

Golf Ball Tracking Application for Putting Stroke Analysis

Peter Lee
Dept. of Mechatronics Engineering
University of Canterbury
Christchurch, New Zealand
cle102@uclive.ac.nz

Richard Green
Dept. of Computer Science and Software Engineering
University of Canterbury
Christchurch, New Zealand
richard.green@canterbury.ac.nz

Abstract— This paper proposes methods for tracking golf balls during putting, utilising mask filters, region of interest segmentation, morphology tools, and circle approximation algorithms. Given the surge in interest in golf post-COVID, there is a need for effective feedback mechanisms to improve putting skills, which significantly impact overall golf scores. Three main approaches are evaluated: Hough Circle, Differencing, and Contour Circularity. Among these, the Contour Circularity method demonstrates the most effectiveness, with a true positive detection rate of 37.5%. This is less effective compared to a prior study which tracked ping pong balls and achieved an average detection accuracy of 80.5%. However, there are distinct differences between the two studies such as the control environment and experiment setup, so a direct comparison is difficult. Challenges persist due to the dynamic outdoor environment and the small size of the golf ball compared to background noise. Future research directions include combining multiple methods for improved accuracy and exploring techniques to identify the putter head to refine the region of interest. Despite limitations, this study provides valuable insights into developing robust systems for golf ball tracking, with potential applications in sports analytics and performance improvement.

Keywords: golf, ball tracking, putting, putting stroke

I. INTRODUCTION

Golf is the most played sport in New Zealand with more than half a million kiwis playing golf every year [1]. Post-COVID, the sport has seen a surge in numbers from both men and women, with membership numbers growing each year. One crucial area of the game, which is often overlooked by beginners, is putting. The “putt” is defined as a light golf stroke made on the putting green in an effort to place the ball into the hole (Fig. 1).



Fig. 1. A golfer attempts to hole a putt on the green. Adapted from [2].

While putting might not be as impressive or bombastic as smashing a ball hundreds of metres off the tee, it can be responsible for up to half of the strokes incurred on the golf course. Inspiring phrases such as, “Drive for show, putt for

dough”, improving a golfer’s putting skills is one of the most effective methods to reduce their score on the golf course.

This paper proposes a method to provide feedback to golfers on their putting stroke, using a video taken from a mobile phone camera. One of the aims of a good putting stroke, is to impart as much topspin on the ball at impact, which reduces skidding, hopping, and side-spin. Standard putter heads will have a loft of 3-4° from the vertical, therefore, at impact, the ball will launch up into the air with some backspin, skid, bounce, and then start rolling with topspin as shown in Figure 2.

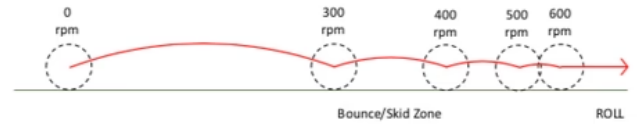


Fig. 2. Path of a golf ball after putter impact. Adapted from [3].

How the ball reacts after impact is influenced by the putter design, green conditions, and the putting stroke (Fig. 3). By tracking the ball path after impact, information about the putting stroke such as the putter head’s position and attack angle can be deduced.



Fig. 3. Incoming putter head position and attack angle affecting the ball’s trajectory. Adapted from [4].

By leveraging image processing algorithms, the paper proposes a robust and accurate system capable of identifying and tracking the golf ball throughout its trajectory, even in outdoor and uncontrolled environments. A methodology for golf ball tracking is proposed, encompassing various stages of image processing and analysis. By employing techniques such as colour space conversion, thresholding, contour detection, and circle fitting, the paper attempts to achieve robust detection and accurate tracking of the golf ball in video sequences captured during putting strokes.

The accurate tracking of objects in video sequences is a fundamental task with applications in various domains including sports analytics. Tracking the movement of a golf ball during putting strokes presents a unique challenge due to the ball's small size and fast motion.

II. BACKGROUND

Ball-tracking in golf was first visualised on TV screens by Toptracer in 2006 [5]. Figure 4 shows Toptracer [6] in action.



Fig. 4. Toptracer being used in a PGA tour event. Adapted from [6].

