Camera calibration Using Deltille Pattern

Erick Tornero Tenorio
Universidad Catlica San Pablo
Walter Zuniga Herrera
Universidad Catlica San Pablo

Abstract—The pattern detection is one important part of the computer vision, for that a correct for that reason to obtain the best results in later steps it is necessary to have the best way to calibrate a camera, this changes depending on the proposed scenario, in this paper we try to implement a calibration form proposed in [1], to obtain more faithful results .

Index Terms—Pattern, Calibration, Machine Vision, Filter

1 Introduction

Camera calibration is one of the main steps in computer vision, it is useful to get information about the world from images. The are several ways to calibrate the Camera, in this work a method using a well known pattern and some images is used, this method was proposed by Zhengyou[1]

There are several ways to perform a detection of patterns, since the selection of what kind of pattern to use, it could be a mesh of squares or like in this work a mesh of concentric rings, to the selection of filter and its parameters.

The purpose of this paper is to evaluate two differents ways to make identification of ellipses, and create an heursitics to remove the external noise in the image, in this case noise will be all the ellipses outside the mesh of patterns.

December 12, 2018

2 Steps

2.1 Space color

The raw image is converted to grayscale for two reasons, to reduce number of computations, and because the color in most situations does not provide any relevant information, in this particular scene to use the following filters is necessary work in just a single channel.

2.2 Preprocessing

Filters are used in order to remove unnecessary information or remove noise from the image or just to get relevant information depending on the application as was done in the calibration of normal patterns you also have to remove the unnecessary noise.

To remove noise for high frequency, a Gaussian filter is applied, See figure 1 Apart of that threshold filter is applied.

2.2.1 Adaptive Threshold

In this case just a Adaptative threshold is aplyed, tha reason id that in the proccess of building the program, find that the method works best when only this method is used, instead using an integral threshold resulted in less quality calibrations images.

Solve the problem of changes and lighting, obtaining a degree of error, but the use of this technique has a high computational cost.

Not all the noise is deleting but, we get better perform of the patter.

2.3 Monky Saddle Detection

After having the iamgen in a form with less noise, we go for the obtaining of the monkey saddle, the pipeline for the obtaining of these is to try to find it with the equation, which is the intersection of the 3 planes, this to have a greater contrast we get a better saddle point and a better position of the true center, even when the image does not have an indicated resolution.

2.4 Identification and tracking of pattern elements

2.5 Ditribution

To have a better calibration, we have to use a distribution method, this is done through circles in the calibration image where we assume the best frame is found.

The method to obtain better frames was to try to have a uniform distribution over the entire visual field of the camera. For this we first divide the window, take the center of the pattern and check if it is within one of the divisions.

We go through different divisions to ensure the best distribution.

2.6 Initial State

In the end we get a distribution equal to this, where we take the best frames of the entire video.

This is the initial distribution window, it will go through different divisions to ensure that it is distributed correctly.

2.7 Final State

Once we obtain the necessary frames for the calibration, a first attempt is made to calibrate the camera, which we will improve iteratively, only taking the frames that we have correctly found, a refinement is made between the Parallel Fronto points and the reprojection .

In the final distribution we can see how the pattern has moved through the image, in the circles they are the places where the frames are better defined.

3 Conclusion

Using the Integral Threshold algorithm consumes much more processing time than using the Adaptive Threshold algorithm, this

