

Documentación Técnica del Sistema de Clasificación de Plantas con IA

Nombres:

Rafael Plata Angulo

Anette Leticia Robles Zamora

Título:

Documentación Técnica del Entrenamiento y Uso del Modelo de Clasificación de Imágenes

Fecha:

30/05/2025

Materia:

Inteligencia Artificial

1. Entrenamiento del Modelo - Generar_modelos.ipynb

El cuaderno Jupyter 'Generar_modelos.ipynb' se enfoca en la creación de un modelo de red neuronal convolucional (CNN) para la clasificación de plantas/frutas. A continuación, se detalla cada etapa crítica del proceso:

1.1 Carga de Librerías

Se importan bibliotecas como TensorFlow, NumPy y KerasCV. Estas son esenciales para manipulación de datos, creación de redes neuronales y entrenamiento eficiente.

1.2 Preparación del Dataset

El dataset está compuesto por imágenes organizadas por clase. Se utiliza ``image_dataset_from_directory`` para cargar imágenes en lotes, redimensionarlas y etiquetarlas automáticamente según el nombre de las carpetas.

Técnicas utilizadas:

- Redimensionamiento de imágenes a 28x28
- Normalización de valores de píxel entre 0 y 1
- Divisiones entre conjuntos de entrenamiento y validación

1.3 Arquitectura del Modelo

Se emplea una red CNN con las siguientes capas:

- Capas convolucionales con activación ReLU
- Capas de pooling para reducción dimensional
- Capas Dense para clasificación final

Se utiliza la función de pérdida ``sparse_categorical_crossentropy`` y el optimizador ``adam``.

1.4 Entrenamiento del Modelo

El modelo se entrena en varias épocas con retroalimentación en cada lote. Se registra la precisión y pérdida tanto en el conjunto de entrenamiento como validación.

1.5 Guardado del Modelo

Finalmente, el modelo se guarda como archivo ``h5`` para ser cargado posteriormente en la aplicación de predicción.

2. Aplicación de Clasificación - app2.py

Este archivo implementa una interfaz gráfica para usuarios utilizando Tkinter. El objetivo es permitir la predicción de imágenes cargadas o capturadas por cámara en tiempo real.

2.1 Carga del Modelo

Se carga el modelo previamente entrenado y almacenado en 'modelos/aprendiendomachinl.h5'. El modelo requiere compatibilidad con capas personalizadas, por lo que se importa 'keras_cv.models.ImageClassifier'.

2.2 Función de Preprocesamiento

La imagen ingresada se convierte a RGB, se redimensiona a 28x28 píxeles y se normaliza. Esto asegura la compatibilidad con los datos usados en el entrenamiento.

2.3 Inferencia del Modelo

La función 'predecir()' hace uso del modelo para calcular la clase más probable de la imagen. Se obtiene la clase con mayor probabilidad ('argmax') y su valor asociado.

2.4 Interfaz Gráfica (Tkinter)

La interfaz tiene las siguientes características:

- Botón para cargar imagen desde archivo
- Botón para capturar imagen con cámara
- Área para mostrar la imagen cargada
- Etiqueta para mostrar el resultado de la predicción

2.5 Lista de Clases

El modelo puede identificar las siguientes clases:

aloevera, banana, bilimbi, cantaloupe, cassava, coconut, corn, cucumber, curcuma, eggplant, galangal, ginger, guava, kale, longbeans, mango, melon, orange, paddy, papaya, peper chili, pineapple, pomelo, shallot, soybeans, spinach, sweet potatoes, tobacco, waterapple, watermelon

3. Flujo General del Sistema

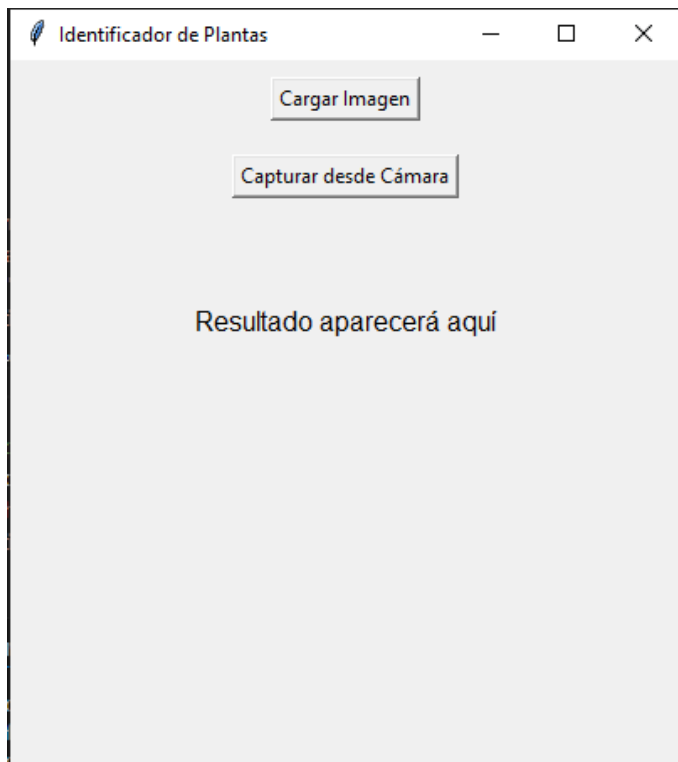
El flujo de trabajo del sistema es el siguiente:

