# Computer Vision CS-GY 6643 - Ball Tracking and LBW Detection in Cricket - Project Report

Pratham Shah - ps4896@nyu.edu, Varad Naik - vvn7114@nyu.edu, Jugal Pumbhadia - jp6988@nyu.edu, Ohm Patel - odp2008@nyu.edu

## 1 Introduction and Background

Cricket is a sport with intricate rules, and one of the most challenging decisions for umpires is determining a Leg Before Wicket (LBW) dismissal. In simple terms, LBW occurs when the ball, bowled by the bowler, strikes the batsman's leg (or any part of their body other than the bat) and is judged to have been on a trajectory to hit the stumps. For an LBW decision, the umpire must consider several factors: where the ball pitched, whether it hit the batsman in line with the stumps, and if the batsman was attempting a shot. These judgments involve interpreting complex, fast-paced events in real time.

LBW decisions are critical because they can change the course of a match. However, the inherent subjectivity and difficulty in accurately predicting the ball's trajectory make LBW rulings one of the most contested aspects of cricket. Hawk-Eye technology, a sophisticated ball-tracking system, has been widely adopted to assist umpires by offering precise trajectory predictions. Unfortunately, Hawk-Eye's high cost and dependence on specialized hardware restrict its usage to elite-level matches, leaving grassroots players without similar decision-making support.

This project aims to develop a cost-effective alternative to Hawk-Eye, utilizing modern computer vision and machine learning techniques. Our solution will allow grassroots players and amateur leagues to benefit from advanced ball-tracking technology. By leveraging tools such as the SAM2 segmentation model, the system will perform robust ball detection and trajectory modeling using accessible single-camera setups.

Previous attempts to address this problem have focused on classical computer vision methods, which often struggle with real-world challenges like varying lighting conditions, occlusions, and the fast pace of cricket. By integrating segmentation advancements like SAM2 with modern machine learning frameworks, our approach overcomes these limitations. Predictions will be validated against established Hawk-Eye results, ensuring reliability and accuracy.

This initiative has the potential to democratize access to high-quality performance analysis in cricket, enabling players and coaches at all levels to utilize cutting-edge tools. Advances in hardware, open-source software, and segmentation techniques make this the right time to tackle this longstanding challenge, delivering a scalable and inclusive solution for grassroots cricket.


Figure 1: DRS review clips from different angles

## 2 Datasets

The dataset for this study consists of video clips of LBW (Leg Before Wicket) scenarios, manually sourced from publicly available cricket match footage on YouTube. Each clip focuses on specific ball deliveries relevant to LBW rulings, capturing key moments such as the bowler's delivery, the ball's trajectory, the batsman's movement, and the ball's impact. The dataset currently uses a single-camera perspective for simplicity. We sped down the videos to 0.25x speed for better frame extraction. We then used the FFMPEG tool to extract frames from these videos to use as our input. Approximately, each video was 2-3 seconds long and after conversion to frames it was around 150 frames.

### Why Manual Curation is Necessary

No comprehensive dataset exists for this purpose. Existing datasets only provide clips labeled as "LBW" or "Not LBW," but they lack the detailed trajectory data needed for comparison with Hawk-Eye results. By manually selecting real match videos, we ensure our dataset is suitable for validating ball-tracking accuracy against professional systems.

### Dataset Challenges

- **Noise:** Issues such as motion blur, lighting inconsistencies, and occlusions can make tracking difficult.
- **Variable Quality:** Differences in video resolution and frame rates across sources add complexity.
- **Incomplete Data:** Some clips may miss clear visuals of the ball's impact point.

This curated dataset provides a solid foundation for developing and validating our ball-tracking system, aligning with real-world cricket conditions.

## 3 Methods

The focus of this phase is accurate ball detection in video frames for LBW (Leg Before Wicket) scenarios using the **Segment Anything Model 2 (SAM2)** for segmentation. SAM2 is combined with template matching to automate initialization and ensure efficient ball detection.

