Camera calibration using OpenCV and using Iterative Refinement of Control Points (Rings Pattern)

Approach Final

Wilderd Mamani & Roxana Soto

Universidad Católica
San Pablo
Arequipa - Perú

Outline for Today

Pipeline

Camera Calibration && RMS

- Patterns: Rings
- RMS analysis without ankur

Ankur approach && implementation

- Front-Parallel
- Paper Review Algorithm Ankur
- Final Results: Opencv vs Ankur
- Results
- Conclusions

Our Pipeline

Canny increase more noise

Noise Reduction :

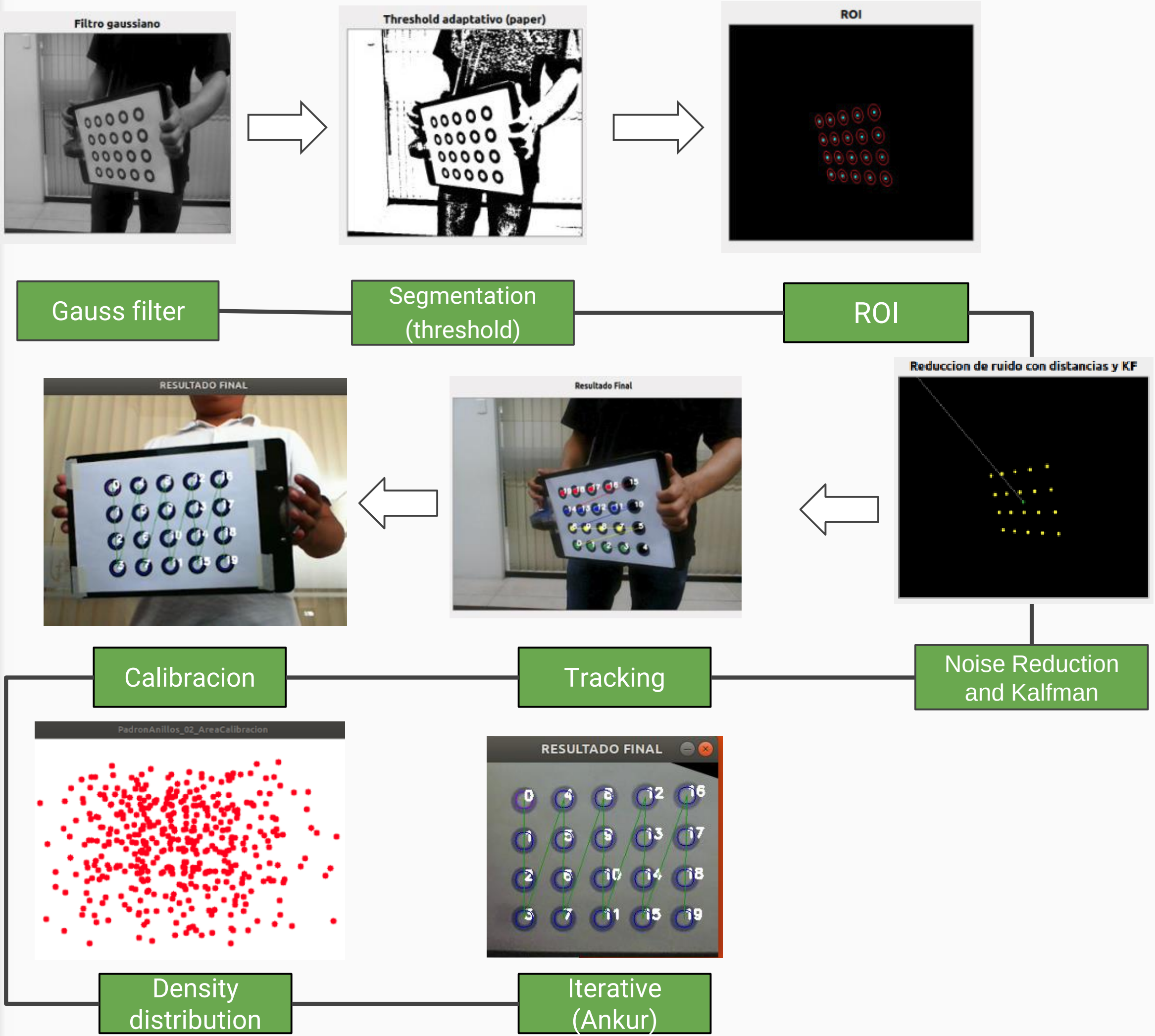
- DFS,MET,AM

Tracking:

