Object Boundary Refinement Using Level Sets

Elyar Esmaeilzadeh*, Sude Onder*, Sepehr Maleki*
Master's Students, Computer Engineering, Bilkent University
**All authors contributed equally to this work.*

***Abstract*—This work presents a method to refine object boundaries using initial contours derived from user-defined inputs or machine learning-based initialization. We explore traditional segmentation approaches, including Active Contours and the Level Set Method, alongside an enhanced Level Set technique that incorporates Sobel filters for improved edge detection. Our analysis compares various initialization strategies, such as user-defined contours, machine learning predictions, and ground truth bounding boxes. The results demonstrate that the Sobel filter-based method excels on synthetic images, while the classic Level Set Method performs better on real-world datasets like Pascal VOC.**

***Index Terms*—Image Segmentation, Active Contour, Level Set, Sobel Filter, Machine Learning, Computer Vision, Pascal VOC, Synthetic Images**

I. Introduction

Image segmentation is crucial in computer vision for identifying the boundaries of objects within images. The Level Set method, which evolves contours implicitly using a level set function, is a widely-used tool for segmentation. However, it can face challenges in noisy or complex environments. Similarly, the Active Contour model minimizes an energy functional to fit a contour to the edges of an object. This paper investigates different ways to initialize contours: A) user-defined contours, B) machine learning-based initialization, and C) default phi initialization. Once the contour is initialized, we apply various segmentation methods, namely: A) Level Set Method, and B) Enhanced Level Set with Sobel Filters on Pascal VOC dataset. Moreover, we compare these methods with Active Contour (Snake Model) on synthetic images. This study aims to compare how initialization strategies impact segmentation performance, particularly in terms of robustness to noise and accuracy on various datasets. Furthermore, it compares the effect of different traditional approaches on real world and synthetic images.

II. Related Work

A. Active Contour Models (Snakes)

Active Contour models, or Snakes, are widely used for image segmentation. These models evolve an initial curve to match the boundaries of the object. The energy functional used in Snakes includes both internal energy (which controls smoothness) and external energy (which draws the curve towards the edges of the image). The energy functional for Snakes is:

$$E_{snake} = \int_{\Gamma} \left(\alpha |\mathbf{v}'(s)|^2 + \beta |\mathbf{v}''(s)|^2 - I(\mathbf{v}(s)) \right) ds \tag{1}$$

Where: - Γ is the contour of the snake. - $\mathbf{v}(s)$ is the position of the snake at curve parameter s. - $\mathbf{v}'(s)$ and $\mathbf{v}''(s)$ are the first and second derivatives of the curve, respectively, controlling the smoothness. - α and β are parameters controlling the snake's internal energy (tension and rigidity). - $I(\mathbf{v}(s))$ is the external energy derived from the image's intensity, typically using edge information like the gradient magnitude. The snake evolves by minimizing this energy functional [1].

(a) Prediction (b) Image

Fig. 1: Segmentation using level sets.

B. Level Set Method

The Level Set method represents contours implicitly using a level set function $\phi(x, t)$. The evolution of the contour is governed by the following equation:

$$\frac{\partial \phi}{\partial t} = \delta(\phi(x)) \cdot (\alpha \nabla \phi \cdot \nabla I + \beta I) \tag{2}$$

Where: - $\phi(x, t)$ is the level set function at time t (evolving over time). - $\nabla \phi(x)$ is the gradient of the level set function, representing the boundary's evolution. - $\delta(\phi(x))$ is the Dirac delta function that forces the curve to evolve at the zero level set (the boundary). - α and β are parameters controlling the smoothness and the effect of the boundary.

The curve evolves under the influence of image forces, with $\nabla\phi \cdot \nabla I$ capturing the gradient information from the image [2].

C. Edge Detection and Sobel Filters

Sobel filters are commonly used to detect edges in images by convolving the image with kernels that calculate the gradient in the horizontal (G_x) and vertical (G_y) directions:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

The gradient magnitude G is calculated as:

$$G = \sqrt{G_x^2 + G_y^2} \tag{3}$$

This gradient magnitude is used as the external energy in both the Active Contour and Level Set methods to attract the contours towards the edges [5], [3].

III. Methodology

This study explores three methods of initializing contours and three segmentation models, as outlined below:

A. Contour Initialization Strategies

The initialization of the level set contour significantly influences the efficiency and convergence behavior of the segmentation process. Previous works have established the effectiveness of level sets for segmentation tasks, particularly with robust initialization strategies [7]. In the level set method, the contour is represented implicitly using a level set function, ϕ, where the zero level set ($\phi = 0$) defines the contour. The values of ϕ indicate regions inside ($\phi > 0$) or outside ($\phi < 0$) the contour. The choice of initialization for ϕ plays a critical role in determining how effectively and efficiently the contour evolves to segment the object.

