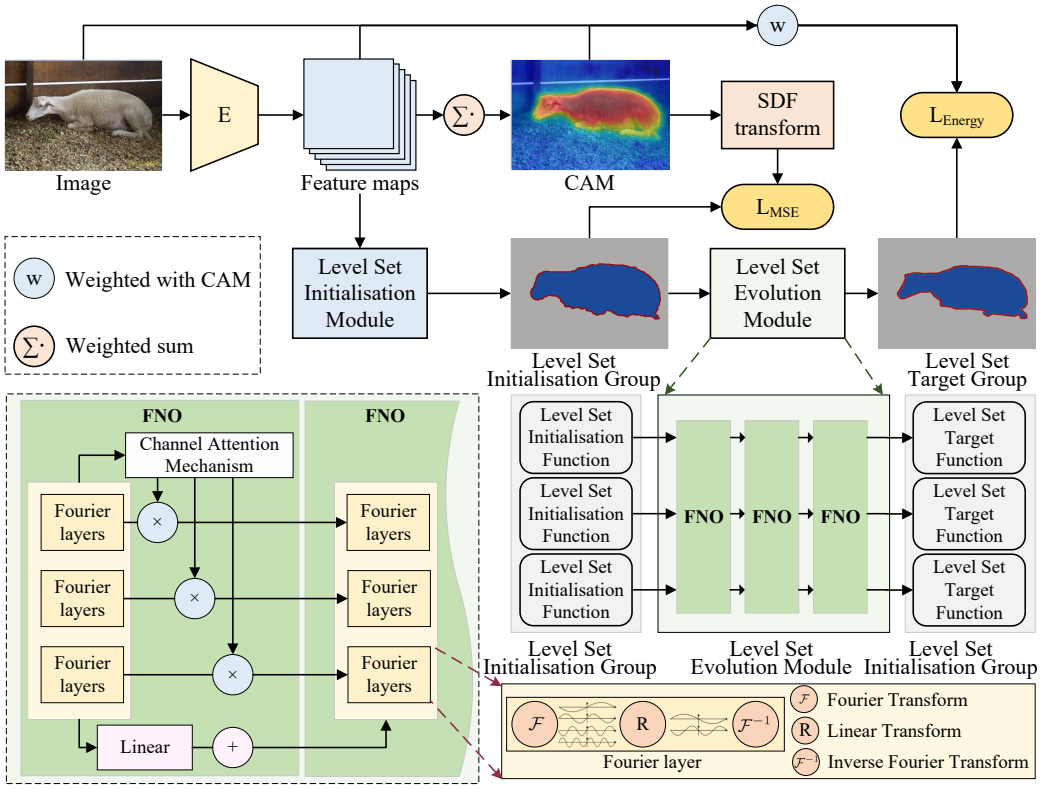
Graphical Abstract

Class activation map guided level sets for weakly supervised semantic segmentation

Yifan Wang, Gerald Schaefer, Xiyao Liu, Jing Dong, Linglin Jing, Ye Wei, Xianghua Xie, Hui Fang



Highlights

Class activation map guided level sets for weakly supervised semantic segmentation

Yifan Wang, Gerald Schaefer, Xiyao Liu, Jing Dong, Linglin Jing, Ye Wei, Xianghua Xie, Hui Fang

- We are the first to apply level sets in image-level WSSS, improving boundary accuracy.
- Our model-agnostic design integrates seamlessly into various WSSS frameworks to improve their performance.
- We introduce Fourier neural operators to accelerate level set evolution, enhancing efficiency.

Class activation map guided level sets for weakly supervised semantic segmentation

Yifan Wang[a], Gerald Schaefer[a], Xiyao Liu[b], Jing Dong[c], Linglin Jing[a], Ye Wei[a], Xianghua Xie[d], Hui Fang[a]

[a]*Department of Computer Science, Loughborough University, UK*
[b]*School of Computer Science and Engineering, Central South University, China*
[c]*The Key Laboratory of Advanced Design and Intelligent Computing, Dalian University, China*
[d]*Department of Computer Science, Swansea University, UK*

Abstract

Weakly supervised semantic segmentation (WSSS) aims to achieve pixel-level fine-grained image segmentation using only weak guidance such as image-level class labels, thus significantly decreasing annotation costs. Despite the impressive performance showcased by current state-of-the-art WSSS approaches, the lack of precise object localisation limits their segmentation accuracy, especially for pixels close to object boundaries. To address this issue, we propose a novel class activation map (CAM)-based level set method to effectively improve the quality of pseudo-labels by exploring the capability of level sets to enhance the segmentation accuracy at object boundaries. To speed up the level set evolution process, we use Fourier neural operators to simulate the dynamic evolution of our level set method. Extensive experimental results show that our approach significantly outperforms existing WSSS methods on both PASCAL VOC 2012 and MS COCO datasets.

Keywords: weakly supervised semantic segmentation, class activation map, pseudo-label, level set, Fourier neural operator

1. Introduction

In computer vision, semantic segmentation, that is, classifying pixels of an image into predefined categories, is a fundamental and crucial task. It plays a vital role in various applications, including environmental perception for autonomous vehicles, lesion detection in medical imaging, and object interactions in robotics [1]. Traditional semantic segmentation methods rely on pixel-level annotations to be used in a fully supervised learning framework, achieving high segmentation accuracy. However, acquisition of precise pixel-level annotations is costly and time-consuming in many real-world applications [2, 3, 4], rendering this paradigm less practical, especially in emerging or specialised fields.

To overcome this limitation of fully supervised methods, weakly supervised semantic segmentation (WSSS) exploits more readily obtainable weak labels, such as image-level labels, bounding boxes, or scribbles, as supervision guidance. Among these, image-level WSSS is particularly appealing due to its minimal annotation requirements. An image-level WSSS method typically comprises three processing stages [5]: (i) the use of methods such as class activation maps (CAMs) to generate initial pseudo-labels guided by image-level annotations; (ii) the refinement of these pseudo-labels via methods such as [5, 6], and (iii) a re-training stage to train a segmentation model using the pseudo-labels.

Recent WSSS work focusses on enhancing the initial label seeds generated for the above-mentioned first processing stage. [7, 2] extend the identifiable regions within CAMs to encompass object parts that are less discriminative

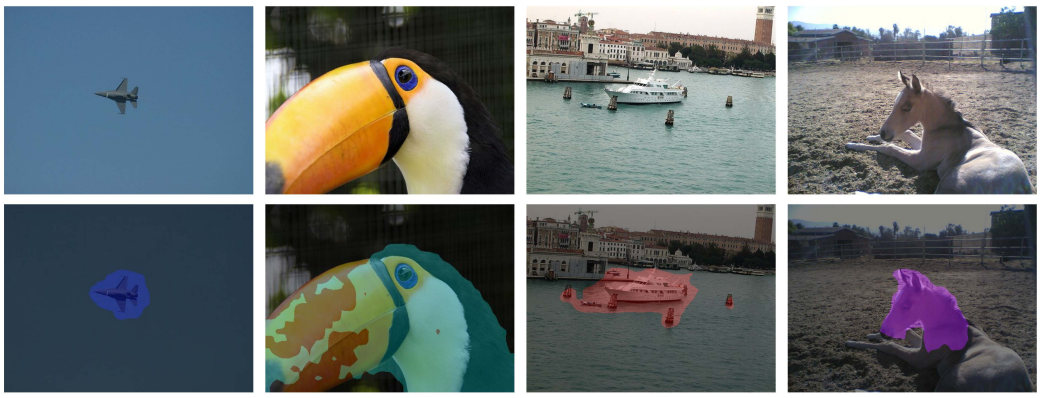


Figure 1: Over-segmentation and under-segmentation are the main challenging issues in WSSS. Top: input images; bottom: images overlaid with semantic segmentations obtained by MCTformer+ [3].

but semantically relevant, thereby enriching the initial seeds with more comprehensive semantic information. [8, 9] refine object boundaries through anti-adversarial manipulations and multi-scale processing techniques, while [10] integrate auxiliary contextually semantic cues to improve the robustness of CAMs and, consequently, segmentation accuracy. Despite these advancements, the inherent limitation of CAM-based approaches, namely the lack of distinctive feature representations across object boundaries, remains a challenge that prevents achieving better segmentation performance. As illustrated in Figure 1, this in turn results in problems, including under-segmentation, missing significant parts of an object, and over-segmentation due to the CAM region extending beyond the actual object boundaries.

