Automated Billboard Replacement in Videos: A Deep Learning Pipeline for Real-Time Content Localization

Emre Belikırık
Department of Computer Engineering
Hacettepe University
Ankara, Turkey
emrebelikirik25@hacettepe.edu.tr

***Abstract*—Manual billboard replacement in video content is a labor-intensive and expensive post-production process that limits content localization and dynamic advertising capabilities. This paper presents an end-to-end automated pipeline for billboard detection, segmentation, tracking, and replacement in video sequences. The system integrates state-of-the-art deep learning models including fine-tuned YOLOv8 variants for detection and segmentation, SAM2 (Segment Anything Model 2) for video object segmentation, and multiple tracking algorithms for temporal mask propagation. A comprehensive comparison was conducted between fine-tuned models and zero-shot approaches using YOLO-World for open-vocabulary detection. Additionally, various tracking strategies, ranging from optical flow to planar Kalman filtering, were evaluated to analyze trade-offs between accuracy and real-time performance. The proposed system enables automated, scalable billboard content replacement suitable for digital advertising and content localization applications.**

***Index Terms*—billboard detection, video object segmentation, deep learning, SAM2, YOLOv8, YOLO-World, perspective transformation**

I. Introduction

The digital advertising industry increasingly demands dynamic content that can be localized, personalized, and updated in real-time. Billboards appearing in video content—whether in sports broadcasts, films, or user-generated media—present significant opportunities for targeted advertising and content monetization. However, the manual process of replacing billboard content in video sequences is prohibitively expensive, requiring skilled visual effects artists and extensive frame-by-frame editing.

Traditional post-production workflows face several challenges: (1) the labor-intensive nature of manual segmentation, (2) difficulty maintaining temporal consistency across video frames, (3) handling perspective distortions as camera angles change, and (4) ensuring seamless blending between replaced content and original footage. These challenges motivate the development of automated solutions that can reduce costs while maintaining visual quality.

This work addresses these challenges by developing an end-to-end automated pipeline for billboard replacement in videos. The proposed approach leverages recent advances in deep learning, specifically:

- **Detection:** Fine-tuned YOLOv8-n/s/m object detection models that predict bounding boxes around billboard regions. Additionally, zero-shot YOLO-World is evaluated for open-vocabulary detection without domain-specific training data. Detection provides the initial localization that guides subsequent processing stages.
- **Segmentation:** Pixel-level mask prediction using fine-tuned YOLOv8-seg models that simultaneously predict bounding boxes and instance masks. Furthermore, SAM2 (Segment Anything Model 2) is integrated as a foundation model that accepts detection outputs (bounding boxes or points) as prompts to generate precise segmentation masks. This hybrid approach combines the detection accuracy of fine-tuned models with SAM2's zero-shot segmentation capability.
- **Tracking:** Multiple tracking algorithms including SAM2 propagation with memory-based attention, Kalman filtering for smooth motion prediction, and optical flow methods for temporal mask consistency across video frames.
- **Perspective Transformation:** Automated 4-corner extraction from segmentation masks and homography-based warping for geometrically correct replacement content insertion.
- **Post-Processing:** Temporal smoothing algorithms including Kalman filtering and IIR cascaded filters to eliminate flicker and ensure visual coherence across frames.

The main contributions of this paper are:

1) A comprehensive comparison of fine-tuned vs. zero-shot detection and segmentation approaches for billboard detection.
2) An extensive benchmark of six tracking algorithms evaluating accuracy-speed trade-offs.
3) An integrated pipeline combining detection, tracking, and replacement with temporal smoothing.

II. Related Work

The process of automated billboard replacement relies on combining several established computer vision tasks. The most basic requirement is object detection to find where a billboard

is in a frame. YOLO (You Only Look Once) [8] is the most popular choice for this because it is fast and easy to implement. Recent versions like YOLOv8 [6] allow us to not only detect boxes but also get pixel-level masks through instance segmentation, which is a key part of our project.

For more complex shapes, the Segment Anything Model (SAM) [7] has changed how we think about segmentation. Instead of training a model for every specific object, SAM allows us to find objects by just giving it a box or a point prompt. Our work uses the newer SAM2 [10] variant, which is designed to handle videos by "remembering" objects from previous frames. This helps solve the problem of flickering masks in video sequences.

