Segmentation challenge

Mila - Safe AI for Humanity (SAIFH) Team

1 Project definition

The goal of the project is to disentangle the Mila logo from any image. To do so, we ask you to use a deep learning approach for semantic segmentation. You are expected to propose, train, and evaluate a model, and then hand in the code you developed, your best model checkpoint, as well as a report describing your approach and findings.

If you do not have access to a GPU to run your experiments, you can use Google Colab or Kaggle; these platforms provide free access to GPUs for a sufficient number of hours. Make sure however that you develop your code and plan your experiments ahead of time, as those platforms may impose time limitations.

Your model should predict if the logo overlaps or not for each pixel of an image. We provide you a dataset of 22,470 RGB images with (height=340, width=512) pixels in JPG format and their associated masks in BMP format. Most are natural images but their content is random. The original logo image is an RGBA image with (height=400, width=797) pixels. We transform the logo with several random transformations:

1. flip in the left/right direction,
2. rotate with an angle $\theta \in [0; 2\pi]$,
3. swirl with a strength in $[-5; 5]$ and a radius of 500 pixels,
4. scale down by a factor in $[0.1; 0.38]$,
5. translate in the image boundary,
6. α-blend with the image with $\alpha \in [0.1; 0.4]$.

Each of the first three transformations is executed with probability 0.5. All parameters are sampled from a uniform distribution over their respective

domain. The α-blend transformation is a convex combination: $(1-\alpha)\cdot\text{img}+\alpha\cdot\text{logo}$. Finally, a logo can appear several times in an image. The number of occurrences is chosen randomly in $\{1, 2, 3\}$. The transformations were performed with the `skimage` 0.14.2 library[1].

2 Deliverables

We expect you to submit high-quality code that would be similar to the code you would write as if you were part of our team; this means the code should be easily readable and well-structured, with clear modularization. If you submit code that you did not write yourself, you must identify it explicitly in your files, and cite the source in the report. Expect to be questioned about your implementation and all design decisions. You are free to use any deep learning framework or toolchain you wish, as long as you are able to describe in detail how it works and how you used it. You should use appropriate built-in functions and classes as needed; no need to reinvent the wheel.

The code must include your entire model training pipeline as well as a script that generates predicted masks using your best model's checkpoint on new images from a given directory:

```
python infer.py <image_dir> <output_dir>
```

The `<output_dir>` will contain the predicted masks associated with each image. The filenames must be the same between `<image_dir>` and `<output_dir>`. We provide an evaluation script, `evaluation.py`, to test your `infer.py`.

We also expect a short report (3-5 pages) describing your methodology, the model architecture and its motivation, the experimental settings, the computational resources used — i.e. the number and model of GPU(s) that were used, a discussion of observed model performance, and ideas to improve your baseline.

Your model will be evaluated with the Intersection over Union (IoU) metric on a separate hidden dataset. The final results will be communicated to you during the hiring process. Note that achieving good performance on this hidden dataset is not the only element guiding our decisions; we also take into consideration the quality of the code, the proposed ideas, and the amount of compute used to conduct your experiments.

The report, best model checkpoint, and code must be submitted in an archive file **at most 48 hours** after receiving this file.

Good luck!

[1] https://scikit-image.org/