ID CARD IDENTITY DETECTION/RECOGNITION

Overall Structure:

In order to solve this problem, we will need to execute the following steps:

1) **Image Capture on a Tablet**
2) **Send Image to a Cloud Server in order to processing it**
3) **Preprocess the image**
4) **Perform Text Segmentation**
5) **Do a Text Recognition**
6) **Analyze the Semantics**
7) **Create a Template and Output the Structured Information**

Text extraction method

The first thing to do is the text extraction. This method is subdivided into two phases: text **localization and text recognition**. The first is used to identify the regions where there is text and the second infers the characters within each text region. We can identify essential **keywords** using the tool **TesseractOCR (https://github.com/tesseract-ocr/).**

Text Localization

First things first, it is essential to do image preprocessing since images can be taken from different angles and light conditions. We can use contrast adjustment, transform to grayscale and apply a median filter to enhance the image and reduce the noise. Also, the text lines can be tilted, causing the OCR failing to process correctly later. So, to prevent that, we will apply a Hough transform to the image already filtered. This filtering can be done through a combination of an adaptive threshold method and morphological operations. The first one will help us convert the grayscale image to binary one, after the classification of foreground and background pixels with respect to a certain threshold. The second one are operations used with square shapes of different sizes in order to filter too small foreground pixel's regions and to group remaining regions that are close to each other. In this way, we can aggregate the regions relative to the text. Horizontal and vertical lines relative to the text area can be now extracted and used as features for rectification. Next we can crop the image in a way that allows us to put the identity document area apart from the rest of the image. Now, we can detect maximally stable extremal regions using a method based on Connected Components. This method detects text and non-text regions and, to get rid of non-text regions, we need to use geometric properties like stroke width. So, after having segmented words and text bounding boxes, we can proceed to text recognition.

Text Recognition

Now that we have already identified the regions of interest (RoI) of the image, these are sent to the OCR engine that, in our case, will be Tesseract OCR. This tool gives for each bounding box a percentage of accuracy that we can use to select the most accurate word among the words processed. The recognition confidence results for each bounding box vary in relation to the image that is processed. Moreover, we can eliminate regions with low accuracy values and those in which there’s only one character.

Semantic Analysis

Sentence Detection

Sentence detection is important because, now that we finished the data extraction, some words that only make sense together might appear separated. So, one way to detect the sentences would be by applying the Euclidean Distance to calculate distances between all the words and classify them as near or distant throughout the same horizontal line (and assuming that adjacent words have similar heights).

Keywords and Semantic Recognition

So, in our case, we are looking for face photo, name, ID number and other information (like date of birth, expiration date, etc) - these are the keywords that we want to recognize in the ID document. These keywords can be located by making use of a dictionary lookup and regular expressions. For that, sentences need to be transformed into alphanumeric text in order to remove every space and special characters that appeared during the OCR phase. Next, a function can determine if this text matches any of the regular expressions in a dictionary of keywords. The matched texts can be stored in a keywords dictionary structure.

Data Collection / Data Annotation/ Train & Test Data

So, to assess the individual performance of each task, the combination of them and the entire system, I would suggest create five sets of data:

- **1st dataset:** set of images of ID Documents with the bounding boxes of the words recognized in order to assess the sentence detection method.
- **2nd dataset:** set of sentences extracted from images of ID Documents in order to assess the Keyword detection method.
- **3rd dataset:** set of images of ID Documents with the bounding boxes of all the sentences and the Keywords identified. This will allow us to evaluate Semantic recognition methods.

- **4th dataset:** contains the same set of images and bouncing boxes of the words recognized of the 1st database. Notwithstanding, this now is intended to assess the Semantic Analysis Process in general and examine the true effect of the sentence detection method on the system output.
- **5th dataset:** set of images with distinct perspectives of ID documents. This dataset is intended to evaluate the end-to-end ID Detection System in a real environment.

