FMDB *Transactions on Sustainable Computing Systems*

From Heuristics to Transformers: A Practical Comparison of UAV Safe Landing Zone Detection Methods with Task-Specific Evaluation Metrics

Abhay Valiyaparambil[1,*], D. Hemavathi[2], Sayyed Khawar Abbas[3]

[1,2]Department of Data Science and Business Systems, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu, India.
[3]Department of Information Systems, Corvinus University of Budapest, Budapest, Hungary.
as3735@srmist.edu.in[1], hemavatd@srmist.edu.in[2], sayyedkhawar.abbas@uni-corvinus.hu[3]

Abstract: Safe landing zone detection from a single downward-facing camera is a practical necessity for autonomous UAVs operating without GPS or prior maps. Existing approaches range from lightweight, training-free heuristics to large, fine-tuned deep networks, yet no study compares them under evaluation criteria relevant to the actual landing task. This paper presents a five-way comparison: A Laplacian flatness detector, an HSV color-consistency heuristic, our proposed Multi-Cue Saliency Safety Score (MC-SSS), SAM zero-shot region scoring, and SegFormer-B0 fine-tuned across three data regimes. Researchers also introduce Landing Zone Accuracy (LZA) and Largest Safe Region Ratio (LSRR), two metrics that assess whether a prediction results in a successful landing rather than simply overlapping with a ground-truth mask. Experiments on 3,939 labeled images from Aeroscapes and UAVid show that color-only methods lose up to 37.2 percentage points of accuracy when terrain changes from rural to urban. At the same time, MC-SSS holds the drop to 18.1 points. SAM, applied without any fine-tuning, achieves LZA below 38% on both benchmarks: it segments scenes correctly but selects the wrong region. SegFormer trained on data from both datasets reaches 98.0% and 78.4% LZA on Aeroscapes and UAVid, respectively. The results give practitioners a concrete, evidence-based basis for choosing a method under different operational constraints.

Keywords: Unmanned Aerial Vehicle (UAV); Semantic Segmentation; Landing Zone Accuracy (LZA); Segment Anything Model; Domain Generalization; Safe Landing Zone; Laplacian Edge Variance.

Received on: 27/05/2025, **Revised on:** 22/07/2025, **Accepted on:** 21/10/2025, **Published on:** 10/08/2026

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSCS

DOI: https://doi.org/10.69888/FTSCS.2026.000719

Cite as: A. Valiyaparambil, D. Hemavathi, and S. K. Abbas, "From Heuristics to Transformers: A Practical Comparison of UAV Safe Landing Zone Detection Methods with Task-Specific Evaluation Metrics," *FMDB Transactions on Sustainable Computing Systems*, vol. 4, no. 3, pp. 165–173, 2026.

1. Introduction

A drone with a low battery and no map must still land somewhere safe. That is the problem researchers study. Given only a single RGB image from a camera pointed at the ground, the system must decide where it is safe to touch down, with no GPS, no prior survey of the area, and no human in the loop. This sounds like a tractable vision problem, and in some settings it is. On rural terrain, grass and open soil look nothing like rooftops and tree canopies. Simple color checks or edge analysis can reliably separate safe from hazardous ground. The difficulty appears when the same system is asked to work in a city. Road

[*]Corresponding author.

surfaces and rooftops share similar colors at typical drone altitudes. The visual gap between safe and unsafe shrinks considerably, and methods that worked in open countryside fail. The field has developed three broad strategies for addressing this. Training- free heuristics require no labeled data and run in milliseconds, but each one encodes a narrow assumption about what safe terrain looks like. Foundation model scoring, as in Kirillov et al. [8], generates region proposals from any image without task-specific training, though it remains unclear whether this translates into useful landing guidance. Supervised segmentation networks achieve the highest accuracy but require annotated UAV data and careful domain matching between training and deployment conditions. No published study has compared all three strategies under the same experimental conditions, using the same datasets and metrics. This gap matters because practitioners making deployment decisions need to know what each approach actually delivers, not what it delivers on its own curated test set. The evaluation gap is also a problem. Standard segmentation metrics, such as IoU, measure pixel overlap, which does not directly tell you whether the predicted landing centroid falls inside a safe region or whether the predicted safe area forms a single coherent pad. A drone cannot land on 200 scattered green pixels [12]. This paper contributes three things:

- MC-SSS, a training-free method that combines LBP texture uniformity, Laplacian edge variance, and HSV color consistency into a single safety map per image, reducing cross-terrain performance loss by roughly half compared to color-only approaches.
- LZA and LSRR are two evaluation metrics that measure whether a predicted landing point is actually safe (LZA) and whether the prediction is a single usable region rather than fragments (LSRR).
- The first controlled five-way benchmark across Aeroscapes and UAVid, covering training-free heuristics, a zero-shot foundation model, and fine-tuned transformers under consistent conditions.

