

Deep Learning - 236606 - Final Project

Shredder Recovery Challenge

January 4, 2019

1 Submission Instruction

1. Submission deadline for the competition: 26/1/2019 23:59
2. The first, second, and third teams in the challenge track will gain 8, 5, 3 points bonus respectively to their final grade (which will not exceed 100 anyway).
3. Final version deadline : 1/3/2019 23:59
4. TA in charge: Yonatan Geifman
5. Submit a zip file containing only your code and a report summarizing your work, in the webcourse site: <http://webcourse.cs.technion.ac.il/236606/>
6. Answers must be submitted as a PDF file, typed using any document editor (not a scan of handwritten answers)
7. Submission is in pairs only
8. No late submissions
9. Download link for the project files - link

2 Introduction

Your task is to develop an algorithm capable of recovering documents and images that have been destroyed using a shredder. You are given two datasets: one for images, and one for documents. You also receive the shredder algorithm. Given crops of images/documents that were generated by the shredder, your algorithm should recover the original version.

3 The Shredder

The shredder has only one parameter: the number of crops per axis (t). It receives an image or a scanned-document and crops it to equally-sized crops vertically and horizontally. The result will be t^2 equally sized crops. For example, when setting $t = 2$, the shredder will crop the image to 4 quartiles (see Figure 1). The shredder algorithm is given in the supplementary files. In the supplemented files you can also find a directory with an image processed by the shredder applied with $t = 4$.

We assume that all input texts/images are given in gray-scale. Thus, the given shredder is transforming all data to gray-scale. Make sure you also do so.

4 Datasets

You can find in the supplied directory two datasets: one for images, consisting of 1300 images at various sizes. The second dataset includes 937 texts of academic papers.

5 Shredder Recovery

You should create an algorithm that gets as input a folder containing jpeg images of a shredded document (a picture or an image of a scanned document) and recovers it to the original text/image. For evaluation, your algorithm should return a label for each crop. The label should be a number in $\{0, 1, 2, 3, \dots, t^2 - 1\}$, representing the crop's position in the original document, assuming that the crops are arranged in a "row-major order" (see Figure 1). For evaluation we will use the supplied script "evaluate.py". You should complete this file with your own prediction method. Your algorithm should be able to recover documents and images of various sizes. You will be evaluated on t values of 2, 4, and 5.

6 Out-of-Distribution (OoD) data

Additionally, the input folder with the jpeg images will contain up to t *out-of-distribution* (OoD) images. That is, there will be t^2 pieces that originally came from the same image or scanned-document, and additional up to t pieces which *do not belong to the main image*. Your algorithm should distinguish these OoD pieces, and label each of them as -1 (minus one). You can assume that pictures will contain picture-OoD distractors, and that documents will contain document-OoD distractors.

7 Other Comments and Restrictions

- We will test your algorithm based on images and documents similar to those in the training set.

second, and third teams in this challenge, will gain 8, 5, 3 points bonus respectively (to their final grade). Note, however, that your final grade for the entire course is bounded by 100.

8.1 Project Evaluation

Your project will be evaluated by the following criteria.

- Novelty - 30%
- Clarity - 10%
- Final performance - 30%
- Correctness of analysis - 20%

The last 10 points will be decided based on overall impression of the project depth.