

# SVGEditBench: A Benchmark Dataset for Quantitative Assessment of LLM’s SVG Editing Capabilities

## Supplementary Material

### 6. Structure of the Prompts

Here, we show how we composed the prompts we input to the LLMs in the proposed benchmark dataset. The following prompt is an example of the **Change Color** task.

An example prompt of the **Change Color** task

```
The following code is the SVG code for the emoji 'movie camera'. Please generate an SVG code that changes the part of the emoji with a #31373D color to red.
```

```
```svg
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 36 36"><path fill="#31373D" d="M32
21v1h-2v-1c0-.446-.09-.867-.225-1.268 2.446-.757 4.224-3.038 4.224-5.733
0-3.314-2.687-6-6-6-1.603 0-3.055.632-4.131 1.656C23.241 6.433 20.405 4 17 4c-3.866
0-7 3.134-7 7 0 2.551 1.369 4.777 3.409 6H13c-2.209 0-4 1.791-4 4H8l-6-4H1v14h116-4
h1v2c0 2.209 1.791 4 4 4h13c2.209 0 4-1.791 4-4v-3h2v1h3v-6h-3z"/><path fill="#66757F"
F" d="M22 11c0 2.761-2.239 5-5 5s-5-2.239-5-5 2.239-5 5-5 2.238 5 5z"/><circle
fill="#CCD6DD" cx="17" cy="11" r="2"/><circle fill="#66757F" cx="27.999" cy="14" r
="4"/><circle fill="#CCD6DD" cx="27.999" cy="14" r="2"/><path fill="#8899A6" d="M17
20h10v10H17z"/><path fill="#31373D" d="M19 22h6v6h-6z"/><circle fill="#8899A6" cx
="12.999" cy="28" r="2"/></svg>
```
```

```
Please respond in the following format. Only return the SVG code.
```

```
```svg
<svg>...</svg>
```
```

The **red part** at the beginning of the prompt describes the editing task. The description includes the emoji’s name (**purple part**). We show the name to specify what the following SVG code represents. The name function of the `unicodedata`<sup>3</sup> Python standard library returns the name of an emoji as defined in Unicode. For the **Change Color** and the **Set Contour** tasks, we must indicate the area to edit. We randomly choose a fill color from the ones in the SVG code and show the color in the **first green part** of the prompt. The LLM should edit the part filled with the chosen color. We also randomly select a target color among red (`#FF0000`), green (`#00FF00`), blue (`#0000FF`), yellow (`#FFFF00`), cyan (`#00FFFF`), magenta (`#FF00FF`), white (`#FFFFFF`), and black (`#000000`). The **second green part** denotes the target color.

The **cyan part** is the SVG code before the edit. The code is surrounded by ````svg` and ````` to explicitly indicate to the LLM that it is the code part. Finally, the **orange part** defines the output format. We also state that the LLM should only return the SVG code to suppress explaining the content of the image or the procedures of editing. We can automatically process the LLM’s output by adding this paragraph. Namely, we regard the text surrounded by ````svg` and ````` as the LLM’s output.

<sup>3</sup><https://docs.python.org/ja/3/library/unicodedata.html>

## 7. Example of Prompts for Each Prompt

This section shows the prompts used in each task except for the **Change Color** task, which we already demonstrated. These prompt examples are for the same emoji as the previous example. Note that we omitted the **cyan part** and the **orange part** since these parts are the same as earlier.

### Set Contour

The following code is the SVG code for the emoji 'movie camera'. Please generate an SVG code that draws a black line around the part of the emoji with a #66757F color.

### Compression

The following code is the SVG code for the emoji 'movie camera'. Please generate a more compact SVG code that represents the same emoji.

### Upside-Down

The following code is the SVG code for the emoji 'movie camera'. Please flip this emoji upside down.

### Transparency

The following code is the SVG code for the emoji 'movie camera'. Please make this emoji transparent by half.

### Crop to Half

The following code is the SVG code for the emoji 'movie camera'. Please trim the right half and keep the left half.