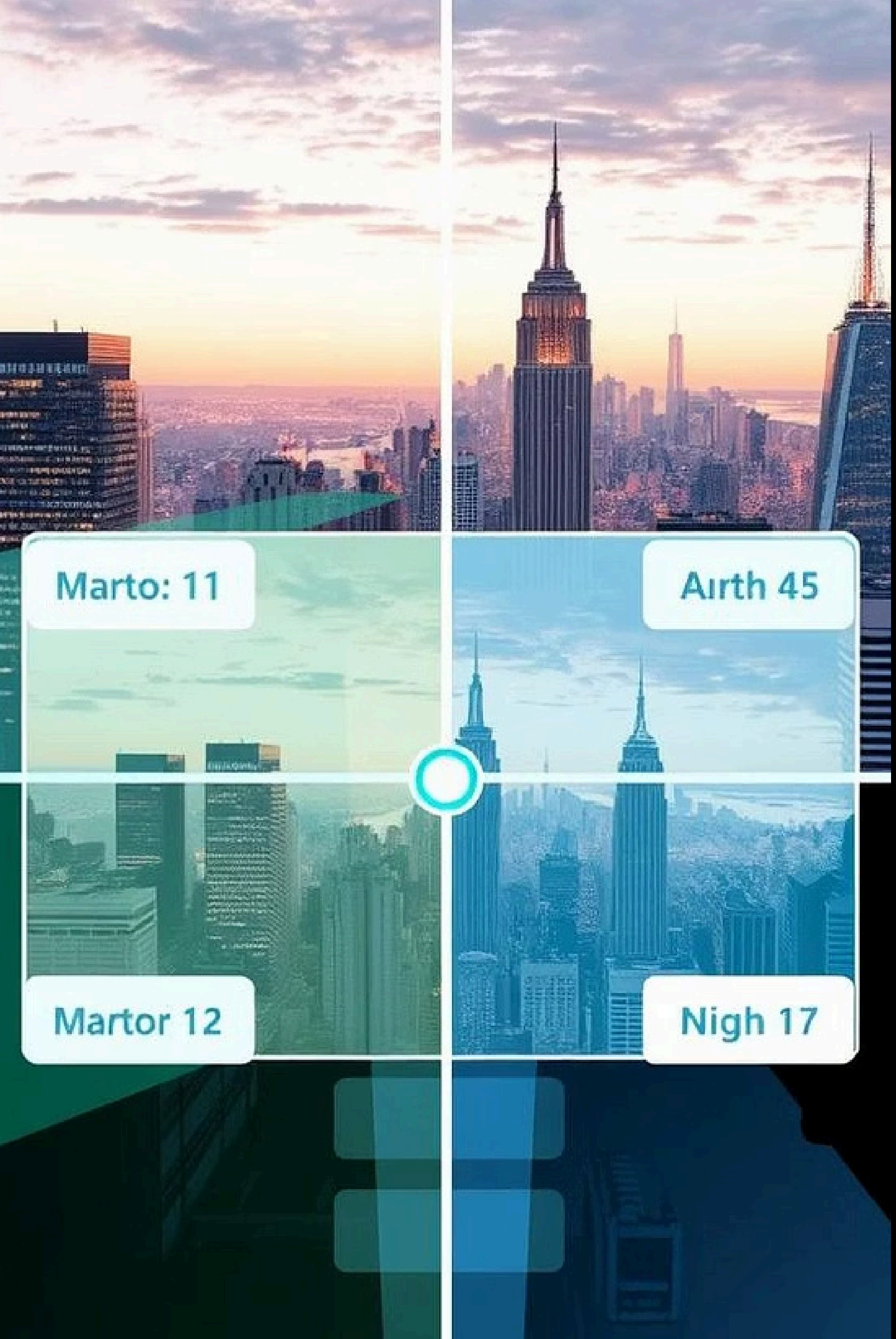
Day or Night? Unveiling the Vision of AI

Explore how Artificial Intelligence learns to perceive the world, just like us, but with a unique digital insight and with **Smart Model**.

Let's ask a question...

Can a computer tell if it’s day or night just by looking at a photo?



But how computer can see photo to tell us if it in day or night?