- Rectangle Rotation
- Using minAreaRect

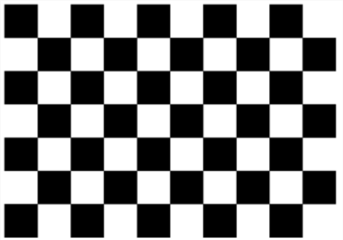
Calibration

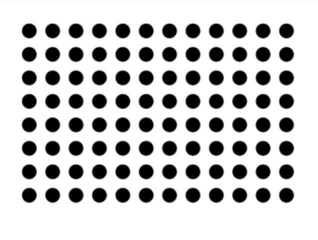
- Using opencv
- Iterative for improve RMS

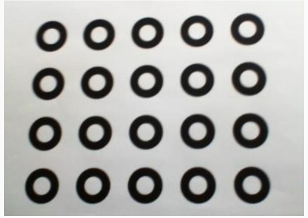


Tracking and Camera Calibration

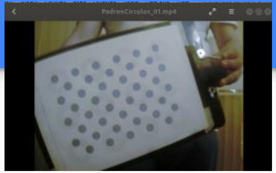
Patterns: CHESSBOARD, CIRCLES_GRID, ASYMMETRIC_CIRCLES_GRID







Calibration Opencv to Circles and Chessboard



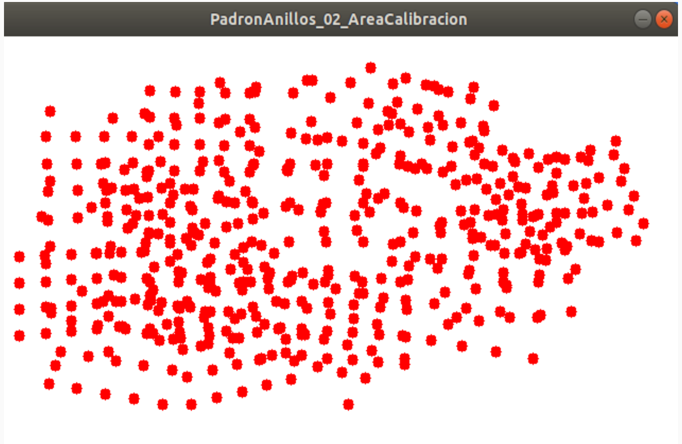
P	F_x	F_y	C_x	C_y	RMS
$N_F = 25$					
0	611.848062	620.398131	319.783601	273.558949	0.54837121
1	677.635412	690.63589	719.55126	658.36266	0.6253254
$N_F = 35$					
0	699.972124	340.587525	701.081777	245.975391	0.222251
1	672.164181	672.164181	672.164181	672.164181	0.5959

