END TO END TRAINABLE ACTIVE CONTOURS VIA DIFFERENTIABLE RENDERING

Anonymous authors
Paper under double-blind review

ABSTRACT

We present an image segmentation method that iteratively evolves a polygon. At each iteration, the vertices of the polygon are displaced based on the local value of a 2D shift map that is inferred from the input image via an encoder-decoder architecture. The main training loss that is used is the difference between the polygon shape and the ground truth segmentation mask. The network employs a neural renderer to create the polygon from its vertices, making the process fully differentiable. We demonstrate that our method outperforms the state of the art segmentation networks and deep active contour solutions in a variety of benchmarks, including medical imaging and aerial images.

1 INTRODUCTION

The importance of automatic segmentation methods is growing rapidly in a variety of fields, such as medicine, autonomous driving and satellite image analysis, to name but a few. In addition, with the advent of deep semantic segmentation networks, there is a growing interest in the segmentation of common objects with applications in augmented reality and seamless video editing.

Since the current semantic segmentation methods often capture the objects well, except for occasional inaccuracies along some of the boundaries, fitting a curve to the image boundaries seems to be an intuitive solution. Active contours, is a set of techniques that given an initial contour (which can be provided by an existing semantic segmentation solution) grow iteratively to fit an image boundary. Active contour may also be appropriate in cases, such as medical imaging, where the training dataset is too limited to support the usage of a high-capacity segmentation network.

Despite their potential, the classical active contours fall behind the latest semantic segmentation solutions with respect to accuracy. The recent learning-based active contour approaches were not yet demonstrated to outperform semantic segmentation methods across both medical datasets and real world images, despite having success in specific settings. ~~are either non-competitive or proven to be effective in the specific settings of building segmentation~~.

In this work, we propose to evolve an active contour based on a 2-channel displacement field (corresponding to 2D image coordinates) that is inferred directly and only once from the input image. This is, perhaps, the simplest approach, since unlike the active contour solutions in literature, it does not involve and balance multiple forces, and the displacement is given explicitly. Moreover, the architecture of the method is that of a straightforward encoder-decoder with two decoding networks. The loss is also direct, and involves the comparison of two mostly binary images.

The tool that facilitates this explicit and direct approach is a neural mesh renderer. It allows for the propagation of the intuitive loss, back to the displacement of the polygon vertices. While such renderers have been discovered multiple times in the past, and were demonstrated to be powerful solutions in multiple reconstruction problems, this is the first time, as far as we can ascertain that this tool is used for image segmentation.

Our empirical results demonstrate state of the art performance in a wide variety of benchmarks, showing a clear advantage over classical active contour methods, deep active contour methods, and modern semantic segmentation methods.

2 RELATED WORK

Neural Renderers A neural mesh renderer is a fully differential mapping from a mesh to an image. While rendering the 3D or 2D shape given vertices, faces, and face colors is straightforward, the process involves sampling on a grid, which is non-differentiable. One obtains differentiable rendering, by sampling in a smooth (blurred) manner (Kato et al., 2018) or by approximating the gradient based on image derivatives, as in (Loper & Black, 2014). Such renderers allow one to ~~Backpropogate~~Backpropagate the error from the obtained image back to the vertices of the mesh.

In this work we employ the mesh renderer of Kato et al. (2018). Perhaps the earliest mesh renderer was presented by Smelyansky et al. (2002). Recent non-mesh differential renders include the point cloud renderer of Insafutdinov & Dosovitskiy (2018) and the view-based renderer of Eslami et al. (2018). Gkioxari et al. (2019) use a differentiable sampler to turn a 3D mesh to a point cloud and solve the task of simultaneously segmenting a 2D image object, while performing a 3D reconstruction of that object. This is a different image segmentation task, which unlike our setting requires a training set of 3D models and matching 2D views.

Active contours Snakes were first introduced by Kass et al. (1988), and were applied in a variety of fields, such as lane tracking (Wang et al., 2004), medicine (Yushkevich et al., 2006) and image segmentation (Michailovich et al., 2007). Active contours evolve by minimizing an energy function and moving the contour across the energy surface until it halts. Properties, such as curvature and area of the contour, guide the shape of the contour, and terms based on the image gradients or various alternatives attract contours to edges. Most active contour methods rely on an initial contour, which often requires a user intervention.

Variants of this method have been proposed, such as using a balloon force to encourage the contour to expand and help with bad initialization, *e.g.*contours located far from objects (Kichenassamy et al., 1995; Cohen, 1991). Kichenassamy et al. (1995) employ gradient flows, modifying the shrinking term by a function tailored to the specific attracting features. Caselles et al. (1997) presented the Geodesic Active Contour (GAC), where contours deform according to an intrinsic geometric measure of the image, induced by image features, such as borders. Other methods have replaced the use of edge attraction by the minimization of the energy functional, which can be seen as a minimal partition problem (Chan & Vese, 2001; Marquez-Neila et al., 2014).

The use of learning base models coupled with active contours was presented by Rupprecht et al. (2016), who learn to predict the displacement vector of each point on the evolving contour, which would bring it towards the closest point on the boundary of the object of interest. This learned network relies on a local patch that is extracted per each contour vertex. This patch moves with the vertex as the contour evolves gradually. Our method, in contrast, predicts the displacement field for all image locations at once. This displacement field is static and it is the contour which evolves. In our method, learning is not based on a supervision in the form of the displacement to the nearest ground truth contour but on the difference of the obtained polygon from the ground truth shape.