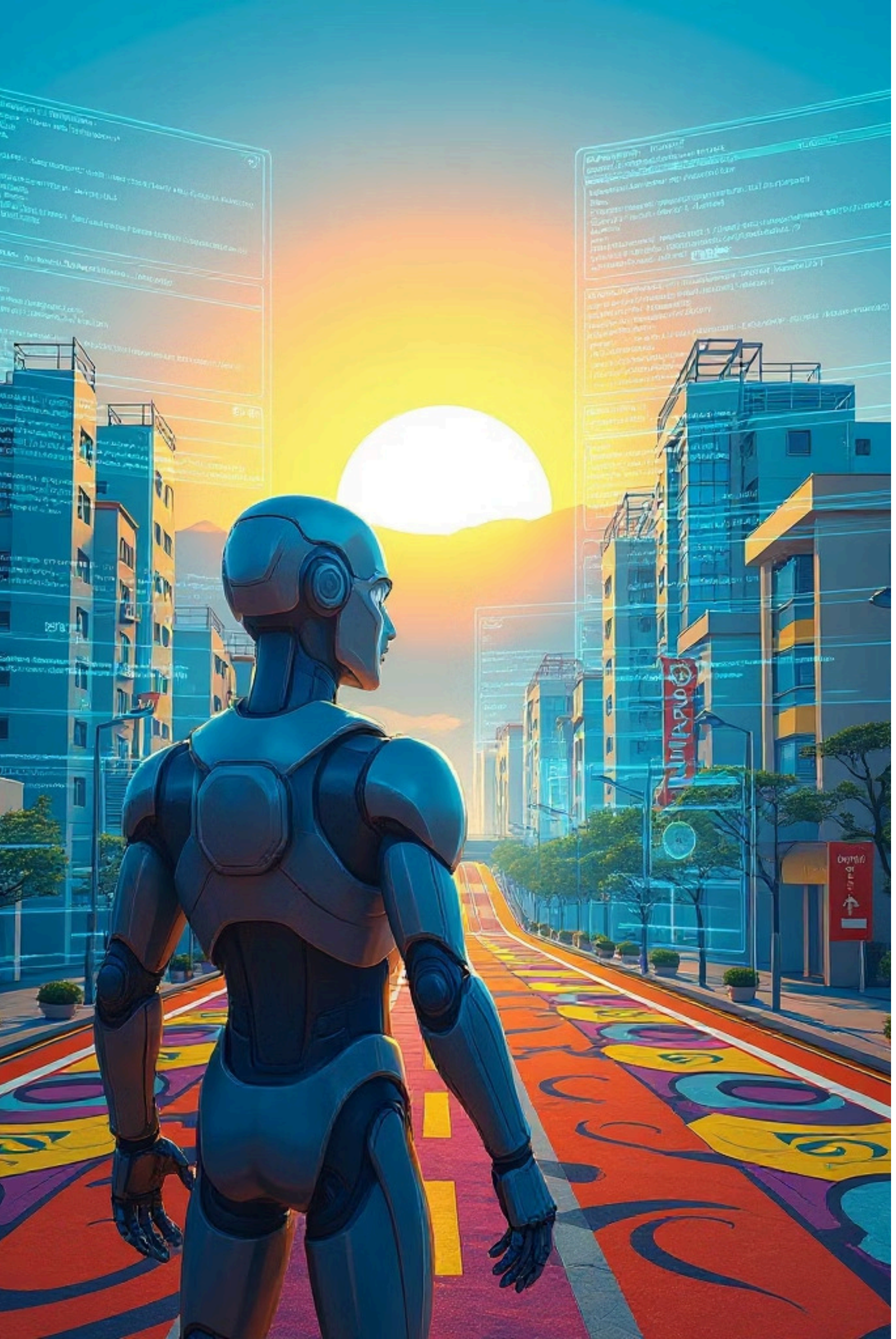
Do Computers See Images Like We Do as HUMANS?

Human Vision

Our eyes capture light, sending signals to the brain which interprets colors, shapes, and context, allowing us to distinguish day from night.

Machine Vision

Computers see images as arrays of numbers (pixels). They require complex algorithms to recognize patterns and make sense of visual data.



From Pixels to Perception: How Computers "See"

Every image is a grid of pixels, each with a numerical value representing color and intensity. Computers process these numbers to extract meaningful features.

Decoding Images: From Pixels to Meaning

To move beyond raw pixel data, computers employ sophisticated methods to interpret what they "see." This process transforms simple numerical values into meaningful insights, much like our brains process visual information.

