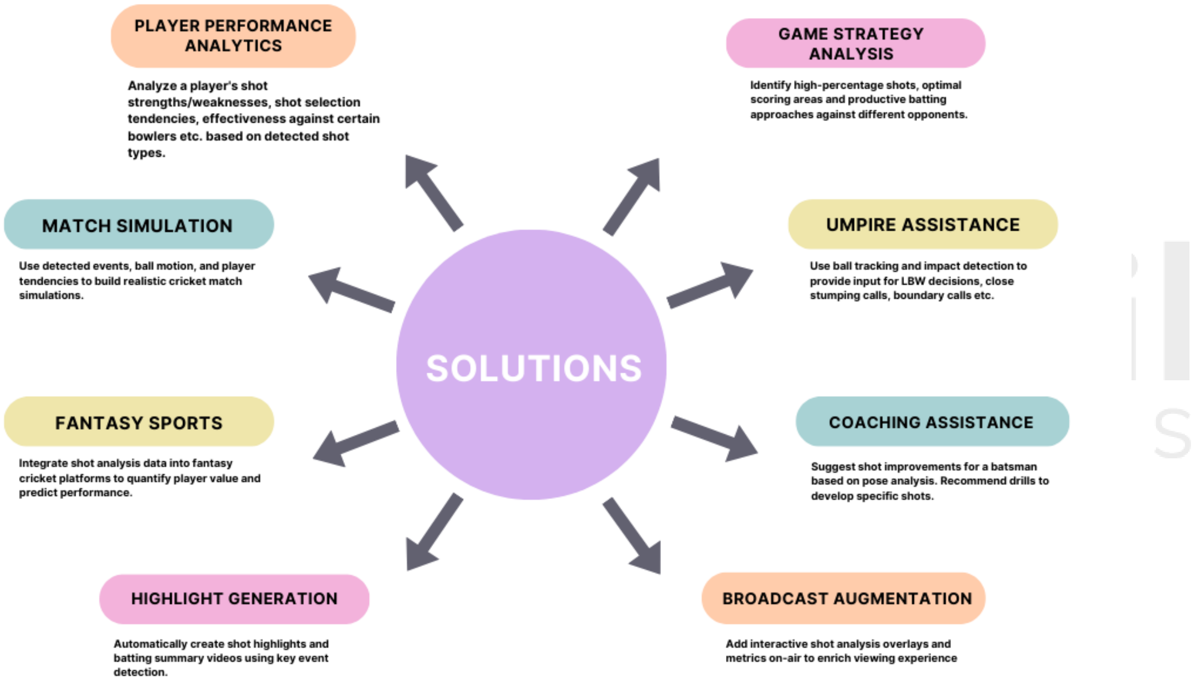
Black Basil
TECHNOLOGIES

WHITE PAPER: REAL TIME CRICKET MATCH ANALYSIS



SOLUTIONS

PLAYER PERFORMANCE ANALYTICS

Analyze a player's shot strengths/weaknesses, shot selection tendencies, effectiveness against certain bowlers etc. based on detected shot types.

GAME STRATEGY ANALYSIS

Identify high-percentage shots, optimal scoring areas and productive batting approaches against different opponents.

MATCH SIMULATION

Use detected events, ball motion, and player tendencies to build realistic cricket match simulations.

UMPIRE ASSISTANCE

Use ball tracking and impact detection to provide input for LBW decisions, close stumping calls, boundary calls etc.

FANTASY SPORTS

Integrate shot analysis data into fantasy cricket platforms to quantify player value and predict performance.

COACHING ASSISTANCE

Suggest shot improvements for a batsman based on pose analysis. Recommend drills to develop specific shots.

HIGHLIGHT GENERATION

Automatically create shot highlights and batting summary videos using key event detection.

BROADCAST AUGMENTATION

Add interactive shot analysis overlays and metrics on-air to enrich viewing experience

Author - Mush-k

Organization - Black Basil Technologies[1]

[1] 1st Floor, A-41, The Iconic Corenthum, Sec 62, Noida - 201301, contact@blackbasiltech.com , (+91) 96507 00345

Introduction

This paper presents a novel computer vision system for automatically analyzing various aspects of a cricket match. A core contribution is a robust object detection model capable of identifying key elements like the batsman, pitch and ball in video footage.