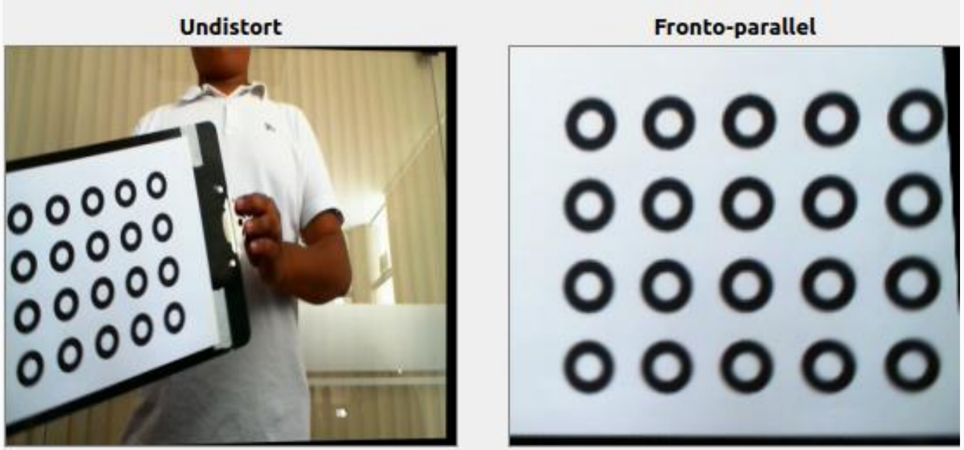
TABLE 1. PS3 CALIBRATION RESULTS

P	F_x	F_y	C_x	C_y	RMS
$N_F = 25$					
0	608.977979	601.998482	342.519779	227.219724	0.23051
2	525.377012	524.315352	358.92688	159.616455	0.459279
$N_F = 35$					
0	509.876941	508.940996	276.055842	206.17429	0.372094
1	520.815631	511.098022	310.493886	164.373091	0.421195

TABLE 2. LIFE CAMERA CALIBRATION RESULTS

Fronto - Parallel





Manual, intervals, RANSAC

findHomografy

warpPerpective

Fronto parallel to Rings

P	F_x	F_y	C_x	C_y	RMS
Ct = PS3 Camera					
0	669.317504	656.649057	389.398628	269.763956	0.53092
Ct= $LifeCamera$					
0	511.257561	510.402286	333.912424	199.896073	0.324985

TABLE 3. CALIBRATION RESULTS USING DENSITY DISTRIBUTION

Review Paper - Ankur

Objective

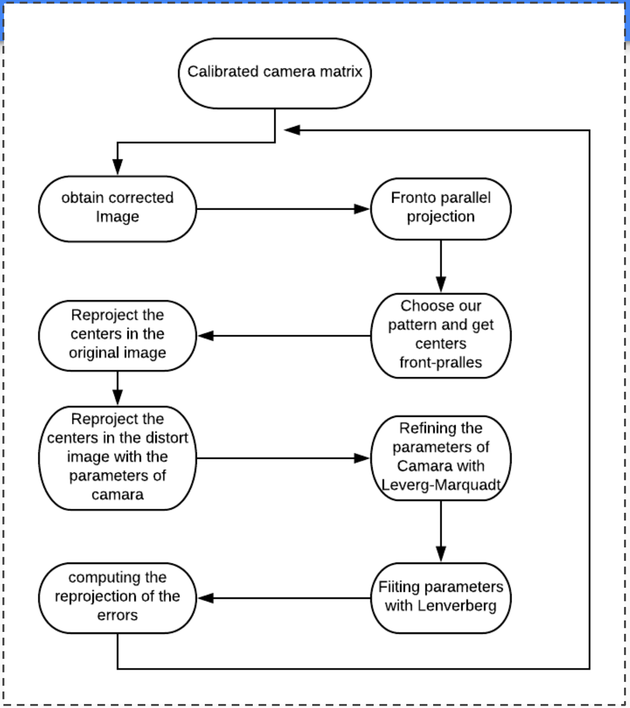
Given N images of the planar calibration grid, estimate the camera parameters.