Feature Extraction

Computers identify basic visual cues such as edges, textures, and color gradients within the image.

Pattern Recognition

These extracted features are then combined to recognize complex patterns, shapes, and ultimately, objects.

Advanced Understanding

For intricate tasks like distinguishing day from night, a more sophisticated approach is required to build a comprehensive perception.

> This is where specialized neural networks come into play, enabling computers to achieve a deeper understanding of visual content. Specifically, **Convolutional Neural Networks (CNNs)** are designed to excel at this complex visual interpretation.

Introducing Convolutional Neural Networks (CNNs)

CNNs are a specialized type of neural network designed to process pixel data in images. They are inspired by the human visual cortex, learning to identify features at different levels of abstraction.

Input Layer

The raw image data (pixels) is fed into the network.

Convolutional Layers

Filters scan the image, extracting features like edges, textures, and patterns.

Pooling Layers

These layers reduce the dimensionality of the feature maps, making the network more efficient and robust.

Fully Connected Layers

The extracted high-level features are then used to make predictions, such as classifying an image.

The "Day or Night" Prediction Model

Our CNN model is specifically trained to classify images as either "Day" or "Night." This seemingly simple task involves complex pattern recognition.

Day Image Input

Night Image Input

Training the Model: Learning from Light and Shadow

The CNN learns by analyzing thousands of labeled images. It identifies subtle differences in light, shadow, color temperature, and object appearance that distinguish day from night.

1

Data Collection

Gather a diverse dataset of day and night images.

2

Labeling

Manually tag each image as "day" or "night."

3

Model Training

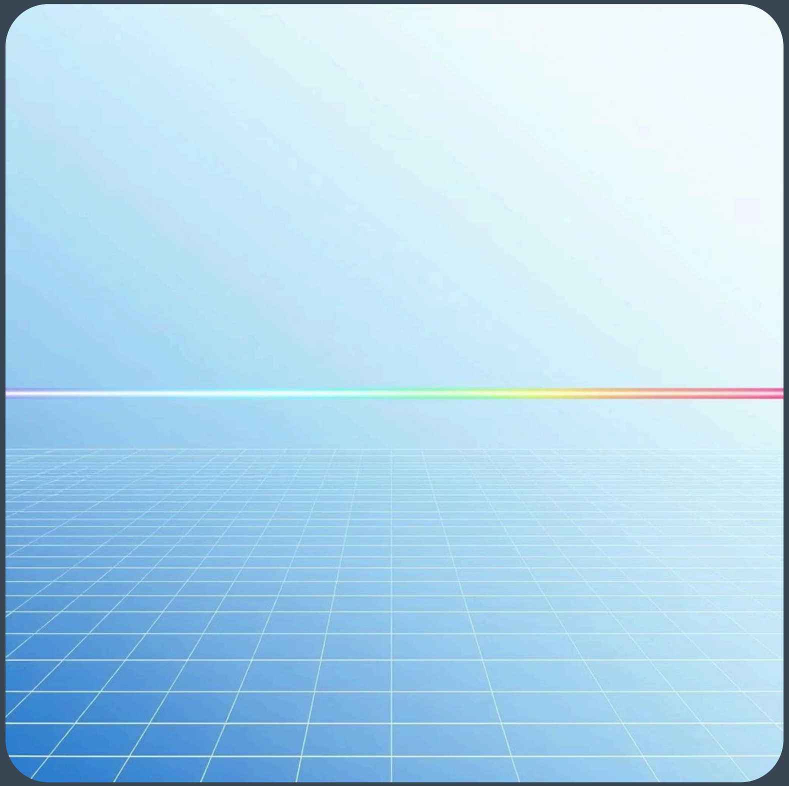
Feed the labeled data to the CNN, allowing it to adjust its internal parameters.

