International Journal of Networking and Computing – www.ijnc.org, ISSN 2185-2847
Volume 16, Number 2, pages 166–181, June 2026

Robust Baseball Spin Rate Measurement using a Smartphone with Outlier Correction

Ryota Yamamoto
Graduate School of Science and Engineering, Saga University
1 Honjo-machi, Saga City, Saga 840-8502, Japan

Susumu Matsumae
Graduate School of Science and Engineering, Saga University
1 Honjo-machi, Saga City, Saga 840-8502, Japan

Received: February 13, 2026
Revised: April 29, 2026
Accepted: June 4, 2026
Communicated by Shuichi Ichikawa

Abstract

Spin rate is a critical performance indicator in baseball pitching. However, professional-grade systems are expensive, and even more affordable alternatives, such as sensor-equipped balls, still pose barriers to widespread implementation. This study aims to develop and validate a method for easily measuring baseball spin rate using a standard smartphone camera and machine learning techniques. The proposed method utilizes YOLOv8 for ball detection and cropping, followed by a CNN model to classify the ball's orientation into four discrete angles ($0°, 45°, 90°, 135°$) by leveraging the seam-pattern symmetry. To address the quantization noise and errors inherent in discrete classification, we introduce two key techniques: 1) an outlier correction algorithm based on temporal continuity, and 2) a cumulative linear regression method that estimates rotation speed from phase-unwrapped angles. Experimental results on 15 test videos demonstrated that the proposed method significantly outperformed the baseline angular difference method. Using a success criterion of error $\leq$ 8.9%, the method achieved a success rate of 90% with an average error of 5.7%, verifying that our approach effectively mitigates discretization artifacts and provides accessible, high-precision measurement.

Keywords: object tracking, spin rate measurement, deep learning, CNN

1 Introduction

In recent years, the utilization of various tracking data has become widespread in baseball training. For pitching, key metrics such as the spin rate of a baseball, its spin axis, and movement are actively used to analyze and improve performance. These data are typically acquired using specialized equipment. However, professional-grade systems, such as TrackMan [1] and Rapsodo [2], are expensive for most users. Although more affordable alternatives such as sensor-equipped balls [3] exist, their cost still poses barriers to widespread adoption, and there remains a need for specialized devices.

To tackle this issue with a more accessible device, prior studies attempted to measure the spin rate using standard smartphone cameras. However, these methods often suffered from instability due to environmental noise or limited data samples.

In this paper, we propose a robust and accessible spin-rate measurement pipeline that overcomes these limitations. Specifically, we introduce an estimation framework that automatically corrects sporadic estimation errors to ensure consistent tracking, combined with an improved calculation algorithm for stable spin rate estimation. The effectiveness of this approach is validated on an expanded dataset of 15 videos, demonstrating that the proposed method achieves practical accuracy and stability significantly superior to conventional approaches.

Our main contributions are as follows:

- We introduce an outlier detection and correction algorithm that exploits temporal continuity to mitigate sporadic frame-level misclassifications in the predicted angle sequence.
- We propose a Cumulative Linear Regression method with phase unwrapping to estimate spin rate robustly from cyclic, quantized angle predictions, and demonstrate improved accuracy over the angular-difference baseline on 15 test videos.

2 Related Work

2.1 Sensor-based Systems

To analyze pitching performance, dedicated hardware systems are widely used. Doppler radar-based systems such as TrackMan [1] and optical tracking systems like Rapsodo [2] provide highly accurate data and are standard in professional baseball. However, these systems are expensive and require large-scale installation. Sensor-embedded balls, such as Technical Pitch [3], offer a more portable alternative. While effective, they are still costly compared to standard baseballs and act as a consumable item, which limits their widespread adoption among amateur players.

2.2 Vision-based Approaches

To address the cost and hardware limitations of dedicated sensors, vision-based methods using cameras have been explored.

Ijiri et al. [4] proposed a system that estimates the spin rate and axis using consumer-grade high-speed cameras (e.g., Sony RX series). Their method utilizes background subtraction and image periodicity analysis to estimate spin parameters from videos recorded at 480 or 960 fps. While their approach significantly lowered the barrier compared to professional equipment, it still requires a specific high-speed camera, which is not as ubiquitous as a smartphone. Furthermore, their reliance on classical image processing techniques assumes high temporal resolution (high frame rate) to track the ball's phase changes accurately.

Nishida et al. [5] attempted to measure the spin rate using only a standard smartphone camera to tackle the hardware limitation with a more accessible device. Their method involved drawing a blue mark on the ball and capturing the pitch with an iPhone's slow-motion feature. Specifically, they employed YOLOv3 for ball detection, while the spin was estimated by extracting the blue mark via HSV color conversion and tracking its centroid across frames. However, this method reported relatively large measurement errors, ranging from 7.9% to 20.3% compared to manual verification. These inaccuracies can be attributed to the difficulty of precisely extracting the blue mark under varying lighting conditions, which resulted in errors in the centroid calculation. Additionally, the requirement to physically mark the ball imposes a preparatory burden on the user.

In our previous work [6], we proposed a markerless method using only a standard smartphone camera and a CNN-based discrete angle classification to overcome the limitations of physical markers. In that preliminary experiment using two test pitches, the Angular Difference method achieved high accuracy with measurement errors of 0.1% to 2.4% relative to a sensor-embedded ball. However, that study also reported that a simple Linear Regression approach resulted in significant errors (e.g., +46.5%) because it did not correctly handle the cyclic nature of the angles (i.e., it lacked phase unwrapping). Furthermore, the evaluation was limited to a very small dataset, leaving the robustness on a larger variety of pitches unverified.

