SolarGuard: Intelligent Defect Detection on Solar Panels

## Project Objective:

To automate solar panel defect detection using a Deep Learning CNN model and deploy it as a Streamlit web app.

## Part 1: Model Training Code (Colab)

# Install required libraries  
!pip install tensorflow opencv-python  
  
# Import libraries  
import tensorflow as tf  
from tensorflow.keras.models import Sequential  
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout  
from tensorflow.keras.preprocessing.image import ImageDataGenerator  
  
# Upload Dataset (Upload your Solar\_Panel\_Dataset.zip)  
from google.colab import files  
uploaded = files.upload()  
  
# Unzip dataset  
import zipfile  
zip\_path = "/content/Solar\_Panel\_Dataset.zip"  
with zipfile.ZipFile(zip\_path, 'r') as zip\_ref:  
 zip\_ref.extractall("/content/y")  
  
# Data generator with validation split  
datagen = ImageDataGenerator(rescale=1./255, validation\_split=0.2)  
  
train\_generator = datagen.flow\_from\_directory(  
 "/content/y",  
 target\_size=(224,224),  
 batch\_size=32,  
 class\_mode='categorical',  
 subset='training'  
)  
  
val\_generator = datagen.flow\_from\_directory(  
 "/content/y",  
 target\_size=(224,224),  
 batch\_size=32,  
 class\_mode='categorical',  
 subset='validation'  
)  
  
# CNN Model  
model = Sequential([  
 Conv2D(32, (3,3), activation='relu', input\_shape=(224,224,3)),  
 MaxPooling2D(2,2),  
 Conv2D(64, (3,3), activation='relu'),  
 MaxPooling2D(2,2),  
 Flatten(),  
 Dense(128, activation='relu'),  
 Dropout(0.5),  
 Dense(6, activation='softmax')  
])  
  
model.compile(optimizer='adam', loss='categorical\_crossentropy', metrics=['accuracy'])  
  
# Train Model  
model.fit(train\_generator, epochs=5, validation\_data=val\_generator)  
  
# Save model  
model.save("solar\_panel\_model.h5")  
  
# Download model to local  
from google.colab import files  
files.download("solar\_panel\_model.h5")

## Part 2: Streamlit App Code (app.py)

import streamlit as st  
import tensorflow as tf  
from tensorflow.keras.preprocessing import image  
import numpy as np  
import cv2  
import os  
  
st.set\_page\_config(page\_title="Solar Panel Defect Detection", page\_icon="☀️")  
st.title("☀️ SolarGuard: Solar Panel Defect Detection")  
st.write("Upload a solar panel image to detect its defect condition.")  
  
@st.cache\_resource  
def load\_model():  
 model = tf.keras.models.load\_model('solar\_panel\_model.h5')  
 return model  
  
model = load\_model()  
  
class\_labels = ['Bird-Drop', 'Clean', 'Dusty', 'Electrical-Damage', 'Physical-Damage', 'Snow-Covered']  
  
uploaded\_file = st.file\_uploader("📤 Upload a Solar Panel Image", type=["jpg", "jpeg", "png"])  
  
if uploaded\_file is not None:  
 file\_bytes = np.asarray(bytearray(uploaded\_file.read()), dtype=np.uint8)  
 img = cv2.imdecode(file\_bytes, 1)  
 img\_resized = cv2.resize(img, (224, 224))  
 img\_array = image.img\_to\_array(img\_resized)  
 img\_array = np.expand\_dims(img\_array, axis=0)  
 img\_array /= 255.0  
  
 st.image(img, caption="📷 Uploaded Image", channels="BGR")  
  
 if st.button("🔍 Predict Defect"):  
 prediction = model.predict(img\_array)  
 predicted\_class = class\_labels[np.argmax(prediction)]  
 confidence = np.max(prediction) \* 100  
  
 st.success(f"✅ Predicted Condition: \*\*{predicted\_class}\*\*")  
 st.info(f"🔍 Confidence Level: {confidence:.2f}%")  
  
st.write("🚀 Developed by Ajay | Capstone Project - SolarGuard ☀️")

## Part 3: How to Run This App

1. Install dependencies:  
  
pip install streamlit tensorflow opencv-python pillow numpy

2. Run app:  
  
streamlit run app.py

## Final Deliverables:

* solar\_panel\_model.h5 - Trained CNN model file
* app.py - Streamlit App Code
* Project\_Report.docx - Project summary and workflow documentation
* screenshots/ - Accuracy, Loss, Confusion Matrix images

## Business Use Case:

Automated solar panel defect detection with AI-powered image classification for smart solar farms — optimizing inspection, maintenance, and maximizing operational efficiency.

## Developed By:

Ajay | Capstone Project — SolarGuard ☀️