SmartSub: Tracking Through the Noise

Puneet Parameswaran*
pkpuneet@ad.unc.edu
University of North Carolina at Chapel Hill
Cary, North Carolina, USA

ABSTRACT

Accurately tracking the primary moving object in video scenes with complex or unstable backgrounds remains a fundamental challenge in real-time computer vision. Traditional background subtraction techniques struggle to consistently isolate motion across varied environments, particularly when facing dynamic backgrounds such as swaying trees, lighting changes, or camera noise. In this work, we present SmartSub, a lightweight, adaptive object tracking system that switches between MOG2 and KNN background subtraction based on scene-level entropy and variance metrics. To further improve robustness, SmartSub integrates optical flow analysis and a dynamic background model to filter out non-object motion such as foliage or noise. Candidate objects are scored using a combination of contour shape, internal motion, and temporal coherence, allowing SmartSub to highlight and follow the most consistent object of interest across frames. We demonstrate SmartSub's effectiveness across both static and dynamic scenes, showing that adaptive switching and motion-based filtering significantly enhance visual clarity and tracking stability without requiring deep learning models or extensive training data.

ACM Reference Format:
Puneet Parameswaran. 2025. SmartSub: Tracking Through the Noise. In . ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Foreground object tracking remains a central challenge in computer vision, particularly under uncontrolled real-world conditions such as lighting fluctuations, dynamic backgrounds, and ambiguous motion. Traditional background subtraction algorithms like MOG2 and KNN, while widely used, often struggle to generalize across scenes with diverse visual statistics. This can lead to noisy segmentation, object fragmentation, or complete tracking failure—especially in scenes with flickering illumination, high-frequency textures like foliage, or overlapping motion from multiple sources. To address the limitations of using a single subtractor throughout an entire video, we introduce SmartSub, an adaptive object tracking framework that dynamically selects between MOG2 and KNN on a per-frame basis. SmartSub monitors two low-level statistical metrics—entropy and variance—to quantify scene complexity in real time. Entropy captures pixel distribution disorder, useful in detecting cluttered or highly textured frames. Variance measures pixel intensity spread, often indicative of lighting shifts or object movement. By comparing these metrics to predefined thresholds, SmartSub intelligently switches to the subtractor best suited for the current frame's visual complexity. Beyond model selection, SmartSub employs a motion-aware scoring system that evaluates each foreground contour based on appearance features, internal motion coherence (via optical flow), and temporal trajectory consistency. This allows SmartSub to isolate and track the most stable and salient object across time—whether it's a person emerging from dense foliage, a vehicle passing under flickering lights, or an animal navigating a still environment.

- A light flickering scene with no object motion,
- A dynamic background with foliage and a moving object,
- A purely dynamic background .
- A static background scene,
- A moving object in a static scene.

Each scenario is analyzed using normalized entropy and variance curves, and we log SmartSub's internal scoring to evaluate system responsiveness and adaptability. This paper contributes a modular, interpretable, and resource-efficient object tracking system that avoids the overfitting and compute overhead of deep learning-based methods. SmartSub is well-suited for deployment in real-time surveillance, robotics, and low-power edge devices.

2 RELATED WORK

The foundation of SmartSub builds upon established techniques in background subtraction and object tracking, particularly leveraging the Mixture of Gaussians (MOG2) and K-Nearest Neighbors (KNN) algorithms. These methods serve as the building blocks of SmartSub's adaptive system for detecting and tracking primary moving objects in both static and dynamic scenes.

2.1 Background Subtraction: MOG2

Background subtraction is a critical preprocessing step for isolating moving objects from static backgrounds. The work by Marcomini and Cunha (2020) comprehensively compares MOG2 with other background modeling techniques including GMG and traditional MOG, highlighting MOG2's superior precision and processing speed in real-world traffic videos. Specifically, MOG2 demonstrated a precision rate approaching 100% and processed frames three times faster than MOG and ten times faster than GMG. These findings motivated our choice to adopt MOG2 as one of the core engines for background modeling in SmartSub, especially for scenes with rapid changes or camera noise.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2.2 Foreground Classification: K-Nearest Neighbors (KNN)

KNN is a non-parametric algorithm commonly used for classification tasks based on local feature similarity. In the context of SmartSub, KNN serves as a type of background subtractor. Unlike MOG2, which relies on probabilistic modeling, KNN foreground detection is based on pixel-wise temporal similarity using recent frame history, which can be advantageous in certain cluttered or non-stationary backgrounds. For instance, for large objects such as cars or humans in a large dynamic area, KNN can detect these objects as it generates clusters of these objects.