In this paper, we propose a novel CAM-based level set approach to address the limitations of existing WSSS methods. Level set frameworks have been widely applied in unsupervised image segmentation due to their advan-

tage of better boundary convergence driven by an energy function minimisation process. A level set approach should therefore be suitable to tackle the under/over-segmentation issues in WSSS. Meanwhile, the CAMs allow to introduce a better initialisation and high-semantic guidance to improve the segmentation quality that level set approaches may suffer. Furthermore, we use Fourier neural operators (FNOs), originally designed for efficiently simulating partial differential equation computations, to improve the convergence efficiency of our level set approach. To our best knowledge, we are the first to deploy a level set framework for image-level WSSS.

Our main contributions in this paper are:

- We design the first level set method for image-level weakly supervised semantic segmentation. Our approach is capable of enhancing the pseudo-label quality, especially yielding better performance on object boundaries.
- Importantly, our method is model agnostic and can thus be readily plugged into various frameworks for improved segmentation performance.
- We introduce Fourier neural operators to accelerate the level set evolution process, addressing efficiency issues of traditional approaches in big-data applications.
- Extensive evaluation on challenging datasets, such as PASCAL VOC 2012, confirms significant performance gains and thus significantly improved segmentation accuracy.

The remainder of the paper is organised as follows: Section 2 reviews related work in the field of weakly supervised semantic segmentation and

level sets. Section 3 then introduces in detail our proposed CAM-based level set approach. Section 4 presents experimental results, demonstrating the effectiveness of our method on benchmark datasets. Finally, Section 5 concludes the paper.

2. Related Work

2.1. Image-level WSSS

Image-level weakly supervised semantic segmentation approaches aim to train semantic segmentation models using only image-level labels. Existing methods typically rely on class activation maps, i.e., heat maps derived from a classification network, to produce pseudo-labels for segmentation network training. However, these techniques often suffer from two significant issues: (i) the generated CAMs tend to cover only the most discriminative regions of objects, leading to partial object coverage, and (ii) CAMs exhibit pseudo-label ambiguities in regions close to object boundaries. Various strategies have been proposed to address these problems. Adversarial learning [2] and equivariant regularisation [11] can be used to enhance attention to non-discriminative regions. Contrastive learning allows to improve feature representations across views [4], while advanced network architectures, such as vision transformers and multi-class tokens, can capture global contexts for class-specific CAMs [12]. Despite these advancements, the challenges of under- and over-segmentation persist.

2.2. Level Sets

Level sets have been widely used for image segmentation for over three decades [13]. Traditional level set approaches use low-level image cues, such

as edges and regions, to construct an evolution function [14]. However, these methods often struggle to accurately initialise and converge to true object boundaries due to the lack of high-level semantic information. Deep learning-based methods allow to overcome the limitations of traditional level set approaches. On one hand, they offer better initialisation of the level set function, for example using a deep belief network to predict initial segmentation contours [15], while on the other hand, deep learning features can be integrated into the level set evolution process, for example by embedding the CNN classification loss into the level set energy function [16], initialising the level set function with CNN-predicted edge probability maps [17], or incorporating CNN-extracted instance-aware features into the level set evolution [18]. By leveraging the obtained high-level semantic information, these methods are able to segment images more accurately.

In summary, integrating strong semantics from deep learning models with the flexible geometric representation and evolution mechanism of level sets yields a strong foundation for image segmentation.In this paper, we propose a novel CAM-based level set approach to tackle the challenging image-level WSSS task.

3. Method

3.1. Problem Formulation

Given an image dataset containing samples $(\mathbf{x}_i, \mathbf{y}_i)$, where $\mathbf{x}_i \in \mathbb{R}^{H\times W\times 3}$ is the i-th input image, $\mathbf{y}_i \in \{0,1\}^C$ is the corresponding image-level label indicating the presence of the C object categories in the image, and H and W denote the image height and width, the goal of WSSS based on image-level

labels is to learn a segmentation model $\mathcal{M}$ that maps an input image $\mathbf{x}_i$ to its corresponding pixel-level segmentation mask $\mathbf{s}_i \in \mathbb{R}^{H \times W \times C}$:

$$\mathbf{s}_i = \mathcal{M}(\mathbf{x}_i), \quad \mathcal{M} : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H \times W \times C}. \tag{1}$$

The three processing stages of WSSS are:

1. *Pseudo-label seed generation:* A classification network f is trained using the image-level labels $\mathbf{y}_i$. The network learns to map input images $\mathbf{x}_i$ to their corresponding class activation maps $\mathbf{s}_i^{seed} \in \mathbb{R}^{H \times W \times C}$, which serve as pseudo-label seeds:

$$\mathbf{s}_i^{seed} = f(\mathbf{x}_i), \quad f : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H \times W \times C}. \tag{2}$$

2. *Pseudo-label refinement:* The pseudo-label seeds $\mathbf{s}_i^{seed}$ are refined using an optimisation algorithm $\mathcal{A}$ to obtain pixel-level pseudo-labels $\hat{\mathbf{s}}_i \in \mathbb{R}^{H \times W \times C}$:

$$\hat{\mathbf{s}}_i = \mathcal{A}(\mathbf{s}_i^{seed}), \quad \mathcal{A} : \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{R}^{H \times W \times C}. \tag{3}$$

   Common optimisation algorithms employed here include PSA (pixel-level semantic affinity) [5], IRN (inter-pixel relation network) [6], and dense CRF (conditional random field) post-processing techniques.

3. *Segmentation network training:* The segmentation model $\mathcal{M}$ is trained using the generated pixel-level pseudo-labels $\hat{\mathbf{s}}_i$ as supervision signals. The model learns to map the input image $\mathbf{x}_i$ to its corresponding pixel-level segmentation mask $\mathbf{s}_i$:

$$\mathbf{s}_i = \mathcal{M}(\mathbf{x}_i), \quad \mathcal{M} : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H \times W \times C}. \tag{4}$$

3.2. Motivation and Method Overview

CAMs form a key component in a typical WSSS approach since they provide pseudo-labels to train the segmentation network. However, the subpar quality of CAMs limits WSSS performance due to their focus on the most discriminant object regions. Intuitively, level sets well complement CAMs due to their focus on boundary convergence driven by an energy minimisation process. In this paper, we therefore design the – to our knowledge – first level set approach for image-level WSSS. An overview of our approach is illustrated in Figure 2, which shows that we employ CAMs as an initialisation strategy for our level set approach, while the level set convergence process enhances the feature quality, which in turn improves the CAM quality to generate better pseudo-labels.