Building on this, an efficient ball tracking algorithm is developed to follow the ball's trajectory over multiple frames. To determine critical gameplay events, the system detects impact between the bat and ball by analyzing their intersection and changes in ball direction.

A key analytical component is a human pose estimation model focused on the batsman to interpret their body position and anticipate shot types. It recognizes common cricket shots including cover drives, defense shots, late cuts, straight drives, lofted drives,sweeps, flicks, hooks, pulls and square cuts.

By combining object detection, motion tracking, physics-based impact modeling and pose analysis, this work enables detailed automatic understanding of cricket batting. It has the potential to provide quantitative player performance analytics and advance computer assistance for umpiring and coaching.

Problem Areas

The areas where problem can be solved are :

Player performance analytics - Analyze a player's shot strengths/weaknesses, shot selection tendencies, effectiveness against certain bowlers etc. based on detected shot types.

Game strategy analysis - Identify high-percentage shots, optimal scoring areas and productive batting approaches against different opponents.

Umpire assistance - Use ball tracking and impact detection to provide input for LBW decisions, close stumping calls, boundary calls etc.

Coaching assistance - Suggest shot improvements for a batsman based on pose analysis. Recommend drills to develop specific shots.

Match simulation - Use detected events, ball motion, and player tendencies to build realistic cricket match simulations.

Fantasy sports - Integrate shot analysis data into fantasy cricket platforms to quantify player value and predict performance.

Highlight generation - Automatically create shot highlights and batting summary videos using key event detection.

Broadcast augmentation - Add interactive shot analysis overlays and metrics on-air to enrich viewing experience

Injury prevention - Detect potential issues with batsman technique that could lead to increased injury risk.

Solutions we built

Integrated end-to-end system - Our solution combines multiple analysis modules (object detection, ball tracking, impact detection, pose estimation) into one unified pipeline rather than separate products.

Rich pose analysis - Detailed batting stance and shot analysis using pose estimation to go beyond just tracking bat swing or ball impact events.

Focus on generating actionable insights - Apply analysis for tangible use cases like strategy, coaching feedback, player development rather than just visualization.

Real-time capabilities - Develop real-time analysis to detect events and shot types with minimal latency for potential use in live matches.

Markerless tracking - Utilize computer vision rather than wearable sensors or markers for non-invasive tracking and analysis.

Customizable algorithms - Allow coaches and analysts to configure models like setting shot type classifiers for individual players.

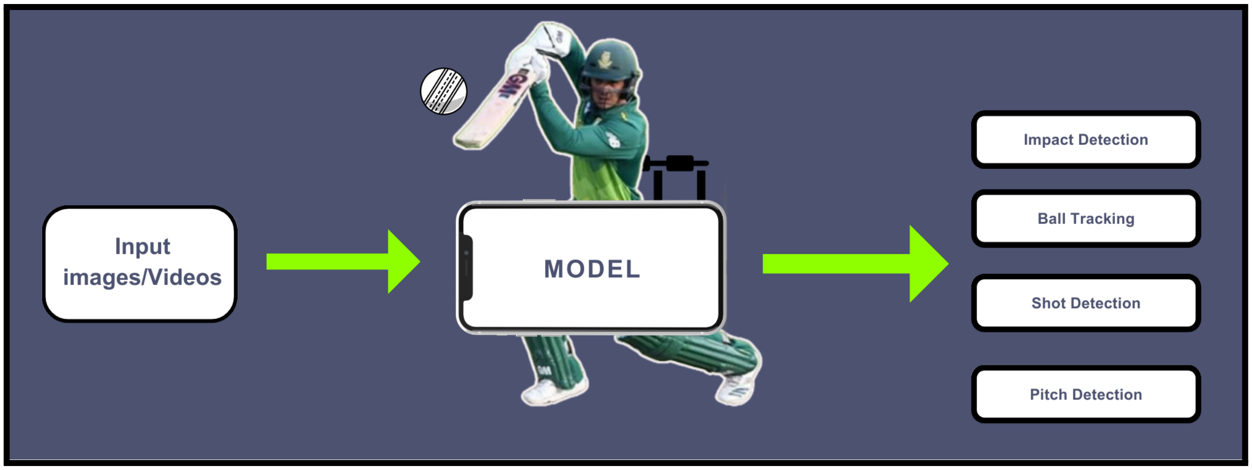
Comprehensive analytics - Combine multiple data sources like video, trajectory data, player stats rather than just computer vision or sensor inputs alone.