In contrast to these existing methods, this paper significantly extends that preliminary work. We introduce an outlier correction algorithm and a Cumulative Linear Regression method that incorporates phase unwrapping. This improved pipeline is evaluated on a larger dataset of 15 videos to demonstrate its effectiveness and robustness compared to the simple Angular Difference method.

3 Proposed Method

Our proposed method measures the spin rate of a baseball from video recorded by a standard smartphone. Specifically, video data were captured from behind the catcher's position using the slow-motion feature of an iPhone XR (240 fps, 1080p HD). To create a focused view for the subsequent analysis, these captured videos were manually cropped to extract the region containing the ball. Figure 1 shows an example of this preliminary cropping.

Figure 1: An example frame from the captured video, with a rectangle indicating the manually cropped region.

Following this preprocessing, the overall workflow consists of three main stages: (1) Ball Detection and Angle Classification, (2) Outlier Detection and Correction, and (3) Spin Rate Calculation. The details of each stage are described below.

3.1 Ball Detection and Angle Classification

For the detection stage, YOLOv8 [7], a deep learning-based object detection model, is employed. Specifically, the pre-trained YOLOv8x model is used to detect and track the baseball in each pre-processed video frame. Figure 2 illustrates an example in which YOLO identifies the ball with a bounding box. In our actual system, however, the detection coordinates are directly used to crop the ball image, rather than displaying the bounding box.

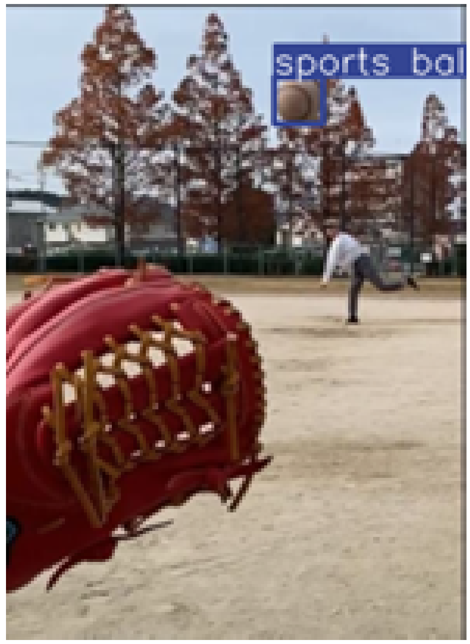


Figure 2: An example of ball detection using YOLOv8.

A collection of the resulting cropped ball images, which are used for the subsequent angle estimation, is shown in Figure 3. All subsequent procedures are conducted on this sequence of still images, rather than on the video data.

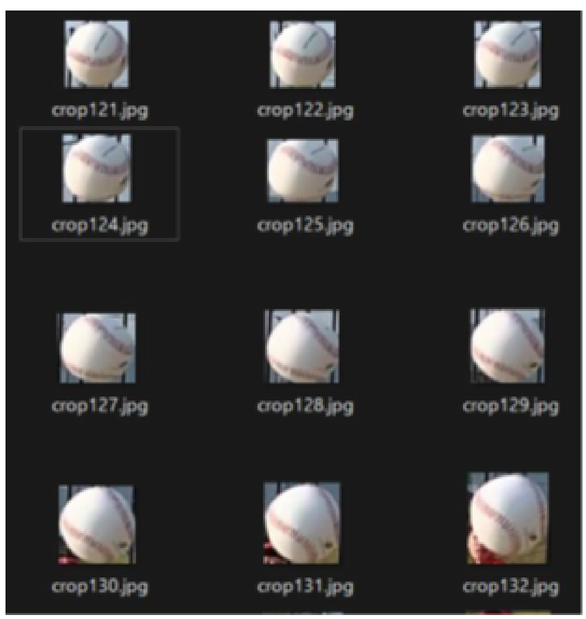


Figure 3: A collection of cropped ball images for subsequent analysis.

To estimate the ball's rotational angle, a convolutional neural network (CNN) was developed. Four discrete angle classes (0°, 45°, 90°, and 135°) are defined for the rotation of a standard fastball with a horizontal spin axis, as viewed from the catcher's perspective. See Figure 4. This classification is based on the 180° rotational symmetry of the baseball's seam pattern, since a rotation of 180° is visually indistinguishable from 0°.

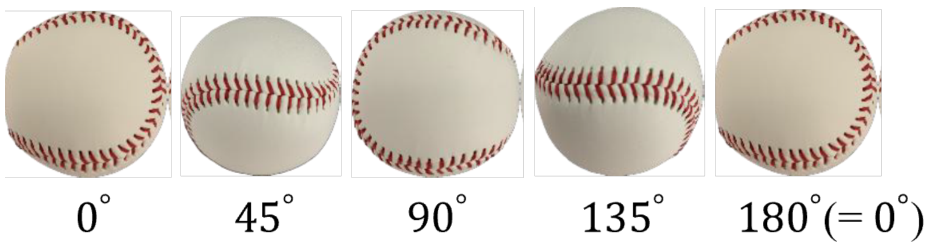


Figure 4: Ball orientations at discrete angles (0°, 45°, 90°, 135°, and 180°, with 180° regarded as identical to 0°).

The CNN, implemented with the Keras [8] library, takes a cropped ball image as input and outputs a prediction for one of these four angles. To ensure robustness against natural variations in pitch trajectory, the training dataset was augmented with ball images captured at a horizontal tilt (see Figure 5). Vertical tilt is managed by adjusting the camera position during the initial recording. The performance of this model was then evaluated in our experiments.