Algorithm

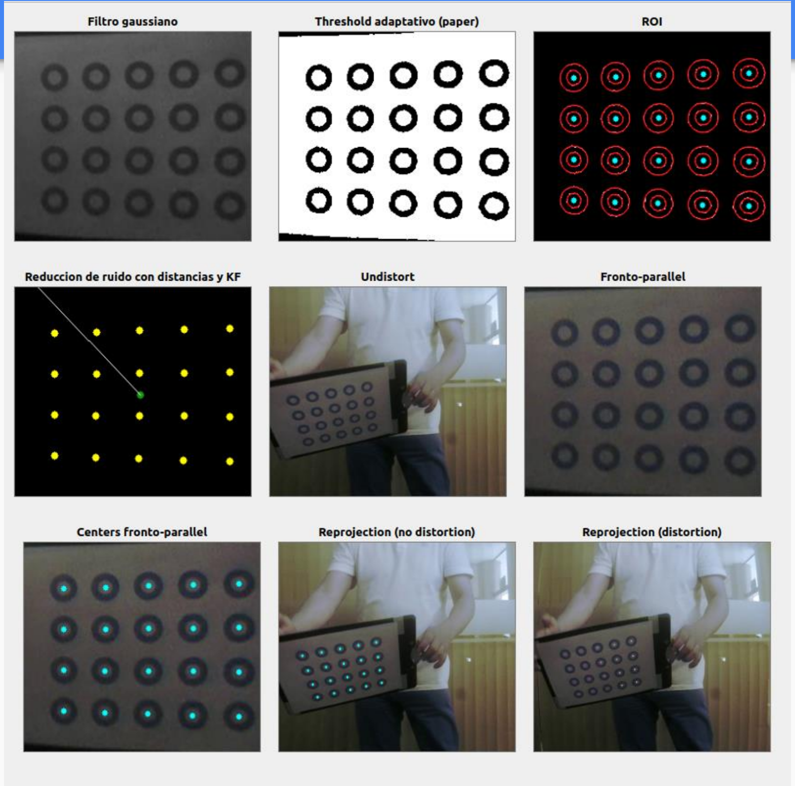
1. **Detect control points**: Detect calibration pattern control points (corners, circle or ring centers) in the input images.
2. **Parameter Fitting**: Use the detected control points to estimate camera parameters using Levenberg-Marquardt [1].

Do until convergence

1. **Undistort and Unproject**: Use the camera parameters to undistort and unproject input images to a canonical pattern.
2. **Localize control points**: Localize calibration pattern control points in the canonical pattern.
3. **Reproject**: Project the control points using the estimated camera parameters.
4. **Parameter Fitting**: Use the projected control points to refine the camera parameters using Levenberg-Marquardt [1].



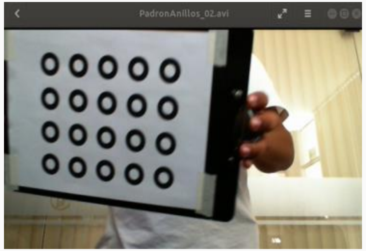
Iterative Method



Results Opencv vs iterative method

	Chessboard	Assim. disks		Concc. rings	
	RMS	RMS	iterative	RMS	iterative
30	0.6253254	0.548371206	0.513531	0.678611	0.303971

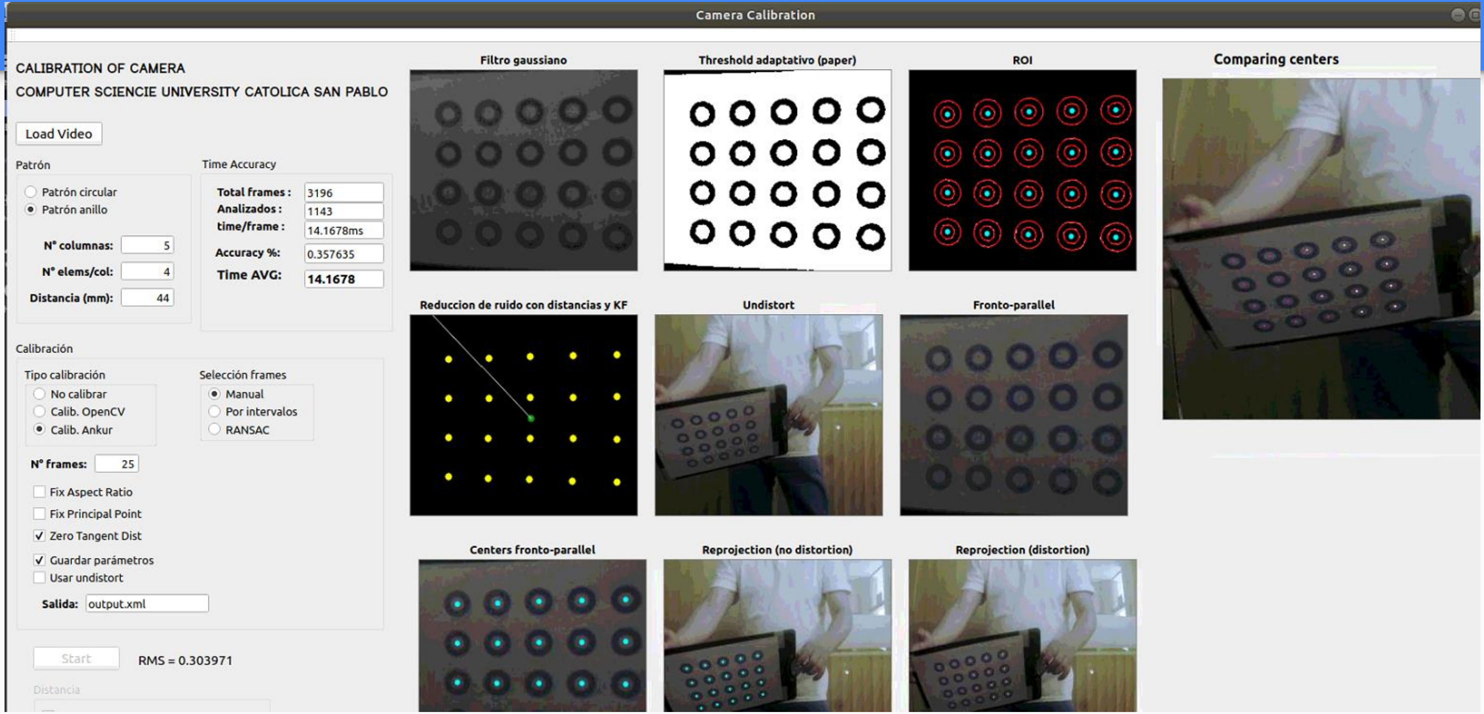
TABLE 4. ITERATIVE REFINEMENT CALIBRATION RESULTS USING DENSITY DISTRIBUTION CAPTION FOR PS3 CAMERA



