Comparison of Segmentation and Detection in the First Robotics Competition

Aarush Rajagopalan

Abstract

Advancements in computer vision have transformed robotic perception, with segmentation models offering unprecedented precision over traditional object detection methods. This paper explores the application of Meta AI's Segment Anything Model (SAM) in the context of the 2024 FIRST Robotics Competition (FRC) game, CRESCENDO. SAM delivers high-resolution, pixel-level object masks with minimal input, operating in a zero-shot manner that eliminates the need for extensive retraining. In robotics domains characterized by clutter and occlusion—such as competitive FRC gameplay—segmentation models like SAM and RISE have proven highly effective, especially when labeled data is scarce or conditions evolve over time. We compare SAM's segmentation capabilities with conventional detection models, highlighting its advantages in spatial awareness, contextual understanding, and engineering simplicity. Real-world FRC examples demonstrate how segmentation-based systems enhance localization, alignment, and obstacle avoidance. We propose a vision framework that fuses SAM's segmentation with sensor-based data to improve reliability and strategic autonomy, presenting SAM as a scalable and adaptable vision solution in dynamic robotics environments.

Intro

Advancements in computer vision have revolutionized numerous domains, enabling automation and enhanced decision-making in complex environments. One of the most significant recent innovations is the Segment Anything Model (SAM), developed by Meta AI. SAM represents a breakthrough in image segmentation, offering the ability to generate high-precision, pixel-level object masks with minimal user input (Kirillov et al., 2023). Unlike traditional detection models that rely on bounding boxes, SAM provides detailed segmentation, allowing for more accurate scene understanding.

SAM is designed as a foundation model for segmentation, capable of generalizing to a wide range of tasks without requiring task-specific training. It achieves this by leveraging a powerful image encoder, a flexible prompt encoder, and a lightweight mask decoder. The model is trained on an extensive dataset, SA-1B, which includes over a billion segmentation masks. (Kirillov et al., 2023) Unlike conventional segmentation approaches that require extensive manual labeling, SAM can operate in a zero-shot manner, producing accurate masks from minimal user interaction. This ability makes

SAM a highly adaptable tool for robotics applications, where real-time segmentation and adaptability to new environments are crucial.

While object detection models are effective at identifying the presence and location of objects, they lack the granularity required for precise interactions with a dynamic environment. Detection models typically return bounding boxes, which provide coarse localization but fail to account for an object's exact shape, size, and contours. (Sun et al., 2024) In contrast, segmentation models like SAM deliver pixel-level accuracy, offering a more refined understanding of object boundaries. This level of precision is particularly valuable in robotics, where decisions often depend on precise positioning and spatial awareness.

This paper explores the potential of SAM within the FIRST Robotics Competition (FRC), an annual event where teams design, build, and program robots to complete dynamic game tasks. Each season, FRC presents a new challenge that requires teams to strategize, adapt, and optimize their robots for success in a fast-paced, competitive environment.

In 2024, the FRC game CRESCENDO challenges teams to navigate a music-themed competition, where robots must score notes (orange disks) in speakers (high goal) and the amp (low goal), amplify their speaker, and climb onstage for endgame bonuses. The game features both autonomous and teleoperated phases, demanding precise robot alignment, object tracking, and strategic movement. With SAM's segmentation capabilities, teams can enhance their robots' perception, enabling more accurate positioning, real-time obstacle avoidance, and improved path planning—key factors for success in this dynamic and competitive environment.

One of the most promising applications of SAM in FRC is improving robot alignment from the driver station. By segmenting field elements, game pieces, and surrounding structures, SAM provides real-time visual feedback that enables precise robot positioning. Operators can choose between a traditional camera feed or an augmented display showing an overlayed path, tailoring the experience to their personal preferences.

SAM's approach integrates data from multiple sources, not just the raw input from onboard cameras. This fusion of sensor data creates a richer, more comprehensive view of the field, enhancing the overall perception of the robot's surroundings. As a result, the system offers a more detailed and versatile understanding than basic camera feeds alone.

Unlike traditional object detection methods that simply draw bounding boxes around targets, SAM's segmentation delivers a finer level of detail. This precision allows for more intuitive robot adjustments and reduces the need for frequent code updates, as the system adapts seamlessly to changes in the field environment.

In autonomous navigation, SAM enables robots to recognize and track both static and dynamic obstacles, facilitating more effective path planning and obstacle avoidance.