2. Related Work

2.1. Training-Free Heuristic Approaches

Keipour et al. [1] showed that HSV color segmentation works well for finding landable zones in rural fixed-wing footage. The method performs adequately when safe terrain has a clearly different color from hazardous terrain, which is often true in open countryside but rarely in cities. Xin et al. [2] took a different approach, using a Laplacian-of-Gaussian edge density to identify flat surfaces, on the basis that flat ground produces fewer strong edges than trees or buildings. The method is sensitive to lighting and struggles to distinguish a flat rooftop from flat soil. Sanchez-Lopez et al. [3] proposed a multi-channel approach combining color histograms with gradient maps, reporting improvements over single-cue baselines on a proprietary dataset. Neither paper quantified how performance changes when the test terrain differs from the training assumptions, which is the comparison this work is built around.

2.2. Deep Learning for UAV Segmentation

Ma et al. [5] applied DeepLab-v3+ to UAV images for obstacle detection, reporting good in-domain IoU but significant drops when tested on data from a different source. Wang et al. [6] fine-tuned the U-Net on UAVid and achieved an F1 score above 0.85 on the same dataset, despite cross-terrain degradation. Xie et al. [4] improved on these results with a hierarchical Mix Transformer encoder that processes images at four spatial scales, paired with a simple MLP decoder. This multi-scale representation better handles the range of feature sizes in aerial images than single-scale approaches, which is why it serves as the supervised backbone in this study. An earlier transformer approach used a similar idea but at considerably higher computational cost [7].

2.3. Foundation Models in Remote Sensing

Kirillov et al. [8] trained on over a billion segmentation masks and can partition any image into coherent regions without task-specific prompting. Chen et al. [9] tested SAM on remote sensing tasks, including road and building segmentation. Their finding was that SAM's boundaries are geometrically accurate, but its outputs lack the semantic meaning needed for task-specific decisions. Houichime and El Amrani [10] attempted to add domain-specific prompts for infrastructure inspection and observed some improvement over unprompted SAM, though their performance did not match that of supervised methods. Ravi et al. [11] extended SAM to video in SAM 2, with better mask quality but the same fundamental limitation. No prior study has specifically tested SAM for UAV landing zone detection.

2.4. The Evaluation Gap

IoU and F1 are the standard metrics for segmentation. Still, they do not directly answer the question a drone operator cares about: will this prediction guide the vehicle to a safe landing spot? A method can achieve high IoU while placing its predicted

centroid in the middle of a building. Conversely, a prediction that is slightly over-segmented at the edges but correctly centered on safe ground will be penalized by IoU. LZA and LSRR were designed to close this gap.

3. Methodology

3.1. Problem Formulation

Given a UAV image $\mathbf{I} \in \mathbf{R}^{H \times W \times 3}$, the task is to produce a binary mask $\mathbf{M} \in \{0, 1\}^{H \times W}$ where $\mathbf{M(i, j) = 1}$ means the pixel belongs to safe terrain (road, low vegetation, bare soil) and $\mathbf{M(i, j) = 0}$ means it does not (building, tree, vehicle, sky). From this mask, a landing centroid $c = (c_x, c_y)$ is computed as the geometric center of the largest connected safe region. A prediction is considered operationally successful when c falls within the ground-truth safe area.

3.2. Training-Free Baselines

- **Laplacian-only Detector:** The Laplacian L(I) = ∂2I/∂x2+∂2I/∂y2 produces strong responses at edges and near-zero values over flat homogeneous surfaces. Patch variance of the Laplacian is computed across a grid; patches below the mean (0.5×std) are labeled safe.
- **HSV Color Consistency Heuristic:** Hue and Saturation channel histogram variance is computed per patch and inverted through a sigmoid to produce a safety score. High color consistency within a patch suggests one terrain type and probable flatness. The Value channel is left out because shadow can lower brightness even on perfectly safe ground.