Figure 5: Examples of balls with tilted spin axes.

3.2 Outlier Detection and Correction

As identified in our previous work [6], the CNN classifies each frame independently without considering temporal context. Consequently, sporadic estimation errors (outliers) occurred, which were a primary source of instability in the preliminary study. To overcome this limitation and ensure robust estimation, a post-processing algorithm based on the physical continuity of the ball's rotation was implemented.

First, the rotational periodicity is defined. Since the baseball's seam pattern has 180° rotational symmetry, the angle θ is defined in the range $[0°, 180°]$, where 180° is equivalent to 0°. Our system estimates the angle sequence in reverse chronological order (from the last detected frame to the first). This strategy is adopted because the ball approaches the camera in the later frames, resulting in a larger apparent size and clearer seam visibility, which ensures more stable classification. In this reverse sequence, the expected rotation cycle is $135° \to 90° \to 45° \to 0° \to 135°$, and so on.

To evaluate the continuity between frames, the directed angular distance $d(x, y)$ between two angles x and y considering the cyclic boundary is defined as:

$$d(x, y) = (x - y) \pmod{180} \tag{1}$$

where the modulo operator handles the wraparound at 0°/180°. For example, the distance from 0° to 135° is calculated as $(0 - 135) \pmod{180} = 45°$, correctly reflecting the shortest rotational path in the reverse sequence.

Using this distance metric, outliers are detected by examining three consecutive frames: the previous frame P, the current frame C, and the next frame N. If the estimation is correct, the current angle C must lie on the directed rotation path from P to N. This condition satisfies the triangular equality:

$$d(P, C) + d(C, N) = d(P, N) \tag{2}$$

If this equation does not hold (i.e., the left-hand side is greater than the right-hand side), it implies that C deviates from the expected trajectory and is identified as an outlier.

When an outlier is detected, C is corrected by replacing it with the value of the previous frame P.

$$C_{corrected} \leftarrow P \tag{3}$$

Figure 6 schematically illustrates this process. The blue node represents an outlier detected by the algorithm, and the red node shows the corrected state. For example, if a sequence is predicted as $135° \to 45° \to 90°$, the algorithm calculates $d(135, 45) + d(45, 90) = 225$, while $d(135, 90) = 45$. Since Eq. (2) does not hold ($225 \neq 45$), the middle frame (45°, corresponding to the blue node) is corrected to the previous value (135°), resulting in a smoother sequence.

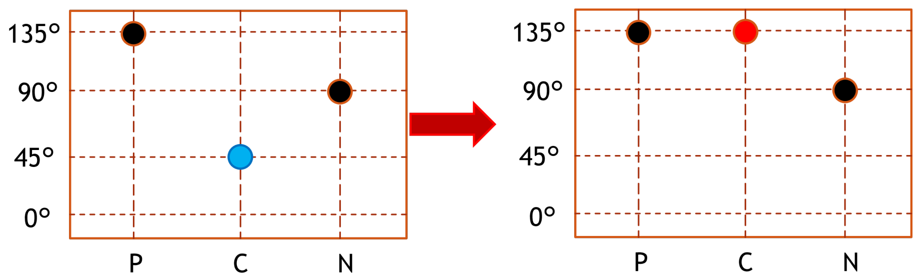


Figure 6: Schematic illustration of the outlier detection and correction logic.

3.3 Spin Rate Calculation

After classifying a sequence of frames into angles, the spin rate (in rpm) is calculated. In this study, two different calculation methods are applied over a fixed analysis window of 12 frames. This window size was selected to ensure a sufficient number of samples for stable estimation.

[Approach 1 (Angular Difference Calculation)]

This method calculates the spin rate by accumulating the angular changes between consecutive frames.

1. Calculate the angular difference $d(\theta_t, \theta_{t+1})$ between each consecutive pair of frames using Eq. (1).
2. Sum these differences to obtain the total rotation Θ_{total} over the 12 frames.
3. Compute the spin rate:

$$\text{(spin rate)} = \frac{\Theta_{total}/360}{\text{(Frame Count)} - 1} \times 240\,\text{fps} \times 60\,\text{s/min}. \tag{4}$$

Figure 7 shows an example of the angle differences. The graph shows a consistent difference of 45°, verifying that the CNN functions as expected. However, this method is sensitive to quantization error. Since the system only detects 45° steps, the calculated speed depends heavily on the timing of the first and last frames, leading to fluctuations ("staircase effect") even within a 12-frame window.

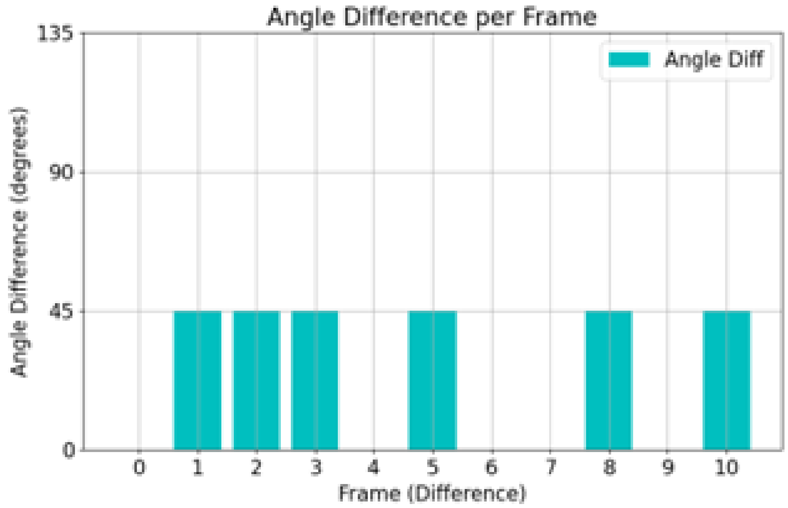


Figure 7: Example of calculated angle differences. The discrete 45° steps are verified.