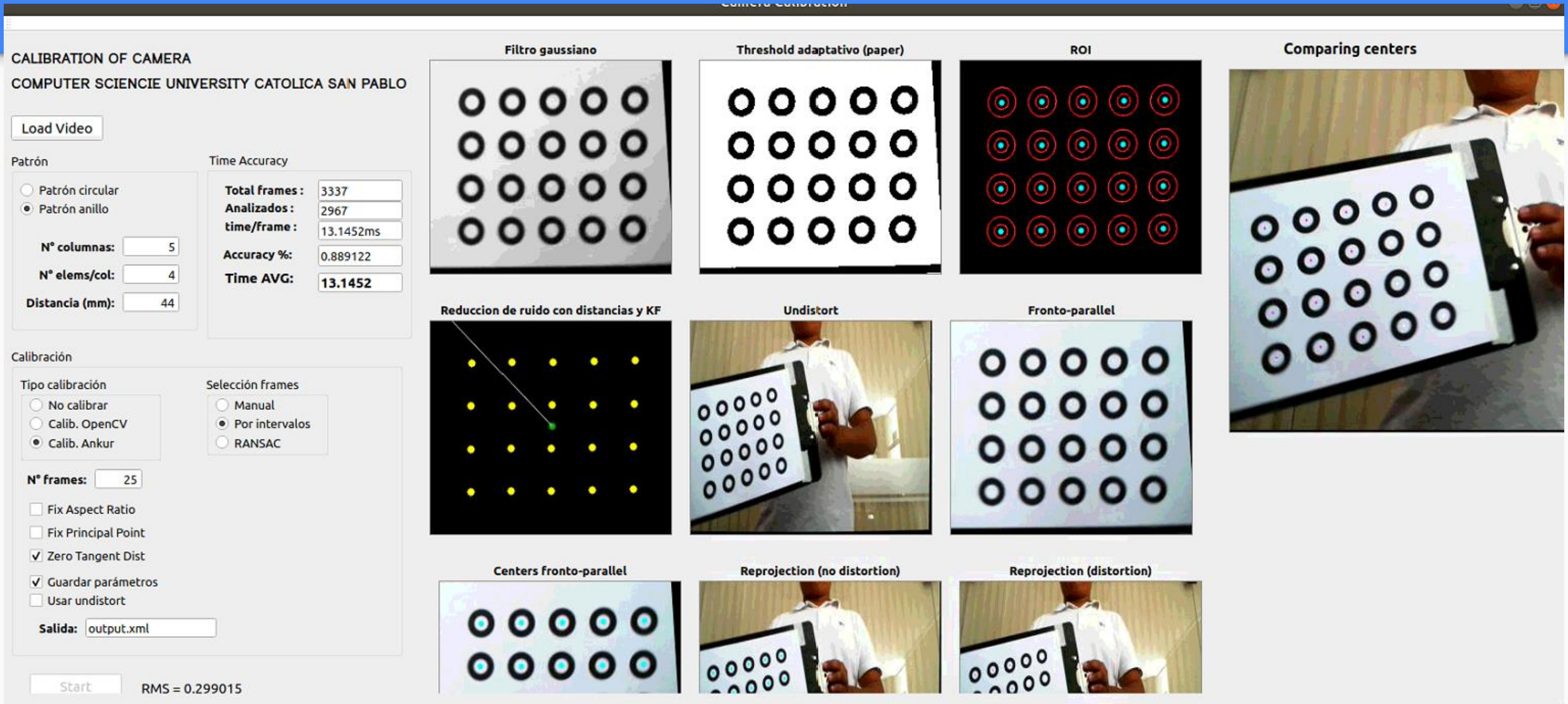
	Chessboard	Assim. disks		Concc. rings	
	RMS	RMS	iterative	RMS	iterative
30	0.372094	0.421195	0.313727	0.324985	0.299