A level-set approach for active contours has also gained popularity in the deep learning field. Work such as (Wang et al., 2019; Hu et al., 2017; Kim et al., 2019) use the energy function as part of the loss in supervised models. Though fully differentiable, the level-sets do not enjoy the simplicity of polygons and their ability to fit straight structures and corners that frequently appear in man-made objects (as well as in many natural structures). The use of polygon base snakes in neural networks, as a fully differentiable module, was presented by Marcos et al. (2018); Cheng et al. (2019) in the context of building segmentation.

Building segmentation and recent active contour solutions Semi-automatic methods for urban structure segmentation, using polygon fitting, have been proposed by Wang et al. (2006); Sun et al. (2014). Wang et al. (2016); Kaiser et al. (2017) closed the gap for full automation, overcoming the unique challenges in building segmentation. Marcos et al. (2018) argued that the geometric properties, which define buildings and urban structures, are not preserved in conventional deep learning semantic segmentation methods. Specifically, sharp corners and straight walls, are only loosely learned. In addition, a pixel-wise model does not capture the typical closed-polygon structure of the buildings. Considering these limitations, Marcos et al. (2018) presented the Deep Structured Active Contours (DSAC) approach, in order to learn a polygon representation instead of the segmentation mask. The polygon representation of Active Contour Models (ACMs) is well-suited for building

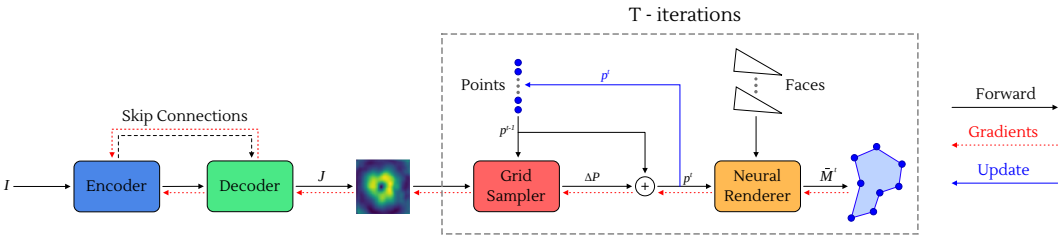


Figure 1: Illustration of our method. The input image I is encoded using network E and decoded back by the decoder D to provide a 2D displacement field J. The vertices of the polygon at time $t-1$ are updated by the displacement values specified by J, creating the polygon of the next time step. During training, a neural renderer reconstructs the polygon shape, based on the polygon vertices and the output of triangulation process. A loss is provided by comparing the reconstructed shape with the ground truth segmentation mask.

boundaries, which are typically relatively simple polygons. ACMs are usually modeled to attract to edges of the reference map, mainly the raw image, and penalize over curvature regions. The DSAC method learns the energy surface, to which the active contour attracts. During training, DSAC integrates gradient propagation from the ACM, through a dedicated structured-prediction loss that minimizes the Intersection-Over-Union.

Cheng et al. (2019), extended this approach, presenting the Deep Active Ray Network (DARNet), based on a polar representation of active contours, also known as active rays models of Denzler & Niemann (1999). To handle the complexity of rays geometry, Cheng et al. (2019) reparametrize the rays and compute L1 distance between the ground truth polygon, represented as rays and the predicted rays. Furthermore, in order to handle the non-convex shapes inherit in the structures, Cheng et al. (2019) presented the Multiple Sets of Active Rays framework, which is based on the deep watershed transform by Bai & Urtasun (2017).

Castrejón et al. (2017) and Acuna et al. (2018) presented a polygonal approach for objects segmentation. Given an image patch, the methods use a recurrent neural network to predict the ground truth polygon. Acuna et al. (2018) additionally made use of graph convolution networks and a reinforcement learning approach. Ling et al. (2019), introduced the Curve-GCN, an interactive or fully-automated method for polygon segmentation, learning an embedding space for points, and using Graph Convolution neural networks to estimate the amount of displacement for each point. Ling et al. (2019) also presented the use of differentiable rendering process, using the non-differentiable OpenGL renderer, and the method of Loper & Black (2014) for computing the first order Tylor approximation of the derivatives. Curve-GCN supports both polygon and spline learning, but also added supervision by learning an explicit location for each point, and edge maps.

Both DSAC and DARNet enjoy the benefit of a polygon-based representation. However, this is done using elaborate and sophisticated schemes. Curve-GCN on the other hand, benefits from the use of differentiable rendering, but suffers from a time consuming mechanism that restricts it to apply it only in the fine-tuning stage. Our method is considerably simpler due to the use of a fast differentiable renderer, and a 2-D displacement field in the scale of the input. It uses only ground truth masks as supervision, and two additional loss terms that are based on time-tested pulling forces from the classical active contour literature: the Balloon and Curvature terms.

3 METHOD

Our trained network includes an encoder E and decoder D. In addition, a differential renderer R is used, as well as a triangulation procedure L. In this work we use the Delaunay triangulation.

Let S be the set of all training images. Given an image $I \in \mathbb{R}^{c \times h \times w}$, an initial polygon contour P^0 is produced by an oracle A, and the faces of this shape are retrieved from a triangulation procedure L, which returns a list F of mesh faces, each a triplet of polygon vertex indices. In many benchmarks in which the task is to segment given a Region of Interest (ROI) extracted by a detector or marked

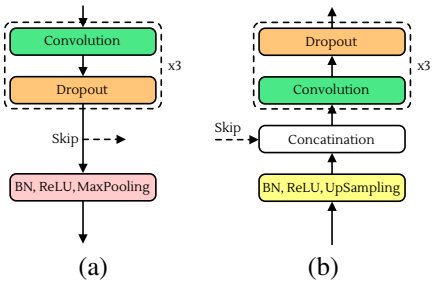


Figure 2: Illustration of the (a) encoder and (b) decoder blocks.

