Automatic Origin and Direction Selection on Arthroscopic Images

Rui Melo

November 17, 2009

Algorithm Outline

The objective is to automatically define an origin and direction in a set of arthroscopic images in order to keep a constant reference frame along different images. The approach proposed uses two squares of the calibration grid painted in different colors as markers to detect the desired points (see figure 1).

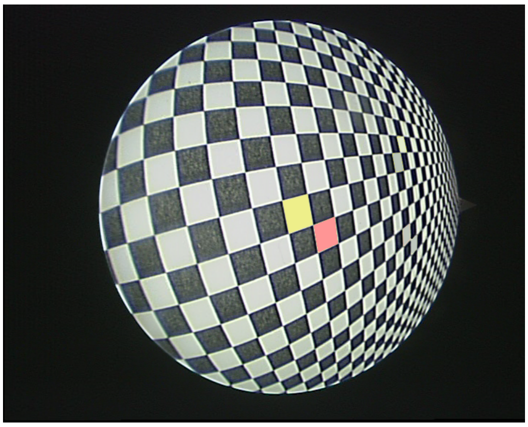

Figure 1: Synthetic image used to test the algorithm. The grid has two painted squares (one yellow and other pink) used in the origin and direction detection.

The data used as input to the algorithm are the RGB image and the grid corners detected at calibration time. The outline of the algorithm can be described as follows:

- Image segmentation by colors and labeling of the yellow and pink regions;
- Origin detection using the 4 nearest corners to each centroid of the colored squares;
- Direction detection according to a defined convention.

Color Segmentation

In order to achieve a better color segmentation the HSV (Hue, Saturation and Value) color system was used. Although other color decomposition were not tested, we can say that the image segmentation using this color system is robust and discriminative enough. Note that we are assuming that the arthroscopic images were captured using the calibration box with the grid back lighted.

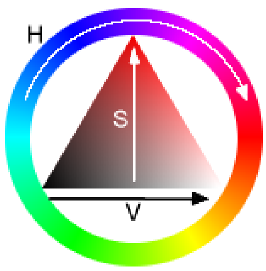

Figure 2: HSV color system.

The color segmentation is quite simple. By defining some HSV thresholds we can robustly segment the different colored squares. Therefore, a small learning step is required in order to define the HSV thresholds for each color. The result of the color segmentation step is illustrated in figure 3:

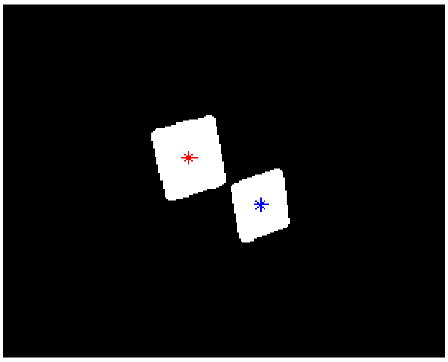

Figure 3: Color segmentation of the arthroscopic image. The blue and red points are the centroids of the two detected blobs.

As the segmentation is not always perfect and the blobs corresponding to each color can be glued together an *erode* morphological operation followed by a *dilate* were applied to the image. This way the blobs are completely separated and the *regionprops* MATLAB function can be used to retrieve the blobs centroids.

Origin Location

Having the corner points of the calibration grid, the four nearest corners to each centroid are found. These correspond to the boundaries of the colored squares. The origin is simply defined as the common corner between the two colored squares, as shown in figure 4.

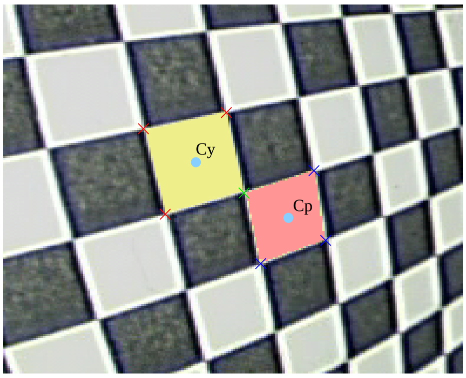

Figure 4: Origin detection. The origin (green cross) is the common corner of the yellow and pink colored squares.

Direction Definition

To define a constant direction for all the arthroscopic images the approach represented in figure 5 was used:

We define a vector (V_c) that connect the yellow square centroid computed in the segmentation process to the origin. Two other vectors are defined (V_1 and V_2) as being the vectors that connect the centroid to the two nearest corners to the origin.

$$\Theta_1 = atan2\,(V_c.y, V_c.x) - atan2\,(V_1.y, V_1.x) \qquad \Theta_2 = atan2\,(V_c.y, V_c.x) - atan2\,(V_2.y, V_2.x) \tag{1}$$

Using equation 1 we can find the angles between the vectors V_1 and V_2 with respect to V_c. It is expected that those angles have opposing signs. We convention that the corner to be used as the direction point is

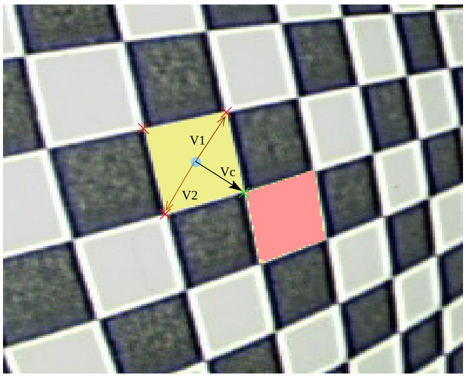

Figure 5: Direction definition. The direction is defined by finding the yellow corner point on the left side of the vector joining the centroid to the origin corner.

defined by the vector with a positive angle. This way the direction is kept constant according to the colored squares disposition. Figure 6 illustrate the final result of the algorithm:

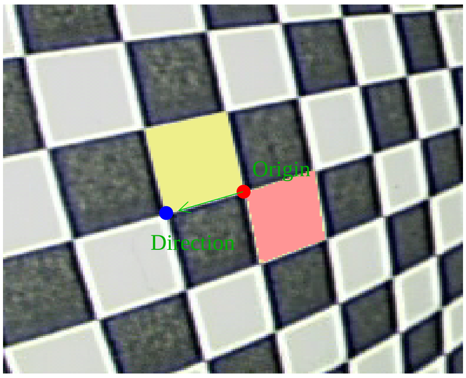

Figure 6: Final result image of the algorithm.

Final Considerations

The algorithm presented is able to robustly identify the origin and direction of the reference frame as long as the markers (yellow and pink squares) are kept in the central region of the arthroscopic image. Since the algorithm is dependent of distance measurements, if the markers are in the periphery of the image the algorithm fails due to the radial distortion effect. Note that the tests were only performed for synthetic images (where the colored squares were added to the original image). In real images (where a grid is printed with the colored squares on it) we expect to only have to adjust the segmentation thresholds. As long as the two colors are correctly segmented and the markers are in the central region of the image the algorithm should not fail.