Although, this was not entirely accurate as the ball flight path would be estimated and drawn on manually. Nowadays, technology has improved greatly and golf ball-tracking is much more accurate, but it still remains a challenge to track a small white ball travelling at over 150 kmph outdoors. Along with Toptracer, another company - Shotracer [7], has provided a mobile app solution that is accessible to everyday golfers, allowing them to trace the ball flight on a video taken of their golf shot.

Prior research have looked at analysing the putting stroke to obtain a unique swing signature [8], or looking at golf swings to track either the club head or shaft [9], or for pose and joint tracking [10]. Other research has been done on tracking and estimating golf ball flight during full shots [11], however, none of these past research papers were looking at tracking a golf ball during a putting stroke. The most similar and applicable research that could be found were from those tracking other sports balls such as squash, cricket, tennis, and even beer pong balls [12].

A common limitation of these papers is that the ball-tracking algorithms only work in strict controlled environments (usually indoors), and cannot easily deal with the dynamic and chaotic conditions of nature outdoors. Previous research in object tracking has primarily focused on techniques such as optical flow, Kalman filtering, and deep learning-based methods. However, these approaches often struggle with tracking small, fast-moving objects like golf balls accurately. Limitations of existing methods include sensitivity to noise, occlusions, computational complexity, and requiring strictly controlled environments.

III. PROPOSED METHODS

A video will be taken of the golfer's putting stroke, and then the video will be processed and analysed frame by frame.

A. Equipment setup

Since the golf ball is small compared to the rest of the background, pixel quality and frame rate are important parameters in capturing a sufficient image quality suitable for processing. To track the golf ball, the camera should be positioned face-on from the golfer as shown in Figure 5.



Fig. 5. Camera orientation and setup.

An earlier study tracking a squash ball indoors [13] discusses further methods for calculating camera placement locations, minimum frame rate, and minimum camera resolution.

B. Ball detection methods

1) *Hough Circles*: A common method of finding circular shapes in an image, Hough Circles uses the Hough Gradient method which is a variant of the Hough Transform. Before applying the Hough Transform, a Gaussian Blur is used to smooth the image, reducing noise and suppressing small variations in pixel intensity. Then, edge detection algorithms like Canny Edge are typically used to extract edges from the image. This reduces the problem of detecting circles to the problem of detecting edges.

Since the environment is outdoors and not strictly controlled, there is a lot of noise and false positives that are detected when using the function, as shown in Figure 6.

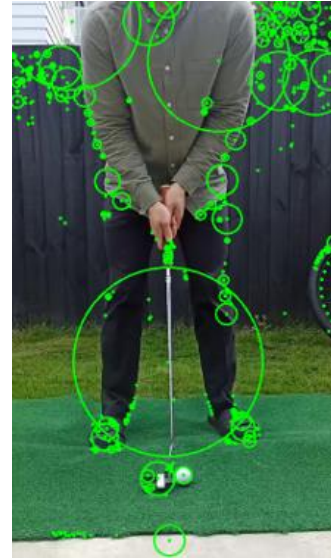


Fig. 6. False circles detected from Hough Circle function.

Parameters such as minimum and maximum radiiuses can be optimised so that only the golf ball will correctly be detected. However, this method requires parameter optimisation for any change in the video setup, which is not ideal when it is aimed to work for an outdoor environment. Also, after impact, when the ball inevitably suffers from motion blur, the algorithm struggles to detect the ball due to deformation from motion blur.

2) *Contour Circularity*: This proposed method detects and extracts contours from binary images. First, the image is

transformed into the HSV colour space [14] as shown in Figure 7.

