# Trimap Matting

Scuola d'Arti e Mestieri di Trevano (SAMT)
Documentation

Paolo Bettelini

# Contents

# 1 Introduction

## 1.1 Abstract

## 1.2 Information

This is a project of the Scuola Arti e Mestieri di Trevano (SAMT) under the following circumstances

- **Section**: Computer Science
- **Year:** Fourth
- **Class:** Progetti Individuali
- **Supervisor:** Geo Petrini
- **Title:** Trimap Matting
- **Start date**: 2022-09-29
- **Deadline**: 2022-12-07

and the following requirements

- **Documentation**: a full documentation of the work done
- **Diary**: constant changelog for each working session
- **Source code**: source code of the project

All the source code and documents can be found at https://github.com/paolobettelini/trimap-matting [1].

# 2 Requirements

| **Req-00** | |
|---|---|
| **Name** | CLI tool |
| **Priority** | 1 |
| **Version** | 1.1 |
| **Notes** | none |
| **Description** | A CLI tool to execute background removal must be developed |
| **Subrequirements** | |
| **Req-00__0** | The target image must be specified |
| **Req-00__1** | The trimap image can be specified |
| **Req-00__2** | The soft mask can be specified |
| **Req-00__3** | Either the soft mask or the trimap must be specified |
| **Req-00__4** | The program can save the generated background mask |
| **Req-00__5** | The program can remove the background and replace it with an image |
| **Req-00__6** | The program can remove the background and fill it with a color |
| **Req-00__7** | The program can remove the background and leave it transparent |

| **Req-01** | |
|---|---|
| **Name** | Image formats |
| **Priority** | 1 |
| **Version** | 1.0 |
| **Notes** | none |
| **Description** | Multiple image formats must be supported |
| **Subrequirements** | |
| **Req-00__0** | The JPG format must be supported |
| **Req-00__1** | The PNG format must be supported |
| **Req-00__2** | The WebP format must be supported |

| **Req-02** | |
|---|---|
| **Name** | Size check |
| **Priority** | 1 |
| **Version** | 1.0 |
| **Notes** | none |
| **Description** | The executable must assert that the target image and trimap are of the same size |

| **Req-03** | |
|---|---|
| **Name** | GUI |
| **Priority** | 1 |
| **Version** | 1.0 |
| **Notes** | none |
| **Description** | A GUI application must be developed in other to interact with the program features |

# 3 To use

```
let output = log!(
    "Generating soft mask",
    args.verbose,
    matting::generate_mask(&target, &trimap)? // heavy lifting
);
```

```
<div>
    <h1>Title</h1>
</div>
```

```
function ciao() {
    console.log(extendedchars"extendedcharseeextendedchars");
}
```

```
# asd
hello = "valore"
```

```
server {
    server_name = _;
}
```

# 4 CLI

## 4.1 Compilation

The executable can be compiled using the `cargo` package manager.

```
    $ cd matting-cli
    $ cargo build --release
```

This will generate an executable (`matting-cli`) in `./target/debug`. In order to make this executable globally available we can move it into a folder in the $PATH enviroment variable, such as `/usr/bin`. We may also modify the executable file name to change its invokation name.

```
    $ sudo mv target/release/matting-cli /usr/bin/
```

We executable can now be invoked by just writing

```
    $ matting-cli
```

## 4.2 Usage

The followings shows the output of the command upon setting the `--help` or `-h` flag.

```
Matting CLI

Usage: matting-cli [OPTIONS] --target <TARGET>
                   <--mask <MASK>|--trimap <TRIMAP>>

Options:
  -i, --target <TARGET>        Target image
      --mask <MASK>            Background mask image
      --trimap <TRIMAP>        Trimap image
      --save-mask <SAVE_MASK>  Save mask path
  -o, --output <OUTPUT>        Output image
  -f, --fill <FILL>            Fill background action
  -t, --transparent            Transparent background action
  -r, --replace <REPLACE>      Replace background action
      --verbose                Verbose flag
  -h, --help                   Print help information
  -V, --version                Print version information
```

The `--target` parameter specifies the image on which the operation needs to be applied. This parameter is mandatory.

The `--trimap` parameter specifies the trimap image which will be used to generate the alpha mattes.

The `--mask` parameter specifies the image containing the alpha mattes to use.

The parameter `--trimap` and `--mask` are mutually exclusive and one of them is mandatory.

The advantage of using `--mask` over `--trimap` is that the alpha mattes are already given rather than having to be computed. This can save lots of computational times. The alpha mattes image can be saved on the file system by specifying the `--save-mask` parameter.

There are 3 different operations that can be applied to the background of the result: `--transparent`, `--replace` or `--fill`. These operations are mutually exclusive and if one is specified, the `--output` parameter must also be set to specify the path where the resulting image will be saved. Similarly, if the `--output` parameter is set an operation must also be specified.