This study investigates four distinct initialization strategies: an initial bounding box (default phi), user-defined contours, and machine learning-based predictions using XGBoost and Random Forest models. Each method was evaluated for its impact on iteration count, convergence time, and segmentation accuracy.

1) Initial Bounding Box (Default Phi): The initial bounding box serves as the simplest and most widely used initialization strategy, providing a baseline for comparison. This method initializes the contour using a default phi function, with two distinct modes:

- Mode 1 (center initialization): The contour is initialized as a small positive region centered within the object. This mode is effective for objects with a clear, central location and ensures the level set evolves outward to capture the object boundary.
- Mode 2 (outside initialization): The contour is initialized as a narrow band outside the object, encompassing its boundary. This mode is ideal for objects with well-defined edges and ensures inward evolution of the contour.

Although this method consistently converged to accurate segmentations, it required more iterations and longer convergence times compared to user-defined and machine learning-based methods. Its simplicity makes it a reliable fallback for cases where prior knowledge about the object's location or advanced model predictions are unavailable.

2) User-Defined Contours: User-defined contour initialization involves manual input from the user to specify the starting contour. Two approaches were implemented:

- Bounding rectangle: The user selects a rectangular region to define the initial contour. This approach is simple and effective, leveraging the user's domain knowledge to provide precise initialization.
- Freeform contour: The user draws a freeform region directly on the image, offering greater flexibility for irregularly shaped objects. This method allows users to fine-tune the contour based on visual inspection, making it particularly useful for complex structures.

Both approaches preserved segmentation accuracy while significantly reducing the number of iterations required for convergence. Among these, the freeform contour achieved the lowest iteration count, demonstrating the advantage of precise initialization in computational efficiency. However, reliance on manual input limits scalability for large datasets.

3) Machine Learning-Based Initialization: To automate the initialization process, machine learning models—XGBoost and Random Forest—were trained to predict bounding boxes for initializing the level set contour.

a) Feature extraction: Features were extracted using Histogram of Oriented Gradients (HOG) combined with pixel intensity values:

- All images were resized to a fixed size of 128×128 for consistency.
- Images were converted to grayscale for HOG feature extraction.
- Pixel intensity values were flattened into a 1D array.
- HOG features were computed with the following parameters:
  - 9 orientations,
  - 16×16 pixels per cell,
  - 2×2 cells per block.
- The final feature vector combined pixel intensity and HOG features.

b) Model comparison: Two models were evaluated for predicting bounding boxes:

- Random Forest: This ensemble model uses a collection of decision trees trained on random subsets of the data. It provided consistent predictions but lacked the precision needed for complex bounding box estimation.
- XGBoost: The gradient-boosted decision tree model outperformed Random Forest in accuracy. Its ability to focus on misclassified samples during training and its regularization mechanisms likely contributed to its superior performance. XGBoost's optimization for speed and accuracy made it better suited for bounding box prediction.

Experimental results showed that XGBoost achieved higher accuracy in predicting bounding boxes compared to Random Forest in terms of MSE, leading to faster convergence in the level set method. Machine learning-based initialization reduced iteration counts relative to user-defined and default initialization, offering a scalable alternative.

However, it was observed that the bounding boxes predicted by the models often failed to fully encompass the object to be segmented. This limitation resulted in some parts of the object lying outside the initial bounding box. When the level set method was applied to such incomplete initializations, these regions outside the bounding box were excluded from the segmentation process, leading to partial or incomplete segmentation of the object. Improving the bounding box prediction model, either by training on more diverse datasets

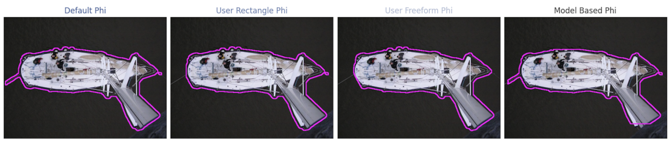


Fig. 2: Segmentation results for different initialization methods.

TABLE I: Iteration Count Comparison for Different Initialization Methods

Initialization Method	Number of Iterations
Default Phi	142
User Rectangle Phi	122
User Freeform Phi	45
Model Based Phi	108

or incorporating post-prediction adjustments, is essential to ensure more complete object segmentation and to fully leverage the advantages of machine learning-based initialization in the level set method.