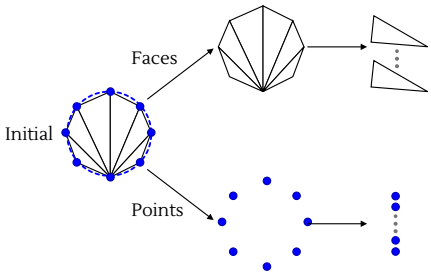


Figure 3: Illustration of initial contour generation process.

by a user, the oracle simply returns a fixed circle in the middle of the ROI. Fig 3 illustrates the initial contour generation process.

The contour evolves for T iterations, from P^0 to P^1 and so on until the final shape given by P^T. It is represented by a list of k vertices $P^t = [p_1^t, p_2^t, \ldots, p_k^t]$, where each vertex is a two dimensional coordinate. This evolution follows a dual-channel displacement field:

$$J = D_1(E(I)) \in \mathbb{R}^{c \times h \times w}. \tag{1}$$

For every vertex $j = 1..k$, the update follows the following rule:

$$p_j^t = p_j^{t-1} + J(p_j^{t-1}) \tag{2}$$

where $J(p_j^{t-1})$ is a 2D vector, and the operation $J(\cdot)$ denotes the sampling operation of displacement field J, using bi-linear interpolation at the coordinates of p_j^{t-1}.

Coordinates which fall outside the boundaries of the image are then truncated (the following uses square brackets to refer to indexed vector elements):

$$p_j^t[1] = min(h, max(0, p_j^t[1])), p_j^t[2] = min(w, max(0, p_j^t[2])) \tag{3}$$

The neural renderer, given the vertices and the faces returns the polygon shape as a mask, where all pixels inside the faces F are ones, and zero otherwise:

$$\bar{M}^t = R(P^t, F) \in \mathbb{R}^{h \times w} \tag{4}$$

where $\bar{M}^t$ is the output segmentation at iteration t. This mask is mostly binary, except for the boundaries where interpolation occurs. Because the used renderer works in 3D space, we project all points to 3D by setting their z axis to 1, and use orthogonal projection.

This segmentation mask $\bar{M}^t$ is compared to the ground truth mask M at each iteration, and accumulated over T iterations to obtain the segmentation loss:

$$\mathcal{L}_{\text{SEG}} = \sum_{t=1}^{T} \|\bar{M}^t - M\|_2 \tag{5}$$

where $\|\cdot\|_2$ is the MSE loss applied to all mask values.

The curve evolution in classic active contour models is influenced by two additional forces: a ballooning force and a curvature minimizing force. In our feed forward network, these are manifested as training losses. The Balloon term $\mathcal{L}_{\mathcal{B}}$, maximizes the polygon area, causing it to expand:

$$\mathcal{L}_{\mathcal{B}} = \frac{1}{h \times w} \sum_{x} (1 - \bar{M}^t(x)) \tag{6}$$

where h and w are the segmentation height and width, and x denotes a single pixel in $\bar{M}^t$. Second, the Curvature term $\mathcal{L}_{\mathcal{K}}$ minimizes the curvature of the polygon, resulting in a more smooth form:

$$\mathcal{L}_{\mathcal{K}} = \frac{1}{k} \sum_{j} ||p_{j-1}^t - 2p_j^t + p_{j+1}^t||_2 \tag{7}$$

Algorithm 1 Active contour training of networks ~~E, D_1, D_2~~E, D. Shown for a batch size of one.

Require: $\{I_i\}_{i=1}^n$: Input images, $\{M_i\}$: Matching ground truth segmentation masks, A : Initial guess oracle, R : differential renderer, L : a triangulation procedure, k : number of vertices, T : number of iterations, λ_1, λ_2 : a weighting parameter.

Initialize networks E, D
for multiple epochs **do**
  for i = 1....n **do**
    $P^0 = [p_1^0, p_2^0, .., p_k^0] \leftarrow A(I_i)$ ▷ Initialize the polygon using the oracle
    $F \leftarrow L(P^0)$ ▷ Triangulation to obtain the mesh faces
    $J \leftarrow D(E(I))$
    for t = 1....T **do**
      Let $P^t = [p_1^t, p_2^t, .., p_k^t]$
      for j = 1....k **do**
        $p_j^t \leftarrow p_j^{t-1} + J(p_j^{t-1})$ ▷ Set the vertices of polygon P^t
      $\bar{M}^t = R(P^t, F)$ ▷ The polygon shape as an image
      $\mathcal{L} \leftarrow \mathcal{L} + \|\bar{M}^t - M\|_2 + \lambda_1 \frac{1}{h \times w} \sum_x (1 - \bar{M}^t(x)) + \lambda_2 \frac{1}{k} \sum_j |p_{j-1}^t - 2p_j^t + p_{j+1}^t|_2$
    ~~Backpropogate~~Backpropagate the loss $\mathcal{L}$ and update E, D

where the L_2 norm is computed on 2D coordinate vectors.

The complete training loss is therefore:

$$\mathcal{L} = \mathcal{L}_{\text{SEG}} + \lambda_1 \mathcal{L}_{\mathcal{B}} + \lambda_2 \mathcal{L}_{\mathcal{K}}, \tag{8}$$

for some weighting parameter λ_1 and λ_2. It is applied after each evolution of the contour (and not just on the final contour). See Alg. 1 for a listing of the process.