Given an image dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, we first use a backbone network f to extract feature maps $\mathbf{F} \in \mathbb{R}^{H \times W \times k}$ for each input image $\mathbf{x}$ as

$$\mathbf{F} = f(\mathbf{x}). \tag{5}$$

These feature maps $\mathbf{F}$ generate CAMs and initialise the level set functions. The CAMs $\mathbf{C}$ highlight the regions most relevant to each object category and are obtained as

$$\mathbf{C} = \text{ReLU}\left(\sum_{k} w_k \mathbf{F}_k\right), \tag{6}$$

where k is the index of channels. Simultaneously, the feature maps $\mathbf{F}$ are fed into the Level Set Initialisation Module f_{LSI} to generate the initialised level set functions $\mathbf{\Phi}^0$ as

$$\mathbf{\Phi}^0 = f_{LSI}(\mathbf{F}). \tag{7}$$

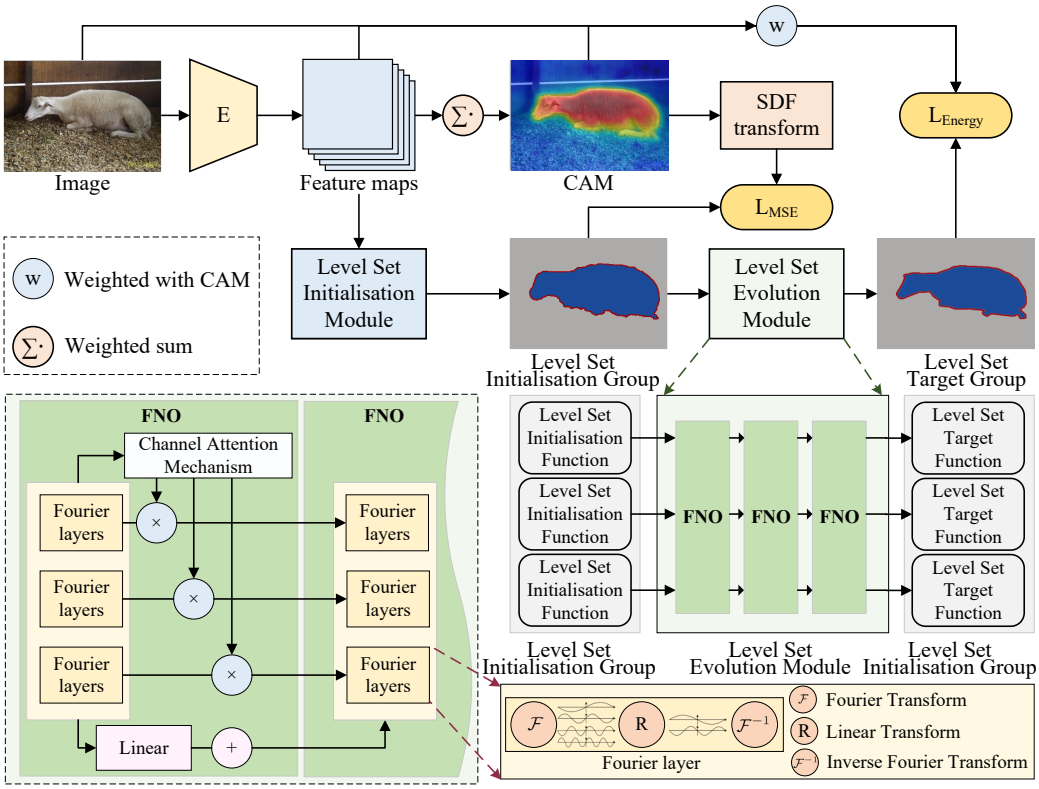


Figure 2: Overview of our proposed approach.

To supervise the initialisation of the level set functions, we transform the CAMs into signed distance functions (SDFs) $\mathbf{D}$ using a distance transform function $\mathcal{T}$. The SDFs serve as labels for the initialised level set functions. Next, the initialised level set functions $\mathbf{\Phi}^0$ are passed to the Level Set Evolution Module f_{LSE} to obtain the target level set functions $\mathbf{\Phi}^T$ as

$$\mathbf{\Phi}^T = f_{LSE}(\mathbf{\Phi}^0). \tag{8}$$

Here, we use an FNO to simulate and speed up the level set evolution process with the FNO parameterising the partial differential equation governing the level set evolution. Finally, the evolved level sets are used to update our

backbone model for better CAM generation.

3.3. CAM-guided Level Set Initialisation

To provide an effective level set initialisation and to accommodate the formulation as a signed distance function, we apply a distance transform function $\mathcal{T}$ that transforms the initial CAM into SDF, guiding the generation of the level set initialisation functions.

Given a CAM $\mathbf{C}$, the class probabilities at pixel (x, y) are computed as

$$\hat{c}(x, y) = \arg\max_{c} \left(\text{softmax}(\mathbf{C}(x, y))\right). \tag{9}$$

Based on these probabilities, a binary mask for each class c is obtained as

$$M_c(x, y) = \begin{cases} 1 & \text{if } \hat{c}(x, y) = c, \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

The SDF for class c is defined by applying the Euclidean distance transform

$$f_{\text{EDT}}(x, y) = \min_{(x', y') \in \{(x,y) | M_c(x,y)=1\}} \|(x', y') - (x, y)\| \tag{11}$$

to each binary mask to yield

$$\mathcal{T}_c(x, y) = \begin{cases} -f_{\text{EDT}}(M_c)(x, y), & \text{if } M_c(x, y) = 1, \\ f_{\text{EDT}}(M_c)(x, y), & \text{if } M_c(x, y) = 0. \end{cases} \tag{12}$$

3.4. FNO-driven Level Set Evolution

We employ Fourier neural operators (FNOs) in our level set evolution framework due to their ability to parameterise continuous operators, enabling end-to-end training of the evolution process and significantly reducing the computational complexity through efficient operations in Fourier space.

An object boundary can be defined as a zero level set Γ of $\phi(x, y)$:

$$\Gamma = \{(x, y) \mid \phi(x, y) = 0\}, \tag{13}$$

where Γ represents the contour or boundary of an object, and $\phi(x, y)$ is a scalar function defined over the image domain.

However, level set methods face challenges when dealing with multiple objects and complex shapes. To address this, we introduce a set of level set functions, called a level set group (LSG), to handle multiple objects. Given an initialised LSG $\mathbf{\Phi}^0 = \{\phi_1^0, \phi_2^0, ..., \phi_c^0\}$, where c is the number of object categories, our goal is to obtain an LSG $\mathbf{\Phi}^T = \{\phi_1^T, \phi_2^T, ..., \phi_c^T\}$ through the Level Set Evolution Module f_{LSE}.

We further adopt Fourier neural operators for improved efficiency of the level set evolution process. An FNO is a type of neural operator, which is introduced to extend neural networks beyond finite dimensions and to enable operator learning to efficiently solve partial differential equations [19]. FNOs use convolutional kernels in Fourier space to significantly improve the efficiency and accuracy of PDE solving [20].

In traditional neural operators, the network processes multiple layers, updating function values in an iterative fashion to approximate PDE solutions. This process starts with iterative updates

$$\phi^{t+1}(x) = \sigma\left(L(\phi^t(x)) + K_\theta(a, \phi^t(x))\right), \quad \forall x \in \mathcal{D}, \tag{14}$$

where $L()$ represents a local linear transformation, σ is a non-linear activation function, and $K_\theta()$ is an integral kernel operator defined as

$$K_\theta(a, \phi^t(x)) = \int_D \kappa_\theta(x, y, a(x), a(y))\phi^t(y)\, dy, \quad \forall x \in \mathcal{D}, \tag{15}$$

where $\kappa_\theta()$ is a kernel function parameterised by θ.

