

Setup and Run 2D Image Generation Program

Caden Miller

2024-06-04

Step 1: Download Files from Google Drive

1. **Navigate to Shared Google Doc:** Go to the repository where the project files are hosted.
2. **Download the Repository:** Download the repository as a ZIP file.
3. **Extract the ZIP File:** Open the zipped folder and extract its contents.
4. **Copy to Desired Location:** Copy the extracted folder to your desired location, e.g., "Documents".

Step 2: Install VSCode

1. **Download VSCode:** Go to the [Visual Studio Code website](#) and download the installer for your operating system.
2. **Install VSCode:** Run the downloaded installer and follow the installation instructions.

Step 3: Install Python

1. **Download Python:** Go to the [Python website](#) and download the latest version of Python.
2. **Install Python:** Run the downloaded installer. During installation, make sure to check the box that says "Add Python to PATH".

Step 4: Set Up VSCode

1. **Open VSCode:** Launch Visual Studio Code.
2. **Install Python Extension:**
 - Click on the Extensions view icon on the Sidebar (or press Ctrl+Shift+X).
 - Search for "Python".
 - Install the official Python extension provided by Microsoft.
3. **Select Interpreter and Create Virtual Environment:**
 - Press Ctrl+Shift+P to open the command palette.
 - Type "Python: Select Interpreter" and select the interpreter from the .venv you created.
 - To create a virtual environment, open a terminal in VSCode (Terminal > New Terminal) and run:

```
python -m venv .venv
```

- Activate the virtual environment:

- **Windows:**

```
.venv\Scripts\activate
```

- **macOS/Linux:**

```
source .venv/bin/activate
```

4. **Open Project Folder:**

- In VSCode, open the folder containing your Python files by going to File > Open Folder and selecting your project directory.

Step 5: Navigating Inside the Virtual Environment

1. **Open Terminal:** In VSCode, open a new terminal by going to Terminal > New Terminal.
2. **Verify Virtual Environment:** Ensure that your terminal prompt starts with (venv) indicating that the virtual environment is activated.

Step 6: Install Dependencies from requirements.txt

1. **Activate Virtual Environment:** Ensure your virtual environment is activated (Step 5).
2. **Run pip to Install Dependencies:** Run the following command in the terminal to install all dependencies listed in the requirements.txt file:

```
sh pip install -r requirements.txt
```

Step 7: Run GUI File

1. **Run the GUI Script:** In the terminal, run the GUI script by executing:

```
sh python gui_image_generator.py
```

Using the GUI

- **View the README:** Click the “Open README” button to open the README.pdf.
- **Select JSON File:** Click “Browse...” to select a JSON file.
 - **Enter Number of Images:** Enter the number of images to create.
 - **Run Image Generation:** Click “Run Image Generation” to start the process.
- **Create a New JSON from Template:** Fill in the form with parameters and click “Create JSON File” to generate a new JSON file.

Following these steps will ensure you have everything set up correctly and can run the program successfully. # GUI Help

Simplified Steps to Modify the Template in the GUI

1. **Open the GUI:**
 - Run the GUI script by executing:

```
python gui_image_generator.py
```

2. **Select JSON File:**
 - Click the “Browse...” button next to “Select JSON File” to choose an existing JSON file.
3. **Modify Parameters:**
 - **Save Directory:** Click the “Browse...” button next to “Save Directory” to select the directory where the generated images will be saved.
 - **Save Name:** Enter the desired base name for the output files in the “Save Name” field.
 - **Background:** Enter the color in RGBA format, width, and height for the background in their respective fields.
 - **Scale:** Select the distribution type and enter the distribution parameters.
 - **Rotation:** Select the distribution type and enter the distribution parameters.
 - **Color:** Select the distribution type and enter the color channel parameters.
 - **Centers:** Enter the minimum distance (r) and attempts (k) for the poisson disc distribution.
 - **Find Images:** Enter the name and depth of the required shapes.
 - **Excluded Images:** Enter the names of the shapes to exclude from random selection.
 - **Size:** Adjust the size of the images by a scalar multiple.
 - **Opacity:** Adjust the opacity of the image
4. **Create JSON File:**
 - Fill in the form with the desired parameters and click “Create JSON File” to generate a new JSON file.