4

Validation

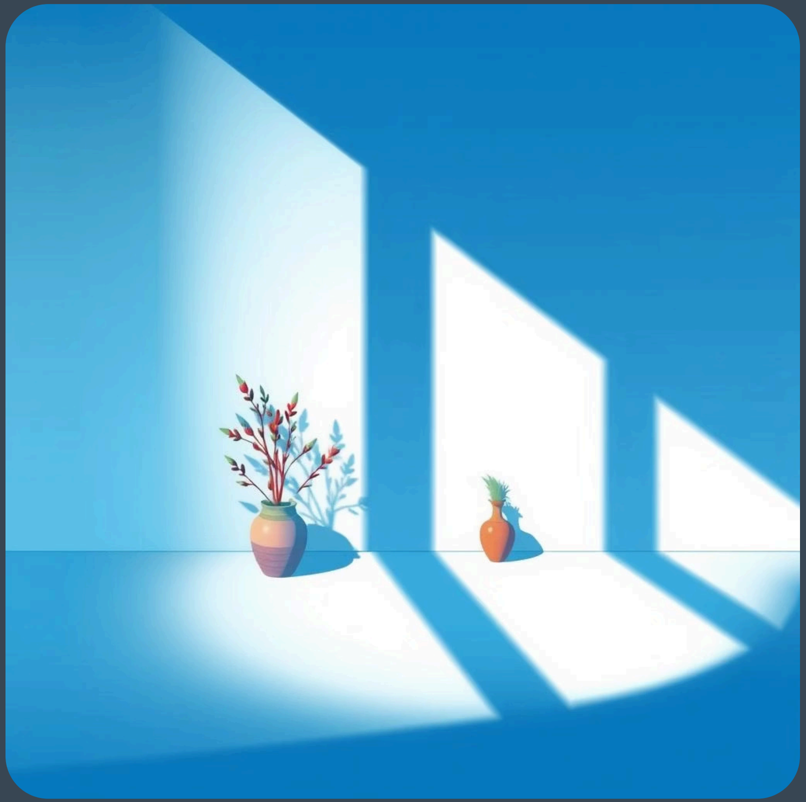
Test the model's accuracy on unseen images.

Key Features for Day/Night Detection



Color Temperature

Day images typically have cooler, bluer tones, while night images often exhibit warmer, yellowish hues from artificial light sources.



Light Intensity & Shadows

Daylight is generally bright and produces sharp shadows. Night scenes are darker with softer, less defined shadows or distinct light sources.

Applications of Day/Night Prediction

Autonomous Vehicles

Adjusting vehicle perception systems for varying light conditions. For example, enabling automatic headlamp activation.

Surveillance Systems

Optimizing camera settings and alerting systems based on ambient light. This could mean activating night vision or motion sensors in the dark.

Energy Management

Controlling smart lighting or solar panel systems more efficiently. This could involve turning off lights during the day automatically.

Smart City Planning

Understanding pedestrian and traffic patterns at different times. This can help optimize public transport or resource allocation.

Challenges and Future Directions

While effective, day/night prediction models face challenges and continue to evolve.

- **Ambiguous Lighting:** Overcast days, dawn, or dusk can present challenges for accurate classification.
- **Data Diversity:** Ensuring the model is robust to various environments and conditions (e.g., snow, heavy fog).
- **Real-time Performance:** Optimizing models for rapid inference on edge devices for immediate decision-making.

Future work involves integrating more contextual information and developing adaptive learning algorithms.

Let's deep now to make our smart model to make the required Classifications

First Our Dataset

- Our dataset it's **https://www.kaggle.com/datasets/hossamahmedsalah/dayandnight/**
- This dataset contain two folders:
  - 📂 [train] folder contain another two folders:
    - 🌅day folder contain (120 image) samples for day photos
    - 🌃 night folder contain (120 image) samples for night photos

  - 📂 [testing] folder contain another two folders
    - 🌅day folder contain (80 image) samples for day photos
    - 🌃 night folder contain (80 image) samples for night photos

Second Choosing the Right Model for Classification

Selecting an appropriate machine learning model is crucial for accurate day/night classification. We evaluated several approaches to determine the best fit for image-based analysis.

Traditional ML Models

Algorithms like SVMs or Decision Trees can classify data, but struggle with the complexity of raw image pixels.

Basic Neural Networks

While capable, standard neural networks often lack the specialized architecture for efficient image feature extraction.