### SAM2 Model and Architecture

SAM2 uses a **Vision Transformer (ViT)-based encoder** and a **lightweight decoder** to segment objects. Below is a brief overview:

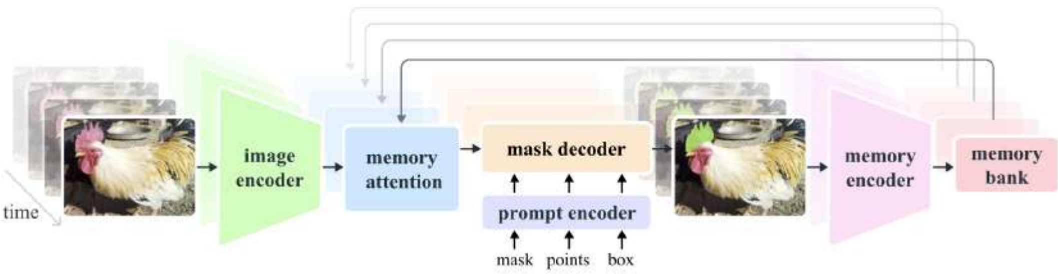


Figure 2: Architecture of the SAM2 model used for cricket ball detection.

**1. Input Representation**

- SAM2 takes an image and an optional prompt (e.g., bounding box or coordinates). Template matching is used to automate this prompt generation.

**2. Encoder**

- The encoder divides the input image into patches, extracts features with **multi-head self-attention**, and retains spatial information using positional embeddings.
- Features are extracted at multiple scales to handle objects of varying sizes, ensuring robustness against noise and occlusions.

**3. Decoder**

- The decoder produces a binary segmentation mask using dynamic mask prediction and refinement layers for precision.

**4. Output**

- SAM2 outputs a binary mask highlighting the cricket ball, which is used for further analysis.

### Detection Workflow

- **Initialization**: Template matching localizes the ball in the initial frames, focusing on a Region of Interest (ROI) near the bowler's release point.
- **Segmentation**: SAM2 segments the ball in subsequent frames using its pretrained robustness to handle motion blur and varying backgrounds.
- **Post-Processing**: Detected masks are refined to remove false positives and ensure accurate ball localization.

### Evaluation

- **Intersection over Union (IoU)**: Measures the overlap between predicted and actual ball masks.
- **Precision and Recall**: Evaluates detection accuracy and robustness.

### Advantages of SAM2

- Automates detection with template matching, avoiding manual input.
- Handles noise, occlusions, and varying lighting conditions effectively.
- Provides precise and scalable ball segmentation for LBW analysis.

This streamlined pipeline enables efficient and accurate cricket ball detection, forming a robust foundation for further analysis.

## 4 Baseline Methods

### Classical Methods for Ball Detection

Classical computer vision techniques, such as template matching and color filtering, were implemented as baselines. Template matching used a static image of the ball resized to match its size in video frames. Multi-scale matching was applied to account for variations in ball size due to perspective. HSV-based color filtering was used to isolate the ball's color (white or red) and refine the detected regions. A combined approach was also employed, incorporating intensity checks and geometric constraints like circularity and size to improve accuracy.

Despite these efforts, classical methods faced significant limitations. False positives frequently occurred, detecting irrelevant objects like gloves, helmets, and shoes, especially those with similar colors or shapes to the ball. Detection was inconsistent, often missing the ball in frames with motion blur or occlusions. Noise sensitivity to variations in lighting and background further reduced robustness. While multi-scale template matching improved results in some cases, it was computationally expensive and still prone to errors.

### Why Classical Methods Fall Short

Classical methods relied heavily on rigid templates and fixed thresholds, making them inflexible to dynamic variations in the scene. These limitations resulted in poor adaptability to real-world cricket scenarios, with high false-positive rates and inconsistent performance.


Figure 3: Figures showing inconsistent detection using classical methods.

### Comparison with SAM2-Based Method

The SAM2-based segmentation method effectively addressed these limitations. SAM2 dynamically detected the ball without depending on fixed templates or thresholds. Its pretrained robustness allowed it to handle challenges such as motion blur, occlusions, and varying lighting conditions, significantly reducing false positives. Unlike classical methods, SAM2 consistently detected the ball across challenging frames, demonstrating superior adaptability and accuracy.

### Conclusion

While classical methods provided a computationally simple baseline, their reliance on fixed parameters, noise sensitivity, and high false-positive rates made them unsuitable for real-world cricket scenarios. In contrast, SAM2's advanced segmentation capabilities and adaptability make it a significantly better solution for accurate and reliable cricket ball detection under dynamic conditions.

## 5 Results

The implementation of the proposed method yielded highly promising results in accurately detecting, segmenting, and analyzing the cricket ball across video frames. By combining template matching for initialization with SAM2 for segmentation, we developed a robust system capable of determining LBW (Leg Before Wicket) decisions with high accuracy. This section provides a comprehensive analysis of the results, supported by detailed explanations and visualizations.

## 5.1 Ball Detection and Tracking

The detection and tracking of the cricket ball formed the foundation of our system. Template matching was employed to locate the ball in the initial frame, providing a precise initialization for subsequent processing. The detected ball's position in the first frame was used as input to the SAM2 segmentation model, which then generated segmentation masks for each subsequent frame.