To enhance computational efficiency, FNOs introduce a Fourier integral operator by transforming the kernel operator into Fourier space as

$$K_\theta(\phi^t(x)) = \mathcal{F}^{-1}\left(R(\mathcal{F}(\phi^t(x)))\right), \quad \forall x \in \mathcal{D}, \tag{16}$$

where $\mathcal{F}()$ and $\mathcal{F}^{-1}()$ denote the Fourier and inverse Fourier transforms, respectively, and $R()$ represents a linear transformation in Fourier space.

As illustrated in Figure 2, we introduce a set of FNOs, called an FNO group f_{LSE}, to handle a level set group. Each FNO consists of multiple FNO layers, each comprising three main components: a Fourier transform layer, a linear transform layer, and an inverse Fourier transform layer. The Fourier transform layer converts the input level set function ϕ_i^{t-1} from the spatial domain to the frequency domain. The linear transform layer is a convolutional layer that applies convolution operations in the frequency domain. Finally, the inverse Fourier transform layer converts the responses back to the spatial domain, yielding the updated level set function $\hat{\phi}_i^t$.

To establish interaction and information exchange within an LSG, we introduce a channel attention mechanism after the FNO layers, inspired by SE-Net [21]. This generates a set of weights $\mathbf{w} = \{w_1, w_2, ..., w_c\}$, which are used to compute a weighted combination of the updated results from different level set functions. Furthermore, we introduce a residual connection mechanism that adds the output of a linear transformation applied to the input level set function ϕ_i^{t-1} to the output of the FNO. This facilitates gradient propagation and improves network training.

The updated level set function $\phi_i^{t+1}(x)$ is thus expressed as

$$\phi_i^{t+1}(x) = \sigma\left(L(\phi_i^t(x)) + w_i \mathcal{F}^{-1}\left(R_i(\mathcal{F}(\phi_i^t(x)))\right)\right), \tag{17}$$

where σ is a non-linear activation function, and w_i is the weight, obtained from the channel attention mechanism, for the i-th level set function.

In an iterative manner, the level set evolution module thus gradually optimises the level set functions $\mathbf{\Phi}^T$, allowing them to converge to the contours of the target objects.

3.5. Energy Loss Function

In image segmentation, the level set method is used to identify and track the dynamic evolution of object boundaries as

$$\frac{\partial \phi}{\partial t} + E|\nabla \phi| = 0, \tag{18}$$

where E is a function representing the velocity field, and $|\nabla \phi|$ denotes the magnitude of the gradient of ϕ.

A prominent application of this method is the Chan-Vese model [22], which simplifies the Mumford-Shah function, and employs

$$\begin{aligned} \mathcal{E}^{CV}(\phi, \mu, \nu) = &\int_\Omega |I(x,y) - \mu|^2 H(\phi(x,y))\, dx\, dy \\ &+ \int_\Omega |I(x,y) - \nu|^2 (1 - H(\phi(x,y)))\, dx\, dy \\ &+ \gamma \int_\Omega |\nabla H(\phi(x,y))|\, dx\, dy \end{aligned} \tag{19}$$

as the energy function for segmenting images, where H denotes the Heaviside function, and Ω represents the image space. The zero crossing contour $\Gamma = \{(x,y) : \phi(x,y) = 0\}$ of the level set ϕ divides the image space into two

disjoint regions: the inside of the contour ($\Omega_1 = \{(x, y) : \phi(x, y) > 0\}$, and the outside the contour $\Omega_2 = \{(x, y) : \phi(x, y) < 0\}$. The first two terms in the energy function aim to fit the data by minimising the squared differences between the image $I(x, y)$ and the mean values μ and ν inside and outside Γ, respectively, while the third term regularises the zero-level contour with a non-negative parameter γ. Image segmentation is thus achieved by finding the level set function $\phi(x, y) = 0$ with μ and ν that minimise the energy $\mathcal{E}^{CV}$.

3.5.1. Incorporating Class-wise Information through CAMs

To incorporate class information into the level set evolution process, we use class activation maps to weigh the original input data. In particular, we employ a weighting function $\mathcal{W}$, which combines the CAM-generated weights $\mathcal{C}_c(x, y)$ with the input $\mathbb{X}_c(x, y)$ compute the weighted input

$$\begin{aligned} \mathbb{I}_c(x, y) &= \mathcal{W}(\mathbb{X}(x, y), \mathcal{C}_c(x, y), \alpha) \\ &= \mathbb{X}(x, y) \left[(1 - \alpha) + \alpha\sigma(\mathcal{C}_c(x, y))\right]. \end{aligned} \tag{20}$$

The weighted input $\mathbb{I}_c(x, y)$ is then fed into the energy function to compute the loss as

$$\begin{aligned} E(\mathbf{\Phi}, \mathbb{I}) = \sum_{c=1}^{C} \Bigg(&\int_c \left|\mathbb{I}_c(x, y) - \mu_c\right|^2 H(\phi_c(x, y))\, dx\, dy \\ &+ \int_c \left|\mathbb{I}_c(x, y) - \nu_c\right|^2 (1 - H(\phi_c(x, y)))\, dx\, dy \Bigg) \\ &+ \gamma \sum_{c=1}^{C} \int_c \left|\nabla H(\phi_c(x, y))\right| \, dx\, dy, \end{aligned} \tag{21}$$

where μ_c is the within-class mean

$$\mu_c(\phi_c) = \frac{\int_c \mathbb{I}_c(x, y)(\phi_c(x, y))\, dx\, dy}{\int_c \sigma(\phi_c(x, y))\, dx\, dy}, \tag{22}$$

and ν_c the between-class mean

$$\nu_c(\phi_c) = \frac{\int_c \mathbb{I}_c(x,y)(1-(\phi_c(x,y)))\,dx\,dy}{\int_c (1-(\phi_c(x,y)))\,dx\,dy}. \tag{23}$$

To effectively exploit both low-level information from images and high-level information from feature maps, our final energy loss, obtained as

$$\mathcal{L}_{\text{Energy}} = E(\mathbf{\Phi}^T, \mathbf{x}) + E(\mathbf{\Phi}^T, \mathbf{F}), \tag{24}$$

incorporates contributions from both types of inputs. This comprehensive approach allows to leverage the distinct yet complementary information from each input type, thereby improving the ability to accurately track object contours during the level set evolution.

3.5.2. Overall Loss

To effectively guide the level set evolution, our overall loss function integrates two key components, a mean squared error (MSE) loss and the energy loss. The MSE loss $\mathcal{L}_{\text{MSE}}$, computed as

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{\Phi}_i^0 - \mathbf{D}_i\|_2^2, \tag{25}$$

with

$$\mathbf{D} = \mathcal{T}(\mathbf{C}), \tag{26}$$

supervises the initialisation of the level set group using the SDFs transformed from CAMs, while the energy loss $\mathcal{L}_{\text{Energy}}$ further drives the evolution process, ensuring accurate object contour tracking.

The overall loss is the defined as

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{MSE}} + \lambda_2 \mathcal{L}_{\text{Energy}}, \tag{27}$$

where λ_1 and λ_2 are weighting factors to balance the contribution of each loss component.

4. Experimental Results

4.1. Experimental Settings

We conduct experiments on the PASCAL VOC 2012 dataset [23], which comprises 21 categories including a background category, and on the MS COCO 2014 dataset [24], which contains 81 categories including a background category. Following [5, 11, 8, 2], the VOC2012 dataset is augmented with the SBD dataset [25], providing a total of 10,582 training images, 1,449 validation images, and 1,456 test images, while the COCO2014 dataset contains 82,783 training images and 40,504 validation images. Only image-level labels are employed for WSSS training. All experiments are performed on an RTX 3090 GPU using the PyTorch framework [26].