Tracking is another area where many methods exist. Simple methods like Optical Flow or Kalman Filters [3] are very fast but often fail if the camera moves too quickly. More advanced methods like XMem [5] or Cutie [4] use deep learning to keep track of objects over long periods. We compare several of these to find the best balance between speed and accuracy for a real-time system.

Finally, for the actual replacement, most commercial systems use manual editing. Some researchers have started using generative models like Stable Diffusion [9] to "paint" new ads into videos. However, for billboards, a simple perspective transformation using the OpenCV library [2] is usually enough to get a realistic result. Our pipeline follows the inspiration of projects like "Track Anything" [12], but adds an automated detection step so the user doesn't have to click on the billboard manually.

III. Methodology

The pipeline consists of four sequential stages: Detection & Segmentation, Tracking, Replacement, and Post-Processing. Figure 1 illustrates the overall system architecture.

A. Stage 1: Detection and Segmentation

The first stage localizes billboards in video frames and generates precise segmentation masks. This stage evaluates multiple approaches for both detection (bounding box prediction) and segmentation (pixel-level mask prediction).

1) Detection Approaches:

a) Fine-Tuned YOLOv8 Detection: YOLOv8 detection variants (nano, small, medium) were fine-tuned on a large-scale billboard detection dataset comprising 3,798 images. The detection models predict bounding boxes and class confidence scores for billboard regions. Training utilized extensive data augmentation including HSV color-space transforms (Hue: ± 0.015, Saturation: ± 0.7, Value: ± 0.4), geometric distortions (rotation $\pm 15°$, shear $\pm 5°$, perspective 0.001), and advanced regularization (Mosaic $p = 1.0$, MixUp $p = 0.15$, Random Erasing $p = 0.4$). The detection loss combines classification and bounding box regression:

$$\mathcal{L}_{det} = \mathcal{L}_{cls} + \lambda_{box}\mathcal{L}_{box} \tag{1}$$

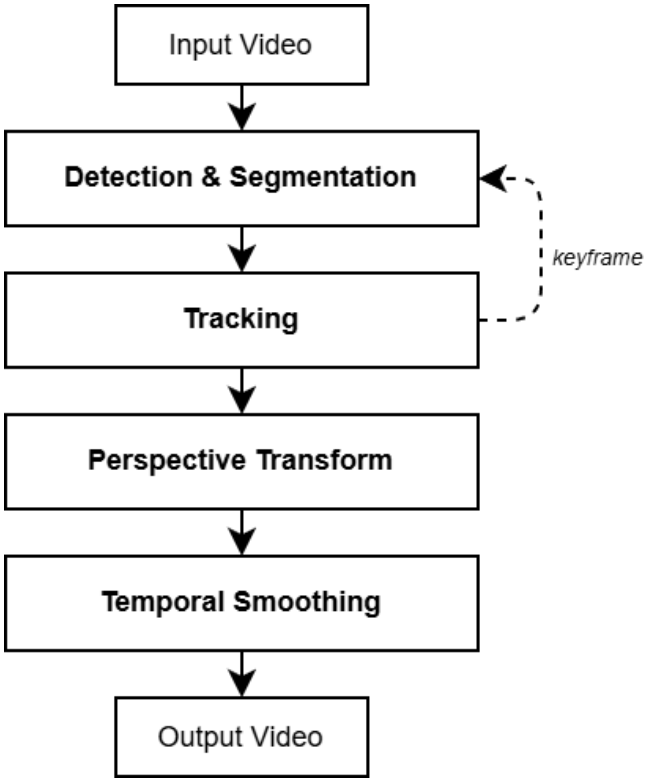

Fig. 1. Overview of the automated billboard replacement pipeline showing the four processing stages.

b) Zero-Shot YOLO-World: For scenarios without domain-specific training data, YOLO-Worldv2 is employed for open-vocabulary detection. The text prompt "billboard" guides the model to detect relevant objects without explicit training on billboard data. While zero-shot approaches offer deployment flexibility, experiments show accuracy gaps compared to fine-tuned models (see Section IV).