[**Approach 2 (Cumulative Linear Regression)**]

In our previous work [6], linear regression was directly applied to the angle sequences. However, that approach failed to handle the cyclic nature of the data (phase wrapping) across multiple rotations, resulting in significant estimation errors (as high as +46.5%). To overcome this limitation and mitigate quantization error, we propose a Cumulative Linear Regression method. Unlike the previous attempt, this method explicitly performs phase unwrapping to convert the cyclic angles into a continuous cumulative rotation value.

The challenge is the cyclic nature of the data. Figure 8 illustrates the basic concept of our regression approach. Figure 8(a) shows the raw predictions, which follow a downward trend but wrap around at the cycle boundary ($0° \to 135°$). Applying linear regression directly to this cyclic data fails, as shown in Figure 8(b), because the wrap-around disrupts the linearity.

In our preliminary approach, a local offset correction was applied as shown in Figure 8(c). Specifically, the discontinuous data points a, b, c, d, e in Figure 8(b) are remapped to a', b', c', d', e' to create a locally linear sequence. However, this method is limited to short continuous segments (e.g., 4–5 frames) and cannot handle multiple rotations.

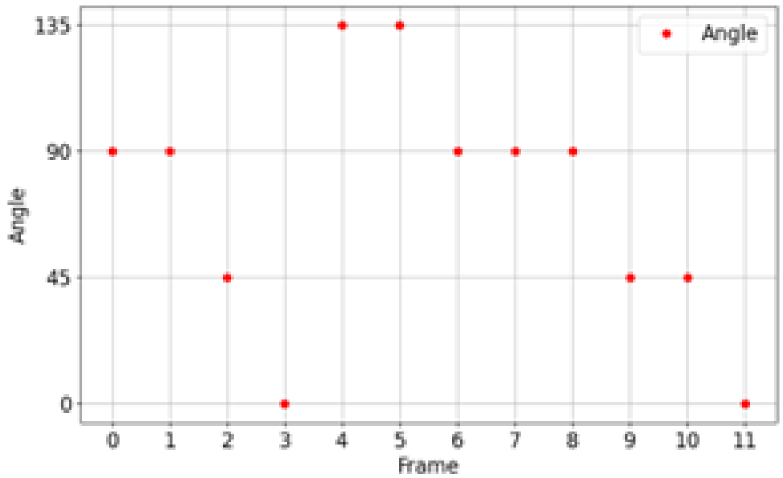
To solve this and utilize the full 12-frame window, this concept is advanced to Cumulative Rotation (Phase Unwrapping). C_0 is initialized to 0, and the angular distance is iteratively added:

$$C_t = C_{t-1} + d(\theta_{t-1}, \theta_t) \tag{5}$$

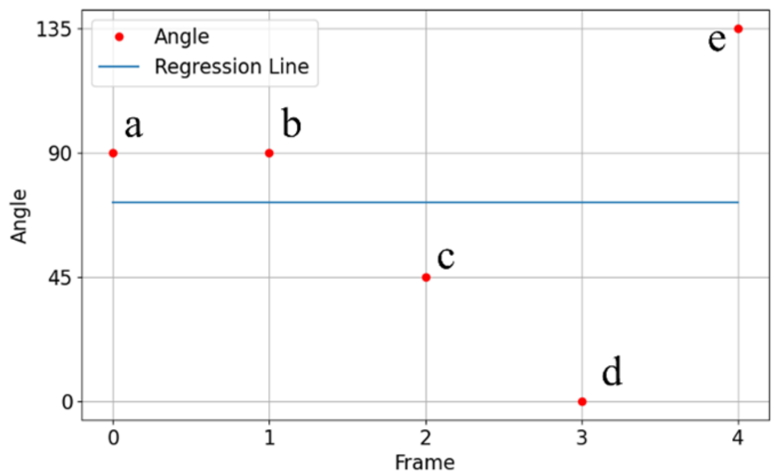
As shown in Figure 9, this transformation creates a continuous monotonic line that extends seamlessly across cycle boundaries. This allows linear regression to be applied to all 12 data points. The spin rate is calculated from the slope α of the regression line ($C_t \approx \alpha t + \beta$):

$$(\text{spin rate}) = \frac{\alpha}{360} \times 240\,\text{fps} \times 60\,\text{s/min}. \tag{6}$$

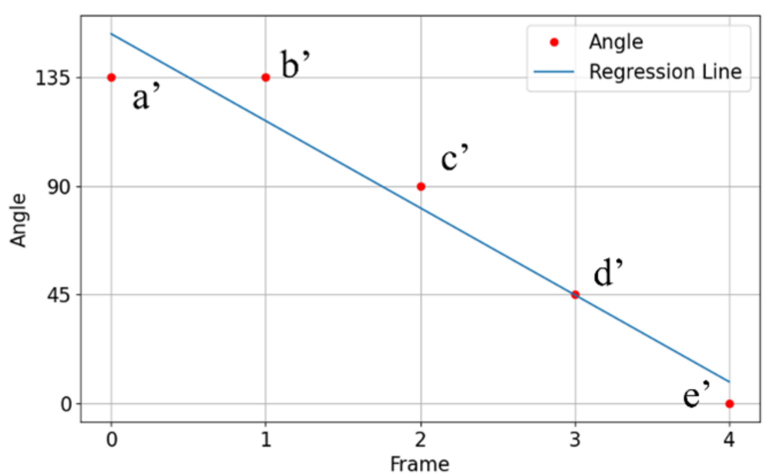
This least-squares fitting over a longer window effectively smooths out the local 45° quantization steps, providing a more robust estimation than Approach 1.