3.1 Architecture and Training

We employ an Encoder-Decoder architecture with U-Net skip connections (Çiçek et al., 2016), which link layers of matching sizes between the encoder sub-network and the decoder sub-network, as can be seen in Fig. 1. The encoder part, as can be seen in Fig. 2(a), is built from blocks which are mirror-like versions of the relative decoder blocks, as can be seen in Fig. 2(b), connected by a skip connection.

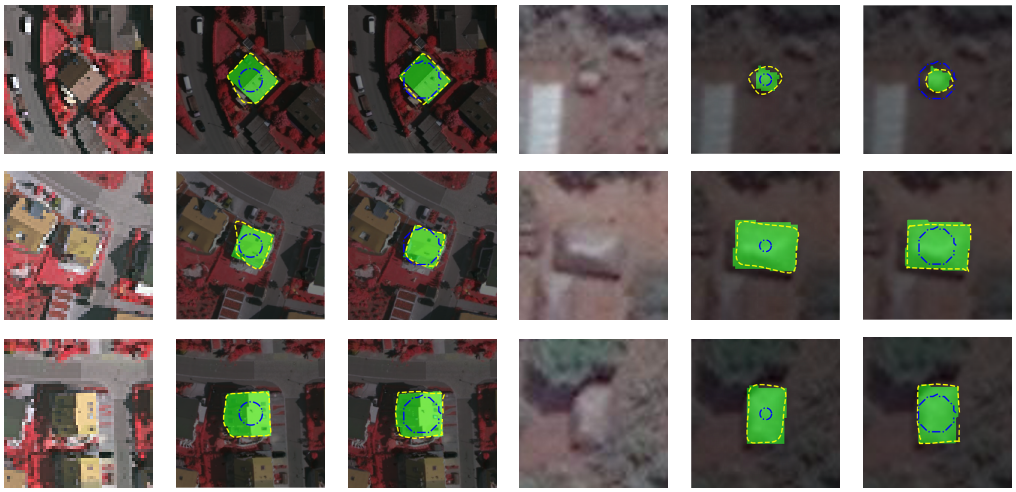
An encoder block consists of (i) three sub-blocks of convolution layer followed by dropout with probability of 0.2, (ii) ReLU, (iii) batch normalization and (iv) max-pooling, to down-sample the input feature map.

~~The decoder blocks~~ A decoder block consists of (i) batch normalization, (ii) ReLU, (iii) bi-linear interpolation, which up-samples the input feature map to the size of the skip connection, (iv) concatenation of the input skip connection and the output of the previous step, (v) three sub-blocks of convolution layer, followed by dropout with probability of 0.2. For the last decoder block, we omit the dropout layer, and up-sample to the input image size using bi-linear interpolation. To get the pixel-wise probabilities, we employ the Sigmoid (logistic) activation.

For training the segmentation networks, we use the ADAM optimizer Kingma & Ba (2014) with a learning rate 0.001, batch size varies depending of input image size, for 64×64 we use 100, 128×128 we use 50. We set $\lambda_1 = 10^{-2}$ and $\lambda_2 = 5 \cdot 10^{-1}$.

4 Experiments

For evaluation, we use the common segmentation metrics of F1-score and Intersection-over-Union (IoU). Additionally, for the buildings segmentation datasets, we use the Weighted Coverage (WCov) and Boundary F-score (BoundF), which is the averaged F1-score over thresholds from 1 to 5 pixels around the ground truth, as described by Cheng et al. (2019).



(a) (b) (c) (d) (e) (f)

Figure 4: Qualitative results of DARNet (Cheng et al., 2019) and our method. Columns **(a)-(c)** show results from the Vaihingen dataset (Rottensteiner et al.), and **(d)-(f)** show results from the Bing huts (Marcos et al., 2018) dataset. **(b) and (e)** - DARNet (Cheng et al., 2019), **(c) and (f)** - Ours. **Blue** - Initial contour. **Yellow** - Final contour. **Green** - GT Mask.

Table 1: Quantitative results on the two buldings datasets of Vaihingen (Rottensteiner et al.) and Bing (Marcos et al., 2018). † denotes the use of DSAC as backbone, and ‡ denotes the use of DARNet as backbone.

Method	Vaihingen				Bing			
	F1-Score	mIoU	WCov	BoundF	F1-Score	mIoU	WCov	BoundF
FCN†	-	81.09	81.48	64.6	-	69.88	73.36	30.39
FCN‡	-	87.27	86.89	76.84	-	74.54	77.55	37.77
DSAC†	-	71.10	70.76	36.44	-	38.74	44.61	37.16
DSAC‡	-	60.37	61.12	24.34	-	57.23	63.09	15.98
DARNet‡	93.65	88.24	88.16	75.91	85.21	75.29	77.07	38.08
Ours	**95.62**	**91.74**	**89.03**	**79.19**	**91.04**	**84.73**	**82.23**	**58.29**

4.1 Building Segmentation

We consider two publicly available datasets in order to evaluate our method, the Vaihingen (Rottensteiner et al.) dataset, which contains buildings from a German city, and the Bing Huts dataset (Marcos et al., 2018), which contains huts from a village in Tanzania. A third dataset named TorontoCity, proposed by Marcos et al. (2018); Cheng et al. (2019) is not yet publicly available (private communication, 2019). **The Vaihingen dataset** consists of 168 buildings extracted from ISPRS Rottensteiner et al.. All images contain centered buildings with a very dense environment, including other structures, streets, trees and cars, which makes the task more challenging. The dataset is divided into 100 buildings for training, and the remaining 68 for testing. The image's size is $512 \times 512 \times 3$, which is relatively high. We experiment with different resizing factors during training. **The Bing Huts dataset** consists of 606 images, 335 images for train and 271 images for test. The images suffer from low spatial resolution and have the size of 64×64, in contrast to the Vaihingen dataset.