**Note:** the argument of `--color` can be any valid CSS color. See [**csscolors**] for the documentation.

The `--verbose` flag is optional and will print additional information about what the program is doing and the elapsed time of each operation.

### 4.3 Examples

The following command generate a mask of the alpha mattes given a trimap.

```
$ matting-cli -i target.jpg --trimap trimap.png
    --save-mask mask.png
```

The following shows the output of the program when the `--verbose` flag is set. This command computes the alpha mattes given a trimap, then it saves the generated mask, fills the background of the image with the color red and then saves the result.

```
$ matting-cli -i target.jpg --trimap trimap.png
    --save-mask mask.png -o out.png --fill red --verbose

Reading target image... Done! [4.021485ms]
Reading trimap image... Done! [1.976477ms]
Generating soft mask... Done! [7.104532642s]
Reading target image... Done! [178.884124ms]
Saving soft mask... Done! [509.050896ms]
Filling background with color... Done! [117.90393ms]
Saving output... Done! [954.085622ms]
```

# 5 Trimap Matting

Matting is a technique used to extract an object from an image. Trimap matting is a term used to refer to the process of generating alpha mattes[2] for an object in an image given an initial approximation of its borders.

The goal of this process is to determine how much each pixel of a target image is part of the object that needs to be extracted. This means that given a pixel $P_{x,y}$ we want to find a value $\alpha \in \mathbb{R}$ such that

$$P_{x,y} = \alpha F_{x,y} + (1-\alpha)B_{x,y}, \quad \alpha \in [0;1]$$

where $F$ represents the foreground color and $B$ represents the background color at a given pixel.

Note that the multiplicative operator here is the scalar vector multiplication. This is because the pixels are represented by a vector of values, usually $\mathbb{R}^3$ or $\mathbb{R}^4$ (for transparent images).

The trimap is an approximation of the alpha mattes. The black dye represents background-only space, the white dye represents foreground-only space whilst the gray one delimits the distinction between the two.

Here are some examples using an image of a plant.

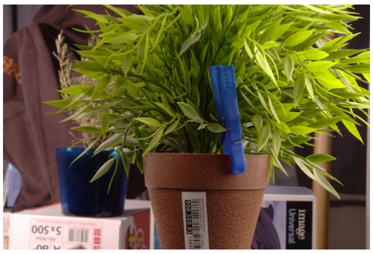

Figure 1: Plant Image

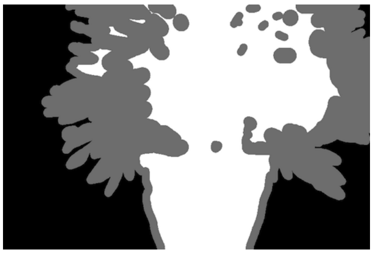

Figure 2: Plant Trimap

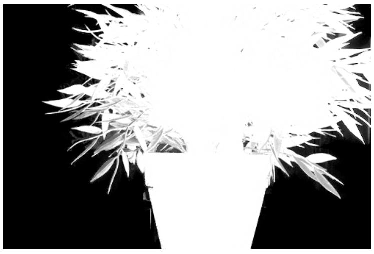

Figure 3: Plant Soft Mask

Once the alpha mattes are generated (soft mask) we can use them to extract the object from the target image. Thereore, we can remove the background behind the object, fill the background with a color, replace it with another image or leave it transparent.

Given a target image with pixels $T_{x,y}$, the alpha mattes $\alpha_{x,y}$ and the pixels of the replacement image $R_{x,y}$ we can compute the pixels of the output image $T'_{x,y}$ as follows

$$T'_{x,y} = \alpha_{x,y} \cdot T_{x,y} + (1-\alpha_{x,y}) \cdot R_{x,y}$$

If we want to replace the background with a color we can consider. $R = (r,g,b)^t$
If we just want to leave the background transparent, meaning $T' \in \mathbb{R}^4$, we have

$$T'_{x,y} = \alpha_{x,y} \cdot T_{x,y}$$

and the alpha value for $T_{x,y}$ is set to $\alpha_{x,y}$

## 6 Opencv

OpenCV[**opencv**] is a library for computer vision. It contains a large amount of tools, from GUIs, video analysis, machine learning, object detection, image processing and many more[**opencvdoc**]. The library also contains a module about *alpha matting*, contains a function to find generate alpha mattes given a trimap [**opencvalphamatting**].

Testo

# List of Figures

# References

[1] Paolo Bettelini. *trimap-matting*. 2022. URL: https://github.com/paolobettelini/trimap-matting.

[2] Wikipedia contributors. *Matte (filmmaking) — Wikipedia, The Free Encyclopedia.* https://en.wikipedia.org/w/index.php?title=Matte_(filmmaking)&oldid=1127451547. [Online; accessed 18-January-2023]. 2022.