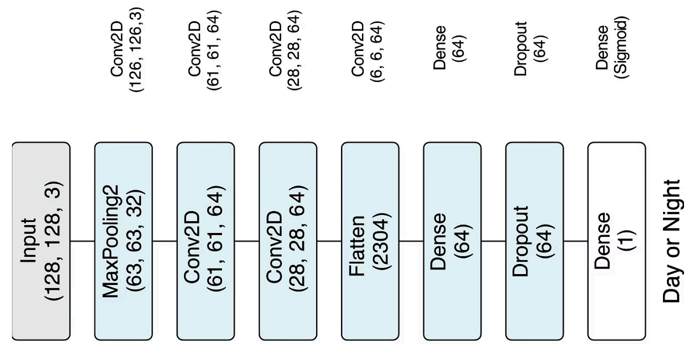
Convolutional Neural Networks (CNNs)

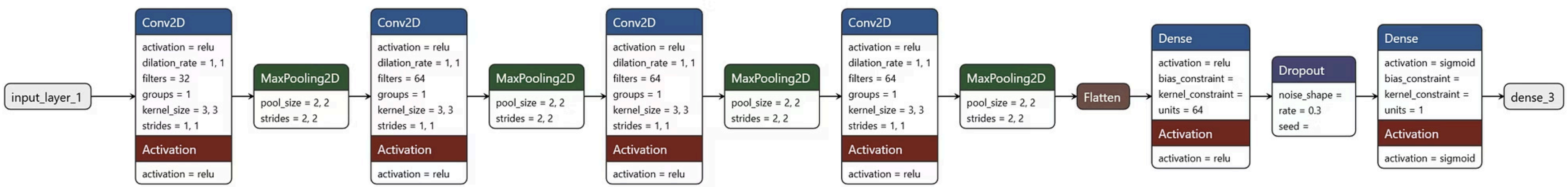
CNNs are purpose-built for visual data, excelling at recognizing patterns and hierarchies in images, making them ideal for our task.

Given the nature of image data and the need for robust feature learning, **Convolutional Neural Networks (CNNs) emerged as the optimal choice** for our day/night prediction model.

Third Building Our CNN Model

Having chosen Convolutional Neural Networks for their superior image processing capabilities, let's look at the architecture of our specific model designed for day/night classification. This model leverages several common CNN layers to effectively learn and classify visual features.





CNN Model structure details

Each layer plays a crucial role in transforming raw pixel data into meaningful predictions:

Convolutional Layers (Conv2D)

These layers apply filters to the input images, detecting patterns like edges, textures, and more complex shapes as data progresses through the network.

Pooling Layers (MaxPooling2D)

Pooling layers reduce the spatial dimensions of the feature maps, helping to simplify the model, reduce computational cost, and make the detected features more robust to slight shifts in the image.

Flatten Layer

After feature extraction, the Flatten layer converts the 2D feature maps into a single, long 1D vector, preparing the data for the fully connected layers.

Dense Layers

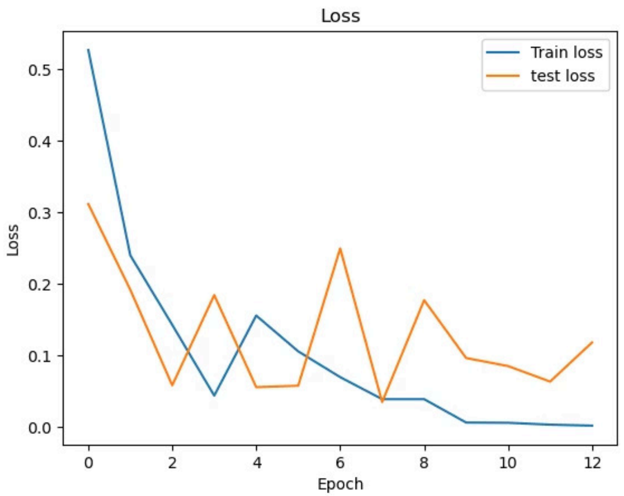
These are standard neural network layers that take the flattened features and perform high-level reasoning to make the final classification. The output layer uses a 'sigmoid' activation for binary (day/night) prediction.

Dropout

A regularization technique that randomly 'drops out' (sets to zero) a fraction of neurons during training. This prevents the model from overfitting to the training data, improving its generalization to new images.