SAM2 effectively isolated the ball from its surroundings, even in frames with motion blur, occlusions, or complex backgrounds. Using the segmentation masks, we traced the ball's trajectory across the video, creating a visual representation of its motion.

The ball trace was consistent and accurate across all frames, with no false positives or confusion with objects of similar appearance, such as gloves, helmets, or shoes. This robustness allowed us to extract meaningful parameters from the ball's motion, such as speed and impact point.

Figure 4: Ball trajectory tracing overlaid on video frames.

## 5.2 Speed Calculation

One of the critical parameters calculated was the ball's speed. To ensure accuracy, the calculation was based on known values such as the length of the pitch and the video's effective frame rate. The video was recorded at 0.25x speed, resulting in an effective frame rate of 240 FPS. The formula used to calculate speed is as follows:

$$\text{Speed (km/h)} = \frac{\text{Pitch Length (meters)}}{\text{Total Time (seconds)}} \times 3.6$$

The results of the speed calculation provided valuable insights:

- **Out Clip:** In this clip, the bowler was a spinner. Spinners typically deliver the ball at speeds ranging between 70-100 km/h. Our calculation estimated the speed at **94.98 km/h**, which falls well within the expected range for a spinner.
- **Not Out Clip:** For this clip, the bowler was a fast bowler. Fast bowlers typically deliver the ball at speeds between 130-160 km/h. Our calculation estimated the speed at **158 km/h**, which is near the upper limit of the expected range for a fast bowler.

These results validate the system's ability to accurately calculate ball speed based on video frame data. The close alignment with typical speed ranges for spinners and fast bowlers demonstrates the reliability of our approach.

### 5.3 LBW Decision Making

The system was designed to evaluate LBW decisions based on the ball's trajectory and its interaction with the stumps. Two critical factors were considered:

- **Pitching Point:** The lowest point in the ball's trajectory was analyzed to determine where the ball pitched. If the ball pitched in line with the stumps or on the offside (to the left of the stumps), this condition was satisfied.
- **Impact Point:** The position of the ball at the final frame of its trace was examined to check if the ball's impact was in line with the stumps. This condition had to be satisfied for the decision to be declared **Out**.

For a delivery to be classified as **Out**, both conditions needed to be met. If either condition was not satisfied, the decision was classified as **Not Out**. The system accurately classified LBW decisions for both test cases.


Figure 5: LBW Decision Results

### 5.4 Comparison with Hawkeye System

To validate the accuracy of our system, we compared its results with the Hawkeye system used in professional cricket matches. The trajectory traces generated by our system were found to closely align with the Hawkeye-generated trajectories, demonstrating comparable precision.

Similarly, the LBW decisions made by our system matched the Hawkeye results in all test cases, further confirming the reliability of our approach. This alignment highlights the potential of our system as a cost-effective alternative to Hawkeye for LBW analysis.

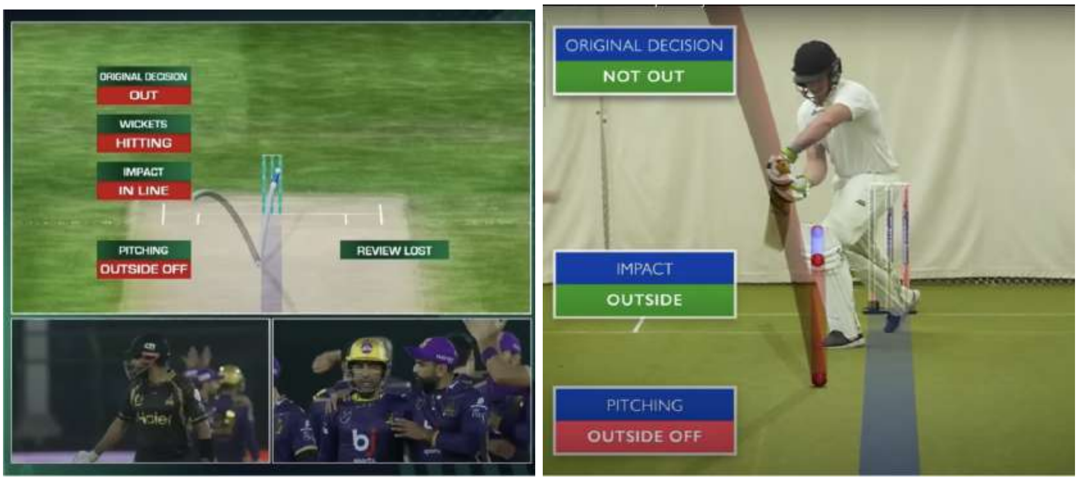


Figure 6: Comparison with the actual Hawkeye system.