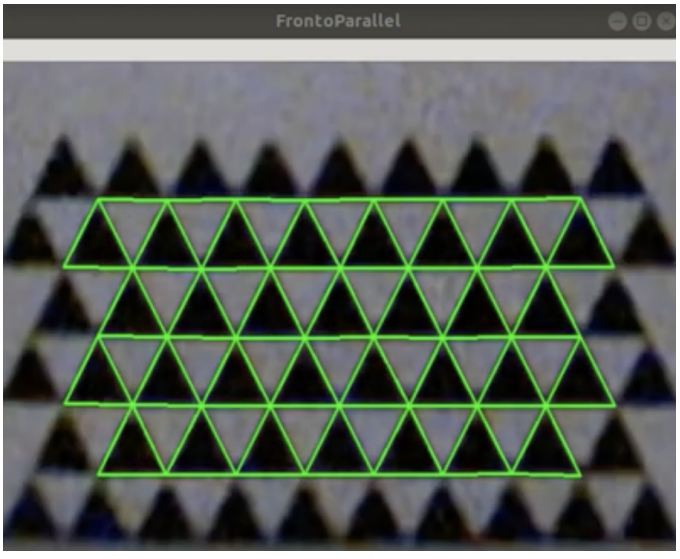

Fig. 1. Finding monkey saddle

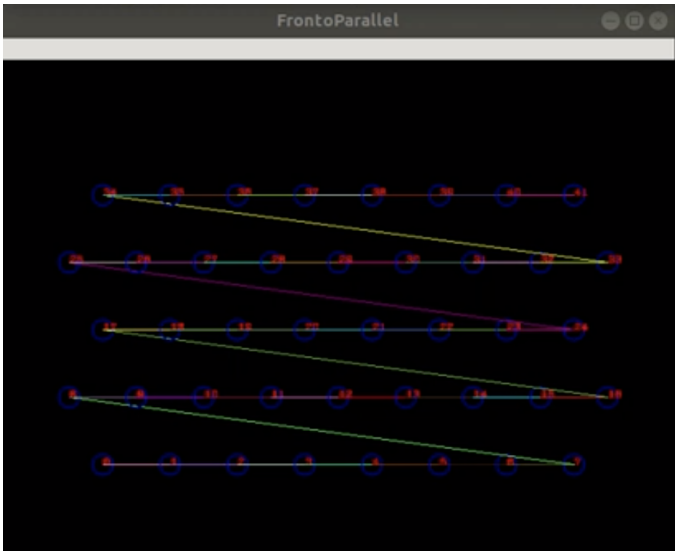

Fig. 2. Applying Adaptative Threshold filter

difference between Thresholding is seen in the 16,716 milliseconds that Integral Threshold takes against the 10.8667 milliseconds of Adaptive Threshold, this difference of 5.8493 milliseconds allows having a processing in real time, and even the first is an algorithm with better results, this can be achieved in the same way with the heuristic of constricted ellipses.

REFERENCES

[1] Zhengyou Zhang, *A Flexible New Technique for Camera Calibration*, 1998

[2] Shubham Rohan Asthana, *Enhanced Camera Calibration for Machine Vision using OpenCV*. In International Journal of Artificial Intelligence, Septiembre 2014.

[3] Dereck Bratley, *Adaptative thresholding using the integral image*. In Journal of Graphics, GPU, and Video Game Tools, August 2014.

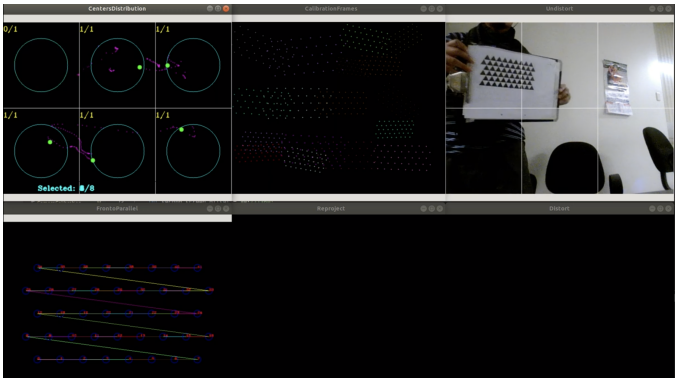

Fig. 4. Final recognition result

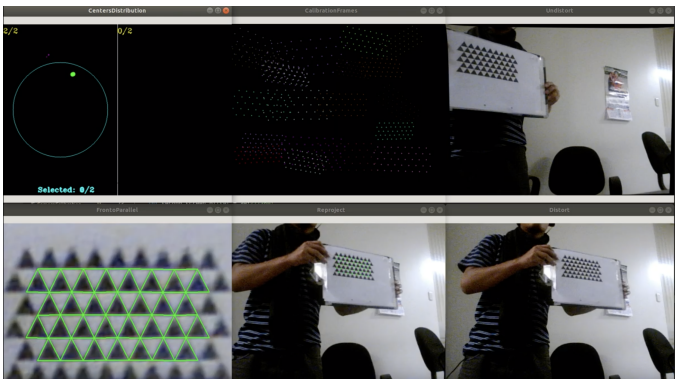

Fig. 5. Final recognition result

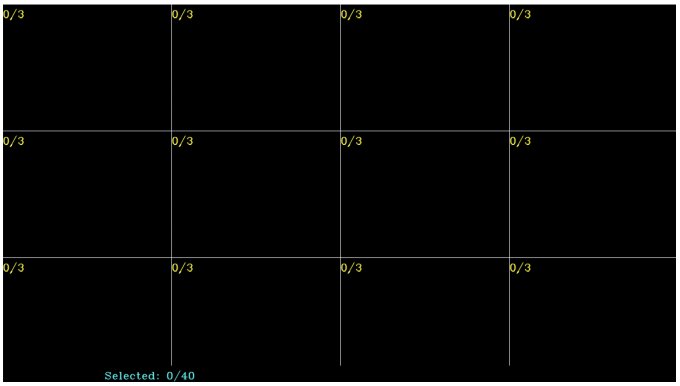

Fig. 3. Final recognition result