In our FNO implementation, the network architecture consists of four Fourier layers, each comprising a SpectralConv2d module with modes=12 and width corresponding to the number of semantic categories in the dataset (20 for PASCAL VOC). Each Fourier layer is followed by a 1×1 convolutional layer (maintaining the category-specific channel dimensionality) and a channel attention module. As activation functions, GELU is used for intermediate layers and ReLU for the channel attention modules.

For model training, to ensure a fair comparison, we strictly follow the original implementations of the respective baseline methods. Input images are resized to 448×448 pixels, and we maintain the optimiser settings, namely AdamW with an initial learning rate of 5e-4 for MCTformer+, 1e-4 for MuS-

CLE, and 0.01 for SEAM. Similarly, we preserve the learning rate scheduling strategies, employing a cosine annealing schedule with a 5-epoch warm-up period, as well as the same data augmentation techniques including random horizontal flipping and colour jittering.

As is standard practise, we use the mean intersection-over-union (mIoU) as the performance measure to evaluate our proposed model and to compare it with a number of state-of-the-art (SoTA) approaches, including the CNN-based IRN [6], SC-CAM [27], SEAM [11], BES [28], LIID [29], OAA [30], RIB [31], URN [32], AdvCAM [2], RCA [33], LPCAM [34], MuSCLe [4], and MDBA [35], the transformer-based AFA [36], MCTformer [12], OCR [8], and MCTformer+ [3], and the text embedding-based CLIMS [37] and CLIP-ES [38].

4.2. Model Performance and Model-agnostic Approach

We start by evaluating the performance of our proposed approach on the PASCAL VOC dataset. One of the advantages of our method is that it is model-agnostic and that it can thus be applied to any WSSS method to yield a performance boost. To demonstrate this, we conduct experiments using three classical WSSS methods, namely SEAM [11], MuSCLe [4], and MCTformer+ [3].

The obtained mIoU results of CAM and pseudo-labels on the VOC training set are reported in Table 1.

As we can see from there, our proposed approach improves yields improved performance of all models and for both seeds and pseudo-labels. Compared to the underlying benchmark methods, we boost the mIoU by 3.1/3.2 (SEAM), 2.5/2.8 (MuSCLe), and 2.3/0.9 (MCTformer+) for (seeds/pseudo-

Table 1: mIoU results on PASCAL VOC *train* set with and without our proposed method.

	seed	pseudo-label
SEAM [11]	55.4	63.6
SEAM + proposed method	58.5 (+3.1)	66.8 (+3.2)
MuSCLe [4]	58.4	66.8
MuSCLe + proposed method	60.9 (+2.5)	69.6 (+2.8)
MCTformer+ [3]	68.8	76.2
MCTformer+ + proposed method	71.1 (+2.3)	77.1 (+0.9)

labels), convincingly confirming that our approach significantly enhances the seed quality and consequently the pseudo-labels.

4.3. Ablation Study for FNO-driven Level Set Evolution

To evaluate the effectiveness of our FNO-driven level set evolution module, including the supervision during initialisation and the energy-based evolution, we conduct an ablation study which investigates the contribution of each component to the overall performance of our model. The results on the VOC2012 *train* set using MCTformer+ with different configurations for level set processes are given in Table 2.

The results there confirm that incorporating level set evolution significantly improves the seed quality, with our effective initialisation further enhancing performance. The combined use of CAM and level set processes leads to an overall improved performance, with the best results, an mIoU improvement from 68.8 to 71.1, obtained when both components are employed.

To evaluate the computational efficiency gain of our proposed FNO-driven

Table 2: Ablation study results on VOC2012 *train* set using MCTformer+ with different level set processes.

MCTformer+	$\mathcal{L}_{\text{MSE}}$	$\mathcal{L}_{\text{Energy}}$	mIoU
✓			68.8
✓	✓		69.1
✓		✓	70.6
✓	✓	✓	71.1

Table 3: Comparison of computational efficiency between traditional level set evolution and FNO-driven evolution (running time in seconds).

traditional evolution				FNO-driven evolution
1 iteration	5 iterations	10 iterations	15 iterations	
0.3316	1.9541	3.4308	5.6777	1.7832

level set evolution, we compare it to the traditional iterative level set evolution process in Table 3. Considering that traditional level set methods typically require at least dozens of iterations to achieve convergence [39, 40], these results demonstrate that our FNO-driven evolution process significantly speeds up the level set process. This, in turn, also makes our FNO-based approach more suitable than conventional level set methods for tasks where computational complexity is important, such as real-time applications or larges-scale data processing.

4.4. Seed and Pseudo-Label Performance vs. SoTA

In Table 4 and 5, we compare the quality of the obtained seeds and generated pseudo-labels of our method with those of other SoTA approaches

on the PASCAL VOC and COCO datasets, respectively.

Table 4: mIoU results on PASCAL VOC *train* set.

	backbone	seed	pseudo-labels
$\text{IRN}_{\text{CVPR19}}$ [6]	ResNet50	48.0	61.0
$\text{SC-CAM}_{\text{CVPR20}}$ [27]	ResNet38	50.9	63.4
$\text{SEAM}_{\text{CVPR20}}$ [11]	ResNet38	55.4	63.6
$\text{BES}_{\text{ECCV20}}$ [28]	ResNet50	50.4	67.2
$\text{RIB}_{\text{NIPS21}}$ [31]	ResNet50	62.9	70.6
$\text{AdvCAM}_{\text{TPAMI22}}$ [2]	ResNet50	55.6	69.9
$\text{MCTformer}_{\text{CVPR22}}$ [12]	DeiT-S	61.7	69.1
$\text{LPCAM}_{\text{CVPR23}}$ [34]	ResNet50	65.3	72.7
$\text{MuSCLe}_{\text{PR23}}$ [4]	EfficientNet	58.4	66.8
$\text{MCTformer+}_{\text{TPAMI24}}$ [3]	DeiT-S	68.8	76.2
MCTformer+ + proposed method	DeiT-S	**71.1**	**77.1**

As is evident from Table 4, and as discussed above, integrating our approach with existing WSSS methods yields improved seed and pseudo-label quality. Our approach achieves an impressive mIoU of 71.1 for seeds and 77.1 for pseudo-labels, clearly outperforming other SoTA methods, including RIB (by 6.5 for pseudo-labels), AdvCAM (7.2), MCTformer (8.0), MuSCLE (10.3), and MCTformer+ (0.9).

Our method also yields improved performance on the COCO dataset, as seen in Table 5, with an mIoU of 44.6 for seeds and 49.2 for pseudo-labels. This represents a clear improvement over MCTformer+ (by 1.8 for seeds and 1.1 for pseudo-labels), further demonstrating the effectiveness of

our approach across different datasets in enhancing seed and pseudo-label quality.

Table 5: mIoU results on COCO *train* set.

	backbone	seed	pseudo-labels
$\text{IRN}_{\text{CVPR19}}$ [6]	ResNet50	33.1	42.5
$\text{SEAM}_{\text{CVPR20}}$ [11]	ResNet38	25.1	31.5
$\text{RIB}_{\text{NIPS21}}$ [31]	ResNet50	36.5	45.6
$\text{AdvCAM}_{\text{TPAMI22}}$ [2]	ResNet50	37.2	46.0
$\text{MCTformer}_{\text{CVPR22}}$ [12]	DeiT-S	36.6	41.6
$\text{LPCAM}_{\text{CVPR23}}$ [34]	ResNet50	42.5	47.7
$\text{MCTformer+}_{\text{TPAMI24}}$ [3]	DeiT-S	42.8	48.1
MCTformer+ + proposed method	DeiT-S	**44.6**	**49.2**

4.5. Segmentation Performance vs. SoTA

After the retraining phase, we obtain segmentation results on the VOC validation and test sets, which we report in Table 6[1]. As is apparent from there, our proposed approach achieves mIoUs of 74.6 and 74.8 on the PASCAL VOC 2012 validation and test sets, respectively, surpassing all other methods.