2) Segmentation Approaches:

a) Fine-Tuned YOLOv8 Segmentation: YOLOv8-seg extends the detection architecture with a segmentation head that produces pixel-level instance masks. These variants (nano, small, medium) were fine-tuned on a dedicated segmentation dataset of 485 images with precise polygon annotations. The segmentation output provides pixel-wise masks directly suitable for replacement. The loss function combines classification, bounding box regression, and mask prediction terms:

$$\mathcal{L}_{seg} = \mathcal{L}_{cls} + \lambda_{box}\mathcal{L}_{box} + \lambda_{mask}\mathcal{L}_{mask} \tag{2}$$

b) Mask R-CNN: As a two-stage alternative, Mask R-CNN first generates region proposals using a Region Proposal Network (RPN), then performs classification, bounding box refinement, and mask prediction for each proposal. This architecture typically produces higher-quality mask boundaries at the cost of increased computational overhead.

3) Hybrid Detection + SAM2 Segmentation: A hybrid pipeline combines the detection accuracy of fine-tuned

YOLOv8 models with SAM2's (Segment Anything Model 2) powerful zero-shot segmentation capabilities. In this approach:

1) **Detection Phase:** YOLOv8-det predicts bounding boxes for billboard regions with high precision.
2) **Prompt Generation:** Detected bounding boxes are converted to SAM2 prompts (box prompts or center point prompts).
3) **SAM2 Segmentation:** SAM2's image encoder processes the frame, and the mask decoder generates precise segmentation masks conditioned on the detection prompts.

SAM2 employs a Vision Transformer (ViT) backbone for image encoding and a lightweight mask decoder with cross-attention mechanisms. The model supports multiple prompting strategies:

- **Box Prompts:** Detected bounding boxes directly guide mask generation.
- **Point Prompts:** Center points of detections provide coarse object location.
- **Mask Prompts:** Previous frame masks can refine current predictions for video applications.

This hybrid approach leverages SAM2's ability to produce high-quality masks from minimal prompts while benefiting from domain-specific detection training. Importantly, SAM2's video object segmentation mode enables mask propagation across frames using a memory attention mechanism, which is utilized in the tracking stage (Section III-B) for temporal consistency.

B. Stage 2: Tracking

Temporal consistency is critical for video applications. Rather than processing each frame independently, the initial segmentation mask is propagated through subsequent frames using tracking algorithms. The system implements an adaptive detection-tracking balance:

$$\text{Mode}_t = \begin{cases} \text{Detect} & C_t < \tau \text{ or } t \bmod K = 0 \\ \text{Track} & \text{otherwise} \end{cases} \tag{3}$$

where C_t is the tracking confidence at frame t, τ_{low} is the confidence threshold (0.5), and I_{key} is the keyframe interval (30 frames).

1) Tracker Implementations: Six tracking approaches are evaluated:

- **SAM2Tracker:** Combines SAM2's promptable segmentation with a custom temporal history mechanism. It utilizes multiple prompting strategies, including bounding boxes and point sampling from previous masks, while incorporating an optical flow-based motion prediction fallback to maintain mask continuity during detection failures.
- **HybridFlowTracker:** Combines dense optical flow with feature-based matching for robust correspondence.
- **PlanarKalmanTracker:** Models billboard corners as a 16-dimensional state vector $\mathbf{x} = [x_1, y_1, ..., x_4, y_4, \dot{x}_1, \dot{y}_1, ..., \dot{x}_4, \dot{y}_4]^T$ with constant velocity dynamics.
- **FeatureHomographyTracker:** SIFT/ORB feature matching with RANSAC-based homography estimation.
- **ECCHomographyTracker:** Enhanced Correlation Coefficient maximization for sub-pixel registration accuracy.
- **AdaptiveOpticalFlowTracker:** Lucas-Kanade optical flow with forward-backward error checking for point validation.

2) Sanity Checking System: Each tracked mask undergoes multi-level validation before acceptance:

1) **Geometric checks:** Convexity (solidity > 0.7), aspect ratio bounds
2) **Temporal checks:** Area change $< 50\%$ per frame, center movement threshold
3) **Consistency checks:** Similarity with recent mask history using IoU

C. Stage 3: Replacement

Given a validated segmentation mask, we extract the billboard quadrilateral and warp the replacement image using perspective transformation.