We compare our method to the relevant previous works, following the evaluation process, as described in Cheng et al. (2019), using the published test/val/train splits. The evaluated polygons are scaled, according to the original code of Cheng et al. (2019). For both datasets, we augment the training data (of the networks) by re-scaling in factors of $[0.75, 1, 1.25, 1.5]$, and rotating by $[0, 15, 45, 60, 90, 135, 180, 210, 240, 270]$ degrees.

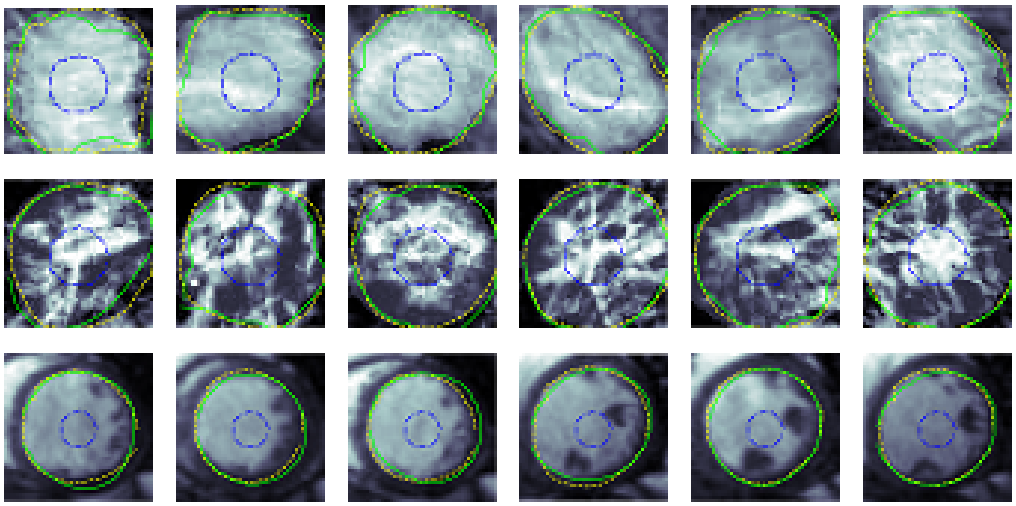


Figure 5: Qualitative results on the mammographic and cardiac datasets. **Top** - INBreast (Moreira et al., 2012), **Middle** - DDSM-BCRP (Heath et al., 1998), **Bottom** - SCD (Radau et al., 2009). **Blue** - Initial contour. **Yellow** - Final contour. **Green** - GT Mask.

Table 2: Quantitative results on the two mammographic datasets of INBreast (Moreira et al., 2012) and DDSM-BCRP (Heath et al., 1998). Reported results are the F1-Score.

Method	INBreast	DDSM-BCRP
Ball & Bruce (2007)	90.90	90.00
Zhu et al. (2018)	90.97	91.30
Li et al. (2018)	93.66	91.14
Singh et al. (2020)	92.11	-
Ours	**94.28**	**92.32**

Table 3: Results on the cardiac MR Left Ventricle segmentation dataset of SCD (Radau et al., 2009), F1-Score on the entire test set.

Method	F1-Score
Queirós et al. (2014)	0.90
Liu et al. (2016)	0.92
Avendi et al. (2016)	0.94
Ngo et al. (2017)	0.88
Ours	**0.95**

As can be seen in Tab. 1 our method significantly outperforms the baseline methods on both building datasets. Fig. 4 compares the results of our method with the leading method by Cheng et al. (2019).

4.2 Medical Imaging

We evaluate our method using two common mammographic mass segmentation datasets, the INBreast (Moreira et al., 2012), DDSM-BCRP (Heath et al., 1998), and a cardiac MR left ventricle segmentation datasets, the SCD (Radau et al., 2009). For the mammographic dataset, we follow previous work and use the expert ROIs, which were manually extracted, and the same train/test split as Zhu et al. (2018); Li et al. (2018). **INBreast dataset** consists of 116 accurately annotated masses, with mass size ranging from $15mm^2$ to $3689mm^2$. The dataset is divided into into 58 images for train and 58 images for test, as done in previous work. **DDSM-BCRP dataset** consists of 174 annotated masses, provided by radiologists. The dataset is divided into into 78 images for train and 5788 images for test, as done in previous work. **SCD dataset** The Sunnybrook Cardiac Data (SCD), the MICCAI 2009 Cardiac MR Left Ventricle Segmentation Challenge data, consist of 45 cine-MRI images from a mix of patients and pathologies. The dataset is split into three groups of 15, resulting in about 260 2D-images each, and report results for the endocardial segmentation.

Tab. 2 and 3 show that our method outperforms all baseline methods on the three medical imaging benchmarks. Sample results for our method are shown in Fig. 5.

(bike) (bus) (motorcycle) (person) (truck)

Figure 6: Sample results from the Cityscapes dataset.