Segmentation results on the COCO 2014 validation set are given in Table 7. We can see from there that our proposed method yields an mIoU of 46.1, outperforming all other methods.

[1]Note, that, following the original papers, some models use, compared to Table 4, a different backbone here and for the COCO dataset.

Table 6: mIoU segmentation results on PASCAL VOC *val* and *test* sets.

	backbone	*val*	*test*
$\text{SC-CAM}_{\text{CVPR20}}$ [27]	ResNet101	66.1	65.9
$\text{SEAM}_{\text{CVPR20}}$ [11]	ResNet38	64.5	65.7
$\text{BES}_{\text{ECCV20}}$ [28]	ResNet101	65.7	66.6
$\text{LIID}_{\text{TPAMI20}}$ [29]	ResNet101	66.5	67.5
$\text{OAA}_{\text{TPAMI21}}$ [30]	ResNet101	66.1	67.2
$\text{RIB}_{\text{NIPS21}}$ [31]	ResNet101	68.3	68.6
$\text{URN}_{\text{AAAI22}}$ [32]	Res2Net101	71.2	71.5
$\text{AdvCAM}_{\text{TPAMI22}}$ [2]	ResNet101	68.1	68.0
$\text{RCA}_{\text{CVPR22}}$ [33]	ResNet38	72.2	72.8
$\text{MCTformer}_{\text{CVPR22}}$ [12]	ResNet38	71.9	71.6
$\text{AFA}_{\text{CVPR22}}$ [36]	MiT-B1	66.0	66.3
$\text{CLIMS}_{\text{CVPR22}}$ [37]	ResNet50	70.4	70.0
$\text{MuSCLe}_{\text{PR23}}$ [4]	EfficientNet	66.6	68.8
$\text{MDBA}_{\text{TIP23}}$ [35]	ResNet101	70.0	70.2
$\text{OCR}_{\text{CVPR23}}$ [8]	ResNet38	72.7	72.0
$\text{CLIP-ES}_{\text{CVPR23}}$ [38]	ResNet101	73.8	73.9
$\text{MCTformer+}_{\text{TPAMI24}}$ [3]	ResNet38	74.0	73.6
MCTformer+ + proposed method	ResNet38	**74.6**	**74.8**

We also compare with text embedding-based semantic segmentation methods [37, 38], which leverage large-scale, pre-trained language–vision models to capture global semantic cues, thereby enhancing recognition for complex or ambiguous object categories. However, these methods rely chiefly on seman-

Table 7: mIoU results on COCO 2014 *val.* set

	backbone	mIoU
$\text{SEAM}_{\text{CVPR20}}$ [11]	ResNet38	31.9
$\text{RIB}_{\text{NIPS21}}$ [31]	ResNet101	43.8
$\text{URN}_{\text{AAAI22}}$ [32]	Res2Net101	41.5
$\text{AdvCAM}_{\text{TPAMI22}}$ [2]	ResNet101	44.4
$\text{RCA}_{\text{CVPR22}}$ [33]	VGG16	36.8
$\text{MCTformer}_{\text{CVPR22}}$ [12]	ResNet38	42.0
$\text{AFA}_{\text{CVPR22}}$ [36]	MiT-B1	38.9
$\text{MDBA}_{\text{TIP23}}$ [35]	ResNet101	37.8
$\text{OCR}_{\text{CVPR23}}$ [8]	DeiT-S	42.5
$\text{CLIP-ES}_{\text{CVPR23}}$ [38]	ResNet101	45.4
$\text{MCTformer+}_{\text{TPAMI24}}$ [3]	ResNet38	45.2
MCTformer+ + proposed method	ResNet38	**46.1**

tic embeddings rather than boundary features, which can pose difficulties for delineating fine object edges. In contrast, our approach employs CAM-driven level set evolution to refine object boundaries more precisely, although it is sensitive to the quality of CAMs and may not fully capture global contextual information.

Overall, the results obtained on both the PASCAL and COCO datasets convincingly demonstrate the effectiveness of our proposed method, leading to performance improvements in WSSS tasks across datasets.

4.6. Visual Examples

Figure 3 showcases some qualitative segmentation results of our proposed method compared to its MCTformer+ backbone, while Figure 4 presents some representative segmentation results for several of the evaluated methods, namely RCA, SEAM, MuSCLe, MCTformer+, and our proposed method.

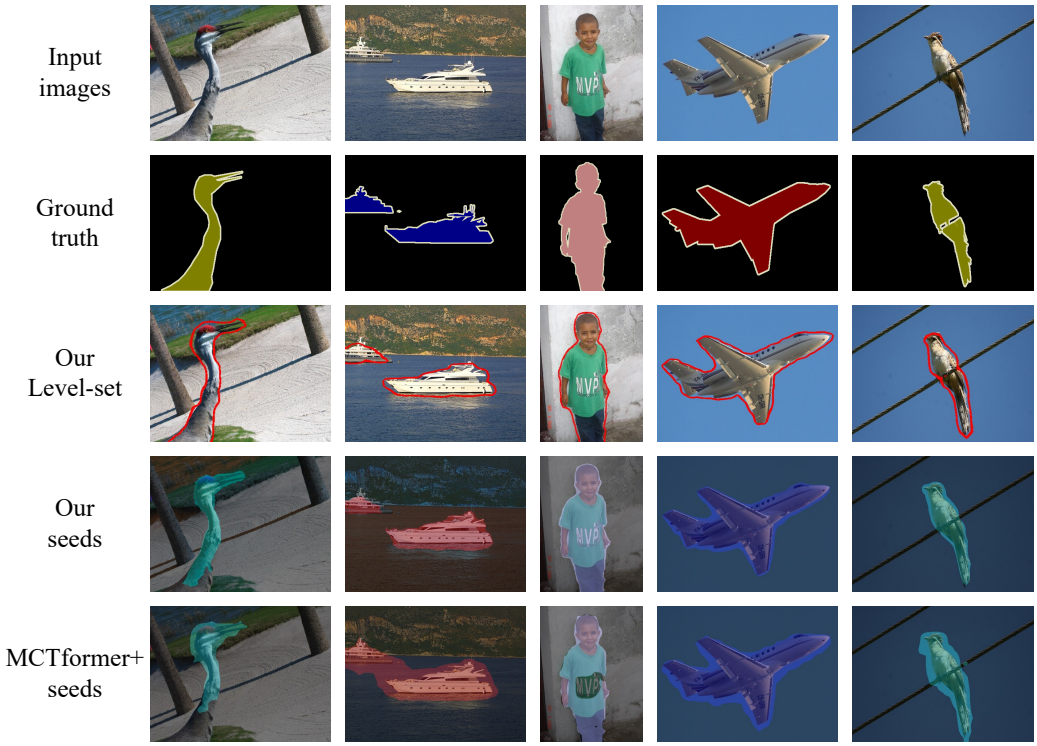


Figure 3: Example seeds and level set results.

As we can observe from Figure 3, the seeds generated by our method are of higher quality and provide more comprehensive coverage of object details and boundaries compared to the backbone model, further demonstrating the effectiveness of our approach in enhancing seed quality.

