Classic Methods on Color Based Ball Tracking

Breno Leite
Guilherme Leite
Instituto de Computação — UNICAMP

Abstract

Abstract

1. Introduction

Ball tracking is a classical problem present in a diverse range of applications. Since the early days of robotics where it was used to track balls in a controlled environment up to today's sports coverage with cluttered background, occlusion, many players in field and camera's shot with spectators in the background. To cope with such noisy situations ball tracking nowadays has to make use of a range of techniques to operate, like training a neural network to learn and detect a ball at a given frame, use a physics model to estimate the probable trajectory of the ball or use tiny details such as the way a player moves when in possession of the ball.

In this work we focus on the classical methods that lead these researches up to this point today, in Section 2 we cover some the literature around the problem, classical and modern approaches to solve the problem, in Section 3 we explain our step by step into constructing the solution, in Section 4 we cover the experiments performed, explaining the weakness and strength of the solution for each experiment and how they were solved in the next one, at last we show how the solution performs in a real world environment, a Robocup's soccer match and in Section 5 are related the importance of implementing this work and how far this branch of computer vision has come.

2. Related Work

Ball Tracking has been studied for a long time, in a diverse range of applications. Some projects are interested in soccer's ball tracking [9, 13, 2], others are interested in volleyball [1] and basketball [6]. Normally, These works use a wide range of techniques to overcome the problems, in which motion estimation, Hough circle transform and Kalman filter are included.

In this project the focus is on colored balls, which was very popular in the robot-cup competition [4, 10, 5]. In order to overcome the problem some usual techniques were studied. Lucas and Kanade motion flow [12, 11] was studied in order to reduce total cost of the algorithm. It predicts the motion flow, which helps the algorithm of detection to be faster and reduce computational cost.

Another technique used in the project is the Hough circle transform [3], which enable the algorithm to find circles in the data turning it more robust to noise present in the scene. And lastly, Kalman particle filter was studied in order to overcome occlusion problems [7], a vast number of tracking problems use the power of Kalman filter to improve the tracking performance [8].

3. Methodology

Initially the tracking was done alone by a color detector., using the color space HSV to select a color range that would create a mask. In this mask every pixel in the range is set to a white color and everything else to black, afterwards two erosion and dilation are applied to reduce noise. In sequence, the connected components are extracted from the resulting mask in order to filter out small areas, the resulting components are possible balls found in the scene. Each of these selected regions has a center of mass, the coordinate of this point is the supposed ball's position in the scene, later on this information will be called "tracking position".

After detecting these supposed areas ("balls") a Hough Circle Transform is used, and it tries to fit circles in these areas. The Hough Circle Transform depends of two parameters passed to the Canny Edge Detector, which is used inside the Hough algorithm. One of this parameters is directly related to the number of edges found by the Canny algorithm, high number of edges means more circles found, and low number of edges indicates the opposite.

In the videos a high value ($param2 = 200$) was used to this parameter, and it was decreased while no circles were found. This approach ensures the algorithm to get best circle predictions, but it has a really slow performance. In order to make the system to work on real time, this parameter needs to be set as a small value ($param2 = 80$), which could decrease the precision of the detection. Using Hough

Circle Transform the tracking narrows down to track only the roughly rounded shapes in the mask.

Applying the color detection every frame is not efficient, even worse when using Hough transform to filter the results. In order to overcome this problem the LK motion flow was implemented, enabling the detection to be applied in every N frames, which is called re-sampling. There is a trade-off using the LK motion, as higher the re-sampling worse is the precision of the model, and better is the efficiency.

LK motion flow is enough to estimate the ball position, but it always depends on the last frame. In a scenario in which the ball is occluded for more than a couple frames LK motion will completely lose its estimative. In order to overcome this problem, Kalman filter is used to keep predict the ball trajectory when it is occluded using previous estimates.

4. Results

The experiments were designed to stress the effect of each new feature that was incrementally added. A controlled environment was set to perform these experiments, including a blank background, artificial white lighting, fixed camera and three balls of the same size and material, two yellow and a blue one.

Single color detection:

The goal was to precisely track a single ball throughout the camera's view.

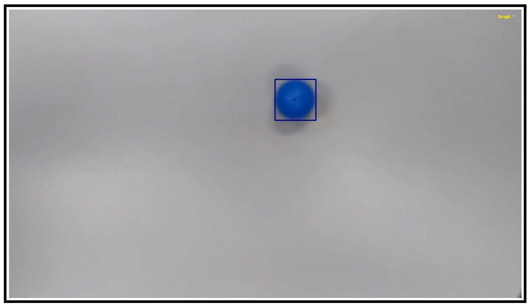

Figure 1. Detection of a colored ball.

The color detection applied to every frame was able to track the ball throughout the camera's view. This solution showed itself insufficient when some objects were introduced, with different shapes other than of a sphere but same color as the ball.