3.3. MC-SSS: Multi-Cue Saliency Safety Score

The idea behind MC-SSS is that any single visual cue for surface flatness has a failure mode on some terrain type. Color fails in cities. Edge variance fails on flat rooftops. Combining them reduces the failure rate. The image is divided into a 16 × 16 grid of cells, and each cell receives three scores:

- **LBP Texture Score SLBP:** Local Binary Patterns compare each pixel to its 8 circular neighbors at radius 1, producing an 8-bit code per pixel. Codes with at most 2-bit transitions are called uniform, and they appear much more often on flat, smooth surfaces than on rough or structured ones. SLBP is the fraction of uniform codes in each cell. Higher means smoother.
- **Laplacian Score SLAP:** The Laplacian variance within each cell is normalized per image and then inverted, so flat, low-edge cells get high scores regardless of overall scene contrast.
- **HSV Score SHSV:** H and S channel histogram variance within each cell is inverted and normalized. Consistent color within a cell indicates one terrain type. The V channel is excluded for the same reason as in the baseline.

3.3.1. Score Fusion and Mask Generation:

The composite score for cell k is:

$$S(k) = 0.40\, S_{\mathrm{LBP}}(k) + 0.35\, S_{\mathrm{Lap}}(k) + 0.25\, S_{\mathrm{HSV}}(k) \quad (1)$$

LBP texture gets the most weight because it is the least terrain-specific: a flat surface in a city and a flat surface in a field both produce uniform LBP codes, even if their colors and edge profiles differ. HSV gets the least weight because it is the most terrain-specific. The cell score map is upsampled to full resolution, thresholded at mean − 0.2× std, and cleaned with a 25×25 morphological closing kernel to remove fragments too small for a drone to land on.

3.4. SAM Zero-Shot Region Scoring

Kirillov et al. [8] is run in automatic mode with a 16×16 prompt grid and a stability threshold of 0.92. This produces between 50 and 200 region proposals per image. Each region is scored on three geometric criteria: area within 1–60% of the image (excluding tiny fragments and sky-level regions), mean Laplacian variance (lower values indicate flatter regions), and compactness (4π · area/perimeter2). The top-scoring region is the predicted landing zone. SAM's weights are not changed [15].

3.5. SegFormer Fine-Tuning

- **Architecture:** SegFormer-B0 uses a MiT-B0 encoder that produces feature maps at 1/4, 1/8, 1/16, and 1/32 of the input resolution, followed by an all-MLP decoder [4]. The design follows the broader shift toward patch-based

transformer encoders for vision tasks [16]. The pretrained 150-class ADE20K head is replaced with a 2-class (Safe/Hazard) linear layer. Researchers initialize from ADE20K segmentation weights rather than ImageNet classification weights because segmentation pretraining already teaches the encoder to produce pixel-level features rather than just image-level class scores.

- **Training:** AdamW with lr = 6 × 10−5 and weight decay 0.01, cosine annealing, 50 epochs, batch size 4, input 384 × 384. Augmentation includes horizontal and vertical flips, rotation (±30◦), brightness and contrast jitter, and a random hue shift of ±20◦ in HSV space. The hue shift is the most consequential augmentation: it forces the network to classify terrain without relying on color, which is exactly the cue that fails when moving from rural to urban environments.
- **Data Configurations:** Three setups are compared: Cross-domain (trained on Aeroscapes, tested on UAVid) to simulate deployment with no target-domain labels; in-domain (trained and tested within UAVid, 80/20 split) as an upper bound; and combined (Aeroscapes plus UAVid 80%, tested on UAVid 20%) to measure whether source-domain data helps.

3.6. Evaluation Metrics

3.6.1. Landing Zone Accuracy (LZA)

The centroid of the largest predicted safe region is extracted for each image and checked against the ground-truth mask:

$$\text{LZA} = \frac{1}{N}\sum_{i=1}^{N} M_{gt}\left(c_x^i, c_y^i\right) = 1 \quad (2)$$

A method that picks the right location but gets the edges slightly wrong still scores 100% LZA. This is by design: edge precision matters less than landing in the right place.