(a) Raw angle data (cyclic).



(b) Incorrect regression on raw data.



(c) Local offset correction (Limited to short segments).

Figure 8: Visualization of the challenges in linear regression on cyclic data.

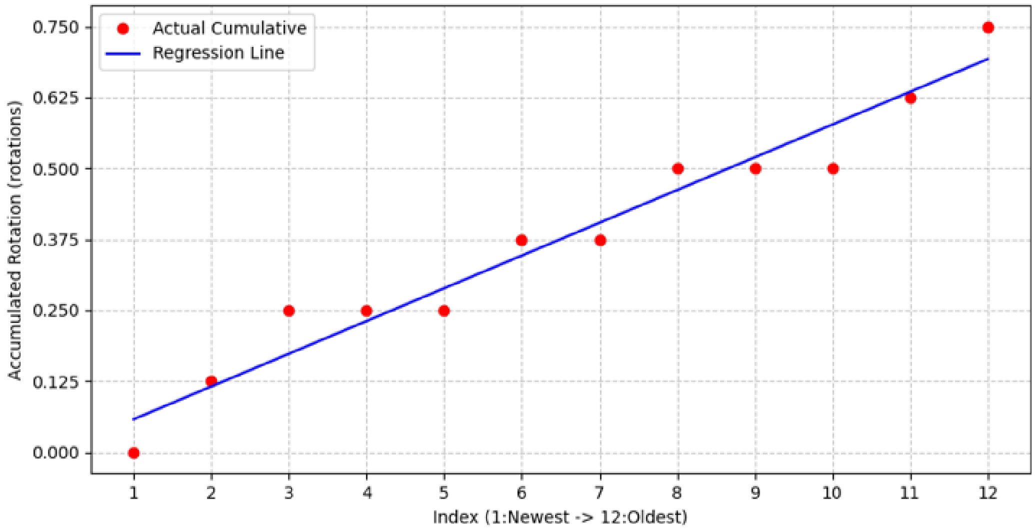


Figure 9: Proposed Cumulative Linear Regression. By unwrapping the phase, a regression line can be fitted over a fixed 12-frame window, enabling robust estimation.

4 Experimental Results

4.1 Experimental Setup

To validate the effectiveness of our proposed system, experiments were conducted using a Technical Pitch ball [3] as a baseline. Although this ball contains embedded sensors to measure spin data, it is manufactured in accordance with the size and weight standards of the Official Baseball Rules. This setup allowed the flight to be simultaneously recorded with a smartphone (iPhone XR, 240 fps, 1080p) and the ground truth spin rate to be acquired from the sensor. The pitches were thrown at speeds of approximately 60–80 km/h. To facilitate the initial ball detection and seam extraction, the recordings were conducted under moderate lighting conditions, avoiding direct sunlight to minimize glare and harsh shadows.

The analysis window was set to 12 frames for both Approach 1 and Approach 2. It should be noted that the system's performance was evaluated in stages: first, the success rates of ball detection and angle classification were assessed, and then the spin rate calculation accuracy was evaluated using only the successfully processed data.

4.2 Evaluation Metrics

The percentage error between the estimated spin rate and the reference value from Technical Pitch is used as the primary evaluation metric:

$$\text{Error}(\%) = \frac{\hat{\omega} - \omega_{ref}}{\omega_{ref}} \times 100, \tag{7}$$

where $\hat{\omega}$ is the estimated spin rate (rpm) and ω_{ref} is the reference (rpm). In addition, because the proposed pipeline is a multi-stage process, success rates at each stage (ball detection and angle estimation) are reported to clarify how many videos are available for the final spin-rate evaluation.

4.3 Evaluation of Outlier Correction

4.3.1 Rationale for Using a Fixed 12-frame Window

In preliminary trials, it was observed that using too few frames can lead to unstable spin-rate estimates because a small number of frames amplifies discretization noise and makes the estimate sensitive to the exact start/end timing. In particular, when the effective window becomes as short as 5–7 frames, the resulting estimate can deviate significantly even if only one frame is misclassified. Therefore, to enable a fair comparison between approaches and to stabilize the estimation, a fixed analysis window of 12 frames is used for both Approach 1 and Approach 2.

4.3.2 Error Sources and Limitations in Discrete Angle Classification

Our angle estimation uses only four classes (0°, 45°, 90°, and 135°), which introduces a quantization error. This discretization may lead to overestimation or underestimation depending on the timing of the sampled frames. For example, if the ball rotates from a state close to 45° to a state close to 0°, the system registers a full 45° rotation, even if the actual rotation is considerably smaller.

This quantization error can accumulate in Approach 1 (Angular Difference), which directly sums discrete steps. Approach 2 (Cumulative Linear Regression) mitigates this effect by least-squares fitting over a fixed window, but large consecutive misclassifications can still break the monotonicity of the cumulative rotation and cannot be fully corrected by the current outlier correction.

4.3.3 Performance of Image Processing steps

Before calculating the spin rate, the performance of the preparatory steps (ball detection and angle classification) was evaluated. Table 1 summarizes the processing results for all 15 videos.

