Groupe: Adriano Ruberto, Emmanuel Kiegaing , Guimel Mendoza Betancourt, Patrick Kanzler

The following presents the approach used to segment the input images and to build the ground-truth from the provided database. Our approach follows the following flow :

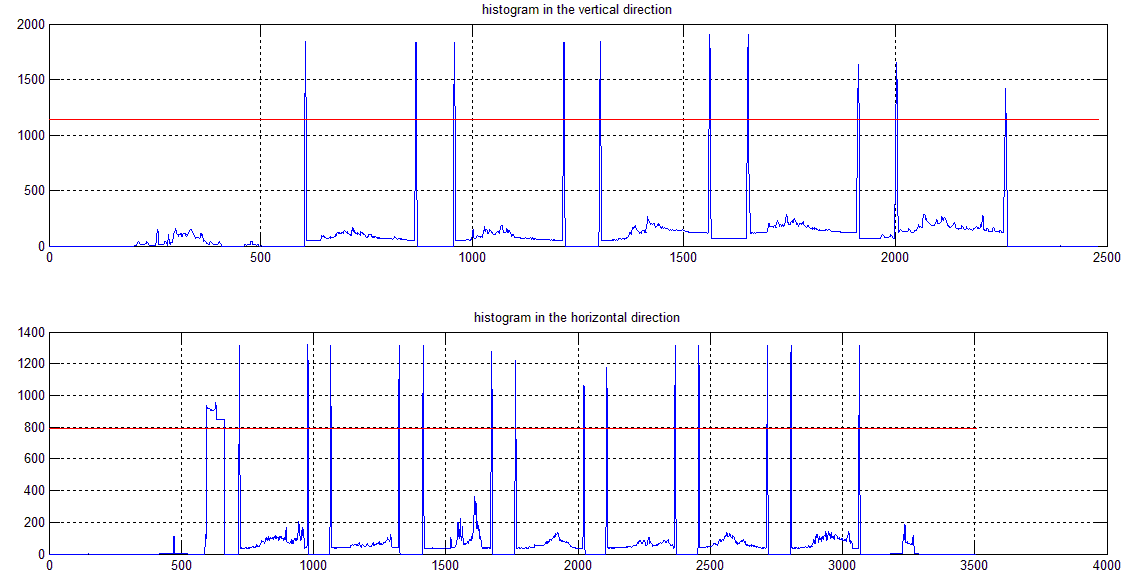
* Pre-processing of the image
* Objects detection and isolation
* Icons recognition with the of aim classifying and create names for sub-images
* Saving sub-images (under the .png extension) and text description files (.txt)

# Image pre-processing

After the image has been red it passes through a pre-processing step aiming at removing noise bands coming from the scan processes. These noise bands when they (exist) are located in a band of pixels at the edges of the image (top, bottom, left and right). The input image is then converted to gray scale and inverted afterward the segmentation of the image can start.

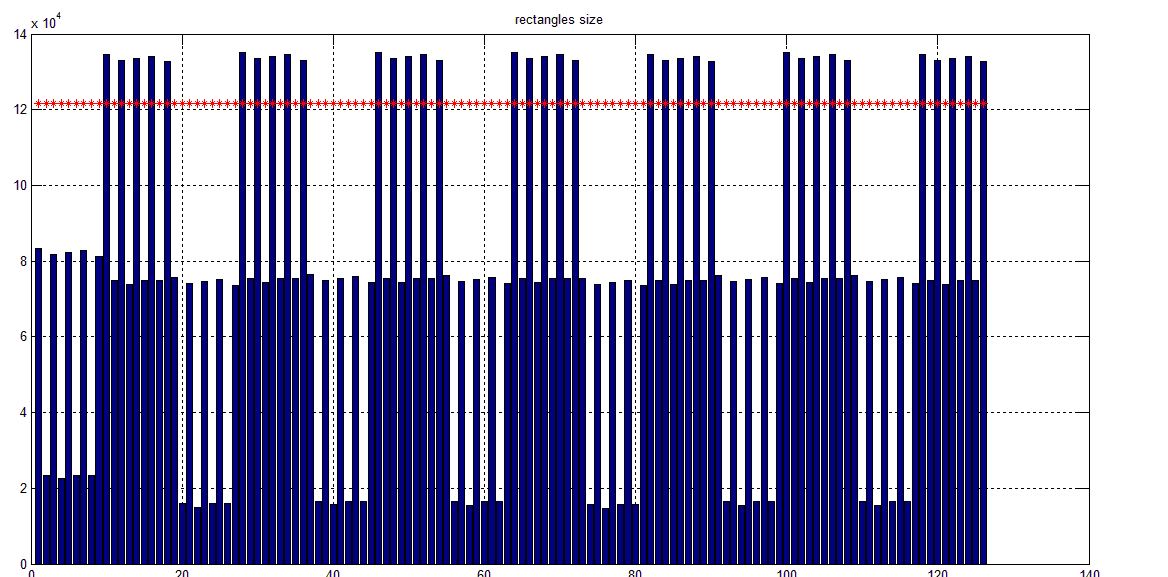
# The segmentation step

This step is based on statistical methods thus benefits from the robustness such methods provide. The detection of lines is mainly based on the computation of the histograms representing the sum of pixels value according to vertical and horizontal axes. Two histograms are computed and they display peaks at positions corresponding to probable lines. The graph below shows an example of histogram for a given input image :



Şekil 1 : Histograms with their associated threshold

To reduce the burden of computation in the detection of peaks threshold values are set and the research for peaks will be done only in the range of points having a value greater than the thresholds. At the end of this process we obtain all probable lines forming the rectangles containing the objects we want to isolate.



Şekil 2 : Rectangles size and their associeted threshold

The next step will consist in removing lines which are no really part of a rectangle containing an object. To do this all the possible rectangles are constructed with their positions and sizes (size of their diagonals) and here we base the rectangle filtering process on the fact that the rectangles containing objects will always be the bigger thus a new threshold related to maximum value of rectangle sizes is chosen and the rectangles having a size bellow this threshold are discarded. The following graph gives a view of the of rectangles size and displays the good insulation of “good” rectangles from the others.

At the end of this process we obtain the coordinates for all the rectangles containing right objects. The next step will be to read the icon at beginning of each row and identify their type use the matchTemplate

# Icons recognition

This process consists of two steps. In the first step a region of interest is cut out. The position on the Y-axis is determined by computing the arithmetic mean of the upper lines of the rectangles in a line. The position on the X-axis is determined by moving a fixed length (coded as the fraction of the image-width) to the left starting from the leftmost rectangle.

This region of interest is then analysed with the method matchTemplate. For this we have cut out the icons out of images in the database. When cutting these images we made sure to make them as small as possible in order to not waste processing time. With the method MinMaxLoc the probability for a match is obtained for every icon. The icon with the highest probability is taken. If the probability is below 0.5 we make no decision and mark the image as undecided. The same workflow is applied in order to find the size-marker (small, medium, large) under the icons.