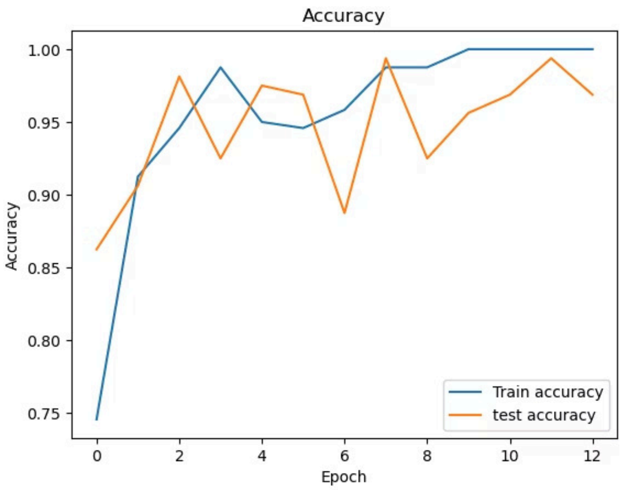
Finally Evaluating Model Performance

Training
Loss Over Epochs



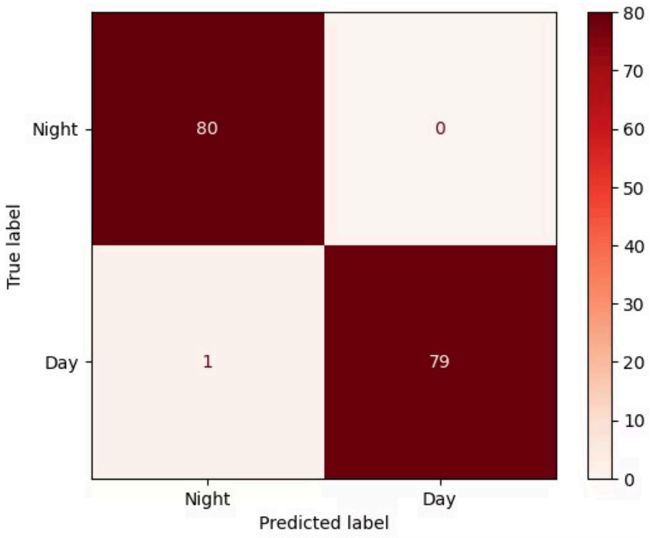
Loss at last epoch(13): 0.1179

Training
Accuracy Over Epochs



Accuracy at last epoch (13): 0.9688

Confusion matrix with a HEATMAP



It just predict a one day image as it night that's the only wrong answer

A Remarkable Achievement

99% Accuracy

Our CNN model has achieved an outstanding **99% accuracy** on the unseen testing dataset, demonstrating its exceptional ability to differentiate between day and night images with near-perfect precision.

This high performance confirms the robustness and reliability of our model, showcasing its strong generalization capabilities beyond the training data. It means that for every 100 test images, our model correctly classifies 99 of them, proving its effectiveness in real-world scenarios.

Bringing the Model to Life: Our Streamlit App

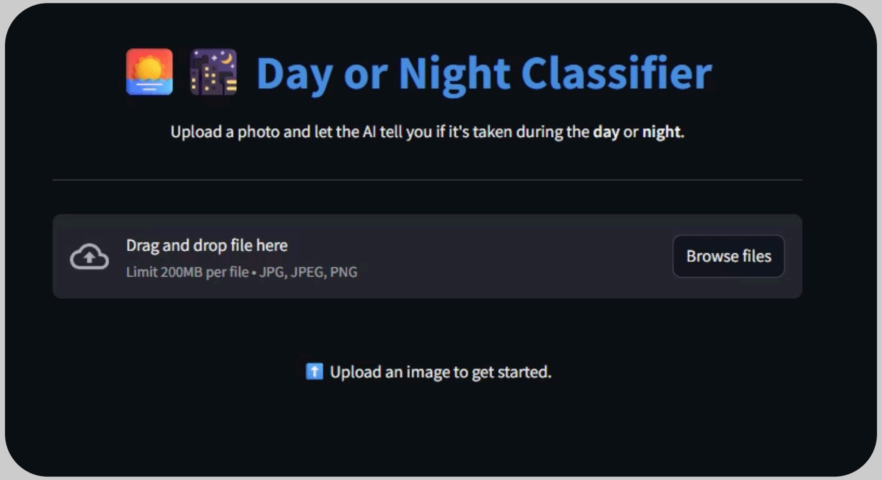
To make our day/night classification model accessible and easy to use, we've developed an interactive web application using Streamlit. This intuitive graphical user interface (GUI) allows anyone to upload an image and instantly see the model's prediction.

Simple Upload & Predict Interface

Users can effortlessly upload their own images through a user-friendly interface. The model processes the image in real-time, displaying whether it's day or night.

