**Curve Regularization Project**

**Overview**

This project aims to identify, regularize, and beautify curves in 2D Euclidean space. The main tasks include recognizing regular geometric shapes, identifying symmetry, and completing incomplete curves. The output is visualized using SVG and PNG formats.

**Requirements**

* Python 3.x
* Required Python packages:
  + numpy
  + matplotlib
  + svgwrite
  + cairosvg

You can install the required packages using pip:

bash

Copy code

pip install numpy matplotlib svgwrite cairosvg

**Project Structure**

* curve\_regularization.py: The main script that performs curve regularization, symmetry identification, and curve completion.
* examples/: Directory containing example CSV files for testing.
* output/: Directory where generated SVG and PNG files are saved.
* README.md: This file.

**How to Run**

1. **Navigate to the Project Directory:**

Open a terminal or command prompt and change to the project directory:

bash

Copy code

cd path/to/your/project/problems

1. **Run the Script:**

Execute the curve\_regularization.py script using Python:

bash

Copy code

python curve\_regularization.py

1. **Check Outputs:**

After running the script, you will find the following:

* + **Plots**: Displayed in a window showing the curves from the CSV files.
  + **SVG Files**: Saved in the output/ directory. These files contain vector graphics of the curves.
  + **PNG Files**: Also saved in the output/ directory. These files are rasterized versions of the SVG files.

**Files**

* **Input CSV Files:** CSV files should be placed in the same directory as the script. Each CSV file should contain polyline data with points defining the curves.
* **Output Files:** Generated SVG and PNG files will be saved in the output/ directory.

**Example Usage**

**Input**

Place your CSV files in the project directory. Example filenames include:

* isolated.csv
* frag0.csv
* frag1.csv
* occlusion1.csv
* occlusion2.csv

**Output**

The script will generate SVG and PNG files corresponding to each input CSV file, showing the visualized curves.

**Code Details**

* **read\_csv(csv\_path)**: Reads the CSV file and converts it to a list of polyline paths.
* **plot(paths\_XYs)**: Plots the polyline paths using Matplotlib.
* **polylines2svg(paths\_XYs, svg\_path)**: Converts the polyline paths to SVG and PNG formats.

**Troubleshooting**

* **File Not Found Error:** Ensure that the CSV files are in the correct directory and are named correctly.
* **Missing Packages:** Install missing packages using pip.