Parameter Details

1. **Save Directory (save_dir):**
 - **Purpose:** Specifies where the generated images will be saved.
 - **Input Example:**
 - Relative Path: "save_dir": "Image_Output/"
 - Absolute Path: "save_dir": "C:/Users/YourName/Documents/Project/Image_Output/"
2. **Save Name (save_name):**
 - **Purpose:** Defines the base name for the output files.
 - Save Name: <save_name>.png
3. **Background (background):**
 - **Purpose:** Sets the color (in RGBA format), width, and height of the background image.
 - **Inputs:**
 - **Color:** "color": [245, 245, 220, 255] (RGBA format)
 - **Width:** "width": 1080

- **Height:** "height": 1080
4. **Scale (scale):**
 - **Purpose:** Determines the size of the generated shapes as a proportion of the background height [0-1].
 - **Inputs:**
 - **Distribution:** "dist": "T" (Triangular) or "dist": "U" (Uniform)
 - **Parameters:**
 - Triangular: "params": [0.15, 0.2, 0.25] (Lower Bound, Peak, Upper Bound)
 - Uniform: "params": [0.2, 0.3] (Lower Bound, Upper Bound)
 5. **Rotation (rotation):**
 - **Purpose:** Defines the rotation angle of shapes in degrees [0-360].
 - **Inputs:**
 - **Distribution:** "dist": "U" (Uniform) or "dist": "T" (Triangular)
 - **Parameters:**
 - Triangular: "params": [-15, 0, 15] (Lower Bound, Peak, Upper Bound)
 - Uniform: "params": [0, 360] (Lower Bound, Upper Bound)
 6. **Color (color):**
 - **Purpose:** Specifies the color distribution for shapes.
 - **Distribution Options:**
 - **Uniform (U):** Using the uniform distribution, each channel of red, green, blue, and alpha will be independently distributed using their own upper and lower bounds. See example:

Color Distribution: U
Color 1 Args: [10, 255]
Color 2 Args: [10, 255]
Color 3 Args: [10, 255]
Color 4 Args: [140, 170]
 - **Triangular (T):** This distribution is input very similar to the uniform one. Each channel is given the lower bounds, peak, and upper bounds. See example:

Color Distribution: T
Color 1 Args: [50, 70, 90]
Color 2 Args: [200, 210, 255]
Color 3 Args: [0, 50, 60]
Color 4 Args: [140, 155, 170]

- **Mode (M)** This is a distributed mode to allow for specific colors to be chosen. The inputs are each a "mode" that could be randomly selected. This supports 1 or more mode inputs. See the example that will generate shapes that are all the same color:

Color Distribution: M

Color 1 Args: [160,195, 93, 155]

Color 2 Args:

Color 3 Args:

Color 4 Args:

This will generate images with 2 colors:

Color Distribution: M

Color 1 Args: [160,195, 93, 155]

Color 2 Args: [30, 150, 38, 140]

Color 3 Args:

Color 4 Args:

7. **Centers (centers):**

- **Purpose:** Sets parameters for the poisson disc distribution to generate center points.
- **Inputs:**
 - **Minimum Distance:** "r" (the smaller the value, the more dense the image)
 - **Attempts:** "k" (32 is a good standard value for this, we want it to be big but any increase will substantially increase computation time)

8. **Find Images:**

- **Purpose:** Specifies shapes to find and their placement depth. This can be left blank if you don't want a target image, or multiple target images can be selected by using comma separated values.
- **Inputs:**
 - **Find Image Name:** "name": "bat01.jpg"
 - **Find Image Depth:** "depth": 0.4

9. **Excluded Images:**

- **Purpose:** Lists shapes to exclude from random selection. Should usually be the variants of your target image.
- **Input Example:** "bat01.jpg,bat02.jpg,bat03.jpg,etc."

10. **Size:**

- **Purpose:** Modify the size of the images by a scalar multiple. This parameter should only be adjusted if you want to adjust the size of all images. If you wish to change the distribution of sizes, you should use the scale parameter.

11. **Opacity:**

- **Purpose:** Adjust the parameter for the opacity of the images. 1 is completely solid. .7 seems to be a good value.

RGBA Color Codes

Here is a link to find the RGBA codes which you'll need to enter to change the background cover.

<https://rgbacolorpicker.com/>

Some Notes

1. A newly generated file (.png or .json) that has the same name as another file in the output folder will be overwritten with no warning.
2. Since the random sampling is done during the execution of this code, you can rerun the same code with the same .json files and it will output new images. This is helpful when by chance the image that you are looking for is placed so close to the edge that it is not possible to find.