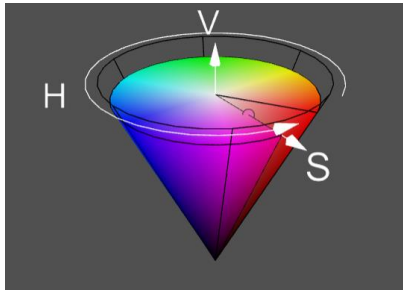


Fig. 7. HSV colour space. Adapted from [14]

Since the function requires a binary image to be fed into the argument, it requires pre-processing steps consisting of:

- BGR to HSV colour conversion
- Specify minimum and maximum HSV threshold
- Binary masking using HSV threshold range
- Morphological operations

The resulting binary mask is used to obtain a list of contours from the image. Other ball-tracking studies using contours usually assume the biggest contour is the ball due to having a strictly controlled environment, but in this case, the contour for a golf ball is small compared to other contours generated from background noise. Figure 8 demonstrates why this is problematic as when circles are drawn over each contour, the circle for the golf ball is much smaller compared to the circles generated due to background noise.



Fig. 8. Circles (red) drawn over each contour (blue).

3) *Deep Learning - Instance Segmentation*: Utilising the power of deep learning, and open-source models pre-trained on the vast COCO dataset, an inference can be run on any supplied image (Fig. 9).



Fig. 9. Instance segmentation.

An inference can be run on the first frame of the video to identify and locate the golf ball, and a boundary drawn around the region of interest (ROI). If a binary mask was applied to delete the background outside the ROI, it would make it easier to keep track of the ball and save processing power. Alternatively, an inference could be run on every frame if the machine running the code was not restrained by processing limitations.

C. Ball Tracking methods

1) *Differencing*: The difference between two images can be shown by subtracting the pixel values of one image from the corresponding pixel values of another image, for example, consecutive frames from a video. Mathematically, for each pixel (x,y) in the two images, the difference can be calculated as shown in (1):

$$\text{Difference}(x,y) = \text{image1}(x,y) - \text{image2}(x,y) \quad (1)$$

After differencing, thresholding is applied to identify pixels of significant change. Thresholding converts the difference image into a binary image where pixels with values above a certain threshold are set to white (255), indicating a significant change, while pixels below the threshold are set to black (0). The resulting binary difference image can be used for various purposes such as background subtraction, motion detection, or in this case, object tracking. Figure 10 shows a binary difference image between two consecutive frames, immediately after impact to show the motion of the ball. It is common to apply iterations of median blurring to the image to remove “salt and pepper” noise from the background.



Fig. 10. Image showing the difference between two consecutive frames.

It is worth noting that although the differencing method is good for detecting moving objects, it is unable to detect stationary objects.

2) *Pythagorean distance*: Given two circle (x,y) positions, (2) calculates the Pythagorean distance between a circle detected in a previous frame and a newly detected circle:

$$Distance = \sqrt{((x_2 - x_1)^2 - (y_2 - y_1)^2)} \quad (2)$$

In the new frame, it can be assumed that the circle closest to the previously detected circle is the golf ball.

3) *Kalman Filter*: The Kalman Filter is a recursive algorithm used for state estimation in systems with uncertain, noisy measurements and dynamics. It predicts the current state of a system based on its previous state and the dynamics model and updates the prediction using new measurements to produce a more accurate estimate. This process results in an optimal estimate of the system state, balancing between the predicted and observed information to minimize estimation errors.

D. *Ball path contrail*: To visualise the path of the ball, contrails can be drawn by appending the centre locations of each circle to a list [15] as shown in Figure 11.



Fig. 11. Drawing the contrail of ball path.

IV. RESULTS

Table 1 shows the specifications of hardware and software that were used for the experiments.

Table 1: Specifications of hardware and software.

Item	Details
Mobile phone	Xiaomi Redmi Note 8 Pro
Video camera	1080 p @ 60 fps
Putter (golf club)	Odyssey Versa Seven (3° loft)
Golf ball	Titleist ProV1
PC (Laptop)	HP Probook 445 G10
CPU	AMD Ryzen 7 7730U, 2 GHz, 6 core, with integrated Radeon Graphics
RAM	16 GB DDR4
OS	Windows 11
IDE	Visual Studio Code
Language	Python 3.9
OpenCV	Version 4.9

From the proposed methods, the Contour Circularity method was explored further. The steps taken to evaluate its performance were:

A. Green detection

As shown in Figures 6 and 8, a lot of background noise causes false detection of contours and circles. In [16], Krishna proposes a method to detect the green region from the input video since the golf ball will always be on the green area, hence, the rest of the background information can be ignored and discarded.

