**🗺️ VISIÓN GENERAL DEL FLUJO**

El proyecto se divide en 3 grandes fases:

**🔸 Fase 1: Preparación del Dataset**

**🔸 Fase 2: Entrenamiento del modelo en Google Colab**

**🔸 Fase 3: Uso del modelo entrenado en VSCode con OpenCV**

**🔸 FASE 1: Preparación del Dataset**

| **Paso** | **Tecnología** | **Objetivo** |
| --- | --- | --- |
| 1.1 | Kaggle (descarga manual o API) | Obtener el dataset de manos anotadas en CSV (labels.csv). |
| 1.2 | Python + Pandas + os + PIL | Cargar labels.csv, procesar las rutas y validar imágenes. |
| 1.3 | Python script personalizado | Convertir el CSV a TFRecord, formato que entiende TensorFlow. |
| 1.4 | TensorFlow (label map) | Crear el archivo label\_map.pbtxt que mapea la clase hand a un ID. |

**🔸 FASE 2: Entrenamiento en Google Colab (con GPU de Colab Pro)**

| **Paso** | **Tecnología** | **Objetivo** |
| --- | --- | --- |
| 2.1 | Google Colab + Git | Instalar TensorFlow Object Detection API desde GitHub. |
| 2.2 | Colab + TensorFlow 2.x | Configurar entorno, Protobuf, dependencias, etc. |
| 2.3 | TensorFlow Model Zoo | Descargar un modelo preentrenado (ej. SSD MobileNet v2). |
| 2.4 | Python | Configurar el archivo .config del modelo: ruta a TFRecord, batch size, learning rate... |
| 2.5 | TFOD API (model\_main\_tf2.py) | Entrenar el modelo con tu dataset. |
| 2.6 | TFOD API (exporter\_main\_v2.py) | Exportar el modelo entrenado a un directorio listo para usar. |
| 2.7 | Colab + gdown o zip | Descargar el modelo entrenado desde Colab a tu PC (para usarlo en VSCode). |

**🔸 FASE 3: Uso en VSCode con OpenCV y TensorFlow**

| **Paso** | **Tecnología** | **Objetivo** |
| --- | --- | --- |
| 3.1 | VSCode + TensorFlow + OpenCV | Cargar el modelo entrenado en tu PC (formato SavedModel). |
| 3.2 | OpenCV (cv2.VideoCapture) | Acceder a tu webcam o una imagen/video. |
| 3.3 | TensorFlow + NumPy | Procesar los frames y hacer inferencias con el modelo. |
| 3.4 | OpenCV (cv2.rectangle, cv2.putText) | Dibujar los cuadros delimitadores (bounding boxes) sobre la mano detectada. |
| 3.5 | Python Script final | Integrar todo en una app que detecte y marque manos en tiempo real o desde imágenes. |