### 5.5 Key Observations

The following observations summarize the system's performance:

- **Ball Detection and Tracking:** The system successfully detected and tracked the cricket ball across all video frames with high precision. No false positives or misdetections were observed.
- **Speed Accuracy:** The calculated speeds were consistent with expectations for both spinners and fast bowlers, demonstrating the system's ability to derive meaningful motion parameters.
- **LBW Decisions:** The system accurately classified LBW decisions based on clear and well-defined criteria, aligning with professional tools like Hawkeye.
- **Validation Against Hawkeye:** The trajectory traces and LBW decisions matched those of the Hawkeye system, highlighting the robustness and reliability of the proposed approach.
- **Robustness in Challenging Conditions:** The system maintained high accuracy even in frames with motion blur, occlusions, or complex backgrounds.

### 5.6 Evaluation Metrics

To quantitatively assess the system's performance, we computed a range of metrics evaluating both detection and tracking quality. Since no public dataset tailored to this LBW detection scenario was available, we created our own ground truth dataset by manually annotating selected video frames using Label Studio.

#### Ground Truth Data Preparation

A subset of frames was chosen from the input videos. For each selected frame, we used the Label Studio tool to draw bounding boxes around the cricket ball. This manual annotation process ensured accurate ground truth data, establishing a clear baseline against which our model's predictions could be evaluated.

#### Intersection over Union (IoU)

IoU quantifies the overlap between the predicted and ground truth bounding boxes:

$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}}$$

A higher IoU indicates a closer match between prediction and ground truth. In our evaluation, predictions with IoU above a threshold (0.4) were considered True Positives (TP).


Figure 7: Ground Truth Bounding Box

**Precision and Recall**

- **Precision**: $\frac{TP}{TP+FP}$ measures how many predicted detections are correct.
- **Recall**: $\frac{TP}{TP+FN}$ measures how many ground truth instances are correctly identified.

**Tracking Accuracy**

Tracking accuracy quantifies how closely predicted ball positions follow the ground truth across frames. We compute the Euclidean distance between predicted and ground truth centroids. To handle frames without direct detections, we employed linear interpolation to generate continuous trajectories for both prediction and ground truth. Lower average error indicates more accurate tracking.

**Trajectory Smoothness**

Trajectory smoothness is approximated by examining the ball's acceleration profile. After computing velocities between consecutive frames, we derive accelerations. A lower average acceleration magnitude suggests a smoother, more physically consistent trajectory.

| **Metric** | **Out Case** | **Not Out Case** |
|---|---|---|
| Precision | 0.9474 | 0.9091 |
| Recall | 0.9474 | 0.9091 |
| Mean IoU | 0.5810 | 0.7697 |
| Tracking Accuracy (pixels) | 1.91 | 18.52 |
| Trajectory Smoothness (pixels/frame$^2$) | 0.38 | 5.47 |

Table 1: Evaluation Metrics for Out and Not Out Cases

For the *Out* scenario, high precision and recall (0.95) confirm robust detection performance, while the mean IoU (0.5810) indicates moderate spatial overlap. The low tracking error (1.91 pixels) and low acceleration magnitude (0.38 pixels/frame$^2$) reflect stable, accurate tracking.

In contrast, for the *Not Out* scenario, while detection remains strong (precision and recall 0.91) and mean IoU is improved (0.7697), the tracking accuracy is lower (18.52 pixels) and trajectory smoothness is higher (5.47 pixels/frame$^2$), suggesting a need for refinement under more challenging conditions.

### 5.7 Results Conclusion

The proposed system demonstrates a reliable and efficient method for detecting, tracking, and analyzing cricket ball trajectories for LBW decision-making. By leveraging advanced segmentation techniques through SAM2 and incorporating physics-based calculations, it achieves high accuracy and robustness. The close alignment of results with the Hawkeye system validates its potential for real-world applications.

Future improvements could include:

- Enhancing temporal consistency across frames to further improve trace accuracy.
- Reducing reliance on manual initialization for template matching to make the system fully automated.
- Extending the system to analyze additional cricket scenarios, such as catches and runouts.

Overall, this system offers a cost-effective and reliable alternative to traditional tools for cricket analytics.