3.6.2. Largest Safe Region Ratio (LSRR)

$$LSRR = \frac{|largest\ safe\ component|}{|all\ predicted\ safe\ pixels|} \quad (3)$$

LSRR = 1.0 means that all predicted safe pixels form a single connected region. A low LSRR indicates the prediction is fragmented. A drone needs a single contiguous pad, so high LSRR combined with high LZA is the target.

3.7. Implementation and Experimental Setup

3.7.1. Datasets

Nigam et al. [13] contain 3,269 images from rural and peri-urban environments, captured at altitudes between 50 and 300 m, with 11 semantic classes. Pixels belonging to class 0 (background), 9 (road), and 10 (low vegetation) are labeled Safe; everything else is Hazard. Lyu et al. [14] provide 670 frames from dense urban areas in Wuhan, China, at altitudes of 20 to 60 m, with 8 classes. Road (128, 64, 128), background (0, 0, 0), and low vegetation (128, 128, 0) are Safe; all other classes are Hazard. UAVid is considerably harder: roads are narrow, buildings are dense, and the ratio of safe to hazardous pixels is lower than in Aeroscapes. All images are resized to 512×512 for storage. Splits are made along temporal sequence boundaries so that consecutive frames from the same scene cannot appear in both training and test sets.

3.7.2. Hardware and Software

Experiments were run on a single machine with an NVIDIA RTX 4060 Laptop GPU (8 GB), an Intel Core i7-13700H, and 32 GB RAM. Software: Python 3.13, PyTorch 2.6.0 with CUDA 12.4, HuggingFace Transformers for SegFormer, OpenCV 4.9 for preprocessing, scikit-image for LBP, and the official segment-anything package. Latency Figures are averages over the full test set after a 50-frame warm-up run.

4. Results and Discussion

4.1. Overall Comparison

Table 1 gives the full numbers. Figure 1 shows the same data visually. The sections below discuss the three main patterns in the results.

Table 1: Results on Aeroscapes and UAVid; LZA: Landing zone accuracy (%); LSRR: Largest safe region ratio; Bold: Best per column; †: Best overall; SF = segformer-b0; Aero N =499; UAVid N =134 test images

Method	Aero LZA	Aero F1	UAVid LZA	UAVid F1	LSRR	ms/f
Laplacian-only	73.7	0.396	70.7	0.514	0.68	9.4
HSV-only	98.2	0.980	61.0	0.553	0.99	1.8
MC-SSS (proposed)	74.1	0.475	56.0	0.501	0.81	70.3
SAM zero-shot	37.5	0.054	36.7	0.067	1.00	1294
SF cross-domain	—	—	65.7	0.661	0.96	34.9
SF in-domain	—	—	75.4	0.873	0.81	32.4
SF combined†	98.0	0.995	78.4	0.862	0.80	41.5

Figure 1 compares Landing Zone Accuracy (LZA) and Segmentation F1 Score of the four UAV safe landing zone identification algorithms on the Aeroscapes and UAVid datasets. Both measures reveal that the SegFormer mixed model performs better. HSV-only performs well on Aeroscapes but poorly on UAVid, highlighting the difficulty of generalizing color-based approaches across terrains.

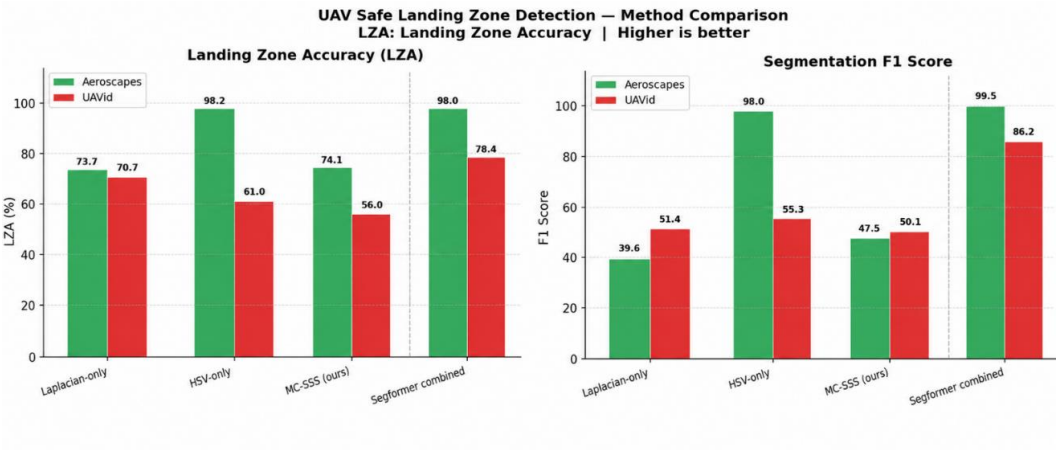

Figure 1: LZA and F1 across all methods on both datasets; the former trained on combined data leads in every column, SAM sits well below the other methods despite its strong segmentation reputation.