1) Multi-Strategy Corner Extraction: Adaptive corner detection is employed:

1) **Polygon approximation:** Using Douglas-Peucker algorithm with adaptive ϵ
2) **Minimum area rectangle:** Fallback for non-polygonal masks
3) **Extreme point detection:** Project convex hull points in 8 directions, cluster to 4 corners
4) **Sub-pixel refinement:** Apply `cornerSubPix` for precise localization

2) Perspective Transformation: The homography matrix H is computed from source corners (replacement image) to destination corners (billboard region):

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \tag{4}$$

3) Advanced Blending Techniques: Three blending strategies are implemented:

Distance-Transform Feathering: Creates smooth alpha masks based on distance from edges:

$$\alpha(p) = \min\left(1, \frac{d(p)}{w_{feather}}\right) \tag{5}$$

where $d(p)$ is the distance from pixel p to the mask boundary.

Multi-Resolution Pyramid Blending: Blends Laplacian pyramids at each resolution level:

$$L^k_{blend} = G^k_{mask} \cdot L^k_{src} + (1 - G^k_{mask}) \cdot L^k_{tgt} \tag{6}$$

Color Harmonization: Adjusts replacement content to match scene lighting:

$$I_{adjusted} = \frac{\sigma_{target}}{\sigma_{source}}(I - \mu_{source}) + \mu_{target} \tag{7}$$

computed in LAB color space for perceptual accuracy.

D. Stage 4: Post-Processing

1) Enhanced Kalman Smoothing: An enhanced 16-dimensional Kalman filter with constant velocity model is used. The state vector is ordered as $\mathbf{x} = [x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4, \dot{x}_1, \dot{y}_1, ..., \dot{x}_4, \dot{y}_4]^T$, where the first 8 elements represent the (x, y) positions of the 4 billboard corners and the last 8 elements represent their velocities:

$$\mathbf{x}_t = \begin{bmatrix} I_8 & \Delta t \cdot I_8 \\ 0 & I_8 \end{bmatrix} \mathbf{x}_{t-1} + \mathbf{w} \tag{8}$$

$$\mathbf{z}_t = \begin{bmatrix} I_8 & 0 \end{bmatrix} \mathbf{x}_t + \mathbf{v} \tag{9}$$

where I_8 is the 8×8 identity matrix, and $\Delta t = 1$ for consecutive frames. The filter includes adaptive noise scaling based on innovation magnitude and automatic reset on large displacements (> 200 pixels).

2) IIR Multi-Stage Smoothing: As an alternative, cascaded Infinite Impulse Response (IIR) filtering is applied:

$$\hat{p}_t = \alpha \cdot p_t + (1 - \alpha) \cdot \hat{p}_{t-1} \tag{10}$$

with $\alpha = 0.3$ applied across 2-3 stages for aggressive smoothing without excessive lag.

E. Datasets and Experimental Setup

1) Detection Dataset: A large-scale detection dataset was constructed comprising 3,798 images sourced from multiple public repositories (Roboflow). The dataset is split into 3,038 training, 549 validation, and 211 testing images. It captures a wide variety of billboard types, sizes, and environmental conditions. To enhance model robustness, extensive data augmentation was applied during training, including HSV color-space transformations (Hue: ± 0.015, Saturation: ± 0.7, Value: ± 0.4), geometric distortions (rotation $\pm 15°$, shear $\pm 5°$, perspective 0.001), and advanced regularization techniques such as Mosaic ($p = 1.0$), MixUp ($p = 0.15$), and Random Erasing ($p = 0.4$).

2) Segmentation Dataset: For segmentation tasks, a dedicated dataset of 485 images annotated with precise polygon masks was utilized. This dataset combines samples from billboard video frames and sports broadcasts to represent complex real-world scenarios. Similar augmentation strategies were employed, with the addition of Copy-Paste augmentation ($p = 0.3$) to improve instance segmentation performance.