1. Entrenamiento del modelo con imágenes etiquetadas
2. Guardado del modelo en archivo .h5
3. Carga del modelo desde una aplicación gráfica
4. El usuario carga o captura una imagen
5. Se realiza preprocesamiento a la imagen
6. El modelo predice la clase de la imagen
7. Se muestra el resultado al usuario

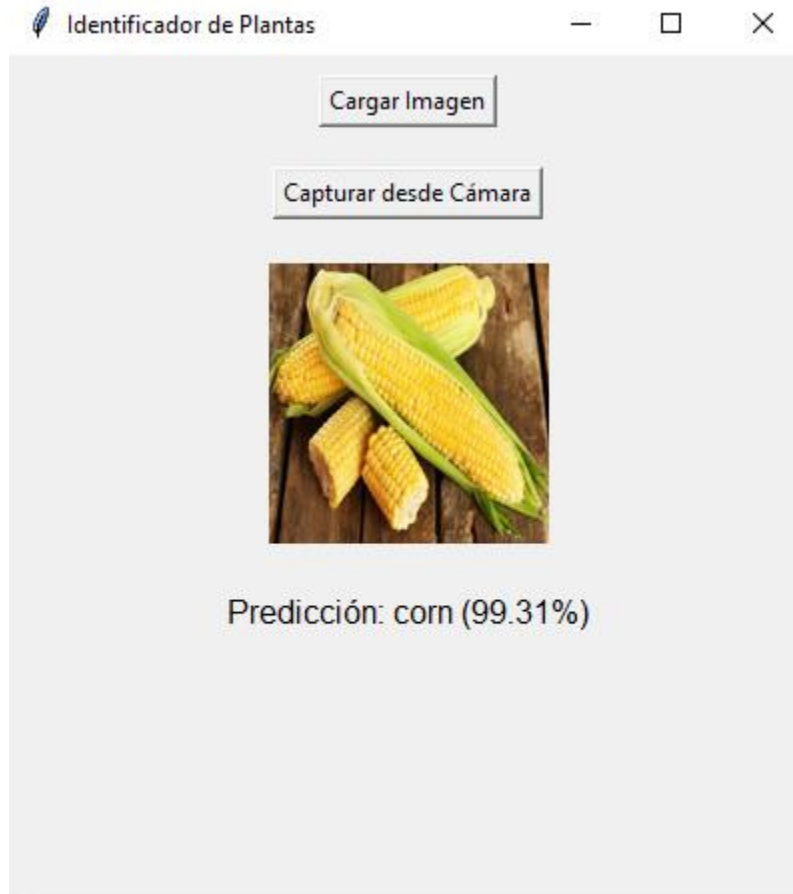
Este flujo permite una separación clara entre la fase de desarrollo del modelo y su uso productivo.

4. capturas del funcionamiento:

Menu principal



Funcionamiento con imagenes cargadas



Cargar Imagen

Capturar desde Cámara



Predicción: orange (99.29%)

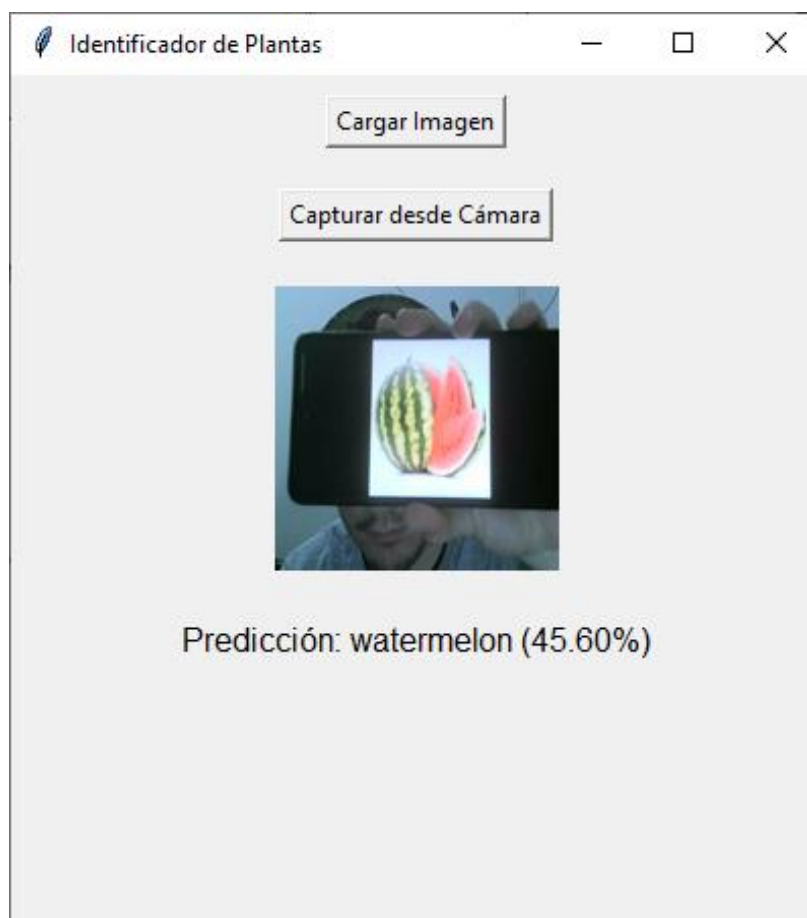
Cargar Imagen

Capturar desde Cámara



Predicción: watermelon (87.95%)

Funcionamiento con camara





Identificador de Plantas



Cargar Imagen

Capturar desde Cámara



Predicción: orange (99.96%)

Cargar Imagen

Capturar desde Cámara



Predicción: corn (83.58%)