4.2. The Terrain Sensitivity Problem

HSV-only reaches 98.2% LZA on Aeroscapes. That number drops to 61.0% on UAVid, a loss of 37.2 percentage points. The reason is straightforward: in rural scenes, grass and soil have different colors from rooftops and roads. In UAVid, roads and rooftops are both grey. The scorer cannot distinguish between them by color alone, so it often picks the wrong one (Figure 2).

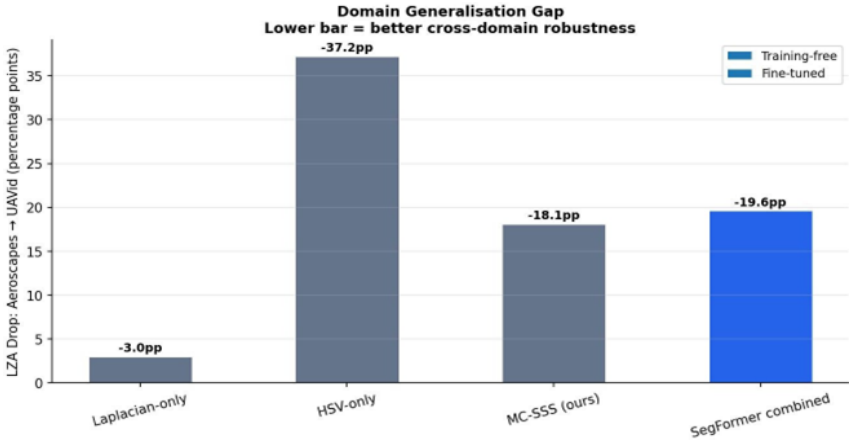

Figure 2: LZA drop from Aeroscapes to UAVid per method; HSV-only loses 37.2 points; MC-SSS loses 18.1; adding texture to the scoring roughly halves the terrain sensitivity.

MC-SSS holds the drop to 18.1 points by adding LBP texture to the score. Rooftop materials (tiles, gravel, tar paper) produce non-uniform LBP codes because their surface structure changes rapidly at the pixel scale. Asphalt roads produce more uniform codes (Table 2).

Table 2: Drop in LZA from Aeroscapes to UAVid; A smaller drop means the method holds up better across terrain types

Method	Aero LZA	UAVid LZA	Drop (pp)
Laplacian-only	73.7%	70.7%	−3.0
HSV-only	98.2%	61.0%	−37.2
MC-SSS (Proposed)	74.1%	56.0%	−18.1

This textural difference partially compensates for the color similarity, as the method-by-method distribution of UAVid F1 versus LZA in Figure 3 makes clear.

4.3. Why SAM Fails?

SAM achieves 37.5% LZA on Aeroscapes and 36.7% on UAVid. On a dataset where roughly half the pixels are safe, random centroid placement would score around 50%. SAM does worse than random. At the same time, LSRR is 1.00 on UAVid and 0.999 on Aeroscapes. The scorer picks a single, large, coherent region each time. The problem is not the segmentation; it is the scoring.