Table I presents a comparison of the number of iterations required for running different initialization methods on Fig. 2. The results demonstrate that the User Freeform Phi method significantly reduces the number of iterations, requiring only 45, compared to the Default Phi method, which takes 142 iterations. The Model Based Phi and User Rectangle Phi methods also show improvements with 108 and 122 iterations, respectively. This highlights the importance of selecting an appropriate initialization strategy to optimize iteration performance and improve computational efficiency while keeping segmentation accuracy.

B. Segmentation Models

Once the initial contour is defined, we apply the following segmentation models:

1) Active Contour (Snake Model): This model evolves the initial contour by minimizing an energy functional, which includes terms for both internal smoothness and external edge attraction.
2) Level Set Method: This method implicitly represents the contour using a level set function and evolves based on image features such as gradients and edge information. The evolution of the contour in our model is governed by the formula below:

$$u(t+1) = u(t) + \Delta t \cdot v \cdot |\nabla u| \tag{4}$$

where v is the external energy defined by:

$$v = \frac{1}{1 + k \cdot |\nabla I|} \tag{5}$$

and $|\nabla I|$ is the gradient magnitude of the image and it defined as:

$$|\nabla I| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \tag{6}$$

Moreover, $|\nabla u|$ is the gradient magnitude of the level set function:

$$|\nabla u| = \sqrt{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2} \tag{7}$$

We also use Gaussian Blur in order to get rid of the initial image to get rid of some of the noise:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-\mu_x)^2 + (y-\mu_y)^2}{2\sigma^2}\right) \tag{8}$$

Following the segmentation process, post-processing techniques are applied to eliminate noise, thereby improving the overall quality of the results.
3) The Enhanced Level Set method with Sobel filters integrates edge detection into the segmentation process. Sobel filters compute image gradients in the horizontal and vertical directions, capturing edge information. This gradient magnitude is then used to enhance the external energy in the Level Set evolution. The modified Level Set equation incorporates this edge information, helping the contour evolution focus on actual boundaries while minimizing the impact of noise. This method improves segmentation accuracy, especially in noisy environments, by guiding the contour more effectively toward true object edges.

IV. Datasets and Experiment Setup

We evaluated our segmentation approach on two datasets:

- Pascal VOC: A well-known real-world dataset with complex backgrounds and diverse object classes. It is particularly useful for testing the performance of segmentation models in natural environments. [4]
- Synthetic Images: Computer-generated images designed to test the segmentation method's robustness in controlled environments with simple, clean object boundaries.

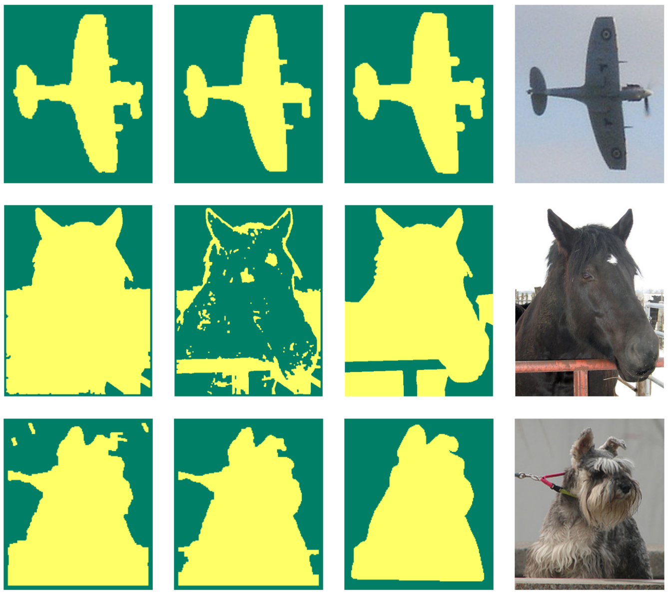
Gaussian noise was added to synthetic dataset to evaluate the robustness of the enhanced level set method under varying noise conditions. The noise levels were categorized as light, medium, and heavy.

V. Segmentation Results

The results of applying different methods and segmentation models are summarized below. We evaluate the performance based on segmentation accuracy, particularly in noisy conditions.