Table 1: Summary of Processing Results for All 15 Videos

Pitch ID	Ball Detection	# Outliers in 12 frames	Usable for Spin Rate
1	Success	Many	No
2	Fail	–	No
3	Success	Many	No
4	Success	0	Yes
5	Success	0	Yes
6	Success	0	Yes
7	Success	2	Yes (after correction)
8	Success	0	Yes
9	Success	Many	No
10	Success	Many	No
11	Success	1	Yes (after correction)
12	Success	0	Yes
13	Success	1	Yes (after correction)
14	Success	0	Yes
15	Success	0	Yes

First, the ball detection based on YOLOv8 achieved a success rate of 93% (14 out of 15 videos). In the single failed case (Pitch 2), the detection was missed in only one frame. Although the recording conditions for this video were consistent with the other successful videos, this momentary drop prevented the extraction of the complete image sequence required for the subsequent analysis.

Next, the angle classification accuracy was evaluated on the 14 successfully detected videos. Initially, without the outlier correction described in Section 3.2, only 7 videos (50%) were correctly classified with a consistent rotation sequence.

In this paper, a "failure" in angle estimation is denoted as a misclassification at a single frame within the fixed 12-frame analysis window (i.e., an outlier in the 12 predicted labels). For example, "1 failure" means that exactly 1 out of the 12 frames was misclassified, and "2 failures" means that 2 frames were misclassified.

In our dataset, three videos contained only a small number of such sporadic outliers: Pitch 11 and Pitch 13 each had 1 failure, and Pitch 7 had 2 failures. By applying our proposed Outlier Correction algorithm, these 3 videos were successfully rescued by correcting the sporadic errors based on temporal continuity. Consequently, the number of usable videos for spin rate calculation increased to 10 (71%).

The remaining 4 videos (Pitch 1, Pitch 3, Pitch 9, and Pitch 10) contained too many errors or inconsistent sequences to be corrected and were excluded from the subsequent spin rate analysis.

4.4 Comparison of Spin Rate Calculation Methods

Finally, the spin rate measurement accuracy on the usable videos was evaluated. The baseline approach (Approach 1: Angular Difference) and the proposed approach (Approach 2: Cumulative Linear Regression) were compared.

To demonstrate the necessity of our proposed improvements, the measurement results obtained with the earlier version of our pipeline are first presented. This baseline version did not use outlier correction and employed a simple angle-based linear regression without phase unwrapping. Table 2 summarizes these results. At this stage, due to the lack of correction for sporadic errors, only 7 videos (Pitch ID 4, 5, 6, 8, 12, 14, and 15) out of 15 were usable for spin-rate computation. In Table 2, the linear regression results are reported together with the number of frames actually used for the regression (shown next to the error). As shown, the simple regression was often limited to short segments instead of the full 12-frame sequence, leading to unstable estimates (e.g., +116.2% error for Pitch 15).

Table 2: Measurement Results Before Improvements (without Outlier Correction and Phase Unwrapping)

Pitch ID	**Reference**	**Angular Diff.**	**Linear Reg. (Angle-based)**
4	983 rpm	982 rpm (-0.1%)	1440 rpm (+46.5%, 5 frames)
5	1105 rpm	1145 rpm (+3.6%)	1260 rpm (+14.0%, 5 frames)
6	1210 rpm	1145 rpm (-5.4%)	977 rpm (-19.3%, 6 frames)
8	799 rpm	818 rpm (+2.4%)	750 rpm (-6.1%, 8 frames)
12	1064 rpm	1309 rpm (+23.0%)	1260 rpm (+18.4%, 5 frames)
14	1238 rpm	1309 rpm (+5.7%)	1620 rpm (+30.9%, 4 frames)
15	1249 rpm	1473 rpm (+17.9%)	2700 rpm (+116.2%, 3 frames)
Avg. Error	-	**8.3%**	**35.9%**
Success Rate	-	**71% (5/7)**	**14% (1/7)**

Next, the performance of the proposed pipeline incorporating the outlier correction and Cumulative Linear Regression (Approach 2) is evaluated. As described in Section 4.3.3, the outlier correction increased the number of usable videos to 10 by rescuing sequences with sporadic errors.

A quantitative criterion for a "successful measurement" was established based on the prior vision-based smartphone study [5], which reported a best-case error of 7.9%. In this study, success was defined as achieving an error rate of 8.9% or less. This threshold includes results strictly better than the prior study ($\leq 7.9\%$) and those comparable to it, allowing for a margin of up to 1.0%. For instance, the result of Pitch 15 using Approach 2 (8.8% error) falls within this comparable range and is considered a successful measurement.

Table 3 details the measurement results for the 10 usable videos (Pitch ID 4, 5, 6, 7, 8, 11, 12, 13, 14, and 15). Using Approach 1 (Angular Difference), only 6 out of 10 pitches (60%) satisfied the success criteria ($\leq 8.9\%$). As discussed, this approach is sensitive to quantization noise, leading to larger errors in some cases.