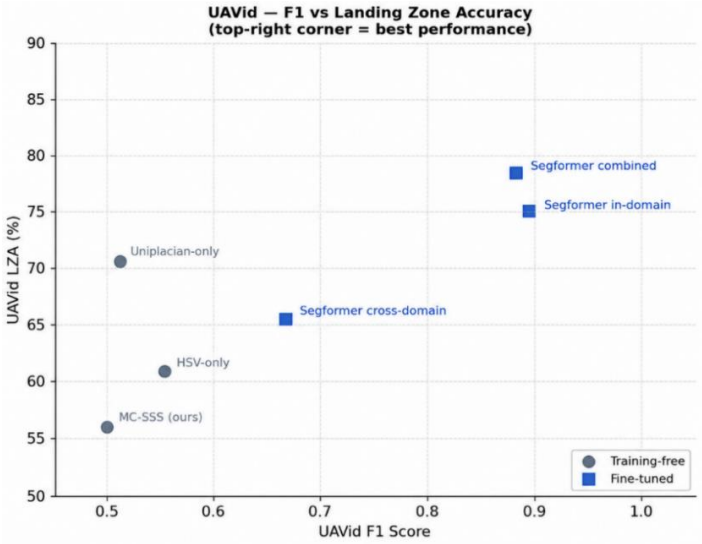

Figure 3: UAVid F1 against LZA for each method; the three SegFormer configurations sit in the upper right, the three training-free methods sit in the lower left; SAM is the outlier at bottom left: High LSRR but LZA below chance.

Figure 4 illustrates what goes wrong. SAM correctly draws separate masks for roads, rooftops, and trees. The geometric scorer then ranks them by area, flatness (Laplacian variance), and compactness. A rooftop is large, has low Laplacian variance because its surface is uniform, and is often more compact than a winding road. It scores higher than the road and gets selected. The result is a drone being directed at a building. The fix is semantic, not geometric. A classifier that knows "road = safe, rooftop = hazard" at the region level would work. Training such a classifier on a few hundred annotated SAM regions would cost far less than building a full pixel-level segmentation dataset.

4.4. SegFormer: What Training Data Does

Cross-domain SegFormer (trained on Aeroscapes, tested on UAVid) reaches 65.7% LZA with zero UAVid training examples. That already beats every training- free method on UAVid. The model learned something about flat surfaces from rural imagery that transfers to urban roads. Adding UAVid training data (combined configuration) brings this to 78.4%, a 12.7-point increase. Interestingly, the combined model slightly outperforms the in-domain model (75.4%), even though the in-domain model saw proportionally more UAVid examples. The Aeroscapes images appear to act as regularisation: the network cannot overfit to the specific appearance of UAVid roads while also handling rural vegetation and soil. The hue-shift augmentation reinforces this by making color an unreliable cue during training, forcing the network to rely on texture and geometry.

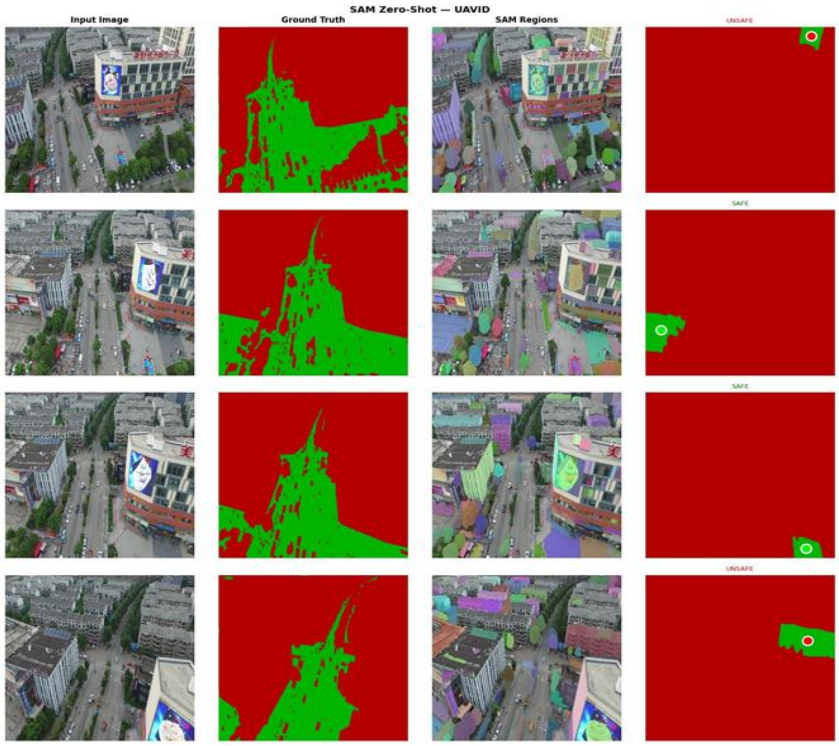

Figure 4: SAM qualitative results on UAVid. The region proposals are geometrically accurate: Roads, buildings, and vegetation are cleanly separated. The scorer then picks a rooftop over a road because it is larger, flatter, and more compact by geometric criteria.