3) Video Test Set: Four real-world videos were used for testing purposes, each exhibiting distinct characteristics. The first video contains a stationary billboard with dynamic background changes. The second video features a billboard occupying the majority of the frame with slow camera movement, as shown in Figure 2. The third video depicts a zoom-in sequence on a distant billboard, testing detection performance across varying scales. The fourth video contains a digital LED billboard with constantly changing content (Figure 3), presenting additional challenges for tracking algorithms that rely on appearance consistency across frames. These diverse scenarios facilitated a detailed performance comparison of both the detection models and tracking algorithms.

IV. Experiments and Results

A. Detection Results

Table I presents detection performance across model variants. YOLOv8m achieves the best AP50 of 0.736, outperforming both smaller variants and zero-shot YOLO-Worldv2.

TABLE I
Billboard Object Detection Performance

Model	P	R	AP50	AP50-95
YOLOv8n	0.752	0.643	0.723	0.473
YOLOv8s	0.787	0.612	0.731	0.479
YOLOv8m	0.818	0.600	**0.736**	**0.480**
YOLO-Worldv2 (Zero-shot)	0.330	0.690	0.598	0.408

The zero-shot YOLO-Worldv2 achieves higher recall (0.690) but significantly lower precision (0.330), resulting in many false positives. Fine-tuned YOLOv8m outperforms zero-shot by +0.138 AP50.

1) Key Findings in Detection:

- **Impact of Fine-tuning:** Fine-tuned models significantly outperformed zero-shot YOLO-Worldv2, with YOLOv8m achieving a **+0.138 higher AP50**, validating the necessity of domain-specific training for billboard localization.
- **Precision-Recall Trade-off:** While YOLO-Worldv2 exhibited the highest recall (**0.690**), its low precision (**0.330**) resulted in excessive false positives; whereas YOLOv8m maintained a superior balance with **0.818 precision**.
- **Scalability:** Detection performance scaled with model size, where YOLOv8m provided the best overall accuracy (**0.736 AP50**), suggesting that detection is less sensitive to overfitting than pixel-level segmentation on this dataset.

B. Segmentation Results

Table II compares segmentation approaches. Fine-tuned YOLOv8n-seg achieves the best performance with 0.672 AP50.

TABLE II
Billboard Segmentation Performance

Model	P	R	AP50	AP50-95
YOLOv8n-seg	0.786	**0.684**	**0.672**	0.542
YOLOv8s-seg	0.819	0.512	0.586	0.442
YOLOv8m-seg	0.693	0.605	0.636	0.462
Mask R-CNN	**0.867**	0.605	0.626	**0.531**
YOLOv8+SAM2	0.407	0.512	0.360	0.284

1) Key Findings in Segmentation:

- **Optimal Architecture:** YOLOv8n-seg achieves the highest AP50 (**0.672**) with the best recall (**0.684**), demonstrating that lightweight models generalize effectively on domain-specific datasets with limited samples (485 images).
- **Mask R-CNN Performance:** After proper AP calculation using PR curves, Mask R-CNN shows strong

results with the highest precision (**0.867**) and competitive AP50-95 (**0.531**). This indicates that two-stage detectors can achieve high-quality mask boundaries when properly trained.

- **Boundary and Mask Fidelity:** YOLOv8n-seg leads in detection-weighted accuracy (AP50: 0.672), while Mask R-CNN excels in stricter IoU thresholds (AP50-95: 0.531), suggesting complementary strengths for different application requirements.
- **YOLOv8+SAM2 Pipeline:** The detection+segmentation pipeline shows lower benchmark scores (AP50: 0.360) due to error propagation from detection to segmentation. However, qualitative testing reveals that SAM2 produces high-quality masks when provided with accurate bounding boxes, making it valuable for video applications where per-frame detection is feasible.

