Algorithm for camera calibration: Pattern localization

Alvaro Rojas and Israel Chaparro

***Abstract*— We describe our algorithm to detect a pattern used for the calibration process of the camera. The developed algorithm is based on the detection of concentric ellipses that require to generate image components that can be qualified as ellipses.**
To get this we use an adaptative thresholding algorithm that reduce the problems caused by illumination. Currently the algorithm allows the detection of the pattern and the tracking of the ellipses in each frame.

I. INTRODUCTION

The process of camera calibration is a really important stage when you want to carry out projects that involve exact estimates of three-dimensional pixel location. Examples like augmented reality require a correct creation of meshes from specific pixel location estimates, which tend to be very inaccurate due to the effects of radial and tangential distortions by example.
The current target is to achieve a correct detection of a pattern that will help as to calculate intrinsic and extrinsic matrices of the camera. This both transformation matrices allow to generate a the corrected images to use them for any process where is required a precise localization of the pixels.

II. PATTERN DETECTION

Given that the algorithm is based in the ellipses detection the process is the next:

* Grayscale detection
* Gaussian smothing
* Integral image thresholding
* Contours localization
* Fitting ellipses
* Heuristics to reduce false positives

A. Grayscale transformation:

The first algorithm used is a grayscale transformation because for nor is important that the algorithm works in real time.
Using grayscale transformation we avoid to repeat unnecessarily process for each channel (R,G,B) in the images and get ready for the thresholding algorithm.

B. Gaussian Smothing:

Usually to prevent that the noise of the present in the original image from passing to the thresholded image is used smothing algorithms that normalize some conditions in nearby pixels.
In this case we use a Gaussian Kernel.

C. Integral image threshold

In the previous update of our algorithm we used fixed values to thresholdize the image but in some cases for inclinations of the pattern the illumination varied a lot and that makes complicate to find a correct value for the threshold. This algorithms use an adaptive threshold based on the mean of nearby pixels and depending if reaches an percentage of the mean is classified.

D. Contours localization

To localize ellipses is required to transform the objects generated into edges that can be adjusted into ellipses. There's multiple algorithms that cant be used because the thresholding stage reduce the complications to generate correct edges. Given that we used the OpenCV findContours functions that return the edges as objects the can be analyzed with with fewer complications.

E. Fitting Ellipses

In this stage we use the fitEllipse OpenCV's function to approximate all the contours resultants to ellipses. Given that this algorithm generates a considerable percentage of false positives is required to use heuristics that allow as to get a final result with the minimum possible percentage of false positive detections.

F. Heuristics to reduce false positives

At this point there's a considerable number of false positive results for which we apply a double filter.

* Size filter: Fist We eliminate the ellipses that has radius that aren't in a interval of allowed values.

* Parent-son filtering: Given that findContours generate a hierarchic (father-son) group of edges each ring generate 2 concentric ellipses and if one is eliminated in the previous step we can eliminate his father-son.

* Pair of centers: The third reduction was got calculating the centers and getting pairs with centers very close.

III. Pattern tracking

We developed an heuristic to match the centers in each frame and using that we got to track the rings in the video. We made the next procedure:

* Pair matching: We start taking the pairs matched in the false positive's reduction heuristic.
* Mean center: Given two center by each ring we define the center of the ring as the mean of the centers of the ellipses.
* Pair matching between frames: We find pairs of centers with a little maximum distance.
* Re-localization: In some frames a little percentage of the circles disappear and after that we use the unmatcheds ellipses as candidates.

IV. Results

We apply all the algorithms mentioned before and test the results in high-low illumination and considerable inclinations of the pattern. The results are shown below:

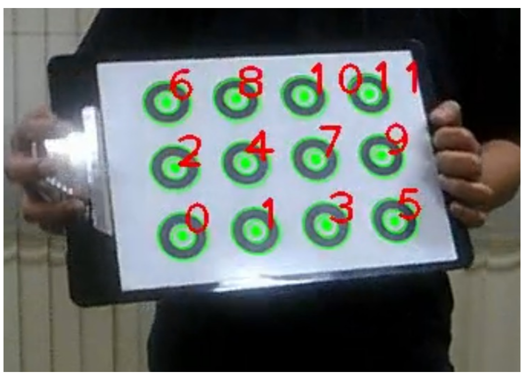

Fig. 1. Hight ilumination conditions

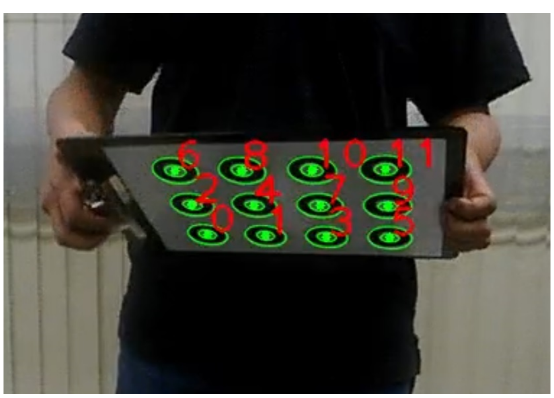

Fig. 2. Low ilumination conditions y inclination

Finally we calculate the accuracy finding all the rings and plotted the result:

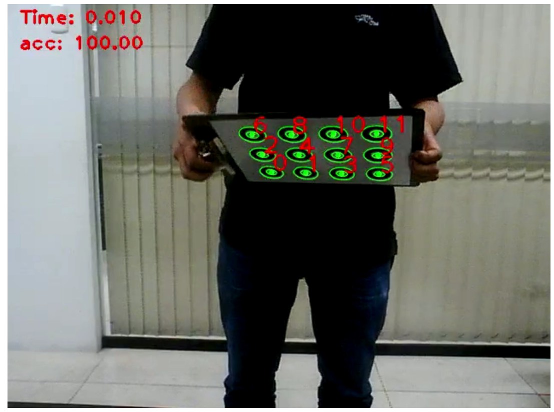

Fig. 3. Low illumination conditions y inclination

V. Conclusions

The implemented adaptive threshold algorithm was very useful allowing correct segmentations in multiple illumination conditions.
The algorithms used for reducing false negative results have good results. Even appear some false negatives but only in a little percentage of the frames.
Similarly the method used for tracking achieve the goal with the rings in the video. In the process is necessary a relocation algorithm that find the rings lost in any frame.