TABLE 5. ITERATIVE REFINEMENT CALIBRATION RESULTS USING DENSITY DISTRIBUTION CAPTION FOR LIFE CAMERA

Results video 1



Results video 2



Demo - videos.mp4

demoCaliAnkur1.rar

Conclusions

In this implementation it is observed that for the recognition and calibration of cameras we have to apply a series of filters and algorithms to obtain a recognition of the sample template, in addition to adjusting the appropriate parameters we have to use an empirical way, segmentation and noise reduction should be applied to the image, finally when applying the algorithms to find the rings and circles on the real time video , we have to apply a heuristic of Ellipse fit for to find our images and a series of criteria to be taken into account and all characteristics were established according to the
empirical experiment

As seen in the results, the asymmetric circles and rings patterns are better than the chessboard because they reduce the error by detecting their centers instead of edges and vertices, which usually tend to fail. This generates an error in the calibration calculus obtaining very variableand distant results. This phenomenon is corrected using the iterative method but it does not apply to the rest of patterns. Between the assymetric circles and concentric rings, the last one have several concentric circles that help reduce the error when calculating the centers of the pattern. The iterative method does not imply a great advantage (in comparison with results obtained selection density function), it helps to reduce the error but it is not very significant, only in cases of cameras with greater distortion it is possible to appreciate like in ps3 case and for patterns that do not have much noise like concentric rings. About results presented on [7], which we are trying to reproduce or at least understand and extrapolate to our case (PS3 and LifeCam cameras), we conclude that improvement percentages are proportional to the distortions level presented in cameras used.So if a video has little distortion (LifeCam) will not be improved on the corrected image, but for videos with high distortion (PS3) this correction
becomes most notorious.

References

1] Zhang Z. (1998) A flexible new technique for camera calibration. Technical Report MSR-TR-98-71, Microsoft ReSearch.
[2] Meier, Thomas, and King Ngi Ngan. ”Automatic segmentation of moving objects for video object plane generation.” IEEE Transactions on Circuits and Systems for Video Technology 8.5 (1998): 525-538.
[3] Cucchiara, Rita, et al. ”Detecting moving objects, ghosts, and shadows in video streams.” IEEE transactions on pattern analysis and machine intelligence 25.10 (2003): 1337-1342.
[4] Yilmaz, Alper, Omar Javed, and Mubarak Shah. ”Object tracking: A survey.” Acm computing surveys (CSUR) 38.4 (2006): 13.
[5] Yang, Changjiang, Ramani Duraiswami, and Larry Davis.”Fast multiple object tracking via a hierarchical particle filter.” Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on. Vol. 1. IEEE, 2005.
[6] Szeliski, Richard. Computer vision: algorithms and applications. Springer Science and Business Media, 2010.
[7] Datta, Ankur, Jun-Sik Kim, and Takeo Kanade. ”Accurate camera calibration using iterative refinement of control points.” Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on. IEEE, 2009