Model	Bicycle	Bus	Person	Train	Truck	Motorcycle	Car	Rider	Mean
Polygon-RNN++ (with BS)	63.06	81.38	72.41	64.28	78.90	62.01	79.08	69.95	71.38
PSP-DeepLab	67.18	83.81	72.62	68.76	**80.48**	65.94	80.45	70.00	73.66
Polygon-GCN (with PS)	66.55	85.01	72.94	60.99	79.78	63.87	81.09	71.00	72.66
Spline-GCN (with PS)	67.36	**85.43**	73.72	64.40	80.22	64.86	81.88	71.73	73.70
Ours	**70.28**	85.31	**75.74**	**74.31**	79.72	**71.75**	**81.96**	**71.80**	**76.35**

Table 4: **Cityscapes dataset.** Quantitative results reported in mean IoU. **BS** indicates that the model uses beam search, **PS** indicates that the model train with explicit points supervision.

4.3 Street View Images

Following Ling et al. (2019), we employ the Cityscapes dataset (Cordts et al., 2016) to evaluate our model in the task of segmenting street images. The dataset consists of 5000 images, and the experiments employ the train/val/test split of Castrejon et al. (2017). Results are evaluated using the mean IoU metric.

The Cityscapes dataset, unlike the other datasets, contains single objects with multiple contours. Similarly to Curve-GCN (Ling et al., 2019), we first train our model on to segment single contours, given the patch that contains that contour. Then, after the validation error stops decreasing, we train the final network to segment all contours from the patch that contains the entire object. Note that the network outputs a single output contour for the entire instance. We follow the same augmentation process as presented for the buildings datasets, using an input resolution of 64×64.

We compare our method with previous work on semantic segmentation, including PSP-DeepLab (Chen et al., 2017), and the polygon segmentation methods Polygon-RNN++ (Acuna et al., 2018) and Curve-GCN (Ling et al., 2019).

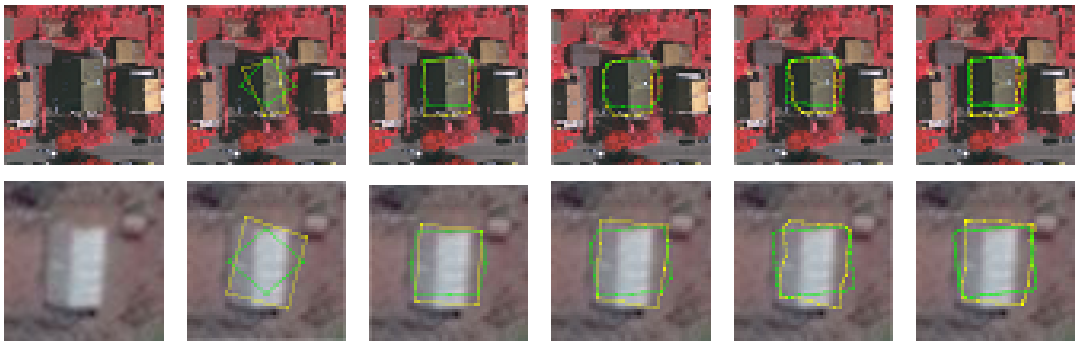
As can be seen in from Tab. 4, our model outperforms and all other methods in six out of eight categories, and achieves the highest average performance across classes by a sizable margin that is bigger than the differences between the previous contributions (the previous methods achieve average IoU of 71.38–73.30, we achieve 76.35) . Unlike Curve-Net (Ling et al., 2019), we do not use additional supervision in the form of explicit point location and edge maps. Sample results can be seen in Fig. 6.

4.4 Model Sensitivity

To evaluate the sensitivity of our method to the key parameters, we varied the number of nodes in the polygon and the number of iterations. Both parameters are known to effect active contour models.

Number of Vertices We experimented with different number of vertices, from simple to complex polygons - [4, 8, 16, 32, 64, 128]. In Fig. 8 - top row, we report the Dice and mIoU on all datasets, including results on DARNet (Cheng et al., 2019) on their evaluation datasets. As can be seen, segmenting with simple polygons yields lower performance, while as the number of vertices increases the performance quickly saturated at about 32 vertices. A clear gap in performance is visible between our method and DARNet (Cheng et al., 2019), especially with low number of vertices. Fig. 7 illustrates that gap on the two buildings datasets.

Number of Iterations The effect of the iterations number is show to be moderate, although a mean increase is seen over all datasets, saturating at about 3 iterations.



Input image 4-vertices 8-vertices 16-vertices 32-vertices 64-vertices

Figure 7: Varying number of vertices. ~~Green~~Yellow - Our method. ~~Yellow~~Green - DARNet (Cheng et al., 2019)

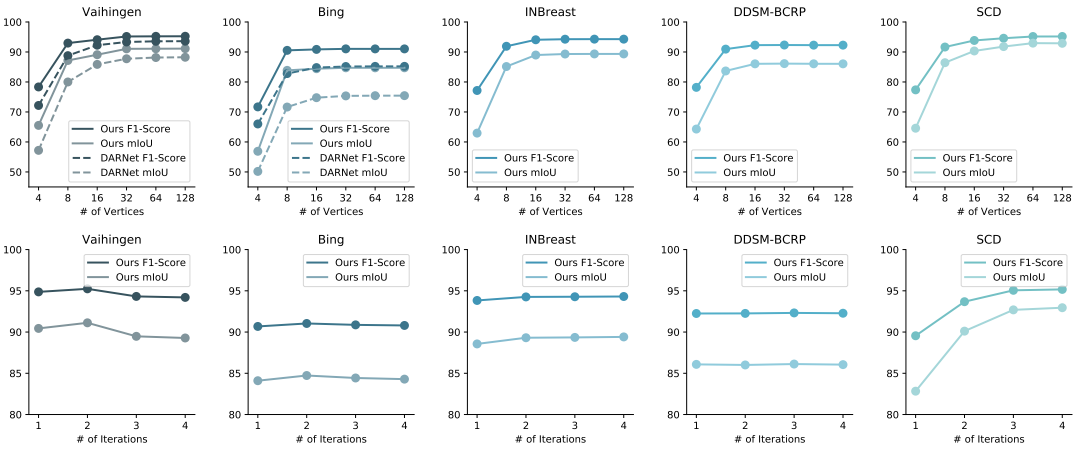


Figure 8: **Top** - different number of vertices, and **Bottom** - different number of iterations. Results for DARNet (Cheng et al., 2019) are available for the buildings datasets.