On Aeroscapes, the combined SegFormer achieves F1 = 0.995 and LSRR = 1.000: every safe pixel is in a single connected region, and the centroid is correctly located in 98% of images.

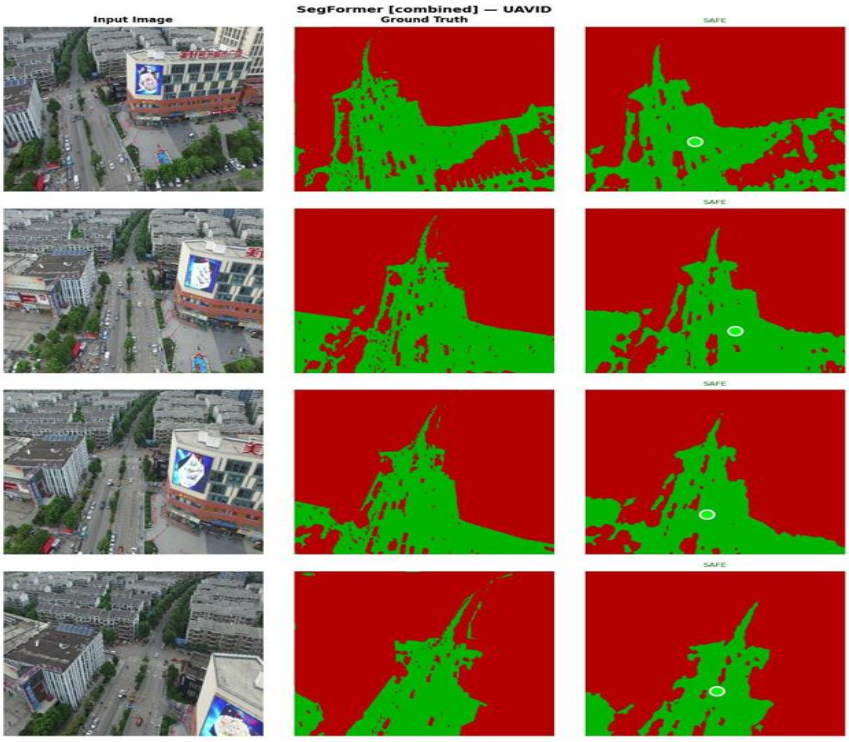

Figure 5: SegFormer combined results on UAVid; the predicted centroid (White Circle) lands inside a ground-truth safe region in 78.4% of test images, and the model reliably picks roads over rooftops in dense urban scenes.

Latency is 41.5 ms per frame on the GPU used, which is adequate for real-time descent control. Deployment on embedded hardware (Jetson class) would require quantization. Qualitative predictions on UAVid are shown in Figure 5.

4.5. Choosing a Method in Practice

The results point to a clear decision tree. If no training data is available and the terrain is consistent (all rural or all urban), HSV-only is hard to beat: 98% LZA at under 2 ms. If the terrain varies, MC-SSS halves the performance loss due to domain changes without requiring additional data. If labeled UAVid-style data and a GPU are available, SegFormer combined gives the best numbers across both datasets. SAM zero-shot, used as a standalone detector with geometric scoring, should be avoided until a semantic region classifier is added.

5. Conclusion

Five methods for UAV safe landing zone detection were compared on two datasets under two task-specific metrics. The main findings are as follows. Color-only scoring is highly effective on rural terrain (98.2% LZA) and falls apart on urban terrain (61.0%). Adding texture through MC-SSS cuts that loss roughly in half. SAM generates accurate region proposals but selects the wrong one based solely on geometric criteria; its LZA is below chance on both benchmarks. SegFormer trained on data from both terrain types achieves 98.0% and 78.4% LZA, with the combined training set slightly outperforming in-domain training. The LZA and LSRR metrics give the community a way to evaluate landing detection that is directly tied to whether a landing would succeed, rather than how closely a predicted mask matches a reference mask. The results and code will be made publicly available.