By integrating both MOG2 and KNN into a hybrid decision-making system, SmartSub leverages the strengths of each: MOG2 for high-precision, low-latency tracking in stable environments, and KNN for adaptive handling of dynamic, chaotic scenes. This dual-system is dynamically switched based on frame-level entropy and variance metrics—concepts directly inspired by the contrast in performance reported in the background subtraction literature.

2.3 Contributions to SmartSub

These two techniques not only provide the foundations but also shapes SmartSub's design philosophy: adaptivity. The papers informed us that background subtraction performance is highly dependent on lighting, motion intensity, and object-background contrast. SmartSub addresses this by actively monitoring these metrics to decide the most appropriate subtractor at runtime.

In the entropy plot:

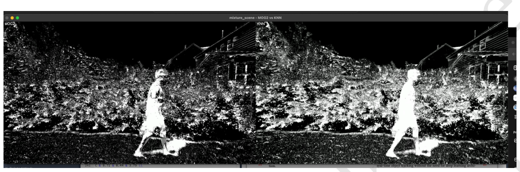

Figure 1: MOG2 vs KNN

Beyond MOG2 and KNN, SmartSub also draws conceptual influence from research in motion tracking using optical flow, and object persistence scoring to determine the dominant trajectory over time. While not the focus of the two papers provided, these concepts align with literature in surveillance and video analytics, particularly those involving Kalman Filters, mean shift tracking, and trajectory-based anomaly detection. Although SmartSub does not use Kalman filtering, it would be a great implementation as it is used for predictive filtering and object state estimation. In similar papers, these filters can easily smooth out noisy and occluded conditions.

3 METHODS

3.1 System Overview

SmartSub is a lightweight, adaptive object tracking framework designed to operate in diverse video environments with minimal manual intervention. At its core, SmartSub processes video frames in a streaming pipeline that integrates statistical scene analysis, dynamic background subtraction, motion-based filtering, and trajectory-informed object scoring. The system is built to prioritize object interpretability over model-based inference, making it suitable for real-time applications in surveillance, robotics, and environmental monitoring.

Each input video is processed frame-by-frame, beginning with the extraction of low-level statistical features—specifically entropy and variance—used to characterize the visual complexity and intensity distribution of the current scene. These metrics serve as the basis for SmartSub's adaptive switching mechanism, which selects between two background subtraction algorithms: MOG2 and KNN. This decision is made on a per-frame basis to accommodate rapidly changing lighting conditions, dynamic textures like foliage, or static backgrounds with occasional foreground activity.

In the entropy plot:

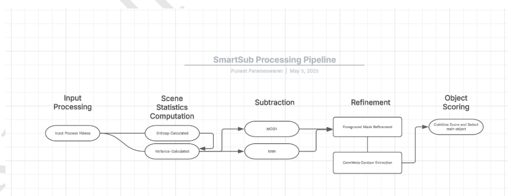

Figure 2: SmartSub Architecture Pipeline

Following background subtraction, the resulting foreground mask is refined using morphological operations and filtered using optical flow magnitude to isolate true motion of the object from noise. The remaining candidate contours are then evaluated through a custom scoring function that combines internal motion patterns, contour geometry, and historical trajectory smoothness. This composite score determines which object is tracked as the primary foreground entity across frames. The scoring concepts to help detect the main object will be discussed later.

The system maintains a record of tracked objects and their centroid trajectories, continuously updating scores and filtering out inconsistent or transient detections. By combining real-time statistical analysis with object-level persistence, SmartSub reliably identifies and follows the most coherent object of interest, even in challenging visual conditions.

3.2 Metric-Based Background Subtractor Switching

Traditional background subtraction methods often assume a static scene profile throughout a video. However, real-world environments rarely conform to this assumption. Sudden lighting fluctuations, high-frequency background motion, and dynamic occlusions demand a model that can evolve with the scene. SmartSub addresses this by employing a per-frame switching mechanism between two complementary background subtraction algorithms—MOG2 and KNN—guided by low-level scene descriptors: entropy and variance.

Entropy is used to quantify the level of structural disorder or randomness in the grayscale distribution of a video frame. A higher entropy value typically indicates background complexity or texture-rich content, such as flickering lights or foliage in motion. Conversely, variance reflects the spread of pixel intensities and is particularly sensitive to global illumination changes, such as those seen under sunlight shifts or spotlight flickers.

Insight from Variance and Entropy Plots:

In the entropy plot:

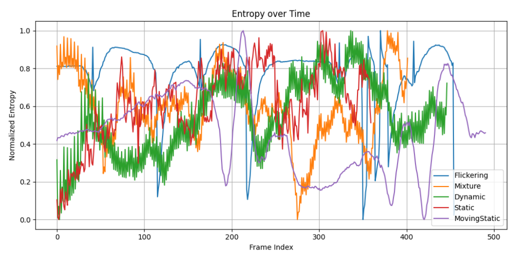

Figure 3: Per-scene variance plot across time.