The detailed segmented maps help robots maintain a clear trajectory, even in challenging conditions, ultimately contributing to more reliable performance during competition.

Comparing Utility and Tradeoffs between Segmentation and Detection

In the field of computer vision, segmentation and detection serve as two fundamental approaches for identifying and analyzing objects within an image. While both methods contribute to decision-making in robotics, they differ significantly in their execution and the level of detail they provide. Instance segmentation classifies every pixel in an image, effectively dividing it into distinct regions corresponding to specific objects or backgrounds (Kirillov et al., 2023). This fine-grained approach is particularly useful for robot localization, as it enables precise identification of field elements, other robots, and obstacles—contributing to better path planning, collision avoidance, and strategic decision-making. Multiple studies in autonomous navigation and robotics have demonstrated that increased spatial detail from segmentation can lead to measurable improvements in these areas (Kimhi et al., 2025).

Table 1: Comparison of Segmentation vs. Detection for System Design Decisions

	Segmentation	Detection
Definition / Summary	Classifies each pixel in an image, generating masks that outline exact shapes and boundaries of objects	Identifies objects and draws bounding boxes around them to indicate approximate location
Speed	Generally slower per frame due to higher computational load, especially on less powerful hardware	Typically faster and more lightweight, especially suitable for real-time applications on edge devices
Accuracy	High pixel-level accuracy; better in cluttered scenes and for precise localization	Lower spatial precision; struggles with occlusion, dense clusters, and irregular shapes
Ease of use	Zero-shot models like SAM reduce setup effort, integration may require geometry-based post-processing	Simpler outputs but often requires custom training, bounding box tuning, and NMS

In contrast, detection identifies and classifies objects by placing bounding boxes around them, providing a more generalized representation of their locations without delineating exact boundaries. Although detection is commonly used to locate game pieces, obstacles, or competing robots, it often struggles with distinguishing individual objects in tightly clustered scenarios (Sun et al., 2024). Research indicates that detection models may merge adjacent objects into a single bounding box, while segmentation generally offers improved separation of closely spaced items, although in very dense clusters even segmentation can face challenges.

From a technical standpoint, detection pipelines are often more code-intensive, as they require custom models and complex processing steps—such as generating, refining bounding boxes, and applying Non-Maximum Suppression (NMS) to remove duplicates—to achieve high accuracy (Sun et al., 2024). While modern zero-shot detection models are emerging, many still demand additional fine-tuning and exhibit variable performance across different datasets. In contrast, segmentation models like SAM operate in a zero-shot manner with minimal retraining. Furthermore, although segmentation outputs provide pixel-level detail, useful metrics such as object size, centers, and derived bounding boxes can be efficiently extracted through computational geometry methods, thereby transforming raw masks into actionable information without extensive postprocessing.

Additionally, segmentation is critical for localization and mapping tasks such as visual odometry and SLAM. By segmenting key field features—boundary lines, goal zones, and obstacles—robots can generate detailed reference landmarks for accurate positioning without relying on heavily pre-trained detection models. One significant advantage of segmentation is the rich contextual information it provides; rather than merely reporting object positions, segmentation delivers detailed shape and texture data (Kirillov et al., 2023) that enables robots, and their vision systems to infer spatial relationships between objects. This additional context is invaluable when distinguishing between allied and opposing robots or closely spaced obstacles, thereby optimizing navigation, shooting angles, and autonomous routines. In contrast, as highlighted by Sun et al. (2024), detection pipelines often involve multi-stage processes—such as designing anchor boxes, hyperparameter tuning, and applying post processing techniques like non-maximum suppression—that demand extensive custom coding and model-specific engineering. While detection may offer a lighter per-frame computational footprint, these additional engineering challenges can hinder adaptability in dynamic competition environments.

