BINYAS

Showmik Bhowmik, Soumyadeep Kundu, Ram Sarkar
Department of Computer Science and Engineering, Jadavpur University, India.

Description: Our input is a color image I which we first convert into its grayscale counterpart I_g and then we subsequently use a contrast stretching operation and a region filling operation which gives us I_{fill}. We binarize I_{fill} with the help of a global threshold based method and get I_b. We also consider another binarized image I_b' which we get by applying a local threshold where the window size is chosen dynamically. Next we check all the connected components (CCs) in I_b in terms of their height, width and aspect ratio and based on that we take a decision to exclude the separators and margins from both I_b and I_b'. After that we again examine the CCs of I_b to classify the input document as geometrically complex or simple.

For complex document, we partition the components present in it into two images - I_{large} and I_{small}. We generate I_{large} from I_b and I_{small} from I_b'. After that we perform text and non-text classification on both the I_{large} and I_{small} separately. From I_{large} we get text only image I_t^l and non-text only image I_{nt}^l. Similarly from I_{small} we get I_t^s and I_{nt}^s. Then we combine I_{nt}^l and I_{nt}^s to generate the final non-text only image I_{nt}^{final}.

We further classify the CCs in I_{nt}^{final} into table, bar chart, image, and inverted text. From I_t^l we identify the drop capital and separate it. Apart from that, we perform stroke-width analysis on the components of I_t^s to generate I_{thick} and I_{thin}. After that we form the regions from I_t^l, I_{thick} and I_{thin}separately. For this we employ an iterative and adaptive morphology based approach using a rotating structuring element. We then refine all the regions to get the closest polygon and the paragraph separator. At the final step we combine all the segmented text images to get I_t^{final}.

However, if we find the input image as a geometrically simple one, then we avoid the component partition step, rather we directly perform text non-text separation on the I_b' to get I_t^s and I_{nt}^{final}. We perform rest of the operations on these images as mentioned earlier.