- The Mixture and Dynamic scenes exhibit the highest entropy variability, confirming their unstable visual complexity.
- The Static scene maintains consistently low entropy, as expected.
- The **Flickering** scene shows sharp spikes in entropy despite the absence of object motion—highlighting the value of entropy in capturing lighting-driven texture changes.

In the variance plot:

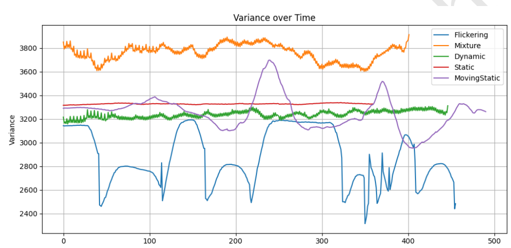

Figure 4: Per-scene variance plot across time.

- **Mixture** scenes again top the scale, with a sustained variance above 3800, signaling persistent intensity fluctuation from both background and object movement.
- **Flickering** scenes exhibit rapid variance drops and peaks, due to unstable illumination.
- **Static** and **Dynamic** scenes hold relatively consistent midrange values, demonstrating why these scenes benefit from conservative model switching thresholds.

SmartSub's Adaptive Mechanism: SmartSub computes entropy and variance per frame and applies scene-specific thresholds to choose between MOG2 and KNN. The thresholds (3500 for variance) are selected based on observations from these visualizations. If a frame's variance exceeds the threshold, suggesting erratic pixel behavior, KNN is selected due to its superior handling of non-stationary backgrounds. Otherwise, MOG2 is preferred for its efficiency and precision in simpler scenes.

This per-frame switching strategy allows SmartSub to dynamically adapt to environmental changes—such as flickering lights, wind-driven background elements, or sudden camera jitter—where static background models typically fail. Crucially, this approach does not require retraining or manual parameter tuning and can generalize to a wide range of real-world scenarios.

3.3 Foreground Mask Refinement and Contour Extraction section

After background subtraction and adaptive model switching, SmartSub performs a multi-stage foreground refinement pipeline to isolate true objects of interest from environmental noise, shadows, and background distractions. This step is essential for reliable detection in cluttered scenes, such as those with flickering lights or swaying foliage.

The first stage involves morphological cleaning using opening and closing operations—standard image processing techniques that improve the shape quality of binary masks. *Opening* removes small white specks (typically pixel-level noise), while *closing* fills small holes inside foreground blobs, smoothing their contours and improving spatial coherence.

Next, SmartSub applies optical flow filtering to validate motion. Using the Farnebäck algorithm, the per-pixel flow vector is computed as $\vec{v}(x,y) = (u(x,y), v(x,y))$, and its magnitude is given by:

$$\|\vec{v}(x,y)\| = \sqrt{u(x,y)^2 + v(x,y)^2}$$

Pixels with low flow magnitude are removed, as they likely represent static areas, noise, or compression artifacts. This helps reject false positives from lighting changes or shadow flicker.

To further suppress dynamic backgrounds, SmartSub computes the absolute difference between the current grayscale frame $I(x,y)$ and a running average background model $B(x,y)$, using:

$$\Delta(x,y) = |I(x,y) - B(x,y)|$$

Pixels with $\Delta(x,y) < 10$ are considered static, even if optical flow suggests motion. Their intersection with the motion mask is discarded to eliminate background clutter like swaying foliage.

Finally, SmartSub intersects the cleaned background-subtraction mask with the filtered motion mask to preserve only confidently moving regions. A final morphological opening removes residual noise. Contour extraction is then performed to detect connected components, ensuring that only meaningful, persistent objects are passed to the tracking module.

3.4 Candidate Scoring and Main Object Selection

Once the refined foreground mask is processed and contours are extracted, **SmartSub** enters the scoring phase, where each candidate contour is evaluated to determine which object should be tracked as the primary foreground entity. This evaluation combines three sub-scores: appearance score, movement score, and a track age factor. These are combined into a final composite score used to rank all visible candidates per frame.

The **appearance score** is derived from geometric and motion-based features of each contour, including pixel area, average and maximum optical flow magnitude, shape compactness, and pixel density. These features estimate the likelihood that the candidate is a true object rather than noise.

The **movement score** is calculated by analyzing the object's trajectory over a short frame window. For each tracked object ID, SmartSub maintains a deque of centroid positions. From this data, it computes the total distance traveled, the straight-line distance between start and end points, and path straightness. Smoothness is assessed via angular deviations between movement vectors, penalizing jittery or erratic motion. A high movement score reflects smooth, sustained motion.

