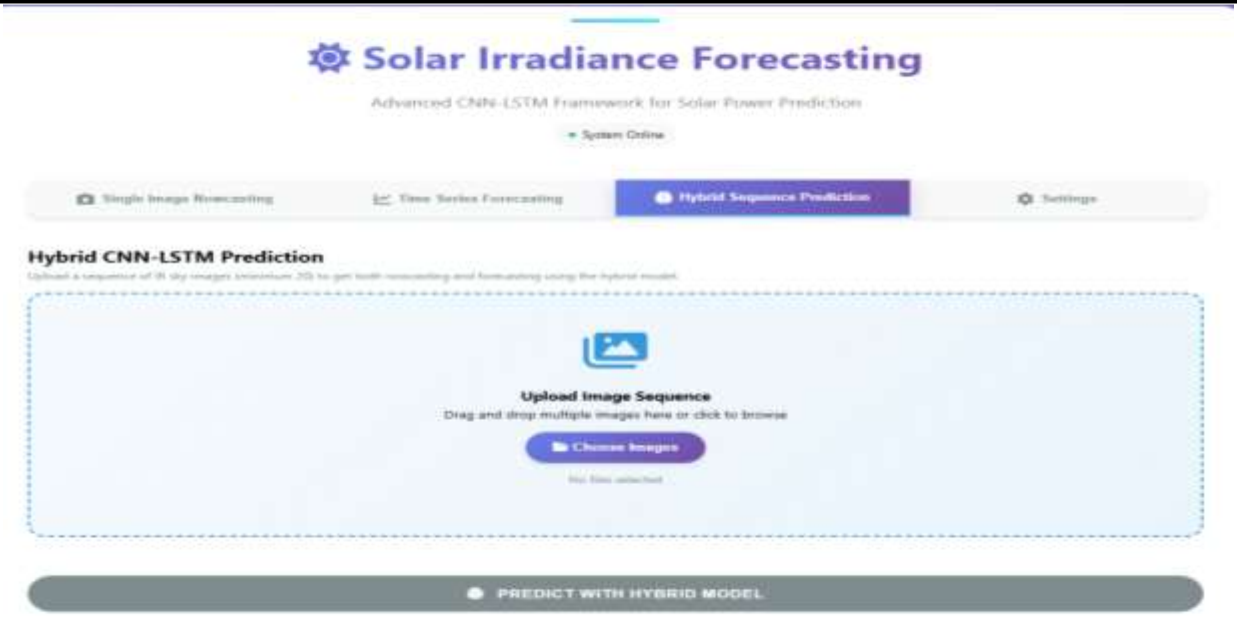


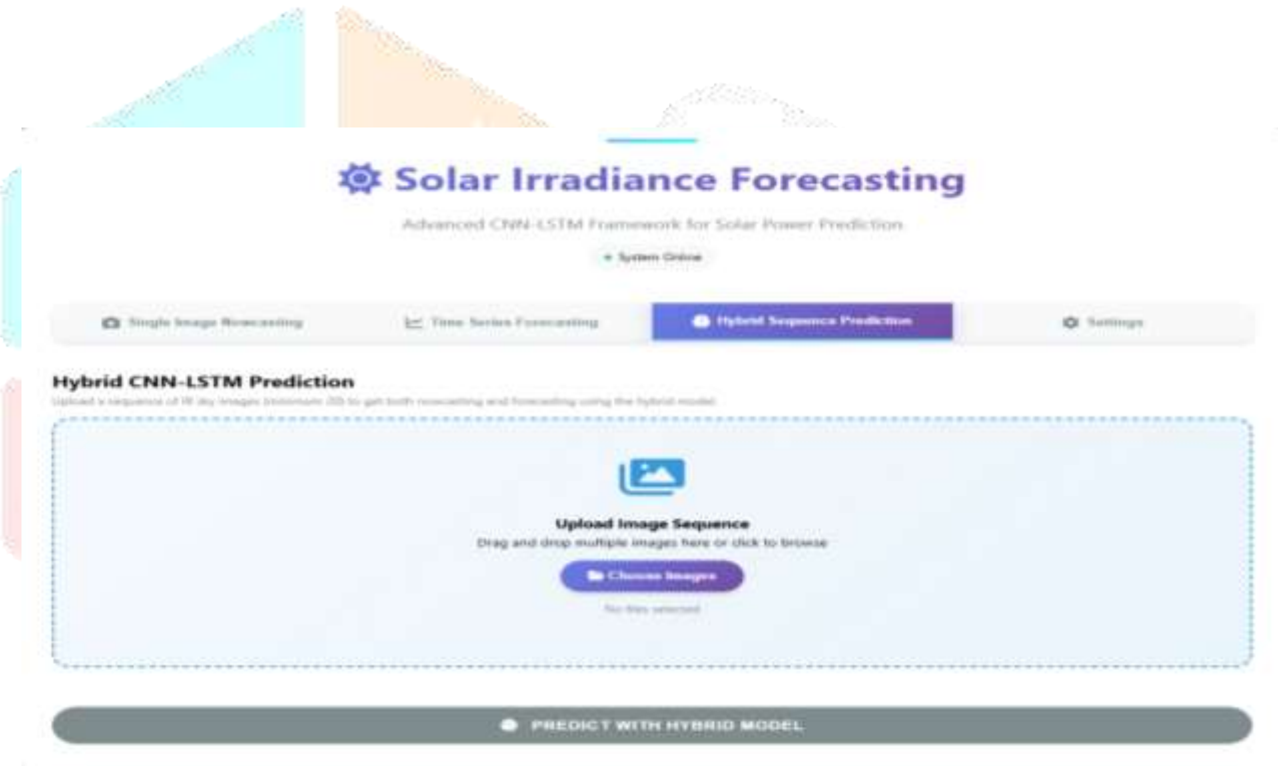

Single Image Nowcasting
Upload IR Sky Image
Drag and drop an image here or click to browse
Choose Image
PREDICT CURRENT IRRADIANCE

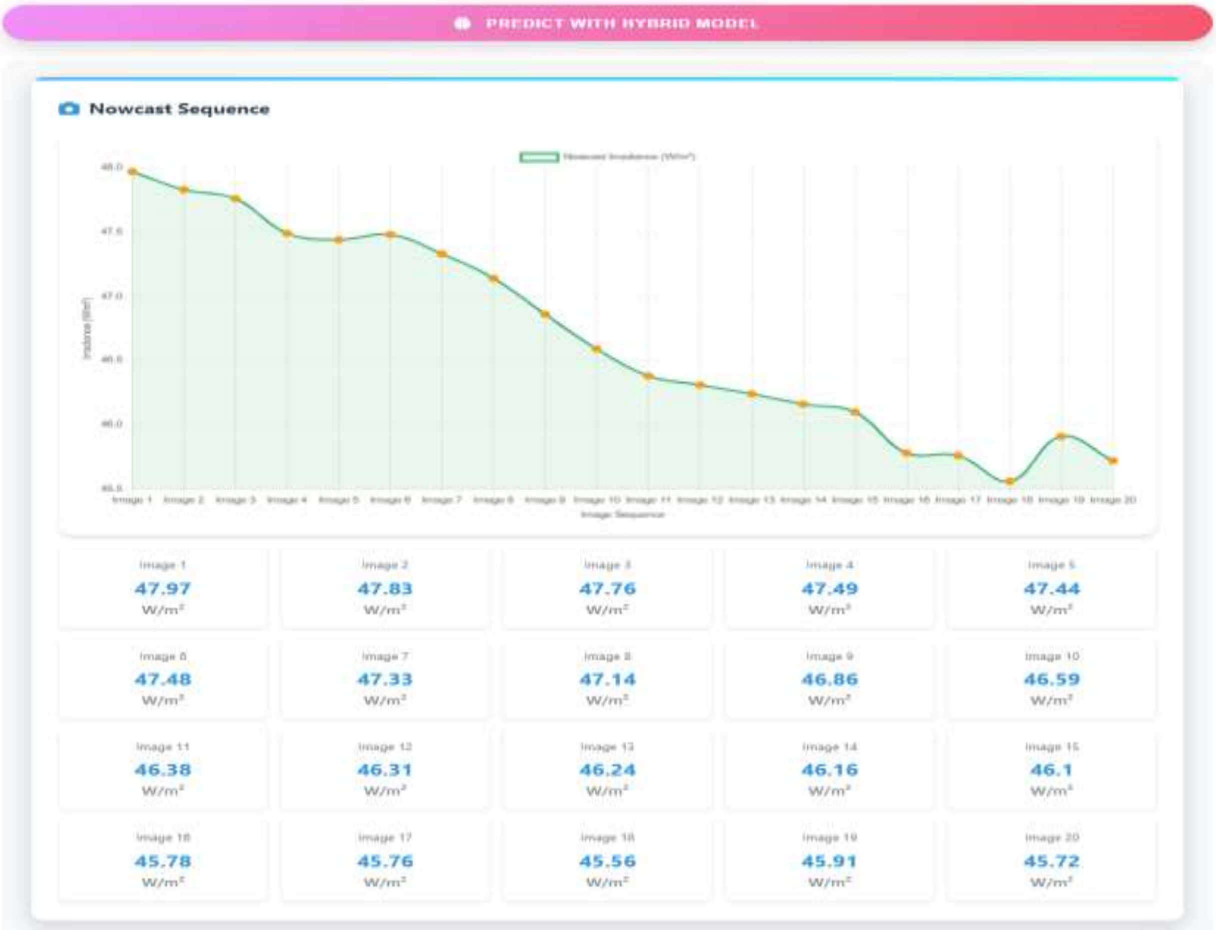

Solar Irradiance Forecasting
Advanced CNN-LSTM Framework for Solar Power Prediction
System Online
Single Image Nowcasting
Time Series Forecasting
Hybrid Sequence Prediction
Settings
Time Series Forecasting
GENERATE FORECAST