Training Image Resolution All of our models are trained at the resolution of $64 \times 64 pixels$.

We have experimented with different input resolutions on the Vaihingen dataset. As can be seen in Tab. 5, there is a steep degradation in performance below 64, which we attribute to the lack of details. When doubling the resolution to 128×128, the performance slightly degrades, however, that model could improve with further training epochs.

Ablation Study In Tab. 6 we show the effect of different loss combinations on our model performance on the Vaihingen benchmark. The compound loss $\mathcal{L}$ is better than its derivatives. Each the ballooning loss improves performance over not using auxiliary losses at all, while the curvature loss by itself does not. We note that even without ~~no~~ the auxiliary losses, with a single straightforward loss term, our method outperforms the state of the art.

5 Conclusions

Active contour methods that are based on a global neural network inference hold the promise of improving semantic segmentation by means of an accurate edge placement. We present a novel method, which could be the most straightforward active contour model imaginable. The method employs a recent differential renderer, without making any modifications to it, and simple MSE loss terms. The elegance of the model does not come at the expense of performance, and it achieves

Table 5: Evaluation of different image resolutions during training on the Vaihingen dataset.

Resolution	16	32	64	128
F1-Score	13.11	54.86	**95.62**	95.13
mIoU	7.13	39.50	**91.74**	90.85

Table 6: Evaluation of different loss combinations on the Vaihingen dataset.

Loss combination	$\mathcal{L}_{\text{SEG}}$	$\mathcal{L}_{\text{SEG}} + \mathcal{L}_{\mathcal{K}}$	$\mathcal{L}_{\text{SEG}} + \mathcal{L}_{\mathcal{B}}$	$\mathcal{L}$
F1-Score	94.94	94.80	95.13	**95.62**
mIoU	90.31	90.20	90.82	**91.74**

state of the art results on a wide variety of benchmarks, where in each benchmark, it outperforms the relevant deep learning baselines, as well as all classical methods.

REFERENCES

David Acuna, Huan Ling, Amlan Kar, and Sanja Fidler. Efficient interactive annotation of segmentation datasets with polygon-rnn++. 2018.

MR Avendi, Arash Kheradvar, and Hamid Jafarkhani. A combined deep-learning and deformable-model approach to fully automatic segmentation of the left ventricle in cardiac mri. *Medical image analysis*, 30:108–119, 2016.

Min Bai and Raquel Urtasun. Deep watershed transform for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5221–5229, 2017.

John E Ball and Lori Mann Bruce. Digital mammographic computer aided diagnosis (cad) using adaptive level set segmentation. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4973–4978. IEEE, 2007.

Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1):61–79, 1997.

Lluıs Castrejón, Kaustav Kundu, Raquel Urtasun, and Sanja Fidler. Annotating object instances with a polygon-rnn. In *CVPR*, 2017.

Lluis Castrejon, Kaustav Kundu, Raquel Urtasun, and Sanja Fidler. Annotating object instances with a polygon-rnn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5230–5238, 2017.

Tony F Chan and Luminita A Vese. Active contours without edges. *IEEE Transactions on image processing*, 10(2):266–277, 2001.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4): 834–848, 2017.

Dominic Cheng, Renjie Liao, Sanja Fidler, and Raquel Urtasun. DARNet: Deep active ray network for building segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7431–7439, 2019.

Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pp. 424–432. Springer, 2016.

Laurent D Cohen. On active contour models and balloons. *CVGIP: Image understanding*, 53(2): 211–218, 1991.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

Joachim Denzler and Heinrich Niemann. Active rays: Polar-transformed active contours for real-time contour tracking. *Real-Time Imaging*, 5(3):203–213, 1999.

S. M. Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S. Morcos, Marta Garnelo, Avraham Ruderman, Andrei A. Rusu, Ivo Danihelka, Karol Gregor, David P. Reichert, Lars Buesing, Theophane Weber, Oriol Vinyals, Dan Rosenbaum, Neil Rabinowitz, Helen King, Chloe Hillier, Matt Botvinick, Daan Wierstra, Koray Kavukcuoglu, and Demis Hassabis. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018.

Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh r-cnn. In *International Conference on Computer Vision (ICCV)*, 2019.

Michael Heath, Kevin Bowyer, Daniel Kopans, P Kegelmeyer, Richard Moore, Kyong Chang, and S Munishkumaran. Current status of the digital database for screening mammography. In *Digital mammography*, pp. 457–460. Springer, 1998.

Ping Hu et al. Deep level sets for salient object detection. In *CVPR*, 2017.

Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. In *Advances in Neural Information Processing Systems*, pp. 2802–2812, 2018.

Pascal Kaiser, Jan Dirk Wegner, Aurélien Lucchi, Martin Jaggi, Thomas Hofmann, and Konrad Schindler. Learning aerial image segmentation from online maps. *IEEE Transactions on Geoscience and Remote Sensing*, 55(11):6054–6068, 2017.

Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3907–3916, 2018.

Satyanad Kichenassamy, Arun Kumar, Peter Olver, Allen Tannenbaum, and Anthony Yezzi. Gradient flows and geometric active contour models. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 810–815. IEEE, 1995.

Youngeun Kim et al. CNN-based semantic segmentation using level set loss. In *WACV*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Heyi Li, Dongdong Chen, William H Nailon, Mike E Davies, and David Laurenson. Improved breast mass segmentation in mammograms with conditional residual u-net. In *Image Analysis for Moving Organ, Breast, and Thoracic Images*, pp. 81–89. Springer, 2018.

Huan Ling, Jun Gao, Amlan Kar, Wenzheng Chen, and Sanja Fidler. Fast interactive object annotation with curve-gcn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5257–5266, 2019.

Yu Liu, Gabriella Captur, James C Moon, Shuxu Guo, Xiaoping Yang, Shaoxiang Zhang, and Chunming Li. Distance regularized two level sets for segmentation of left and right ventricles from cine-mri. *Magnetic resonance imaging*, 34(5):699–706, 2016.

Matthew M Loper and Michael J Black. Opendr: An approximate differentiable renderer. In *European Conference on Computer Vision*, pp. 154–169. Springer, 2014.

Diego Marcos, Devis Tuia, Benjamin Kellenberger, Lisa Zhang, Min Bai, Renjie Liao, and Raquel Urtasun. Learning deep structured active contours end-to-end. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8877–8885, 2018.

Pablo Marquez-Neila, Luis Baumela, and Luis Alvarez. A morphological approach to curvature-based evolution of curves and surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):2–17, 2014.

Oleg Michailovich, Yogesh Rathi, and Allen Tannenbaum. Image segmentation using active contours driven by the bhattacharyya gradient flow. *IEEE Transactions on Image Processing*, 16(11): 2787–2801, 2007.

Inês C Moreira, Igor Amaral, Inês Domingues, António Cardoso, Maria Joao Cardoso, and Jaime S Cardoso. Inbreast: toward a full-field digital mammographic database. *Academic radiology*, 19 (2):236–248, 2012.

Tuan Anh Ngo, Zhi Lu, and Gustavo Carneiro. Combining deep learning and level set for the automated segmentation of the left ventricle of the heart from cardiac cine magnetic resonance. *Medical image analysis*, 35:159–171, 2017.

Sandro Queirós, Daniel Barbosa, Brecht Heyde, Pedro Morais, João L Vilaça, Denis Friboulet, Olivier Bernard, and Jan Dhooge. Fast automatic myocardial segmentation in 4d cine cmr datasets. *Medical image analysis*, 18(7):1115–1131, 2014.

P Radau, Y Lu, K Connelly, G Paul, A Dick, and G Wright. Evaluation framework for algorithms segmenting short axis cardiac mri. *The MIDAS Journal-Cardiac MR Left Ventricle Segmentation Challenge*, 49, 2009.

Franz Rottensteiner, Gunho Sohn, Jaewook Jung, Markus Gerke, Caroline Baillard, Sbastien Bnitez, and U Breitkopf. International society for photogrammetry and remote sensing, 2d semantic labeling contest. http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html. URL `http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html`.

Christian Rupprecht, Elizabeth Huaroc, Maximilian Baust, and Nassir Navab. Deep active contours. *arXiv preprint arXiv:1607.05074*, 2016.

Vivek Kumar Singh, Hatem A Rashwan, Santiago Romani, Farhan Akram, Nidhi Pandey, Md Mostafa Kamal Sarker, Adel Saleh, Meritxell Arenas, Miguel Arquez, Domenec Puig, et al. Breast tumor segmentation and shape classification in mammograms using generative adversarial and convolutional neural network. *Expert Systems with Applications*, 139:112855, 2020.

Vadim N Smelyansky, Robin D Morris, Frank O Kuehnel, David A Maluf, and Peter Cheeseman. Dramatic improvements to feature based stereo. In *European Conference on Computer Vision*, pp. 247–261. Springer, 2002.

Xiaolu Sun, C Mario Christoudias, and Pascal Fua. Free-shape polygonal object localization. In *European Conference on Computer Vision*, pp. 317–332. Springer, 2014.

Oliver Wang, Suresh K Lodha, and David P Helmbold. A bayesian approach to building footprint extraction from aerial lidar data. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pp. 192–199. IEEE, 2006.

Shenlong Wang, Min Bai, Gellert Mattyus, Hang Chu, Wenjie Luo, Bin Yang, Justin Liang, Joel Cheverie, Sanja Fidler, and Raquel Urtasun. Torontocity: Seeing the world with a million eyes. *arXiv preprint arXiv:1612.00423*, 2016.

Yue Wang, Eam Khwang Teoh, and Dinggang Shen. Lane detection and tracking using b-snake. *Image and Vision computing*, 22(4):269–280, 2004.

Zian Wang, David Acuna, Huan Ling, Amlan Kar, and Sanja Fidler. Object instance annotation with deep extreme level set evolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7500–7508, 2019.

Paul A Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C Gee, and Guido Gerig. User-guided 3d active contour segmentation of anatomical structures: significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128, 2006.

Wentao Zhu, Xiang Xiang, Trac D Tran, Gregory D Hager, and Xiaohui Xie. Adversarial deep structured nets for mass segmentation from mammograms. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 847–850. IEEE, 2018.