Accessibility - Offer as an affordable SaaS platform rather than just high-end broadcast enhancement technology.

Automated highlight generation - Use analysis to auto-generate shareable highlight clips and social media content.

Gamification - Provide performance metrics, shot metrics and advanced analytics in gamified training platforms.

Design Methodology and Component Interactions



Technology Used

State of art models we used

The two popular models from the YOLO family are Yolov5 and Yolov8 for both object Detection and pose estimation.

Yolov5 in Object Detection

YOLOv5 is a popular open source object detection model developed by Ultralytics based on the YOLO (You Only Look Once) family of models.

It follows a one-stage detection approach where the model looks at the image only once and directly predicts class probabilities and bounding box coordinates for the objects present. This makes it fast and efficient.

The YOLOv5 architecture is composed of a backbone feature extractor like CSPNet followed by a neck module and then detection head. The input image goes through the backbone to extract high-level semantic features. The neck further processes these features and the detection head makes the final predictions.

The detection head has three components - bounding box regression, object classification and object confidence. It applies anchor boxes to the feature map and makes predictions for each anchor box. The predictions encode the bounding box offsets, class probabilities and confidence score.

During inference, the model predicts boxes with high confidence and NMS is applied to filter out duplicates.
The final output contains the detected objects with class labels and bounding box coordinates.

YOLOv5 uses various tricks like cross-stage partial connections, multi-scale predictions, and regularization methods to improve accuracy while maintaining high speed which makes it very versatile for real-world applications.

YOLOv8 model

YOLOv8 is the latest version of the YOLO object detector family. Like YOLOv5, it follows a one-stage approach and predicts bounding boxes and class probabilities directly from full images in one pass. The YOLOv8 architecture builds on the strengths of YOLOv5 but adds several improvements:

It introduces a new backbone called CSP-DarkNet which helps improve accuracy. The neck has a modified feature aggregation structure for better multi-scale feature fusion.

It adds a new loss function called IOU (Intersection Over Union) Loss which improves coordinate regression for more precise localization.

The detection head uses cross-scale aggregation to combine different scale predictions better.

It has improved data augmentation techniques like mosaic augmentation and new random filtering to prevent overfitting.

During inference, YOLOv8 follows a similar workflow - the input image goes through the backbone, neck and head to generate box predictions which are then filtered by confidence thresholding and NMS.

Compared to YOLOv5, YOLOv8 achieves better accuracy and speed by optimizing the backbone, neck, loss functions and augmentations while keeping the overall one-stage architecture is similar. The enhancements make it more robust for handling small, large and occluded objects in complex images.

So in summary, YOLOv8 builds on YOLOv5's architecture but has several improvements to the backbone, neck, loss functions and augmentations to achieve better object detection performance. The one-stage approach is kept similar for fast inference.

How YOLOv8 work in pose estimation

In our work, we utilized YOLOv8 for detecting human poses in images and video. Pose estimation involves identifying the locations of key body joints of people and tracking their positions over time. To implement it with YOLOv8, we trained the model on labeled datasets comprising human figures with marked joint locations like elbows, knees, shoulders etc.

YOLOv8's strong object detection abilities allowed it to effectively detect and localize the major body joints in our training data. We used YOLOv8 in a single-stage manner where it directly predicts heatmaps for each joint and offset vectors to refine their coordinates in one pass of the input image.

The predicted heat maps indicate the locations of the joints and offsets fine-tune them. During inference, we aggregate the heatmaps and offsets to determine the final joint locations and connect them to output the full body skeleton pose. We optimized YOLOv8's backbone CNN and neck module to capture multi-scale spatial features, which is important for identifying joints and their spatial relationships.

The single-stage approach also enabled real-time pose estimation. Overall, YOLOv8 provided accurate body joint localization and detected poses for multiple people, which was critical for our application, by leveraging its powerful object detection capabilities.

Using pose estimation how we predict shot detection

We started by implementing pose estimation for cricket batsmen in images using YOLOv8 to detect key body joints. The model was trained to identify relevant joints like wrists, elbows, shoulders, knees and ankles on batsmen figures.

After optimizing the backbone and neck modules, YOLOv8 reliably detected 17 key body points that represent the batsman's pose.

We fed test images into the trained model to output poses with numerical (x,y) coordinates for each joint.