## 6 Challenges and Limitations

Despite the promising results, the proposed system has several challenges and limitations that need to be addressed for broader applicability and improved performance. These include:

### 6.1 Data Constraints

The availability of labeled datasets specific to cricket ball tracking is limited. Moreover, the use of single-camera setups restricts the system's ability to analyze the ball's motion accurately from different perspectives, especially in cases of obstructions or occlusions.

### 6.2 Camera Stability

Unstable camera angles or movements introduce noise into the tracking process. Such instability can lead to inaccuracies in the calculated trajectory, speed, and impact points.

### 6.3 Manual Annotations

The current system relies on manual annotations, such as marking the stump line for reference. This dependence increases the likelihood of human error and limits the scalability of the approach to larger datasets or real-time applications.

### 6.4 Frame Rate Dependency

The accuracy of speed calculations is directly dependent on the frame rate of the input video. Variations or inaccuracies in frame rate can result in significant errors in speed estimation.

### 6.5 Segmentation Errors

In certain frames, the ball's appearance may blend with the pitch or players' uniforms, leading to segmentation errors. This affects the trajectory trace and can potentially lead to incorrect LBW decisions.

### 6.6 Dynamic Perspectives

Single-camera setups often suffer from perspective distortions, particularly in low-angle or side-on views. These distortions make it challenging to accurately map the ball's motion relative to the stumps and pitch.

### 6.7 Classical Method Issues

While template matching and SAM2 segmentation provided robust results, classical methods like template matching may fail under noisy conditions or with limited data, requiring more advanced techniques to ensure reliability.

## 7 Future Work

To overcome the identified challenges and improve the system, several enhancements are proposed. Incorporating multi-camera setups can provide richer data, reducing perspective distortion and occlusion issues. Automating the detection of reference lines, such as stumps and creases, would eliminate the need for manual annotations, while advanced segmentation models tailored to cricket scenarios can address issues with ball blending or obstructions. Preprocessing video with stabilization algorithms and using higher frame rates or frame interpolation methods can further improve tracking accuracy and speed calculations. Building a comprehensive labeled dataset for cricket ball tracking would enable better benchmarking and foster community-driven innovation. Real-time implementation using optimized algorithms and hardware acceleration would enable on-field decision-making, and expanding the system to cover additional cricket use cases, such as catch detection or fielding analysis, would increase its utility in the sport.

## 8 Author Contributions

While specific tasks were assigned to individual authors, much of the work was collaborative, with significant overlap in responsibilities. This ensured that every team member contributed to key aspects of the project and provided support across different stages.

| Task \ Author | Pratham | Varad | Jugal | Ohm |
|---|---|---|---|---|
| Conceptualization | ✓ | | ✓ | ✓ |
| Formal Analysis | ✓ | ✓ | ✓ | |
| Investigation | ✓ | ✓ | | ✓ |
| Methodology | ✓ | ✓ | ✓ | |
| Resources | | | ✓ | ✓ |
| Software | | ✓ | ✓ | |
| Writing - Original Draft | ✓ | ✓ | | ✓ |

Table 2: Author Contributions

## 9 References

## References

[1] Matplotlib Developers, *Matplotlib Documentation*, https://matplotlib.org/stable/contents.html

[2] Meta AI, *Segment Anything Model (SAM)*, https://segment-anything.com/

[3] Overleaf Team, *Overleaf LaTeX Guides*, https://www.overleaf.com/learn/latex/Inserting_Images

[4] OpenCV Developers, *OpenCV Documentation*, https://docs.opencv.org/4.x/

[5] FFmpeg Team, *FFmpeg Documentation*, https://ffmpeg.org/documentation.html

[6] Abbas, K. et al., *Deep-Learning-Based Computer Vision Approach for the Segmentation of Ball Deliveries and Tracking in Cricket*, arXiv preprint, arXiv:2211.12009 [cs.CV], 2022. https://arxiv.org/abs/2211.12009.

[7] Arora, U. et al., *Cricket umpire assistance and ball tracking system using a single smartphone camera*, PeerJ Preprints 5, e3402v1, 2017. DOI: 10.7287/peerj.preprints.3402v1.

[8] Jayalath, L., *Hawk Eye Technology Used In Cricket*, South Asian Journal of Engineering and Technology 3, pp. 55–67, 2021. DOI: 10.36346/sarjet.2021.v03i02.002.

[9] Sharma, R. A., P. K. Sankar, and C. Jawahar, *Fine-Grain Annotation of Cricket Videos*, arXiv preprint, arXiv:1511.07607 [cs.MM], 2017. https://arxiv.org/abs/1511.07607.