EfficientSAM: Leveraged Masked Image Pretraining for Efficient Segment Anything

Isha Kumar
Department of CSE
Indian Institute of Technology Hyderabad
CS24MTECH14001

A minimal guide to get EfficientSAM up and running in VS Code, including environment setup, dependency installation, dataset download, and running the main script.

Contents

1 Prerequisites **2**

2 Project Setup **2**

3 Install Dependencies **2**

4 Kaggle Dataset Configuration **2**

5 Download & Prepare Datasets **3**

6 Usage **3**

1 Prerequisites

- **Python** 3.8 or higher
- **VS Code** (or any Python-capable IDE)
- **Git**
- **Kaggle account** with an API token

2 Project Setup

1. **Clone the repository**

```
git clone https://github.com/yformer/EfficientSAM.git
cd EfficientSAM
```

2. **Create & activate a virtual environment**

```
python -m venv .venv
# Windows
.venv\Scripts\activate
# macOS/Linux
source .venv/bin/activate
```

3. **Open in VS Code**

```
code .
```

Be sure to select the `.venv` interpreter in the bottom status bar.

3 Install Dependencies

```
pip install torch torchvision gradio opencv-python pandas matplotlib
pip install -e .           % Installs EfficientSAM in editable mode
pip install kagglehub      % Optional helper for Kaggle interactions
```

4 Kaggle Dataset Configuration

We'll pull COCO 2017 and LVIS validation sets via the Kaggle API.

1. **Create your `.kaggle` folder**

```
mkdir -p ~/.kaggle
```

2. **Add your API token**

   - In your Kaggle account, go to `Account → Create New API Token`.
   - Download the resulting `kaggle.json` and place it in your project root.

```
mv kaggle.json ~/.kaggle/
chmod 600 ~/.kaggle/kaggle.json
```

5 Download & Prepare Datasets

```
# (Optional) Verify COCO datasets on Kaggle
kaggle datasets list -s coco

# Create a data directory
mkdir -p data

# Download & unzip COCO 2017
kaggle datasets download -d awsaf49/coco-2017-dataset -p data --unzip

# Download & unzip LVIS v1 validation
kaggle datasets download -d alexanderyyy/lvis-v1-valid -p data --unzip
```

6 Usage

- **For Training:** From the project root (with the virtual environment active):

```
python train_eff.py --coco_path /home/ritesh/Desktop/Rishabh/data/
    ↪ coco2017 --batch_size 4 --epochs 40 --max_samples 1000
```

  where path to your coco2017 folder is */home/ritesh/Desktop/Rishabh/data/coco2017* and *train_eff.py* is your training python script.

- **For Testing:** From the project root (with the virtual environment active):

```
python EfficientSAM.py
```

  Or, using an absolute path to your test python script *EfficientSAM.py*:

```
python /home/isha/Desktop/Rishabh/EfficientSAM.py
```