These coordinate locations were saved into dataframes, with each row representing the 34 detected key points for a single batsman image. We compiled datasets containing pose information extracted from over 10,000 labeled cricket batting images depicting various common shot types. The diverse training data encompassed important cricket shot classes including drives, pulls, cuts, sweeps etc.

We cleaned the pose data by removing errors, filtering outliers and handling missing values arising from occlusion. The datasets were split into 80% training and 20% test sets for building and evaluating the shot classification model. We explored Random Forests for pose-based shot recognition as they handle nonlinear data well and provide insight into feature importance.

Random Forest is an ensemble of decision trees built on randomized data subsets and features. We initialized the model with 100 trees and Gini impurity criterion for splits. The detected (x, y) coordinates of the 17 joints served as the feature inputs. The cricket shot type for each image was the classification target variable.

The model was trained on the 80% batsman poses to build decision rules correlating joints' positional relationships with different shots. We tuned hyperparameters like tree depth and leaf size to improve accuracy while avoiding overfitting. Performance was evaluated on the held-out test set poses based on multi-class precision, recall and F1 metrics.

The trained Random Forest classifier achieved a test accuracy of 82% in predicting cricket shot types from just the batsman body joint locations. It learned associations between shots and arm angles, bat positioning, knee bend, body balance etc. Behavior of the ensemble model was analyzed to identify the most informative pose keypoint.

In summary, detecting batsman body poses with YOLOv8 and training a Random Forest provided a practical pipeline for shot classification from spatial joint data. The results demonstrate the viability of using only skeletal key points for recognizing cricket batting strokes.

CONCLUSION

In conclusion, this paper has demonstrated a novel computer vision system for automated, real-time cricket match analysis. The proposed methods allow extraction of quantitative performance metrics, optimal game strategies, coaching insights and more.

Our integrated pipeline analyzing video, trajectory data and player statistics provides comprehensive cricket analytics not possible with any singular approach. With real-time low latency capabilities and an accessible SaaS platform, the system can provide personalized timely feedback to enhance coaching, game strategy, fan engagement and umpiring.

Our distributed architecture and choice of technologies ensure the solution can scale on-demand while maintaining accuracy and speed. The techniques presented have significant potential to transform data-driven decision making in cricket through affordable and customized real-time analytics.

Open source Repository

https://github.com/Open-Black-Basil/cricket-animation.git

REFERENCES

1. Glenn Jocher, et al. Ultralytics YOLO: Real-time Object Detection. Ultralytics, https://docs.ultralytics.com/. Accessed 1 August 2023.

2. Thomas, Graham & Gade, Rikke & Moeslund, Thomas & Carr, Peter & Hilton, Adrian. (2017). Computer vision for sports: Current applications and research topics. Computer Vision and Image Understanding. 159. 10.1016/j.cviu.2017.04.011.

3. Maji, Debapriya, Soyeb Nagori, Manu Mathew, and Deepak Poddar. "Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2637-2646. 2022.

4. T. L. Munea, Y. Z. Jembre, H. T. Weldegebriel, L. Chen, C. Huang and C. Yang, "The Progress of Human Pose Estimation: A Survey and Taxonomy of Models Applied in 2D Human Pose Estimation," in IEEE Access, vol. 8, pp. 133330-133348, 2020, doi: 10.1109/ACCESS.2020.3010248.

5. C. Liu, Y. Tao, J. Liang, K. Li and Y. Chen, "Object Detection Based on YOLO Network," 2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2018, pp. 799-803, doi: 10.1109/ITOEC.2018.8740604.

6. Kamble, Paresh R., Avinash G. Keskar, and Kishor M. Bhurchandi. "A deep learning ball tracking system in soccer videos." Opto-Electronics Review 27, no. 1 (2019): 58-69.

7. Rodriguez-Galiano, V., M. Sanchez-Castillo, M. Chica-Olmo, and M. J. O. G. R. Chica-Rivas. "Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines." Ore Geology Reviews 71 (2015): 804-818.

8. Arceda, V. Machaca, and E. Laura Riveros. "Fast car crash detection in video." In *2018 XLIV Latin American Computer Conference (CLEI)*, pp. 632-637. IEEE, 2018.

9. Shah, Rajiv, and Rob Romijnders. "Applying deep learning to basketball trajectories." arXiv preprint arXiv:1608.03793 (2016).