It is known that the green is a static object of interest. Since the camera is also stationary, green detection only needs to be performed for the first frame of the video, then the coordinates of the green region can be used for the rest of the video frames. To detect the green in the first frame, the image was first converted to the HSV colour space.

1) *BGR to HSV colour space*: The contour method takes a binary image, so the image is converted to the HSV colour space (Fig. 12).



Fig. 12. Conversion from BGR to HSV colour space.

2) *Colour thresholding for binary mask*: Figure 13 shows a window with trackbars to find the HSV threshold range to detect the green [17].

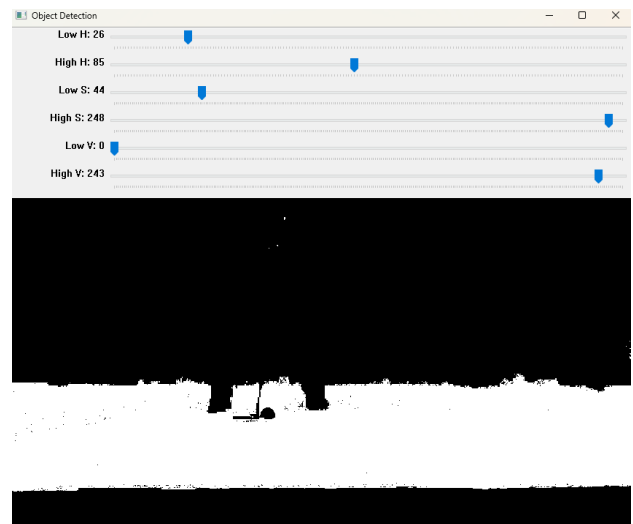


Fig. 13. Finding optimal HSV range for green detection.

Table 2 shows the green HSV ranges obtained for this test.

Table 2: HSV threshold range for ROI.

	Hue	Saturation	Value
Lower green	50	120	70
Upper green	110	255	255

The HSV ranges are then fed into a function which outputs a binary image by setting all pixels within the range to white, and all pixels outside the range to black. Opening and closing morphological operations are applied to clean up the binary mask.

3) *Removing background outside region of interest:* After finding contours from the binary mask, a bounding polygon is drawn around the biggest contour (assuming the green area makes up the biggest contour). Parameters can be adjusted to control how detailed the polygon approximates the contour; smaller values will result in more accurate approximations with more vertices, while larger values will result in fewer vertices with less accuracy (Fig. 14).

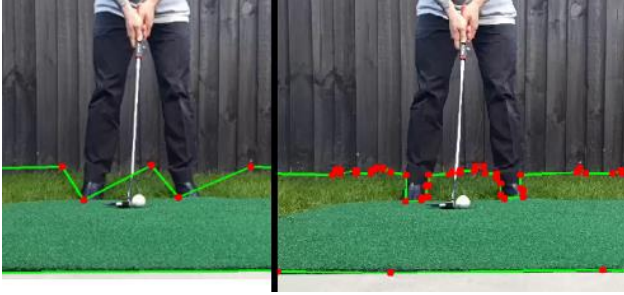


Fig. 14. Two polygons with differing contour approximations.

Now, a polygon boundary has been drawn around the region of interest (the green). This polygon is then used in a bitwise 'AND' function that outputs a mask where pixels within the polygon are kept, and pixels outside the polygon are discarded (Fig. 15).



Fig. 15. Result after background subtraction.

Now, the background noise has been reduced and the problem of finding the golf ball has become simplified. This bounding polygon can also be used for all subsequent frames of the video. After the green detection and background subtraction stage, the processed image can then be used for the ball detection stage.

B. Ball detection

To find the white golf ball in the green region, the same process for green detection is applied.

1) *Colour thresholding for binary mask:* The HSV threshold range is found again, but now for the white golf ball (Fig. 16).