The set of images in datasets 1,3 and 4 can be downloaded from the Internet. The data contained in these pictures about the words' location,and their transcriptions, need to be created. Also, **it is performed a fine tuning of these localizations and transcriptions to obtain ground-truth elements or gold standards annotations**. The 2nd dataset can be generated from a site that creates random identities. IDs utilized in the dataset 5 are dependent upon an assortment of unfavorable conditions, since they are caught under natural conditions (variable light). **The set of images in these datasets need to be, at least, in the order of the hundreds.**

Testing/Evaluation Strategies

Since various tasks are planned to be used in the framework, to quantify the efficiencies of each of these various performance metrics must be utilized.

Sentence detection task

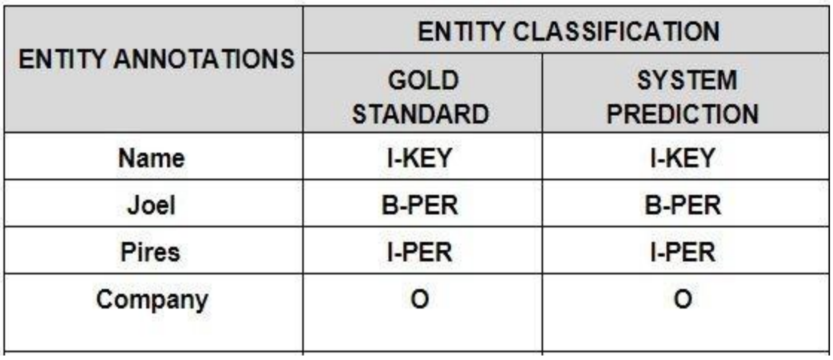
Here, the performance can be measured by Precision, Recall and F-Score of the detected and ground truth bounding boxes of the sentences. We can define the limit used to determine a match between the detected Bouncing boxes and the ground-truth bouncing boxes as when their overlapping ratio is more than 0,85.

Keyword Detection and Semantic Recognition

In this case, albeit both are referenced in two distinct datasets (dataset 2 and 3), the same metrics can be used, since both tasks consist of detecting named entities. These databases contain sentences made out of at least one word, where dataset 2 comprises just text, while database 3, in addition to text, contains the location of the bouncing boxes of each sentence. So as to define which metrics utilize, all possible comparison scenarios of the project's predictions and the gold standard annotations (ground truth) must be investigated to figure out which of these situations relate to each task and consequently realize how to assess it. One possible scheme can consist of labeling with:

- **O** (Outside) those words that don't belong to any named entity
- **I-XXX** (Inside) those that have a place with a named element of type **XXX**.
- At whatever point a gathering of expressions of type **XXX** is inside the same bounding box, the main word is labeled **B-XXX** (Beginning) to show that an entity composed of at least two words labeled with a similar sort and without **O** tokens among them starts.

- In this project, the information contains elements of four kinds: individual names (**PER**), ID numbers (**IDN**), date of birth (**BTH**) and Keywords (**KEY**).
- Let’s see an example of just one possible scenario - if the systems predicts the classification and entity type correctly:



ENTITY ANNOTATIONS	ENTITY CLASSIFICATION	
	GOLD STANDARD	SYSTEM PREDICTION
Name	I-KEY	I-KEY
Joel	B-PER	B-PER
Pires	I-PER	I-PER
Company	O	O

The assessment metric that I would use would be the Language-Independent Named Entity Recognition task to measure the performance of the project as far as Precision, Recall and F-Measure. In this case, the Precision is the percentage of named entities found by the system that are really right; and the Recall, is the percentage of the genuine number of name entities that are found by the system. Also, they only consider a named entity as correct if it is an exact match of the corresponding gold standard entity type.

Semantic Analysis Process

Here I think it’s better to use an alternate performance metric. The inputs of database 4 are equivalent to those in dataset 1 (ground truth bounding boxes of the words contained in the IDs); while the process outputs correspond to the outputs of the Semantic recognition task. Subsequently, this entire process can be considered as a rule-based named-entity recognition system, and the suitable performance metric ought to examine all the potential situations introduced in it. An evaluation method for NERC systems like this that considers all these scenarios is the MUC-5. This evaluation scheme scores a system to find the correct type and the exact Bounding Boxes boundaries of the named entities identified.

End-to-End System

As the outputs of the end-to-end system are the same as the Semantic Analysis Process, the whole system can be considered as a rule-based named-entity recognition system. So, the evaluation method can be the same as that used in the Semantic Analysis