.

Student Satellite Project
Indian Institute of Technology, Bombay
Powai, Mumbai - 400076, INDIA

Website: www.aero.iitb.ac.in/satlab

README - fe_centroiding_rle.m

Electrical Subsystem

---

fe_centroiding_rle(arr_img)

Code author: Millen Kanabar
Created on:
Last modified:
Reviewed by: -
Description:
This function is used to get the *sum of the coordinates* (see the numerator of the expression defining the centroid) and *sizes* of unmerged regions along with the *number of tags and final tags* from a gray-scale image. (The connectivity referred to here is 4-connectivity.) The unmerged centroids are then taken along with the scalars to give out the final coordinates of the centroids of regions (stars) in the input image. The centroid is defined as

$$(x_{centroid}, y_{centroid}) = \left(\frac{\sum_{p \in region} I_p x_p}{n_{pixels}}, \frac{\sum_{p \in region} I_p y_p}{n_{pixels}} \right)$$

where I_p is the intensity of the pixel and (x_p, y_p) are the coordinates of the pixel.
Formula & References:
The function uses the algorithm presented in [1] to find the centroids of the connected sets of pixels in the image. The data is first extracted from a row. The contiguous ranges of pixels in that row are then compared with those in the previous one and the ranges in the current row are tagged accordingly. The ranges in the first row are given new tags. The data from this is added to the final data and the connected regions in the final data and the line data are also merged.
Input parameters:

1. **arr_img** : (matrix) - input image, with pixel location wrt the top left corner as indices ([i, j]); and the reading at the corresponding pixel as the value stored at [i, j]

Output:

1. **arr_centroids:** Array containing the centroids of the connected regions in the input image and their corresponding ids

2. **num_stars:** Number of connected regions in the input image

fe_extract_row_data(arr_row)

Code author: Millen Kanabar
Created on:
Last modified:
Reviewed by: -
Description: Extracts data from a given row of the input image and outputs an array containing that data, along with the number of unbroken ranges of bright pixels
Input:

1. **arr_row:** a row from the input image

Output:

1. **arr_row_data:** array containing the sum of $\sum_{p \in range} I_p x_p$ for a range of bright pixels in the first column, sum of intensities of those pixels in the second, the number of pixels in the third and the start and end pixel of the range in the fourth and fifth respectively[1]
2. **num_ranges:** number of continuous ranges of bright pixels[2]

fe_compare_lines(arr_prev_row, arr_row_data, num_prev_row, num_ranges, num_tags)

Code author: Millen Kanabar
Created on:
Last modified:
Reviewed by: -
Description: This function takes in the data from the current row, the previous row, the number of bright ranges in each row and the tags already allocated, compares the ranges in the current row and tags them with tags of the already tagged ranges in the previous row. The tagging proceeds from the left to the right.
Input:

1. **arr_prev_row:** array containing the first and last pixels of each range along with the corresponding tag for a distinct disjoint range in the previous row in each row. The ranges are ordered going from left to right
2. **arr_row_data:** array containing data from the current row. The column-wise arrangement of the elements is as follows:

$\sum_{pinrange} I_p x_p$	$\sum_{p \in range} I_p$	$\sum_{p \in region} 1 = n$	p_{start}	p_{end}

3. **num_prev_row:** The number of bright ranges in the previous row
4. **num_ranges:** The number of bright ranges in the current row
5. **num_tags:** The number of tags already allotted

Output:

[1] Defaults to an array of zeros if the row has no bright pixels
[2] defaults to 0 if the row has no bright pixels

1. **arr_region_data:** The array containing data for each tag in each row in the following format:

$\sum_{pinrange} I_p x_p$	$\sum_{p \in range} I_p$	$\sum_{p \in region} 1 = n$	flag

   This table is constructed for adding the data from the row to the final output array

2. **arr_regions:** This is a one-dimensional array that indicates the tag for every regions (the i^{th} element will indicate the tag of the i^{th} range from the left.)

3. **arr_merge_regions:** Each row contains tags corresponding to a connected region, the first element indicating the number of tags in that row

4. **num_merge_tags:** The number of nonempty rows in arr_merge_regions

5. **num_tags:** Updated number of tags already allotted

fe_add_centroid_data(arr_centroid_data, arr_centroid_data_new, i_row)

Code author: Millen Kanabar
Created on:
Last modified:
Reviewed by: -
Description: This function takes the data generated by comparing two rows and adds it to the data generated from previous comparisons
Input:

1. **arr_centroid_data:** Data generated from previous iterations over rows

2. **arr_centroid_data_new:** New data to be added to arr_centroid_data

3. **i_row:** The row number for the lower row (from which the data was generated)

Output:

1. **arr_centroid_data:** The updated array containing the data from all rows parsed so far

fe_merge_regions(arr_centroids, arr_merge_regions, num_merge_regions)

Code author: Millen Kanabar
Created on:
Last modified:
Reviewed by: -
Description: This function merges the rows in arr_centroids according to the tags mentioned in arr_merge_regions.
Input:

1. **arr_centroids:** The array containing the data to be worked upon. This is the pre-final data table after every row comparison

2. **arr_merge_regions:** The array containing the details of the tags to be merged. The first element is the number of tags to be merged into one. The subsequent ones are tags to be merged into one.

3. **num_merge_regions:** Number of regions to be merged

Output:

1. **arr_centroids:** The updated array with merged tag data

fe_line2prev(arr_row_data, num_ranges, arr_regions, arr_merge_regions, arr_region_data)

Code author: Millen Kanabar
Created on:
Last modified:
Reviwed by: -
Description: This function moves data stored in the arr_line format and converts it to the arr_line_prev format (see fe_conpare_lines(.) for a detailed description of these formats).
Inputs:

1. **arr_row_data:** array of the form specified in the input description of fe_conpare_lines
2. **num_ranges:** number of non-trivial rows in arr_line
3. **arr_regions:** array containing the tag of each row in arr_line
4. **arr_merge_regions:** The array containing the details of the tags to be merged. The first element is the number of tags to be merged into one. The subsequent ones are tags to be merged into one.
5. **arr_region_data:** Array whose fourth column contains the flag of the tag associated with the tag (which is equal to the row number). Zero if no flag is associated with the tag.

Output:

1. **arr_prev_row:** Array of the form described in fe_compare_lines
2. **num_prev:** Number of ranges in the row being processed

References

[1] Alexander Bochem, Rainer Herpers, and Kenneth Kent. Hardware acceleration of blob detection for image processing. *Advances in Circuits, Electronics and Micro-electronics, International Conference on*, 0:28–33, 07 2010.