5.1. Limitations and Future Work

LZA and LSRR are new metrics, so there are no published numbers from other papers to compare against. The results here are an internal benchmark. Published F1 and IoU Figures are included alongside LZA and LSRR precisely so that future work can contact the existing literature. The binary safe/hazard split involves judgment calls. Rooftops are labeled Hazard even when they are structurally accessible. Roads are labeled 'Safe' even when vehicles are on them. A finer label scheme, distinguishing “always safe”, “conditionally safe”, and “unsafe”, would be more accurate but would require substantially more annotation effort. Both datasets were collected in daylight under reasonable visibility. Whether these methods hold up at dusk, in rain, or under artificial lighting is not known from this data. The most useful near-term extension would be a lightweight semantic scorer for SAM proposals. The segmentation quality is already there; only the region classification is missing. A small MLP trained on SAM region embeddings, with a few hundred labeled examples, might close much of the gap between SAM zero-shot and supervised SegFormer at much lower annotation cost. Multi-frame consistency, either through Kalman filtering of centroid predictions or through Ravi et al. [11], is another straightforward improvement. The LZA and LSRR evaluation code will be released publicly.

Acknowledgment: The authors thank the maintainers of the Aeroscapes and UAVid datasets for making their data available to the research community.

Data Availability Statement: The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Funding Statement: The authors received no financial support for the research, authorship, or publication of this paper.

Conflicts of Interest Statement: The authors declare that there are no conflicts of interest regarding the publication of this research.

Ethics and Consent Statement: This study was conducted in accordance with applicable ethical standards. Informed consent was obtained from all participants involved in the study, where applicable.

References

1. A. Keipour, M. Mousaei, and S. Scherer, “ALFA: A dataset for UAV fault and anomaly detection,” *The International Journal of Robotics Research*, vol. 40, no. 2–3, pp. 515–520, 2021.
2. L. Xin, Z. Tang, W. Gai, and H. Liu, “Vision-Based Autonomous Landing for the UAV: A Review,” *Aerospace*, vol. 9, no. 11, p. 634, 2022.

3. J. L. Sanchez-Lopez, M. Molina, H. Bavle, C. Sampedro, R. A. S. Fernández, and P. Campoy, "A Multi-Layered Component-Based Approach for the Development of Aerial Robotic Systems: The Aerostack Framework," *Journal of Intelligent & Robotic Systems*, vol. 88, no. 5, pp. 683–709, 2017.
4. E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. the 35th Int. Conf. Neural Information Processing Systems*, Sydney, New South Wales, Australia, 2021.
5. P. Ma, C. He, Z. Zhang, Z. Xv, H. Wang, Y. Cai, L. Chen, C. Zhong, and Y. Zhang, "Machine vision based perception for vehicle-mounted UAV autonomous landing under GNSS-denied environments," *Journal of King Saud University Computer and Information Sciences*, vol. 37, no. 11, p. 334, 2025.
6. H. Wang, Y. Shan, L. Chen, M. Liu, L. Wang, and Z. Meng, "Multi-scale feature learning for 3D semantic mapping of agricultural fields using UAV point clouds," *International Journal of Applied Earth Observation and Geoinformation*, vol. 141, no. 7, p. 104626, 2025.
7. S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, and Y. Wang, "Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, Tennessee, United States of America, 2021.
8. A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W. Y. Lo, P. Dollár, and R. Girshick, "Segment Anything," arXiv.org arXiv:2304.02643, 2023. [Accessed by 11/03/2025].
9. K. Chen, C. Liu, H. Chen, H. Zhang, W. Li, and Z. Zou, "RSPrompter: Learning to Prompt for Remote Sensing Instance Segmentation Based on Visual Foundation Model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, no. 1, pp. 1–17, 2024.
10. T. Houichime and Y. El Amrani, "Reinforcement Learning-Based Monocular Vision Approach for Autonomous UAV Landing," arXiv.org arXiv:2505.06963, 2025. [Accessed by 17/06/2026].
11. 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.org arXiv:2408.00714*, 2024. [Accessed by 23/03/2025].
12. T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.
13. I. Nigam, C. Huang, and D. Ramanan, "Ensemble knowledge transfer for semantic segmentation," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Tahoe, Nevada, United States of America, 2018.
14. Y. Lyu, G. Vosselman, G. S. Xia, A. Yilmaz, and M. Y. Yang, "UAVid: A semantic segmentation dataset for UAV imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 165, no. 7, pp. 108–119, 2020.
15. I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, New Orleans, Louisiana, United States of America, 2019.
16. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, Austria, 2021.