Box Module

Gonçalo Afonso

October 10, 2023

1 Introduction

In this document, we will formalize the box module created for OCR engine result generalization and further analysis and processing of these boxes.

2 Box Module

2.1 Box

A box represents a container element for a region in a document. Each container may include other containers of lower levels, with the lowest being a text container.

- **level** : text level of the box. $\{1 : \text{page}, 2 : \text{block}, 3 : \text{paragraph}, 4 : \text{line}, 5 : \text{word}\}$
- **page_num** : only meaningful when multiple pages are processed.
- **block_num** : block identifier in which box is inserted
- **line_num** : line identifier in which box is inserted
- **left** : leftmost horizontal value of the box, relative to the left border
- **right** : rightmost horizontal value of the box, relative to the left border
- **top** : topmost vertical value of the box, relative to the upper border
- **bottom** : bottommost vertical value of the box, relative to the upper border
- **width** : width of the box
- **height** : heigth of the box
- **text** : text recognized inside the box
- **conf** : level of confidence in the text
- **type** : type of box. ['delimiter', 'image', 'text']
- **word_num** : word identifier (applicable if level is word)

2.2 Functions

tesseract_convert_to_box : $Dict \rightarrow [Box]$

Turns a dictionary of tesseract results into a list of boxes.

is_empty_box : $[Box] \rightarrow Bool$

Checks if a box group is empty. Every box of level 5 (word) has to be empty for a positive result.

is_delimiter : $[Box] \rightarrow Bool$

Checks if a box group is a delimiter. A delimiter is an empty box group that follows the rule:

$$p\_box['width'] \geq p\_box['height'] \times 4 \vee p\_box['height'] \geq p\_box['width'] \times 4 \quad (1)$$

where *p_box* is the parent box of the box group.

boxes_to_text : $[Box] \rightarrow Str$

Converts a box group into a string. The string is the concatenation of the text of each box in the group.

append_box : $([Box], Box) \rightarrow [Box]$

Appends a box to a box group.

remove_box : $([Box], Index) \rightarrow [Box]$

Removes a box in index *Index* from a box group.

update_box : $([Box], Index, Box) \rightarrow [Box]$

Updates a box in index *Index* from a box group with a new box.

draw_bounding_boxes :
$([Box], image\_path : Str, draw\_levels : [Int], id : Bool) \rightarrow img : MatLike$

Draws bounding boxes in an image. The image is loaded from *image_path* and the bounding boxes are drawn in the image according with boxes group given and the levels in *draw_levels*. If *id* is true, the id of each box is also drawn in the image.

get_group_boxes : $([Box], id : Str, index : Int) \rightarrow [Box]$

Gets a box group from a box group. The box group is identified by the *id* and the *index*.

line_mean_height : $[Box] \rightarrow Float$

Calculates the mean height of a line.

is_normal_text_size :
$(normal\_text\_size : Float, line : [Box] | line\_height : Float, range : Float) \rightarrow Float$

Checks if a line is normal text size. A line is normal text size if the mean height of the line is within the range of the normal text size. Range is by default 0.3.

analyze_text : $[Box] \rightarrow Dict$

Analyzes a box group. The analysis result returns the value of *normal_text_size*, *normal_text_gap*, *number_lines*, *number_columns* and *columns*.

improve_bounds : $[Box] \rightarrow [Box]$

Improves the bounds of a box group. Not yet finished.

search_text_img_tesseract : $img\_path : Str \rightarrow Dict$

Searches text in an image using tesseract. The result is a dictionary with bounding boxes.

simple_article_extraction_page : $[Box] \rightarrow [Article]$

Extracts articles from a page. Not yet finished.

block_box_fix : $[Box] \rightarrow [Box]$

Fixes the blocks boxes in box group. Eliminates empty, non delimiter boxes and eliminates intersections.

get_delimiter_blocks :
$([Box], search_area : Simple\_Box, orientation : Str) \rightarrow [Box]$

Gets the delimiter blocks in a box group. The delimiter blocks are the blocks that are delimiters and are inside the search area and respect the given orientation.

join_aligned_delimiters : $(delimiters : [Box], orientation : Str) \rightarrow [Box]$

Joins aligned delimiters. The delimiters are aligned if they have the same horizontal or vertical value within a range (*is_aligned* for further reading), depending on the orientation.

estimate_journal_header : $([Box], image\_info : Dict) \rightarrow Simple\_Box$

Estimates the journal header using its box group. The header is estimated by finding the blocks that are delimiters and follow the rule:

$$delimiter['bottom'] \geq image\_info['bottom'] \times 0.5 \wedge delimiter['width'] \geq image\_info['width'] \times 0.3 \quad (2)$$

estimate_journal_columns :
$([Box], image\_info : Dict, header : Simple\_Box?, footer : Simple\_Box?) \rightarrow [Simple\_Box]$

Estimates the journal columns using its box group. The columns are estimated by finding the blocks that are vertical delimiters and are within the area between the header and the footer if they exist (ortherwise within the page).

estimate_journal_template : $([Box], image\_info : Dict) \rightarrow Dict$

Estimates the journal template using its box group. Returns a dictionary with the header and the columns.