Table 3: Measurement Results After Improvements (10 Usable Videos)

Pitch ID	**Reference** (Technical Pitch)	**Approach 1** (Angular Diff.)	**Approach 2** (Cumulative Reg.)
4	983 rpm	982 rpm (-0.1%)	963 rpm (-2.0%)
5	1105 rpm	1145 rpm (+3.6%)	1132 rpm (+2.4%)
6	1210 rpm	1145 rpm (-5.4%)	1127 rpm (-6.9%)
7	857 rpm	982 rpm (+14.6%)	830 rpm (-3.2%)
8	799 rpm	818 rpm (+2.4%)	787 rpm (-1.5%)
11	1025 rpm	982 rpm (-4.2%)	963 rpm (-6.0%)
12	1064 rpm	1309 rpm (+23.0%)	1208 rpm (+13.5%)
13	1172 rpm	1309 rpm (+11.7%)	1259 rpm (+7.4%)
14	1238 rpm	1309 rpm (+5.7%)	1297 rpm (+4.8%)
15	1249 rpm	1473 rpm (+17.9%)	1359 rpm (+8.8%)
Avg. Error	-	**8.9%**	**5.7%**
Success Rate	-	**60% (6/10)**	**90% (9/10)**

In contrast, Approach 2 (Cumulative Linear Regression) achieved a significantly higher success rate. By converting the angle sequence to cumulative rotation and applying linear regression over the full 12-frame window, 9 out of 10 pitches (90%) satisfied the criteria. This result demonstrates that our proposed regression approach effectively smooths out discretization artifacts and provides robust estimation even with low-resolution angle classes.

5 Discussion

5.1 Comparison with Conventional Methods

To clarify the superiority of the proposed pipeline, a comparison with existing vision-based approaches is presented. The most relevant conventional method is the smartphone-based approach proposed by Nishida et al. [5]. While their method achieved measurement errors ranging from 7.9% to 20.3%, it required physical modification of the baseball (drawing a blue mark) to extract color features. This requirement imposes a preparatory burden on users and alters the standard conditions of the equipment. Furthermore, tracking a specific color mark is highly susceptible to lighting variations and motion blur.

In contrast, the proposed pipeline is entirely markerless. By leveraging YOLOv8 for detection and CNN-based angle classification on the natural seam patterns, the need for physical markers is eliminated.

Table 4: Quantitative comparison between the conventional method and the proposed approach.

Method	Marking	Sample Size (N)	Best Error	Mean Error
Nishida et al. [5]	Required	2	7.9%	14.1%
Proposed Approach 2	**Not required**	**10**	**1.5%**	**5.7%**

Table 4 summarizes the quantitative comparison between the conventional marker-based method and the proposed markerless approach. While the conventional method [5] was evaluated on only two samples ($N = 2$) with a mean error of 14.1%, the proposed approach was validated on a larger dataset of 10 usable videos. Through the introduction of the cumulative linear regression approach, the proposed method achieved a significantly lower mean error of 5.7% and a success rate of 90% under the $\leq 8.9\%$ error criterion. Furthermore, the best-case error of the proposed approach (1.5% for Pitch 8) represents a substantial improvement over the best-case error of 7.9% reported in the conventional study. These results clearly demonstrate that the proposed pipeline not only improves user accessibility by being markerless but also provides superior measurement accuracy and reliability compared to the conventional smartphone-based method.

5.2 Failure Mode Analysis

Although the proposed pipeline improved the robustness of spin-rate estimation, some videos remained unusable. From Table 1, the primary failure modes can be summarized as follows.

Detection Failure (Pitch 2)

Pitch 2 failed at the ball detection stage. As noted in Section 4.3.3, this failure was caused by a missing detection in just a single frame. Because the current pipeline strictly relies on a continuous sequence of cropped images for the fixed 12-frame analysis window, even a momentary detection dropout results in a complete pipeline failure. While the specific trigger for this single-frame miss remains unisolated, this highlights the necessity for a tracking mechanism capable of interpolating missing frames, which will be addressed in future work.

Angle Estimation Failure with Many Outliers (Pitch 1, 3, 9, and 10)

Pitch 1, 3, 9, and 10 were excluded due to frequent misclassifications or inconsistent rotation sequences. This indicates that the current CNN-based discrete angle classification can still be unstable for some videos. While several factors (e.g., heavy motion blur, low seam visibility, or strong highlights) may contribute to such instability, controlled experiments were not conducted to isolate the primary cause for each specific failure.

Limitations of the Current Outlier Correction

The proposed outlier correction is designed to handle sporadic errors (e.g., 1–2 outliers within the 12-frame window) by enforcing temporal continuity. Importantly, by design, it cannot reliably recover sequences containing consecutive misclassifications, because the correction rule replaces the current frame with the previous frame based on the assumption of locally consistent transitions. Therefore, when multiple consecutive frames are misclassified or when the predicted sequence deviates from the expected rotation cycle for a prolonged span, the correction fails to recover a valid sequence. These observations motivate future improvements such as confidence-based filtering, re-estimation using multiple hypotheses, or switching from discrete classification to continuous angle regression.

5.3 Practical Considerations for Deployment

A key advantage of our approach is that it only requires a standard smartphone camera. Nevertheless, practical use requires appropriate recording conditions. A complete characterization of the optimal recording setup is outside the scope of this paper; rather, it is noted that the visibility of seam patterns is likely a critical factor for stable estimation.

In addition, the current workflow includes manual cropping of the region of interest. For real-world deployment, automating this step (e.g., by detecting the pitching lane or by tracking the ball across the full frame) would significantly reduce user burden and improve reproducibility.

5.4 Threats to Validity and Scope

This study has several limitations. First, the evaluation was performed using a limited speed range (approximately 60–80 km/h) and a single recording device configuration. Second, the success criterion is defined relative to a prior smartphone study and is intended as a practical benchmark; other applications may require stricter accuracy. Third, the discrete 45° angle representation simplifies learning and inference but inherently introduces quantization noise. Future studies should evaluate broader pitch speeds, multiple recording devices, and alternative learning formulations (e.g., continuous regression or self-supervised rotation tracking).