These improvements in seed generation directly influence the performance

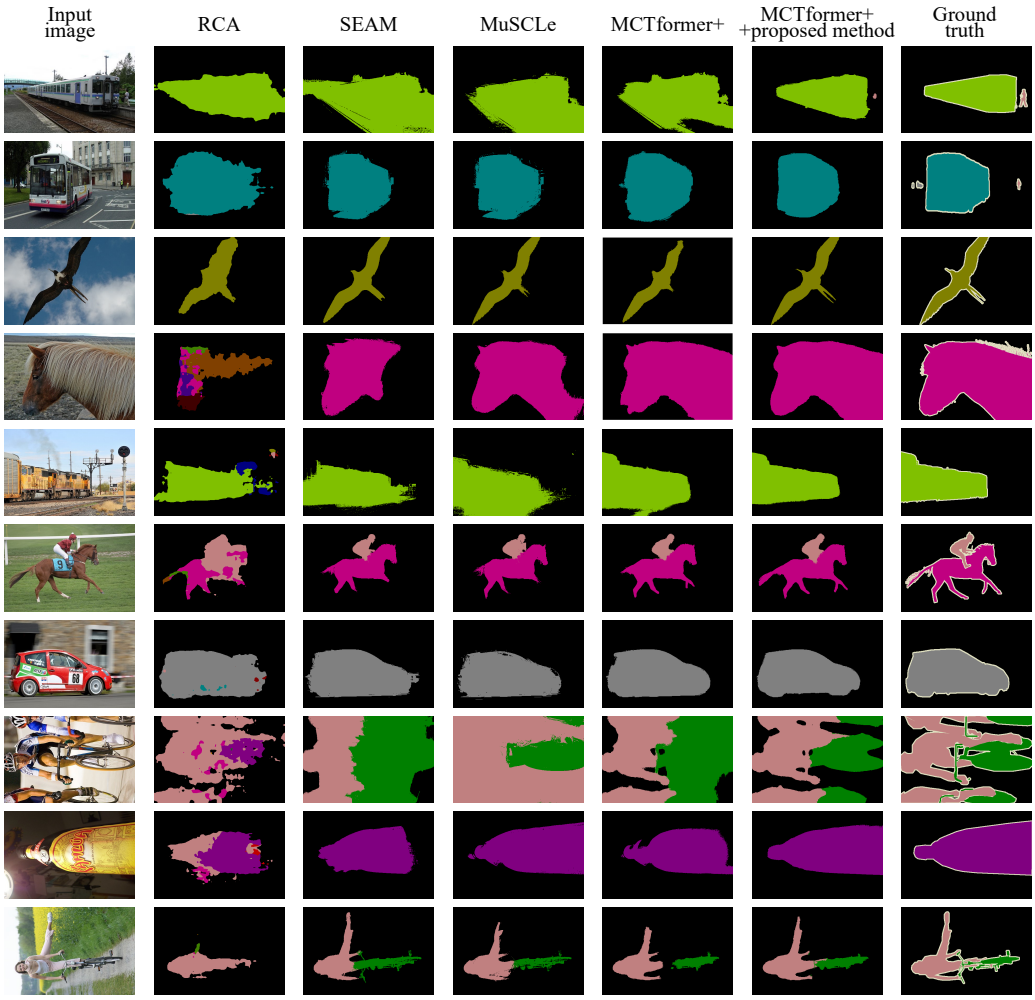


Figure 4: Example segmentations obtained from our proposed approach in comparison with SOTA WSSS methods.

of our level-set segmentation process. As we can observe from Figure 4, our method captures more complex boundaries and significantly reduces noise around object edges, leading to more accurate and detailed segmentations. Furthermore, it demonstrates superior performance in maintaining object continuity and accurately reflecting structural variations. These results fur-

ther validate the effectiveness of our approach in improving boundary refinement for weakly supervised semantic segmentation.

We also show, in Figure 5, some examples of challenging scenarios where our proposed approach achieves only sub-optimal segmentation results. As we can see, our method faces challenges in situations involving multiple objects, complex backgrounds, and intricate contours. Despite these difficulties, our approach demonstrates resilience even here and consistently outperforms other SOTA methods.

We also show, in Figure 5, some examples of challenging scenarios where our proposed approach achieves only sub-optimal segmentation results. Specifically, in scenes involving complex backgrounds or overlapping objects, the quality of CAMs may degrade, providing incomplete or imprecise boundary cues that undermine both level set initialisation and energy function guidance. Moreover, since each object is tracked by a single level set curve, severe occlusions can result in ambiguous overlaps that cannot be adequately separated, leading to suboptimal convergence. Despite these difficulties, our approach demonstrates resilience even here and consistently outperforms other SOTA methods. To address these challenges, potential improvements could include region-based initialisation schemes or additional energy terms to stabilise the evolution when CAM information is insufficient. Furthermore, adopting multi-curve processing could incorporate complementary boundary cues to reduce errors caused by occlusions.

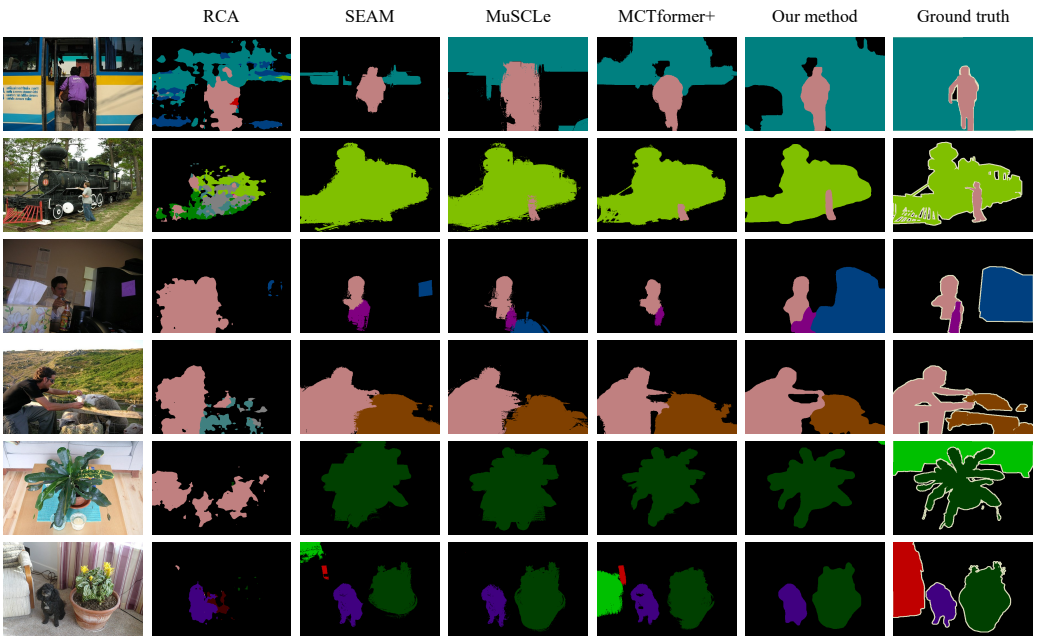


Figure 5: Segmentation results for challenging examples from PASCAL VOC2012.

5. Conclusions

In this paper, we have introduced a new strategy that combines class activation maps with level sets and Fourier neural operators to improve weakly supervised semantic segmentation, addressing the challenges of under-segmentation and over-segmentation in CAM-based methods by incorporating the dynamic boundary evolution of the level set method coupled with the efficiency of FNOs. Extensive experiment on the PASCAL VOC 2012 and COCO 2014 datasets demonstrate our proposed method to improve the quality of pseudo-labels and, consequently, segmentation accuracy, achieving excellent WSSS segmentation performance in comparison to other state-of-the-art WSSS methods.