A. Segmentation on Pascal VOC Dataset

On the Pascal VOC dataset, the Level Set method performs well, producing accurate segmentations with IoU higher than 80.0 in the images with simple background (Fig. 3 and 1). However, it is not able to perform well on images with more complex backgrounds and it achieves an average IoU of 63.3 and F1 score of 75.8 on the Pascal VOC dataset. We have also tried enhanced level set method with sobel filter on the Pascal VOC dataset and observed that the results are the same in most cases as those of the original level set method, and sometimes even worse, and this method achieved IoU of 60.5 and F1 score of 72.9 on the Pascal VOC dataset. Moreover, we have tried different parameter values and compared the results in terms of 3 common metrics (Fig. 4) used for segmentation tasks: IoU(Jaccard Index), F1 Score, and Accuracy and found that dt = 2 works best for the Pascal VOC dataset.



(a) (b) (c) (d)

Fig. 3: Segmentation results on images with simple background in the Pascal VOC dataset. (a) refers to the traditional level set, (b) refers to level set with Sobel filter, (c) refers to ground truth, and (d) refers to the image.

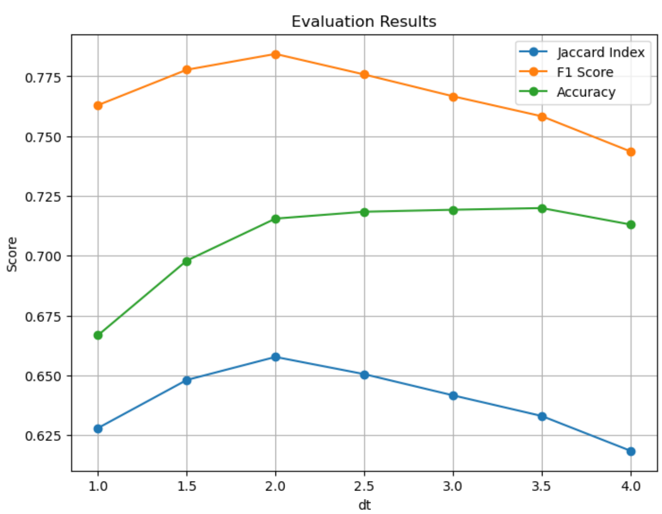


Fig. 4: Effect of changing parameter dt in the level set method on the Pascal VOC dataset.

B. Segmentation on Synthetic Images

On synthetic images, which have well-defined and clean object boundaries, the segmentation performance varies across methods. Figure 5 compares the results from three approaches: Simple Level Set, Improved Level Set (Sobel Level Set), and Snake Model.

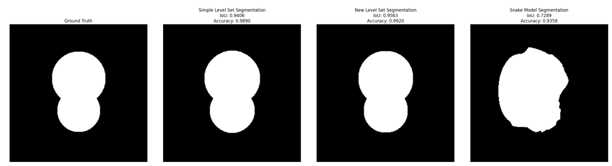


Fig. 5: Segmentation results on synthetic images using Simple Level Set, Improved Level Set (Sobel Level Set), and Snake Model. Level Set methods perform better on clean synthetic shapes.

The Simple Level Set method demonstrates good segmentation performance with an IoU of 94.0 and Accuracy of 98.9. The Improved Level Set (Sobel Level Set) achieves the best results with an IoU of 95.6 and Accuracy of 99.2, excelling at precisely capturing object boundaries. The Snake Model performs noticeably worse, with an IoU of 72.9 and Accuracy of 93.5, showing its limitations for synthetic shapes. To further test the robustness of these models, different levels of noise (Light, Medium, and Heavy) were added to the synthetic images. In Figure 6, the Simple Level Set method struggles under increasing noise levels, failing to produce meaningful segmentations even after adjusting the *dt* parameter.

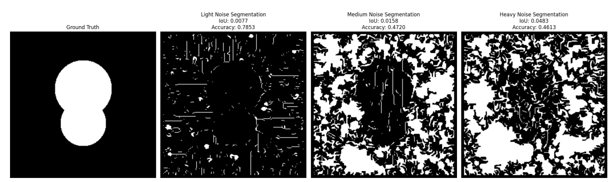


Fig. 6: Segmentation results with noise using the Simple Level Set method. The method fails to handle noise, even with parameter adjustments.

In contrast, the Improved Level Set (Sobel Level Set) performs better across varying noise intensities according to Figure 7, maintaining reasonable segmentation results when the *dt* parameter is properly optimized.

