Instructions

18 April 2019

Data

The character set (label-data) can be downloaded from here:
https://unishare.nl/index.php/s/HUtdrmustBBwUR5

You will find 27 different directories, each containing character-images for one of the letters from the Hebrew alphabet. Please note that the Hebrew alphabet has 22 letters. It does not have an upper or lower case, but five letters (Kaf, Mem, Nun, Pe, Tsadi) have different forms depending on their position within words. For simplicity, we will just consider 27 different classes for the project. The name of the directory is the label for each class.

Here is a list for those 27 characters, with their average images. This will help you to get an idea on how they look like:
https://unishare.nl/index.php/s/yyjckOiJ0gmyZE5

Example test-images can be downloaded from here:
https://unishare.nl/index.php/s/G9Ego9Z822KqF0t

In this directory, you will get 40 images (20 RGB and 20 grey-scaled). Each pair of RGB and grey corresponds to the same physical fragment. The grey images are fused from multiple spectral-band images of each fragment. They have better visual quality in ink-background separation than the coloured ones. For final testing, we will use images similar to the grey images. The coloured images are there for your reference only, and they will give you an idea of how the original fragments look like. The images are already rotated to have the correct orientation of reading.

(If you have a system that works directly on the coloured images then that will be an add-on, but not required. Please discuss and mention this with us before approaching with the coloured images.)

To have an overall idea of how the Dead Sea Scrolls data look like, visit this site: https://www.deadseascrolls.org.il/

For those of you who want to pre-train their model, we recommend using the following font: http://www.historian.net/downloads/habbakuk.ZIP

We have also attached a small python script which provides you with an example of how to create images, similar to your training data, using the font. The script can be downloaded from here:
https://unishare.nl/index.php/s/ULBh4dapeNjKyst

Tasks

1. Preprocessing and character segmentation:

   In order to recognize each character, the first step is to segment them from the test images. The output of the segmentation should be similar to the ones in the training set. The performance can be measured by the overlap (Intersection over Union IoU) between your box and the ground-truth. (one example can be found in page-8 of this paper). In this project, we will not make any evaluation on the bounding box, this is just for your own knowledge of how things work.

2. Character recognition:

   All the characters segmented from the first step should be recognized by your recognizer. You can use the printed characters (font/s) to pre-train your model and then fine-tune using the provided training data set. You can try different models. Note that the total number of characters in the test set is unknown, but similar to the sample images.

   If you do not have enough training samples, some data augment methods might be used. Key words: Character morphing, Elastic morphing.

   You can also train a character recognizer (step 2) and apply it to detect the character box (step 1) using a sliding window strategy. Your main task is the design a character recognizer irrespective of the way you want to do that. In case you want to integrate the linguistic context, we are providing here the n-grams (bi-gram, tri-gram etc.): https://unishare.nl/index.php/s/Yg09054rwwY6tW9

   Anticipatory buzzwords: Binarization, OTSU, HMM, CNN, LSTM.
   Packages: Tensorflow, Theano, Keras, Caffe etc.

Progress updates

Each lecture, starting from the second one,students are expected to give an oral (+powerpoint) presentation/ progress-report. We will randomly assign each group with an identifying number (1,2,3..). Groups with an odd number will present in the second lecture, and with even number at the third lecture, and so on. We will put a schedule once all the groups are formed. The progress-report should at least have the following contents:

1. Overview of articles you have read (including a total number read so far);

2. Amount of text written for literature review & methodology section of the paper;

3. Programming progress (how many modules will the system have, how far is each module completed?);
4. Empirical evaluation progress (both technical and theoretical);
5. Is the overall progress on schedule?

Each component needs to be properly documented and supported by either references or tables and graphs. Show that you read the articles (i.e., show what the article was about and the conclusions), and keep a list, you need it for the References section of your paper.

Each group member should have had the opportunity to show their presentational skills: divide all tasks, including the presentations between the group members equally. Appoint a person for maintaining the progress and updating PPT slides; a person overlooking overall system architecture, a person designing the empirical evaluation (test scripts), etc. Each group will have 5-6 minutes for presentation and 2 minutes for Q&A.

Recognizer

At the end of the course, you are expected to have written a handwriting recognition system for Hebrew characters on the Dead Sea Scrolls data. To test your recognizer, we will (compile and) run your code on a separate, secret test set. We will provide a couple of sample output from the test data, during the third week's tutorial session. To measure the performance, we will use a simplified version of Levenshtein distance.

Report

The report needs to be a scientific paper about the handwriting recognition system that you built during the course.

The report should be around 8-10 pages long, in English (roughly 4000-5000 words);

The main structure should be like in a scientific paper, as follows:
Introduction
Methods
Results
Discussion
References
And, note on individual contribution in the project

The report needs to be written individually, but parts of the report (Methods, Results) can be drafted by the group as a whole (this means your final report needs to be substantially different from the other reports of your group,

especially in the Introduction and Discussion sections). Your personal contribution must be extensive. Coordinate the distribution of focus points among your team.

Along with scientific formalism, your report will be graded on the following subjects:

1. Title + abstract (10);
2. Introduction + literature review (10);
3. Methodology (10);
4. Results + interpretations (10);
5. Discussion + conclusions (10);

Grades

You will be finally graded on your **participation in the group**, on your **presentations**, **empirical evaluation and programming**, and on your **report**.

The final grade appears on Progress. There is no exam other than the final recognizer and written report.

Deadlines

23 June 2019, 23:59
Final version of your recognizer

30 June 2019, 23:59
Final Report

Contacts

Direct your question/s to:
Maruf Dhali, email: m.a.dhali(AT)rug.nl
Jan Willem de Wit, email: j.w.de.wit(AT)student.rug.nl

The use of the Dead Sea Scrolls images used is **restricted**. Please avoid any kind of unwanted distribution of both the labels and the images. Always contact the course instructors before producing any results in the public domain.