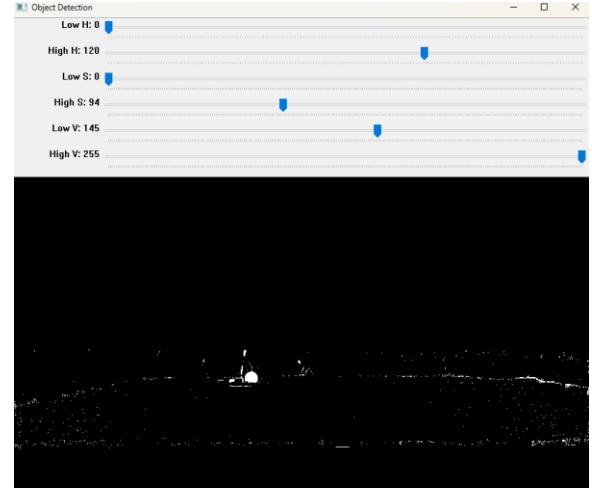


Fig. 16. Finding optimal HSV range for golf ball detection.

Table 3 shows the corresponding white HSV ranges obtained for this test.

Table 3: HSV threshold range for white golf ball.

	Hue	Saturation	Value
Lower ball	50	0	95
Upper ball	150	80	255

Again, opening and closing morphological operations are applied to fill gaps, and clean up noise in the binary mask.

2) *Find contours and draw circles:* Contours are found, and circles are drawn over each contour. As shown in Figure 17, the issue of false circles being detected on the green remains.



Fig. 17. Circles drawn over each detected contour.

A challenge with detecting golf balls is that the ball is relatively small compared to its environment. As seen in Figure 17, bigger contours are detected from the background

noise, hence, it cannot be simply assumed that the biggest contour or circle is the ball.

3) *Finding the best circle shape*: A method was found to calculate how well each contour in a frame fitted to a circle. A circularity score was given to each contour using (3),

$$\text{Circularity} = \frac{\text{Area}_{\text{Contour}}}{\text{Area}_{\text{minEnclosedCircle}}} \quad (3)$$

where a contour with a perfect circle shape would have a circularity score of 1, and non-circular contours will have a score less than 1. For each frame, the contour with the best circularity score would be selected and drawn.

C. Performance

Experiments were conducted by comparing the performances of the Contour Circularity method against other proposed methods. Success was measured by how many times the algorithm could correctly detect the moving golf ball after it had been struck (Fig. 18).



Fig. 18. Successful detection of golf ball after impact.

Table 4 shows the results of how each method performed.

Table 4: Ball detection results for different methods.

Test method	No. Frames	True Positives	False Positives
Hough Circle	32	0	0
Differencing	32	7	5
Contour Circularity	32	12	3

These results show that the out of the proposed methods tested, the Contour Circularity method was the most effective at detecting the golf ball correctly.

V. CONCLUSION

This paper tested three methods for tracking golf balls during putting strokes. The Contour Circularity method was shown to have the best results at correctly detecting an instance of the golf ball with a true positive detection rate of 37.5%, as compared to 0% and 21.9% for Hough Circle and Differencing methods respectively.

For comparison, a similar study that tracked ping pong balls [12] achieved an average detection accuracy of 80.5%. Distinct differences between the two studies is the controlled environment in which the tests were conducted. For example, the ping pong ball tracking was conducted indoors with a specific colour ball that would make detection easier.

A. Limitations

Since golf is an outdoors sport and there are many variables such as: lighting, green conditions, golfer's clothing, and putter type, unless the algorithm is optimised every time, the thresholding range for binary masking has to be inclusive enough to accommodate the different possibilities. However, this introduces a lot of noise into the image since any objects on the green region that are white can be confused for the golf ball.

B. Future research

While each one of the methods tested did not achieve a high success rate on their own, by combining different methods together, the performance could be improved. For example, the shaft or head of the putter can be identified first and knowing that the golf ball is adjacent to the putter head at address, a more precise region of interest could be created.