5.5 Future Work

To further improve the accuracy and usability of the proposed system, technical enhancements in both image preprocessing and automated tracking are being considered.

First, regarding accuracy, the current CNN inputs are raw cropped images that often contain irrelevant background noise and shadows, which can interfere with feature extraction. To address this, the implementation of a preprocessing pipeline that isolates the seam patterns before classification is planned. Specifically, background removal using a circular mask will be applied to eliminate artifacts, followed by a Laplacian Filter [9] to explicitly highlight the seam structure. By training the CNN on these edge-enhanced images, the model is expected to focus more effectively on the geometric features of the seams.

Second, regarding usability, the current workflow requires manual cropping of the Region of Interest (ROI), which is labor-intensive. To automate this process, a dynamic tracking system is currently under development, which integrates three technologies: Pose Estimation (MediaPipe [10]) to identify the pitcher's release point, the latest Object Detection model (YOLO11 [11]) for improved accuracy, and a Kalman Filter [12] for trajectory prediction. By estimating the ball's position and velocity, the system will be able to move the ROI frame automatically and maintain tracking even when visual detection is intermittent. In future studies, this integrated system will be implemented and its effectiveness in reducing user workload and improving robustness against environmental changes will be verified.

6 Conclusion

This paper presented an accessible method for measuring the spin rate of a baseball using only a standard smartphone camera. To improve robustness under practical recording conditions, this study introduced (i) an outlier detection and correction algorithm for sporadic CNN angle misclassifications and (ii) a cumulative linear regression method that estimates rotation speed from phase-unwrapped cumulative rotation rather than raw cyclic angles.

In experiments conducted on 15 smartphone videos, YOLOv8-based ball detection succeeded in 14 cases (93%). Among these, the outlier correction increased the number of usable videos for spin-rate estimation from 7 to 10 (50% to 71%). On these 10 videos, using an error threshold of $\leq 8.9\%$ as a success criterion, the baseline angular-difference method achieved a success rate of 60% (6/10), whereas the proposed cumulative regression achieved 90% (9/10) with a lower average error (5.7% vs. 8.9%). These results indicate that cumulative regression can effectively mitigate discretization noise caused by the 45° angle classes.

This study has several limitations. First, the object detection performance is sensitive to environmental factors such as lighting, shadows, and complex backgrounds. Second, angle classification can degrade due to motion blur and low image quality, and the current outlier correction can only handle sporadic errors.

Future work includes improving the robustness of ball detection, extending outlier handling to cope with consecutive errors, evaluating wider pitch speed ranges, and migrating the entire pipeline onto the smartphone for a fully self-contained real-time measurement application.

References

[1] TrackMan, "Trackman Baseball - Data and Actionable Insights", Trackman, 2026. [Online]. Available: `https://www.trackman.com/ja/baseball`. [Accessed: Feb. 1, 2026].

[2] Rapsodo LLC, "Baseball - Rapsodo Japan", Rapsodo Japan, 2024. [Online]. Available: `https://rapsodo.co.jp/pages/baseball`. [Accessed: Feb. 1, 2026].

[3] Acrodea, Inc., "Technical Pitch", Technical Pitch, n.d. [Online]. Available: `https://technicalpitch.net/`. [Accessed: Feb. 1, 2026].

[4] T. Ijiri, A. Nakamura, A. Hirabayashi, W. Sakai, T. Miyazaki, and R. Himeno, "Automatic spin measurements for pitched Baseballs via consumer-grade high-speed cameras," Signal, Image and Video Processing, vol. 11, no. 7, pp. 1197-1204, 2017.

[5] S. Nishida and H. Kuroda, "Measurement of Baseball Spin Rate and Estimation of Pitch Type Using a Smartphone", in Proc. 83rd National Convention of IPSJ, 2021, pp. 2-143-2-144. (in Japanese)

[6] R. Yamamoto and S. Matsumae, "Measurement of Baseball Spin Rate Using a Smartphone Camera," 2025 Thirteenth International Symposium on Computing and Networking Workshops (CANDARW), Yamagata, Japan, 2025, pp. 356-359,

doi: 10.1109/CANDARW68385.2025.00068.

[7] Ultralytics, "Explore Ultralytics YOLOv8", Ultralytics YOLO Docs, 2026. [Online]. Available: `https://docs.ultralytics.com/models/yolov8/`. [Accessed: Feb. 1, 2026].

[8] Keras, "Keras: Deep Learning for humans", Keras.io, n.d. [Online]. Available: `https://keras.io/`. [Accessed: Feb. 1, 2026].

[9] OpenCV, "OpenCV: Laplace Operator", OpenCV, 2026. [Online]. Available: `https://docs.opencv.org/4.x/d5/db5/tutorial_laplace_operator.html`. [Accessed: Feb. 1, 2026].

[10] Google, "MediaPipe Solutions Guide", Google AI for Edge, 2025. [Online]. Available: https://ai.google.dev/edge/mediapipe/solutions/guide. [Accessed: Feb. 1, 2026].

[11] Ultralytics, "Ultralytics YOLO11", Ultralytics YOLO Docs, 2026. [Online]. Available: https://docs.ultralytics.com/models/yolo11/. [Accessed: Feb. 1, 2026].

[12] MathWorks, "Kalman Filter", MathWorks, 2026 [Online]. Available: https://jp.mathworks.com/discovery/kalman-filter.html. [Accessed: Feb. 1, 2026].