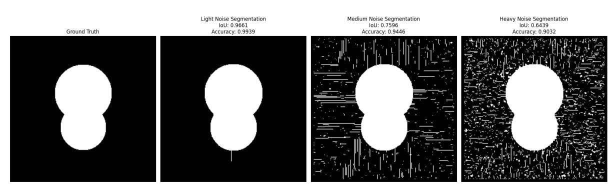


Fig. 7: Segmentation results with noise using the Improved Level Set (Sobel Level Set). The method maintains better performance across different noise levels compared to the Simple Level Set.

These findings highlight that the Improved Level Set (Sobel Level Set) is effective on clean and synthetic shapes, showing robustness against noise when optimized properly. However, when the same models were evaluated on the Pascal VOC dataset, the Improved Level Set (Sobel Level Set) failed to replicate its performance. The results showed significantly lower IoU and Accuracy scores, indicating that while the method excels on clean and synthetic images, it struggles with the complexity and variability of real-world datasets.

VI. Quantitative Evaluation

To quantify the segmentation performance, we calculated the IOU for each method across synthetic datasets. The results are presented in Table II.

TABLE II: Quantitative evaluation of segmentation performance on synthetic images.

Method	**Light Noise**	**Medium Noise**	**Heavy Noise**
LS	0.007	0.01	0.04
LS (Sobel)	0.96	0.75	0.64

From the Table II, we can see that on synthetic datasets, the Improved Level Set (Sobel Level Set) achieves superior scores across all noise levels. Furthermore, we evaluate these methods on Pascal VOC dataset and compare it to DeepLabV3 model with a ResNet50 backbone [6] in Table III.

TABLE III: Quantitative evaluation of segmentation performance on real world images.

Method	**IoU**	**F1 Score**	**Accuracy**
Deeplabv3 (resnet50)	0.74	0.83	0.85
LS	0.63	0.75	0.71
LS (Sobel)	0.60	0.72	0.70

From the Table III, both level set methods give lower scores than the state-of-the-art method on the Pascal VOC dataset.

VII. Discussion

The experiments reveal the advantages and limitations of different segmentation models under varying conditions. For synthetic images with clean boundaries, both Simple Level Set and Improved Level Set (Sobel Level Set) perform well, with the Improved Level Set achieving slightly better accuracy and IoU. Under increasing noise levels, the Simple Level Set method struggles significantly, failing to adapt even with parameter adjustments. The Improved Level Set (Sobel Level Set) demonstrates moderate robustness, maintaining segmentation performance under noise. On real-world datasets like Pascal VOC, the Improved Level Set (Sobel Level Set) fails to generalize, showing significantly reduced accuracy and IoU scores. These results suggest that while the Improved Level Set (Sobel Level Set) is effective for clean and synthetic shapes, it requires further refinement and adaptation to handle the complexity of real-world datasets.

VIII. Conclusion

This paper presents a segmentation approach that evaluates Simple Level Set, Improved Level Set (Sobel Level Set), and Snake Model methods across synthetic datasets with varying noise levels and a real-world dataset (Pascal VOC). Improved Level Set (Sobel Level Set) performs best on clean synthetic images, outperforming both Simple Level Set and Snake Model. Under noise, the Improved Level Set (Sobel Level Set) remains moderately robust, while the Simple Level Set fails to adapt effectively. On the Pascal VOC dataset, the Improved Level Set (Sobel Level Set) shows significantly reduced performance, highlighting its limitations in real-world scenarios. Future work could focus on improving the adaptability of the Improved Level Set (Sobel Level Set) for real-world datasets and exploring hybrid approaches for enhanced robustness across diverse conditions.

IX. Acknowledgments

We would like to thank Prof. Hamdi Dibeklioğlu for their guidance and support during this project.

References

[1] Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, 1(4), 321-331.

[2] Osher, S., & Sethian, J. A. (1988). Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1), 12-49.

[3] Sobel, I., & Feldman, G. (1968). An isotropic 3x3 image gradient operator. *Machine Vision for Three-Dimensional Scenes*, 271-272.

[4] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2), 303-338.

[5] Zhang, H., Wang, Y., Liu, Q., & Huang, D. (2015). MR Image Segmentation Using Active Contour Model Incorporated with Sobel Edge Detection. *Chinese Conference on Computer Vision (CCCV 2015, Part I)*, Communications in Computer and Information Science, 546, 429–437. Springer-Verlag Berlin Heidelberg.

[6] Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). Rethinking Atrous Convolution for Semantic Image Segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, 3041-3049. https://doi.org/10.1109/CVPR.2017.349

[7] Huang, B., Pan, Z., Yang, H., & Bai, L. (2020). Variational level set method for image segmentation with simplex constraint of landmarks. *Signal Processing: Image Communication*, 82, 115745.