C. Tracking Results

Table III presents tracking algorithm performance across the video test set. Evaluation metrics include:

- **Mean IoU:** Intersection over Union between tracked and reference masks.
- **Success@0.5:** Percentage of frames with IoU > 0.5.
- **Failures:** Frames with IoU < 0.5 indicating significant tracking degradation.
- **FPS:** Processing speed benchmarked on an NVIDIA RTX 4070 12GB GPU with AMD Ryzen 5 5600X 6-core processor.

$$\text{IoU} = \frac{|M_{pred} \cap M_{gt}|}{|M_{pred} \cup M_{gt}|} \tag{11}$$

TABLE III
TRACKING ALGORITHM PERFORMANCE (YOLOV8M-DET + SAM2, 379 FRAMES)

Tracker	Mean IoU	S@0.5	S@0.7	Fail	FPS
PlanarKalman	**0.915**	**99.7%**	**96.5%**	**1**	**283.7**
FeatureHomo	0.907	96.2%	92.7%	14	2.9
HybridFlow	0.906	97.6%	88.6%	9	35.7
SAM2	0.859	96.2%	91.3%	14	8.6
ECCHomo	0.684	79.8%	38.0%	76	1.9
OpticalFlow	0.577	63.1%	59.6%	143	32.8

1) Key Findings in Tracking:

- **Best Overall:** PlanarKalmanTracker achieves the highest mean IoU (**0.915**) while simultaneously delivering the fastest processing speed (**283.7 FPS**), with only 1 tracking failure across 379 frames. This makes it ideal for real-time billboard replacement applications.
- **Feature-Based Accuracy:** FeatureHomographyTracker (SIFT-based) achieves 0.907 mean IoU with excellent boundary precision (Median IoU: 0.953), though at significantly lower speed (2.9 FPS) due to computational overhead of feature extraction and matching.
- **Speed-Accuracy Balance:** HybridFlowTracker provides an excellent trade-off with 0.906 mean IoU at 35.7 FPS, combining dense optical flow with feature correspondence for robust tracking across camera motion.
- **Neural Network Tracking:** SAM2Tracker maintains consistent performance (0.859 mean IoU) using memory-based attention mechanisms, with moderate computational cost (8.6 FPS). Its uniform behavior across diverse video conditions makes it suitable when consistency is prioritized over peak accuracy.
- **Classical Methods:** ECCHomography and AdaptiveOpticalFlow showed degraded performance on videos with significant motion or texture-poor billboard surfaces, highlighting the need for hybrid or learning-based approaches in challenging conditions.

D. Qualitative Results

Figure 2 shows example replacement results. The pipeline successfully handles perspective changes, maintains temporal consistency, and produces visually coherent output.

Fig. 2. Comparison of the original video frame (top) and the automated billboard replacement result (bottom).

V. DISCUSSION AND FUTURE WORK

A. Fine-Tuned vs Zero-Shot Detection

Experiments demonstrate that fine-tuned models significantly outperform zero-shot approaches for domain-specific billboard detection. YOLOv8m achieves 0.736 AP50 compared to 0.598 for YOLO-Worldv2, a +0.138 improvement. However, zero-shot models maintain value for rapid prototyping or scenarios where training data is unavailable.

Fig. 3. Additional replacement example demonstrating the pipeline's robustness across different viewing angles and lighting conditions.

B. Tracking vs Per-Frame Segmentation

While the tracking benchmark results (Table III) demonstrate that algorithms such as PlanarKalmanTracker achieve high IoU scores (0.915) with excellent processing speeds (283.7 FPS), qualitative evaluation on the test video sequences revealed an important insight: performing detection and segmentation on every frame consistently produced superior visual quality compared to tracking-based mask propagation.

The primary reasons for this observation include:

- **Drift Accumulation:** Tracking algorithms, even with periodic keyframe re-detection, accumulate small errors over time that manifest as jittery or misaligned mask boundaries.
- **Motion Robustness:** Per-frame detection+segmentation handles rapid camera motion and viewpoint changes more gracefully, as each frame is processed independently without relying on temporal correspondence.
- **SAM2 Performance Gap:** Interestingly, while SAM2 showed relatively lower performance in the static segmentation benchmark (AP50: 0.307 in Table II), it demonstrated significantly better results when used in the detect+segment pipeline with YOLOv8m-provided bounding boxes. This suggests that SAM2's strength lies in its prompt-based segmentation capability rather than standalone detection, making the YOLOv8+SAM2 combination particularly effective for video applications.

These findings suggest that for applications prioritizing visual quality over processing speed, per-frame detection+segmentation may be preferable despite the computational overhead. Hybrid approaches that use tracking for intermediate frames while performing full segmentation at regular intervals offer a practical compromise.