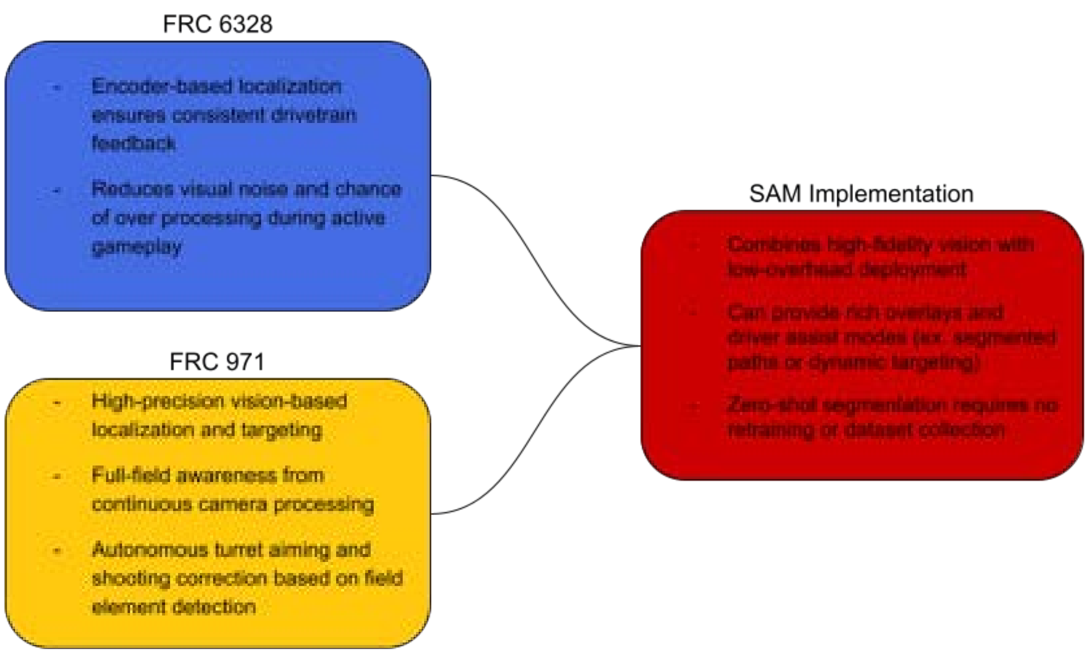
Sample Use Case and Implementation Details

FRC 971, a high school robotics team from Mountain View, California, utilizes a custom machine learning model combined with cameras to determine the exact position of their robot on the field. By leveraging this precise localization, their system can autonomously aim a turret toward the goal and compute

the optimal shooting speed to propel the "note" (an orange disk) accurately. Additionally, FRC 971 employs camera-based object detection to locate the "stage," a trussed structure that can obstruct a direct shot into the goal, in 3D space. If the system detects an obstruction, it either provides a prompt to the driver for manual intervention or autonomously directs the shot over the stage into a predetermined corner of their zone. This approach ensures a high degree of accuracy in scoring while minimizing wasted shots.

FRC 6328, another high school robotics team from Littleton, Massachusetts, follows a slightly different methodology, relying on positional data derived from sensors like magnetic encoders to track the rotation of the wheels in their drive base. Rather than depending on full camera-based localization, their approach triggers alignment only when the shoot button is pressed. Cameras are primarily used to detect "notes" on the ground to activate their intake mechanism. This strategy simplifies the system architecture, reduces computational overhead, and maintains consistency, albeit at the cost of requiring more driver involvement for alignment tasks.

Diagram 1: Comparison of Vision Strategies: FRC 971 vs. FRC 6328 vs. SAM



A hypothetical FRC team integrating the Segment Anything Model (SAM) could achieve a synergy between these two approaches. SAM provides the reliability and high-fidelity spatial awareness seen in FRC 971's camera-based system, while offering the adaptability and simplicity valued by FRC 6328. Through precise segmentation, SAM can identify and outline field elements, game pieces, and even other robots, providing rich pixel-level information for decision-making. Its zero-shot capability removes the need for extensive dataset collection and training, making it ideal for teams that want flexibility and power without a significant development burden.

An alternative and widely used vision solution in FRC is the Limelight system, which offers a robust and production-tested library for both Java and C++. Limelight supports a wide range of targeting, localization, and pose estimation features—ranging from basic retroreflective target detection to advanced AprilTag tracking with MegaTag2 integration. Teams can access simplified targeting information like horizontal offset for basic alignment or use pose estimation methods to update their robot's position using vision data. Limelight also supports dynamic pipeline switching, LED control, Python script integration, and even neural network-based object detection pipelines.

While Limelight excels in structured vision tasks with pre-defined tags and pipelines, it differs from SAM in its approach to adaptability and data richness. SAM provides high-resolution, generalized segmentation of any visible object, enabling applications in game scenarios where elements are not tagged or may vary in appearance. Conversely, Limelight's strength lies in its speed, low latency, and plug-and-play support for retroreflective and fiducial targets, with rich pose estimation tools and simplified deployment via NetworkTables.

