.

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

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

README - fe_pixel_by_pixel.c

Electrical Subsystem

---

fe_pixel_by_pixel.cpp()

Code author: Aravind Bharathi
Converted by: Siddhi Bagul
Created on: 10/06/2021
Last modified: 13/06/2021
Reviwed by: -
Description:
The function ***tag()*** 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.
The function ***merge_tag()*** is used to find the *position of centroids* and *number of stars* found in an image and returns the latter. The input consists of regions, tagged in a particular way, and values pertaining to each tag. The function checks if multiple tags correspond to the same star and groups them accordingly. It then proceeds to find the centroids of these stars. The input parameters consider the top left corner pixel of the image as the origin and the code translates and flips the axes such that the centre of the image becomes the origin. The function *centroiding()* is used to call the two above functions sequentially.

Formula & References:
The algorithm we are referring to basically tags connected pixels with the same tag (which is the index of the array where the sums of coordinates and the number of pixels corresponding to that region are stored) and noting which tagged regions with different tags meet so that they can be merged[**?**]. The algorithm has been modified because we do not need the regions, we only need the centroids. We also include some extra conditions to reduce the complexity and memory requirements. We have five arrays and two scalars as an intermediate output. The centroid of a single tagged region is defined as:

$$\left(\frac{sum\_x_p}{w_p}, \frac{sum\_y_p}{w_p} \right) \forall\ p \in tag,\ final\ tag = 0$$

The centroid of a multi tagged region is defined as:

$$\left(\frac{\Sigma_{p \in r_n} sum\_x_p}{\Sigma_{p \in r_n} w_p}, \frac{\Sigma_{p \in r_n} sum\_y_p}{\Sigma_{p \in r_n} w_p} \right) \forall\, r \in region,\ final\ tag \neq 0$$

The formula to translate and flip the axes is defined as:

$$x_{out} = \left(x_{in} - \frac{length}{2} - 0.5 \right)$$

$$y_{out} = -1 * \left(y_{in} - \frac{breadth}{2} - 0.5 \right)$$

Input parameters:

1. **arr_in_img** : (*unsigned short int* 2D array) - 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** : (*float*) - Centroids of stars in a frame of reference where x+ is rightwards and y+ is upwards centered at the middle of the image. the shape is (no. of stars, 3), with each row having format $(Sr_{no}, x_{center}, y_{center})$.
2. **num_stars** : (int) - Count of number of stars.

Remark: The function outputs the data in a .csv file for each test image. **'fe_tag_merge_centroids_i.csv'** having row format(ID, x_cen, y_cen) for i^{th} test image. Number of test cases can be modified by tuning *'n'* in *int_main* function. Note that, the centroid values are being multiplied with pixel width.The simulation result(expected output) for each image can be found in star matching test cases folder.