Day Prediction Label

GUI Interface



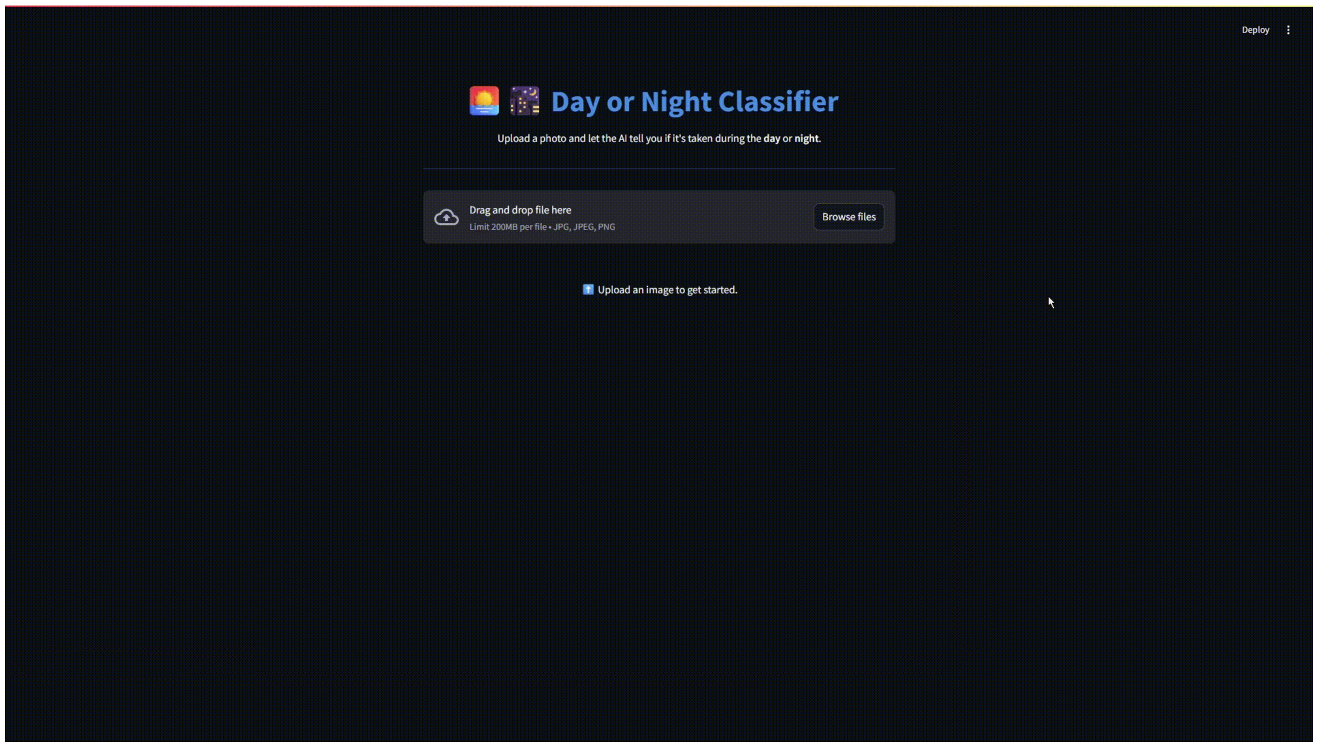
Night Prediction Label

Our Streamlit App Preview

Video of a Real-Trial while Using GUI

This video shows how the GUI lets users upload an image, then the model predicts and displays whether it's Day or Night.

Day



Night

Key Takeaways & Next Steps

As we conclude our journey into AI's vision, here are the key insights and what lies ahead:

1. **The Power of CNNs**

   Convolutional Neural Networks are exceptionally effective for image understanding, excelling at discerning subtle patterns in visual data.

2. **Achieving High Accuracy**

   Our custom CNN model achieved an outstanding 99% accuracy in day/night classification, demonstrating its reliability and generalization capabilities.

3. **AI for Real-World Applications**

   We've shown how powerful AI models can be deployed into user-friendly interfaces, making complex technology accessible for practical, everyday use.

4. **Future Directions**

   Next steps could involve expanding the dataset, exploring real-time video analysis, or integrating with smart home systems for adaptive lighting.

Thank You!

We extend our sincere gratitude for joining us on this exploration of AI and image classification. Your engagement and curiosity are greatly appreciated.

This project was brought to life by the dedicated efforts of:

Zeyad Tarek & Youssef Mahmoud

Explore the full code and project details on GitHub:

View the Project on GitHub