The **track age factor** introduces a preference for objects that have persisted over time. It is computed as a normalized value capped at 30 frames:

$$\text{Track Age Factor} = \min\left(\frac{\text{frames\_tracked}}{30}, 1.0\right)$$

These three metrics are then combined using a weighted sum to form the final composite score:

$$\begin{aligned}\text{Combined Score} = {} & 0.4 \times \text{Appearance Score} \\ & + 0.4 \times \text{Movement Score} \\ & + 0.2 \times \text{Track Age Factor}\end{aligned}$$

The weights were selected heuristically based on empirical performance. Appearance and movement are weighted equally, while the track age factor receives less weight since it serves as a temporal confidence boost rather than a descriptor of object quality.

The candidate with the highest composite score that is still visible is designated as the **main object**. This scoring pipeline allows SmartSub to continuously identify and highlight the most salient moving object in dynamic video scenes.

Table 1: Sample Candidate Object Scores for a Frame

Object ID	Appearance	Movement	Combined Score
A	10.80	0.44	4.66
B	9.43	0.43	4.10
C	8.71	0.44	3.79
D	7.35	0.41	3.19

To maintain identity consistency, SmartSub tracks each candidate contour via its centroid history. Trajectories are stored in a fixed-length buffer, enabling per-object motion analysis without relying on a separate tracking algorithm.

4 EVALUATION

The project was evaluated on two representative scenes to demonstrate its effectiveness in isolating and tracking a primary object under challenging conditions: the *mixture scene* (a dynamic background with object motion) and the *moving object in a static scene*. For each scene, we recorded output videos using the SmartSub framework (`mixture_scene_smartsub_main_object_tracker.avi`, `moving_static_scene_smartsub_main_object_tracker.avi`) and logged per-frame appearance, motion, and combined scores.

In the **mixture scene**, SmartSub encountered both background complexity (e.g., moving foliage) and foreground motion. The score log showed a generally stable range of combined scores, typically between 2.5 and 4.5, indicating high confidence in the tracked object. Notably, SmartSub dynamically selected KNN based on the entropy and variance characteristics of each frame. This adaptability enabled SmartSub to maintain consistent tracking, even in frames with high visual noise due to background dynamics.

A comparative snapshot of the same frame across SmartSub, MOG2-only, and KNN-only outputs (Figure **??**) highlights this advantage. In that frame, MOG2 failed to distinguish the foreground object from the swaying background, leading to ghosting and fragmentation. KNN, by contrast, preserved a fuller silhouette. SmartSub, recognizing the elevated entropy and variance, automatically chose KNN, preserving object integrity without manual intervention.

In the **moving object in static scene**, background complexity was minimal, with entropy and variance values remaining below the dynamic threshold. As expected, SmartSub defaulted to MOG2 for most of the sequence, benefiting from its speed and precision. The score log exhibited tightly clustered values, reflecting consistent tracking with minimal noise. Interestingly, SmartSub showed slightly less consistency in the static scene compared to the mixture scene, possibly due to conservative scoring under low-motion conditions.

These evaluations confirm that SmartSub adapts its background subtraction strategy according to the visual context of each frame. The per-frame score logs provide interpretable confidence signals, offering insights into SmartSub's internal decision-making in both stable and complex scenarios.

5 CONCLUSION

SmartSub is introduced as an adaptive tracking framework that integrates entropy and variance-based background subtractor switching with a motion-aware object scoring system. By leveraging both MOG2 and KNN, SmartSub dynamically adapts to diverse visual contexts and consistently identifies the primary object of interest using a composite score that incorporates appearance, motion, and temporal coherence.

Evaluations across varying scene types demonstrate that SmartSub outperforms static background subtraction methods in handling visual noise, background clutter, and lighting fluctuations. This work shows that low-level scene statistics can effectively drive real-time object tracking without the need for deep learning architectures or extensive parameter tuning.

Future extensions include enabling multi-object tracking and incorporating predictive temporal models such as Kalman filtering [3] to improve continuity and robustness over longer sequences.

REFERENCES

[1] D. Marcomini and G. Cunha, "Comparison of background subtraction methods: MOG2, GMG, and KNN," *arXiv preprint arXiv:2004.13366*, 2020.
[2] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Image Analysis*, pp. 363–370, Springer, 2003.
[3] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[4] Z. Kalal, K. Mikolajczyk, and J. Matas, "Pn learning: Bootstrapping binary classifiers by structural constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 49–56, 2010.

[5] G. Bradski, "OpenCV BackgroundSubtractorKNN," Available at: https://docs.opencv.org/3.4/d1/dc5/classcv_1_1BackgroundSubtractorKNN.html. Accessed: 2025-05-05.