Two different color tracking:

The goal was to track each colored ball throughout the camera's view, not ever mistaking one with another.

The fine tuning of the colors enabled the solution to isolate one ball from the other, tracking them throughout the camera's view. This limits the solution to be tuned at every application, indoor and outdoor needs distinct calibrations.

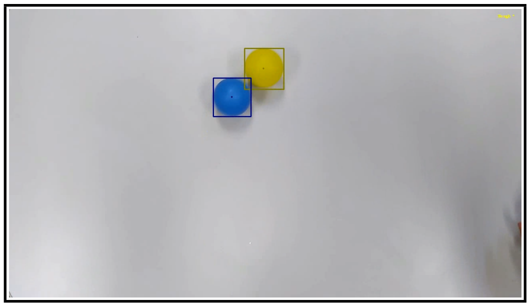

Figure 2. Detection of two balls with different colors.

Two same color tracking:

The goal was to track each ball throughout the camera's view, not ever mistaking one with another, now differentiating between two balls of the same color.

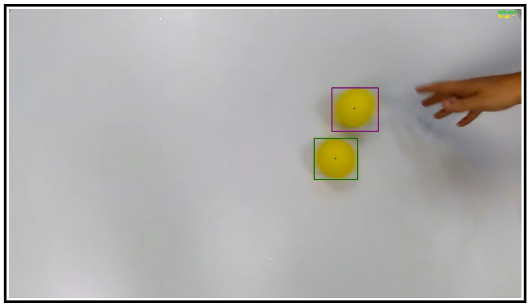

Figure 3. Detection of two balls with the same color.

The size of the connected components regions found in the mask were used the discriminate between the balls, the ratio between the first and second was used and enabled the tracking to work, the solution is limited when the object of interest moves away from the camera and becomes too small to pass the threshold.

At this point the solution is able to detect and track the balls, it lacks the ability to differentiate between $yellow\_ball\_1$ and $yellow\_ball\_2$, it also mistakes other objects for balls and is quite expensive by detecting the balls every frame.

Ball tracking alongside with other shapes:

The goal was to track a ball in a setup with many objects with different shapes, but same color as the ball.

As seen in Figure 4 the solution wasn't able to differentiate the ball from the pen top. It happens because the shape of the object has no weight in the detection process, so every object big enough and in the color range of the chosen color will be detected as a ball. To solve this we used the Hough Circle Transform.

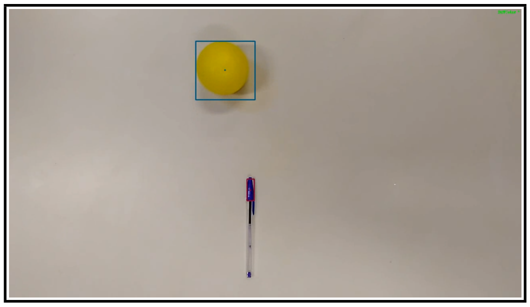

Figure 4. Detection before the Hough Circle Transform.

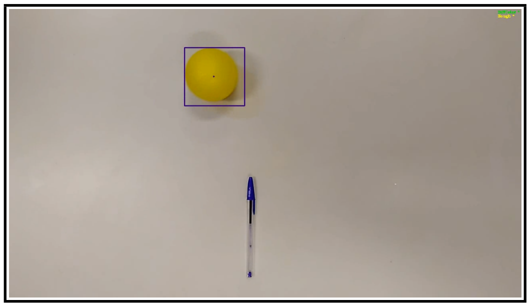

Figure 5. Detection after the Hough Circle Transform.

In Figure 5 using the Hough Circle Transform the solution was able to ignore the pen top and track only the ball shaped object. This solution is still limited by the time it took to adjust the parameters into differentiate a pen top and a ball.

At this point the detection process is producing unambiguous results, but since it happens at every frame it became too expensive to run in real time.

Motion flow tracking compared with color tracking:

The goal was to compare both solutions, Lucas Kanade motion flow tracking and every frame color detection.

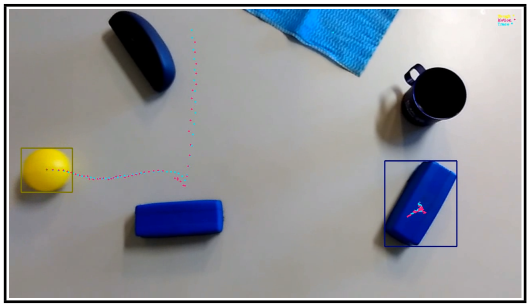

Figure 6. Lucas Kanade motion flow applied every 30 frames.

Using the Lucas Kanade motion flow the solution would only apply the color detection every N frames and estimate the ball position in between detection, this gap eliminated the need to detect it every frame enabling it to perform in real time. The trade off is that the precision of the tracking can diminish substantially depending on the gap size, and this experiment showed that a gap of 30 frames was too large.