Table 2: Comparison of Limelight vs. Custom SAM Integration

	Limelight	Custom SAM
Ease of Use	Very easy to set up and integrate with FRC codebases; robust documentation and plug-and-play support	Requires external hardware and custom code to run inference and integrate outputs.
Benefits	Low latency, optimized pipelines, AprilTag support, direct NetworkTables integration	High-resolution masks of any object, zero-shot segmentation, contextual understanding of scenes.
Drawbacks	Limited to known targets (tags, retroreflective elements); less useful when objects are untagged, for object/gamepiece detection	Slower inference and more complex integration pipeline; needs hardware acceleration for real-time use

Ultimately, both SAM and Limelight represent valuable tools in a team's vision arsenal. SAM excels in flexible, zero-shot segmentation with deep contextual understanding, while Limelight offers a proven, performance-optimized framework for fast and reliable targeting and localization. A well-rounded vision strategy might even combine both—using Limelight for tag-based localization and SAM for game piece segmentation and driver-assist overlays—pushing the boundaries of what is possible in FRC robot perception.

Conclusion

The Segment Anything Model (SAM) from Meta introduces a powerful shift in how robots perceive and interact with their environment, offering a significant upgrade over traditional object detection approaches. Its ability to generate high-resolution, pixel-accurate segmentation masks in a zero-shot manner—with minimal need for retraining or custom datasets—positions it as a highly adaptable tool for competitive robotics applications like the FIRST Robotics Competition (FRC).

Throughout this paper, we have explored how SAM's segmentation capabilities provide key advantages in robot localization, path planning, obstacle avoidance, and decision-making. Unlike detection models, which rely on bounding boxes and often struggle in cluttered or dynamic environments, SAM delivers detailed object boundaries and contextual cues that enhance a robot's spatial awareness. These qualities are particularly valuable in FRC games like CRESCENDO, where precise alignment and real-time responsiveness are crucial for scoring and mobility.

We also compared the engineering demands of segmentation versus detection. While detection may offer a simpler per-frame computational profile, it typically requires more extensive custom coding, complex model tuning, and post-processing to achieve robust performance. SAM, by contrast, minimizes engineering overhead while maintaining high accuracy across diverse scenarios. Its segmentation outputs can be easily converted into actionable metrics—like object centers or bounding boxes—through lightweight computational geometry, bridging the gap between raw visual input and robot control systems.

Real-world examples from FRC teams such as 971 and 6328 illustrate the spectrum of current localization strategies, from advanced camera-based pipelines to simpler sensor-driven systems. A SAM-based solution could bridge these approaches, delivering the robustness of vision-based localization with the ease of deployment typically associated with sensor-based methods. By combining visual segmentation with sensor fusion, teams can enhance both autonomous and driver-controlled performance with minimal added complexity.

Ultimately, SAM represents a foundational shift in how vision can be integrated into robotics. For FRC teams aiming to improve adaptability, precision, and efficiency, SAM offers a scalable, plug-and-play solution that reduces development time while unlocking new levels of strategic potential on the field. As segmentation models continue to advance, their role in robotics will likely expand—enabling smarter, faster, and more responsive autonomous systems in both competition and real-world environments.

References

1. Fang, Y., Liao, B., Wang, X., Fang, J., Qi, J., Wu, R., Niu, J., & Liu, W. (2021). You Only Look at One Sequence: Rethinking Transformer in Vision through Object Detection. ArXiv.org. https://arxiv.org/abs/2106.00666
2. Kimhi, Moshe, et al. "Robot Instance Segmentation with Few Annotations for Grasping." 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 26 Feb. 2025, pp. 7939–7949, https://doi.org/10.1109/wacv61041.2025.00771.
3. Kirillov, Alexander, et al. "Segment Anything." ArXiv (Cornell University), 5 Apr. 2023, https://doi.org/10.48550/arxiv.2304.02643.
4. "Limelight Zero-Code Robotic Perception | Limelight Documentation." *docs.limelightvision.io*, docs.limelightvision.io/docs/docs-limelight/getting-started/summary.
5. Yibo Sun, Zhe Sun, Weitong Chen. "The Evolution of Object Detection Methods." *Engineering Applications of Artificial Intelligence*, vol. 133, 1 July 2024, p. 108458, https://doi.org/10.1016/j.engappai.2024.108458.