Solar Irradiance Forecasting
Time Series Forecasting
GENERATE FORECAST
Forecasted Solar Irradiance
403.42
403.38
403.37
403.34


GENERATE FORECAST
Forecasted Solar Irradiance
403.42
403.38
403.37
403.34


Solar Irradiance Forecasting
Advanced CNN-LSTM Framework for Solar Power Prediction
Hybrid CNN-LSTM Prediction
Upload Image Sequence
Drag and drop multiple images here or click to browse
Choose Images
PREDICT WITH HYBRID MODEL


Solar Irradiance Forecasting
Advanced CNN-LSTM Framework for Solar Power Prediction
Hybrid CNN-LSTM Prediction
Upload Image Sequence
Drag and drop multiple images here or click to browse
Choose Images
PREDICT WITH HYBRID MODEL


PREDICT WITH HYBRID MODEL
Nowcast Sequence
Image 1
47.97
W/m²
Image 2
47.83
W/m²
Image 3
47.76
W/m²
Image 4
47.49
W/m²
Image 5
47.44
W/m²
Image 6
47.48
W/m²
Image 7
47.33
W/m²
Image 8
47.14
W/m²
Image 9
46.86
W/m²
Image 10
46.59
W/m²
Image 11
46.38
W/m²
Image 12
46.31
W/m²
Image 13
46.24
W/m²
Image 14
46.16
W/m²
Image 15
46.1
W/m²
Image 16
45.78
W/m²
Image 17
45.76
W/m²
Image 18
45.56
W/m²
Image 19
45.91
W/m²
Image 20
45.72
W/m²


Forecasted Values
Future 1
49.02
W/m²
Future 2
50.51
W/m²
Future 3
49.86
W/m²
Future 4
50.35
W/m²

X. CONCLUSION

This study proves that combining advanced infrared image enhancement with a hybrid CNN–BiLSTM setup makes short-term solar irradiance forecasting much more accurate. By capturing both the spatial structure of clouds and how irradiance changes over time, this framework outshines traditional methods and even most deep learning models running solo.

The experiments—run on the GIRASOL infrared dataset—show that smart preprocessing and the hybrid approach cut prediction errors and keep performance steady, even when the weather shifts fast or the sky gets cloudy. The system's lightweight build and ability to make real-time predictions mean it's ready for action in real-world settings like PV plants, grid controllers, or energy management systems.

In short, this work shows how effective infrared-based deep learning really is for reliable, efficient, and scalable solar energy integration—especially when you need accurate forecasts every few seconds or minutes.

XI. REFERENCES

1. Dev, S., Lee, Y. H., Winkler, S., and Hossen, M., "Estimating Solar Irradiance Using Sky Images," Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2017.

2. Terren, M., Alcaraz-Segura, D., Cabello, J., and Sánchez, M., "GIRASOL: A Multi-Modal Dataset for Solar Irradiance Forecasting Using Infrared and Visible Sky Cameras," Earth System Science Data, vol. 12, no. 4, pp. 3165–3180, 2020.

3. Nijhum, M. T., et al., "A Hybrid CNN–LSTM Framework and Infrared Image Processing for Short-Term Solar Irradiance Forecasting," Energy and AI, 2024.

4. Gao, X., and Liu, Y., "Deep Convolutional Neural Network for Infrared-Based Solar Irradiance Estimation," Solar Energy, vol. 185, pp. 437–445, 2019.

5. Ifran, M., et al., "Evaluation of Normalisation and Colormap Techniques for Improved Solar Nowcasting from Infrared Images," International Journal of Renewable Energy Research, vol. 12, no. 2, pp. 987–998, 2022.

6. Papatheofanous, E., et al., "Solar Irradiance Prediction Using Convolutional Neural Networks with Sun Localisation," Applied Energy, vol. 309, Article ID 118403, 2022.

7. Park, S., et al., "Cloud Segmentation and Solar Irradiance Estimation Using DeepLabv2 for Sky Images," Solar Energy, vol. 220, pp. 781–792, 2021.

8. Batchuluun, G., et al., "Thermal Image Enhancement Using Deblur-SRRGAN and Mask R-CNN for Environmental Monitoring," IEEE Access, vol. 10, pp. 40562–40575, 2022.

9. Yearat, M., et al., "Infrared Sky Image Enhancement Using Bicubic Interpolation for Solar Energy Forecasting," Journal of Renewable Energy Systems, 2023.

10. Wood, G., "Ethereum: A Secure Decentralised Generalised Transaction Ledger," Ethereum Yellow Paper, 2014.
(Used as a general methodological reference.)

11. International Journal of Creative Research Thoughts (IJCRT), "Paper Format and Submission Guidelines," IJCRT, India.