There are several promising directions for future research. One is to com-

bine our approach with recent text embedding-based models, which could leverage the strengths of both methods. Large-scale language–vision models offer robust semantic understanding, which could be incorporated into the level set energy function, combining global semantics with fine-grained boundary refinement. This integration could improve performance in scenarios requiring contextual awareness and precise edge detection as well as help mitigate our method's reliance on high-quality CAMs. Additionally, exploring the application of our level set framework to different backbone architectures could lead to even more robust WSSS solutions. These extensions could help bridge the gap between the performance of weakly supervised and fully supervised semantic segmentation models.

References

[1] P. de Jorge, R. Volpi, P. H. Torr, G. Rogez, Reliability in semantic segmentation: Are we on the right track?, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7173–7182.

[2] J. Lee, E. Kim, J. Mok, S. Yoon, Anti-adversarially manipulated attributions for weakly supervised semantic segmentation and object localization, IEEE Transactions on Pattern Analysis and Machine Intelligence (2022).

[3] L. Xu, M. Bennamoun, F. Boussaid, H. Laga, W. Ouyang, D. Xu, Mctformer+: Multi-class token transformer for weakly supervised semantic segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence (2024).

[4] K. Yuan, G. Schaefer, Y.-K. Lai, Y. Wang, X. Liu, L. Guan, H. Fang, A multi-strategy contrastive learning framework for weakly supervised semantic segmentation, Pattern Recognition (2023) 109298.

[5] J. Ahn, S. Kwak, Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4981–4990.

[6] J. Ahn, S. Cho, S. Kwak, Weakly supervised learning of instance segmentation with inter-pixel relations, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2209–2218.

[7] J. Li, Z. Jie, X. Wang, X. Wei, L. Ma, Expansion and shrinkage of localization for weakly-supervised semantic segmentation, Advances in Neural Information Processing Systems 35 (2022) 16037–16051.

[8] Z. Cheng, P. Qiao, K. Li, S. Li, P. Wei, X. Ji, L. Yuan, C. Liu, J. Chen, Out-of-candidate rectification for weakly supervised semantic segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 23673–23684.

[9] H. Kweon, S.-H. Yoon, K.-J. Yoon, Weakly supervised semantic segmentation via adversarial learning of classifier and reconstructor, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 11329–11339.

[10] L. Xu, M. Bennamoun, F. Boussaid, W. Ouyang, F. Sohel, D. Xu, Auxiliary tasks enhanced dual-affinity learning for weakly supervised seman-

tic segmentation, IEEE Transactions on Neural Networks and Learning Systems (2024).

[11] Y. Wang, J. Zhang, M. Kan, S. Shan, X. Chen, Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12275–12284.

[12] L. Xu, W. Ouyang, M. Bennamoun, F. Boussaid, D. Xu, Multi-class token transformer for weakly supervised semantic segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 4310–4319.

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

[14] J. Yuan, C. Chen, F. Li, Deep variational instance segmentation, in: Advances in Neural Information Processing Systems, 2020, pp. 4811–4822.

[15] T. A. Ngo, G. Carneiro, Left ventricle segmentation from cardiac mri combining level set methods with deep belief networks, in: IEEE International Conference on Image Processing, 2013, pp. 695–699.

[16] P. Hu, B. Shuai, J. Liu, G. Wang, Deep level sets for salient object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2300–2309.

[17] Z. Wang, D. Acuna, H. Ling, A. Kar, S. Fidler, Object instance annotation with deep extreme level set evolution, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 7500–7508.

[18] N. Homayounfar, Y. Xiong, J. Liang, W.-C. Ma, R. Urtasun, Levelset R-CNN: A deep variational method for instance segmentation, in: European Conference on Computer Vision, 2020, pp. XXIII:555–571.

[19] A. Anandkumar, K. Azizzadenesheli, K. Bhattacharya, N. Kovachki, Z. Li, B. Liu, A. Stuart, Neural operator: Graph kernel network for partial differential equations, in: ICLR Workshop on Integration of Deep Neural Models and Differential Equations, 2020.

[20] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, A. Anandkumar, Fourier neural operator for parametric partial differential equations, arXiv preprint arXiv:2010.08895 (2020).

[21] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.

[22] T. F. Chan, L. A. Vese, Active contours without edges, IEEE Transactions on Image Processing 10 (2) (2001) 266–277.

[23] M. Everingham, L. Van Gool, C. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results, `http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html` (2012).

[24] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common objects in context, in: European Conference on Computer Vision, 2014, pp. 740–755.

[25] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, J. Malik, Semantic contours from inverse detectors, in: International Conference on Computer Vision, 2011, pp. 991–998.

[26] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., PyTorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems, 2019.

[27] Y.-T. Chang, Q. Wang, W.-C. Hung, R. Piramuthu, Y.-H. Tsai, M.-H. Yang, Weakly-supervised semantic segmentation via sub-category exploration, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 8991–9000.

[28] L. Chen, W. Wu, C. Fu, X. Han, Y. Zhang, Weakly supervised semantic segmentation with boundary exploration, in: European Conference on Computer Vision, 2020, pp. 347–362.

[29] Y. Liu, Y.-H. Wu, P. Wen, Y. Shi, Y. Qiu, M.-M. Cheng, Leveraging instance-, image-and dataset-level information for weakly supervised instance segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (3) (2020) 1415–1428.

[30] P.-T. Jiang, L.-H. Han, Q. Hou, M.-M. Cheng, Y. Wei, Online attention accumulation for weakly supervised semantic segmentation, IEEE

Transactions on Pattern Analysis and Machine Intelligence 44 (10) (2021) 7062–7077.

[31] J. Lee, J. Choi, J. Mok, S. Yoon, Reducing information bottleneck for weakly supervised semantic segmentation, in: Advances in Neural Information Processing Systems, 2021, pp. 27408–27421.

[32] Y. Li, Y. Duan, Z. Kuang, Y. Chen, W. Zhang, X. Li, Uncertainty estimation via response scaling for pseudo-mask noise mitigation in weakly-supervised semantic segmentation, in: AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 1447–1455.

[33] T. Zhou, M. Zhang, F. Zhao, J. Li, Regional semantic contrast and aggregation for weakly supervised semantic segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 4299–4309.

[34] Z. Chen, Q. Sun, Extracting class activation maps from non-discriminative features as well, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 3135–3144.

[35] T. Chen, Y. Yao, J. Tang, Multi-granularity denoising and bidirectional alignment for weakly supervised semantic segmentation, IEEE Transactions on Image Processing (2023).

[36] L. Ru, Y. Zhan, B. Yu, B. Du, Learning affinity from attention: End-to-end weakly-supervised semantic segmentation with transformers, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 16846–16855.

[37] J. Xie, X. Hou, K. Ye, L. Shen, Clims: Cross language image matching for weakly supervised semantic segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 4483–4492.

[38] Y. Lin, M. Chen, W. Wang, B. Wu, K. Li, B. Lin, H. Liu, X. He, Clip is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 15305–15314.

[39] D. Cremers, M. Rousson, R. Deriche, A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape, International Journal of Computer Vision 72 (2007) 195–215.

[40] C. Li, C. Xu, C. Gui, M. D. Fox, Distance regularized level set evolution and its application to image segmentation, IEEE Transactions on Image Processing 19 (12) (2010) 3243–3254.