REFERENCES

- [1] Golf NZ, 'Role of Golf New Zealand'. [Online]. Available: <https://www.golf.co.nz/role-of-golf-nz>. [Accessed: 26- Apr- 2024]
- [2] Golf WA, 'WA Open Day 1 Highlights'. [Online]. Available: https://www.youtube.com/watch?v=ySGJWEppokY&list=FLCZjhKkWR6oCZTTiSy_TY0g&index=3. [Accessed: 1- May- 2024]
- [3] CurePutters.com, 'Skid, Roll, Launch and Loft myths'. [Online]. Available: <https://cureputters.com/blogs/news/skid-roll-launch-and-loft-myths-the-plain-truth>. [Accessed: 1- May- 2024]
- [4] Del Mar Golf Centre, 'What loft do you have on your putter? What rise angle do you strike the ball with?'. [Online]. Available: <https://delmargolfcenter.greensidegolfer.com/pages/what-loft-do-you-have-on-your-putter>. [Accessed: 29- Apr- 2024]
- [5] M. McQuade, 'Ball-tracking technology changes way fans consume, practice sport of golf', 2020. [Online]. Available: <https://www.bizjournals.com/phoenix/news/2020/08/15/ball-tracking-technology-changes-golf.html>. [Accessed: 30- Apr- 2024]
- [6] Toptracer.com, 'Toptracer'. [Online]. Available: <https://toptracer.com/tv/>. [Accessed: 22- Apr- 2024]
- [7] Shottracer.com, 'Shottracer'. [Online]. Available: <https://www.shottracerapp.com/>. [Accessed: 22- Apr- 2024]
- [8] M.S. Couceiro, D. Portugal, N. Gonçalves, et al. "A methodology for detection and estimation in the analysis of golf putting". (2013)
- [9] S. Hitchcock and R. Green. "Golf Swing Tracking and Annotation", Computer Vision Lab, University of Canterbury, Tech. Rep., 2023.
- [10] L. Hunn and R. Green. "Golf Swing Coach", Computer Vision Lab, University of Canterbury, Tech. Rep., 2020.
- [11] X. Zhang, T. Zhang, Y. Yang, Z. Wang and G. Wang, "Real-time Golf Ball Detection and Tracking Based on Convolutional Neural Networks," 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 2020, pp. 2808-281
- [12] J. Houghton and R. Green. "Ball Tracking and Shot Trajectory Prediction for Beer Pong", Computer Vision Lab, University of Canterbury, Tech. Rep., 2021.
- [13] J. Allred and R. Green. "Squash Ball Tracking in Three Dimensions", Computer Vision Lab, University of Canterbury, Tech. Rep., 2022.
- [14] L. Stratmann. 'HSV Color'. [Online]. Available: <https://web.cs.uni-paderborn.de/cgwb/colormaster/web>. [Accessed: 10- Apr- 2024]
- [15] Pyimagesearch.com, 'Ball tracking with OpenCV', [Online]. Available: <https://pyimagesearch.com/2015/09/14/ball-tracking-with-opencv/>. [Accessed: 30- Apr- 2024]
- [16] A. Krishna, P. Smith, M. Nicolescu and S. Hayes. "Vision-based Assessment of Instructional Content on Golf Performance". In

Proceedings of the 4th International Conference on Image Processing and Machine Vision (IPMV '22). Association for Computing Machinery, New York, NY, USA, 78–83, 2022.

- [17] OpenCV.com, 'Thresholding Operations using inRange'. [Online]. Available: https://docs.opencv.org/3.4/da/d97/tutorial_threshold_inRange.html. [Accessed: 10- Apr- 2024]
- [18] CodeSavant, 'Ball detection using OpenCV in Python'. [Online]. Available: <https://www.youtube.com/watch?v=RaCwLrKuSIw>. [Accessed: 30- Apr- 2024]
- [19] J. Porter and R. Magill. "Practicing along the contextual interference continuum increases performance of a golf putting task", 2005.
- [20] C. Lyu, Y. Liu, B. Li and H. Chen. "Multi-feature based high-speed ball shape target tracking". In *2015 IEEE International Conference on Information and Automation* (pp. 67–72), 2015.
- [21] H. Ren, W. Liu, F. Li, and C. Zhang. "Compressive tracking with multi-feature weighting based on kalman predictor". In *2016 Chinese Control and Decision Conference (CCDC)* (pp. 3349–3353), 2016.
- [22] M.A. Guadagnoli, W.R. Holcomb and T.J. Weber. *The relationship between contextual interference effects and performer expertise on the learning of putting task*. J Hum Mov Stud 37:19–36, 1999.