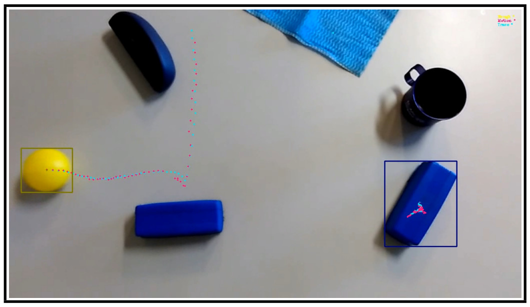

Figure 7. Lucas Kanade motion flow applied every 5 frames.

Using a 5 frames gap the precision of the tracking was close to the color tracking and the solution was still fast enough to operate in real time. Although this solution couldn't keep tracking of the ball when it was occluded.

Occlusion tracking:

The goal was to compare the LK motion flow tracking and kalman filter tracking, both applied towards an occlusion scenario.

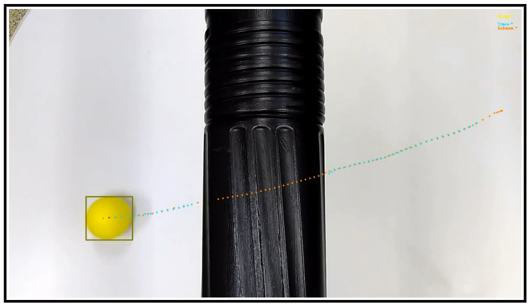

Figure 8. Kalman filter applied to overcome occlusion.

The solution was able to overcome occlusion when the kalman filter was used in such scenarios, since it was able to estimate the ball's trajectory. The filter worked best with balls in low speed and couldn't estimate a position if the ball changed directions whilst occluded.

Real world tracking:

The goal was to hard test all the features implemented in a real world scenario to evaluate their precision.

When applied to a real world scenario the solution can keep tracking up to a point, it fails whenever the ball drastically changes direction, specially occluded, or when it moves too fast.

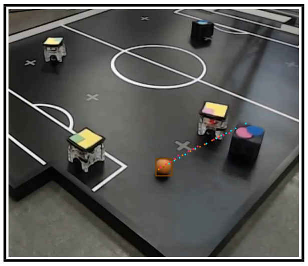

Figure 9. All features applied in a Robocup soccer match.

5. Conclusion

As seen the classic methods were able to track and overcome some of the challenges presented in this project. It could even work good in a real example like the Robocup soccer match, however it is noticeable that the algorithm's precision is worse than in a totally controlled environment.

In harder scenarios, as a soccer, volleyball, and basketball games the algorithm had bad results, in those scenarios physics models are used to describe the parabolic trajectory of the ball, or even more complex models, which uses different clues, like the fact that a basketball player handling the ball o moves differently from the others.

In this project many of the course's topics were covered and applied, clearing our sights towards their weakness and strength in each scenario. As future work, the color based detection could be replaced with a deep neural network in order to locate the initial position of the ball. This approach could improve the overall results, and enable the algorithm to deal with harder environments.

References

[1] H.-T. Chen, H.-S. Chen, and S.-Y. Lee. Physics-based ball tracking in volleyball videos with its applications to set type recognition and action detection. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 1, pages I–1097. IEEE, 2007.

[2] K. Choi and Y. Seo. Probabilistic tracking of the soccer ball. *Statistical Methods in Video Processing*, pages 99–123, 2004.

[3] J. Illingworth and J. Kittler. The adaptive hough transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (5):690–698, 1987.

[4] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa. Robocup: The robot world cup initiative. In *Proceedings of the first international conference on Autonomous agents*, pages 340–347. ACM, 1997.

[5] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawai, and H. Matsubara. Robocup: A challenge problem for ai and robotics. In *RoboCup-97: Robot Soccer World Cup I*, pages 1–19. Springer, 1998.

[6] A. Maksai, X. Wang, and P. Fua. What players do with the ball: A physically constrained interaction modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 972–981, 2016.

[7] B. Ristic, S. Arulampalam, and N. J. Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech house, 2004.

[8] Y. Satoh, T. Okatani, and K. Deguchi. A color-based tracking by kalman particle filter. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 502–505. IEEE, 2004.

[9] Y. Seo, S. Choi, H. Kim, and K.-S. Hong. Where are the ball and players? soccer game analysis with color-based tracking and image mosaick. In *International Conference on Image Analysis and Processing*, pages 196–203. Springer, 1997.

[10] M. Simon, S. Behnke, and R. Rojas. Robust real time color tracking. In *Robot Soccer World Cup*, pages 239–248. Springer, 2000.

[11] C. Tomasi and T. Kanade. Detection and tracking of point features. 1991.

[12] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[13] X.-F. Tong, H.-Q. Lu, and Q.-S. Liu. An effective and fast soccer ball detection and tracking method. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 4, pages 795–798. IEEE, 2004.