C. Current Limitations

Several limitations affect pipeline performance:

- **Quadrilateral Approximation:** Our replacement method uses 4-corner homography, which assumes billboards are rectangular. Non-rectangular or curved billboards require mask-based image warping (e.g., thin-plate spline) to properly fit replacement content to arbitrary mask shapes.
- **Extreme Viewing Angles:** Billboards viewed at acute angles suffer from resolution loss after perspective transformation.
- **Occlusion Handling:** Partial occlusions (e.g., pedestrians, vehicles) cause tracking failures and visual artifacts.
- **Dynamic Content:** LED billboards with changing content confuse tracking algorithms that assume static appearance.
- **Computational Cost:** Transformer-based segmentation models (SAM2) require GPU acceleration with adequate VRAM (12GB+ recommended).

D. Future Improvements

Several directions could enhance pipeline capabilities:

1) Diffusion-Based Image Replacement: The current implementation uses OpenCV's perspective transformation and alpha blending for content replacement. While effective for static image overlays, this approach has limitations such as hard edges at mask boundaries and lighting inconsistencies. Future versions could leverage diffusion models like Stable Diffusion for inpainting, enabling seamless blending with scene context. Text-to-image capabilities would allow generating novel billboard content from prompts without pre-existing assets. The main trade-off is computational cost, as diffusion-based methods require significantly more processing time compared to geometric transformation.

2) Multi-Billboard Tracking: The current pipeline processes a single billboard per video. Extending to simultaneous tracking of multiple billboards would require association algorithms such as SORT or DeepSORT to maintain identity across frames and handle billboard entries/exits from the scene.

3) Occlusion-Aware Replacement: Partial occlusions by pedestrians or vehicles currently cause visual artifacts in the replaced content. Leveraging depth estimation models (MiDaS, ZoeDepth) or instance segmentation could enable proper layering of occluding objects over the replaced billboard content, significantly improving visual realism in complex scenes.

VI. Conclusion

This paper presented an automated pipeline for billboard detection, tracking, and replacement in video content. Comprehensive evaluation demonstrates that fine-tuned YOLOv8

models significantly outperform zero-shot approaches for domain-specific detection (0.736 vs. 0.598 AP50). For tracking, PlanarKalmanTracker achieves both the highest accuracy (0.915 mean IoU) and fastest processing speed (283.7 FPS), with only 1 tracking failure across 379 frames, making it ideal for real-time billboard replacement applications.

The integrated pipeline successfully automates a traditionally manual and expensive post-production task, enabling scalable content localization and dynamic advertising applications. Future work will focus on handling occlusions, improving boundary blending through generative models, and optimizing for real-time deployment.

REFERENCES

[1] D. Bhargavi, K. Sindwani, and S. Gholami, "Zero-shot virtual product placement in videos," in *ACM IMX*, 2023.

[2] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[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] H. K. Cheng, S. W. Oh, B. Price, J.-Y. Lee, and A. Schwing, "Putting the object back into video object segmentation," 2024.

[5] H. K. Cheng and A. G. Schwing, "XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model," in *ECCV*, 2022.

[6] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLO," Jan. 2023.

[7] A. Kirillov et al., "Segment Anything," in *ICCV*, 2023.

[8] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, abs/1506.02640, 2015.

[9] R. Rombach et al., "High-Resolution Image Synthesis with Latent Diffusion Models," in *CVPR*, pp. 10684–10695, 2022.

[10] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "SAM 2: Segment Anything in Images and Videos," arXiv preprint arXiv:2408.00714, 2024.

[11] A. Sharma, M. Weiss, P. Navarathna, A. Mahdavi-Amiri, and Y. Aksoy, "Automated Virtual Product Placement and Assessment in Images using Diffusion Models," arXiv preprint arXiv:2407.01117, 2024.

[12] J. Yang et al., "Track Anything: Segment Anything Meets Videos," arXiv preprint arXiv:2304.11968, 2023.

[13] H. Ye, J. Zhang, S. Liu, X. Han, and W. Yang, "IP-Adapter: Text Compatible Image Prompt Adapter for Text-to-Image Diffusion Models," 2023.