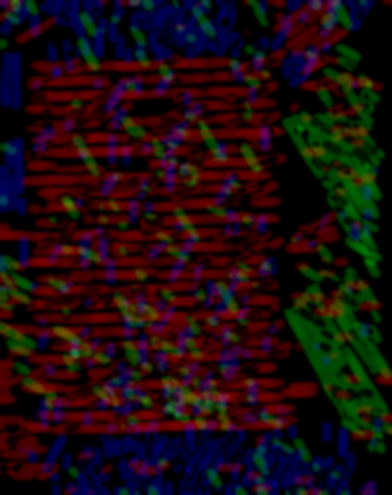
**Web-Based Interactive Page Layout Segmentation**Amit Pintz Amir Arbel

Our mini project is based on an article written by Jihad El-Sana and Majid Kassis, who’s proposed a method to overcome the difficulties in text segmentation. ([link](https://www.cs.bgu.ac.il/~tiip181/wiki.files/InteractiveSegmentation.pdf))

The main idea is to use several Gabor filters with different orientations and combining the response into a single matrix.   
After the combination of the responses of the Gabor filters, we are able to use many segmentation methods, in this project we used Graph-Cut segmentation.

Our program’s steps:  
First we created the Gabor filters with 6 different orientations (number of orientations, along with other settings may be changed via separate settings file).  
The filters were built with python OpenCV library.   
After creating the filters and storing them, we iterate over each one of them, filtering the original image and saving the result.

The second step is to manipulate the responses of the filtered images. We used a kernel for thickening the lines in the response image for better segmentation.  
We also used OpenCV Threshold function for ignoring all irrelevant parts in the responses.  
After the manipulations described, we get to combining the responses into one image.  
We divided each result values by 2, so we can merge the 6 directions who provided 6 responses, into 3 channels, red, green and blue.



Result

Original Image

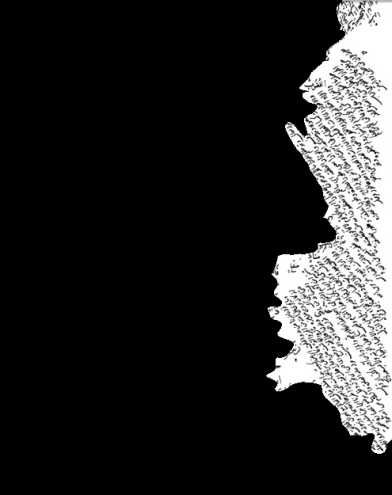
The last step in our program, is the segmentation itself.  
We used Matlab engine and wrote a script using Matlab tools for Image Graph Cut segmentation. In some cases we apply Gaussian filter on the image before sending it to Matlab engine for better result.  
The Matlab script parameters are the image for segmentation and foreground and background pixels extracted from JSON file.

Source code may be found here: <https://github.com/pintza/GaborFilter>.  
Performance examples are below:



Result

Marked Image (A)



Marked Image (B)

Result