DeepVision Crowd Monitor: AI for Density Estimation and Overcrowding Detection

Sanjaykumar

Table of Contents

1. Project Description

The DeepVision Crowd Density Estimation System is a computer vision project designed to analyze and estimate the number of people present in crowded scenes. The objective of this milestone is to preprocess crowd images and generate accurate density maps using annotated ground truth points. These density maps serve as learning targets for deep learning models such as MCNN, CSRNet, and other crowd-counting architectures.

The implemented workflow includes inspecting dataset files, loading images, extracting head annotations from .mat files, generating Gaussian-based density maps, and visualizing the results.

2. Dataset Used

The project uses the ShanghaiTech Crowd Counting Dataset, which contains

two parts:

• Part A: Highly dense crowds

• Part B: Medium-density crowds

For each image in the dataset:

• A .jpg image file is provided.

• A .mat ground truth file contains the coordinates of each person's head

3. Environment Setup

The project is implemented in Python usingVS Code. The key libraries include:

- **NumPy** – numerical array operations
- **OpenCV (cv2)** – image loading and conversion
- **SciPy (scipy.io)** – reading .mat annotation files
- **Matplotlib** – visualization of images and density maps

- **PyTorch (torch, Dataset class)** – organizing images and targets through a custom dataset class

These tools collectively enable structured data loading, annotation extraction, and density map creation.

4. Data Exploration

Before initiating preprocessing, the dataset was explored to verify the directory structure, ensure annotation availability, and inspect sample files.

Verifying Dataset Folder Paths

- **Checking whether the dataset exists in the file system**
  The notebook accessed the dataset path using Python's os module and verified that the folder structure was correctly configured.
- **Viewing sample image paths**
  All .jpg image files in the dataset were listed and sorted. This ensured that the images were correctly named and accessible.
- **Confirming ground-truth file availability**
  Corresponding .mat files containing the head annotations were also listed. The number of image files and annotation files were compared to ensure that each image had a matching .mat file.

5. Data Preprocessing

The preprocessing pipeline converts raw images and annotation data into Gaussian-based density maps. These maps represent crowd distribution spatially and are fundamental supervision signals for crowd-counting models.

5.1 Image Loading

Images are loaded using OpenCV:

- Read in **BGR format**
- Converted into **RGB** for accurate visualization and processing

This ensures consistent color representation and prepares the image for density map overlay.

5.2 Ground Truth Extraction

Each .mat ground truth file contains:

annPoints = [(x1, y1), (x2, y2), ...]

Each coordinate corresponds to a person's head location.

Using SciPy, these coordinates were:

- Extracted,
- Converted to NumPy arrays, and
- Passed into the density map creation function.

This step transforms MATLAB-based annotation structures into a Python-friendly format.

5.3 Density Map Generation

Density maps were generated using Gaussian filtering. The process involved:

1. Creating an empty zero matrix with the same height and width as the original image.
2. Placing a value of **1** at each annotated head location.
3. Applying a **Gaussian filter** over the matrix to spread each head point into a smooth blob.

This produces a heatmap-like representation where:

5.4 Density Map Visualization

Using Matplotlib, the notebook visualized:

- The original RGB image
- The corresponding density map

Heatmaps were displayed to verify that:

- Annotations were correctly mapped
- Gaussian spread aligned with head positions

- Density intensity increased for regions with more people

This serves as a quality assurance step in the preprocessing workflow.

5.5 Bulk Processing of Images

A loop was implemented to:

- Iterate through multiple images
- Load each image and its ground truth
- Generate density maps
- Optionally save processed outputs

This demonstrates the complete preprocessing pipeline applied at scale and prepares the dataset for the training phase.

6. Conclusion

In this milestone, the preprocessing and exploration stages of the DeepVision crowd-counting system were successfully completed. The dataset structure was verified, sample images and annotations were inspected, and a complete pipeline for density map generation was implemented.