# DDL: Deep Deformable Learning for Image Segmentation

**Feiyang Chen**
University of California, Los Angeles
fychen@cs.ucla.edu

**Arjun Kallapur**
University of California, Los Angeles
arjunk@ucla.edu

**Nischal Chandra**
University of California, Los Angeles
nrchandra@ucla.edu

**Sanjeev Venkatesan**
University of California, Los Angeles
sanjeev2@ucla.edu

**Tianyi Xie**
Univeristy of California, Los Angeles
tianyixie77@ucla.edu

**Vaibhav Kumar**
University of California, Los Angeles
vaibhavk@ucla.edu

## Abstract

Image segmentation is a computer vision task of extracting pixel-wise mask segments of objects for an image. While modern Deep Learning-based models have achieved great success in various image segmentation tasks such as scene understanding, medical image analysis, applications for driver-less cars and more, they can still be improved in terms of sheer performance, model explainibility and computational complexity. In this work we will explore the effectiveness of deep learning based models, deformable models and finally the combination of the two over various Image Segmentation datasets. We aim to combine the classical geometry and physics based approach of deformable models with modern large-scale advances in deep learning in order to propose a novel model architecture. We will finally perform an empirical analysis of various models on a suite of datasets and metrics to do a comparative case study.

## 1 Introduction

Image segmentation is a computer vision task of assigning pixels in images into sets. In this paper, we will largely focus on instance segmentation, which is the task of accurately delineating objects in images, and semantic segmentation, which is the per-pixel class labeling of an image [1].

Image segmentation is a fundamental problem in computer vision as it closely resembles how human vision works, and hence there have been concerted decade-long efforts to help machines analyze object shape and motion in real time, just as normal humans "see". One early approach was a family of mathematical models known as "deformable models". These models were motivated by the idea of unifying representations of shape and motion by combining geometry and physics. In their simplest form, these models often combine spline geometry and elastic dynamics by seeking to minimize the energy of a curve in a plane. After their introduction in the 1980s, these models became commonly used following demonstrations of their accuracy [2]

With improvements in hardware capabilities, deep learning approaches are the new state of the art. As we will explore later, these models are able to capture spatially localized features which make them powerful in tasks such as image segmentation.

Preprint. Under review.

In this project, we will be applying deformable model approaches, deep learning approaches, and combined approaches to image segmentation. The specific application we will be focusing on is medical image segmentation. Medical images present interesting application of image segmentation algorithms, particularly in problems such as lesion detection and segmentation. These algorithms can be used to automate stages the diagnoses of diseases, and are hence of practical importance [3]

## 2 Related Work

### 2.1 Deformable Models

The classical deformable model is usually based on image pre-processing to eliminate many limitations of manual slice editing and traditional image processing techniques [4]. The deformable contour model [5] is the primary application of deformable models in medical image analysis, such as snakes [6] to segment structures in 2D images, T-snakes [7] to conduct topological transformations without additional machinery, live-wire [8, 9] to track the boundary through computing and selecting optimal boundaries at interactive rates as the user moves a mouse. These methods treat object boundaries as connected and continuous geometric complete models, the prior knowledge of the shape of the object can be used to constrain the segmentation problem.

Morphological Snakes[10] are yet another family of methods which have proven instrumental for image segmentation. In terms of method, they mimic the popular methodologies Active Contours like Geodesic Active Contours [11].While Active Contour approaches require a PDE, Morphological Snakes use morphological operators -such as dilation or erosion- over a binary array. This makes Morphological Snakes faster and numerically more stable than their traditional counterparts.

### 2.2 Deep Learning

Deep learning approaches to computer vision problems have become increasingly common in recent years. Typically, deep learning architectures for semantic segmentation consist of an encoder network followed by a decoder network. The encoder network often consists of a deep learning backbone architecture pre-trained on a dataset such as ImageNet.

U-Net [12] remains one of the most popular approaches due to its performance on segmentation tasks, particularly in the medical image analysis subspace. Other popular deep semantic segmentation architectures include region-based approaches such as R-CNN [13] and Mask R-CNN [14] and fully convolutional approaches such as DeepLab [15]. Due to compute limitations and to avoid comparing against approaches that have access to external data for pre-training purposes, we will compare our approach against U-Net as a deep learning baseline.

### 2.3 Combined Approaches

More recently, approaches to medical image segmentation that combine elements of deep learning and deformable models have been explored. One such approach, DALS (Deep Active Lesion Segmentation) [16] was able to demonstrate favorable results when compared with other methods. The motivation behind this approach was to combine the segmentation boundary precision of deformable models with the non-linear feature extraction capabilities of deep learning based models.

## 3 Classical Deformable Models

While the goal of this paper of implement a novel Deep deformable model, we also tried visualising the results specifically for **Chan-Vese** [17] and **Geodesic Active contours** [18] using Morphological snakes.

**Chan-Vese** segmentation algorithm helps segment objects with no clearly defined boundaries. This algorithm is based on level sets that evolve iteratively and minimize the energy function. This is defined by weights which correspond to summation of differences which is defined by weighted values corresponding to the sum of differences intensity from the average value outside the segmented region, the sum of differences from the average value inside the segmented region, and a term which

is dependent on the length of the boundary of the segmented region. It relies on three hyperparameters being *lambda1, lambda2* and *nu*.

While Chan-vese performs very well for regions that are characterized by quite different global means, it has limited power in discriminating regions with nearly the same global intensity distribution .

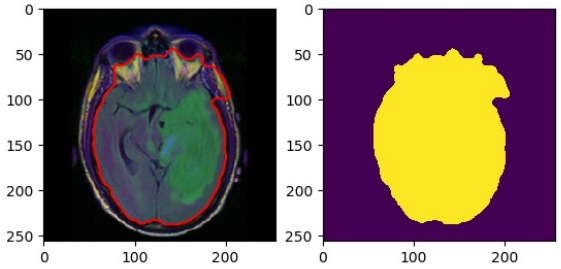


Figure 1: While the green region is expected to be segmented, chan-vese segments the entire brain.

Another approach implemented is using the **geodesic active contours**. Inspired from the classicial approach of snakes, This geodesic approach for object segmentation allows to connect classical **snakes** based on energy minimization and geometric active contours based on the theory of curve evolution.

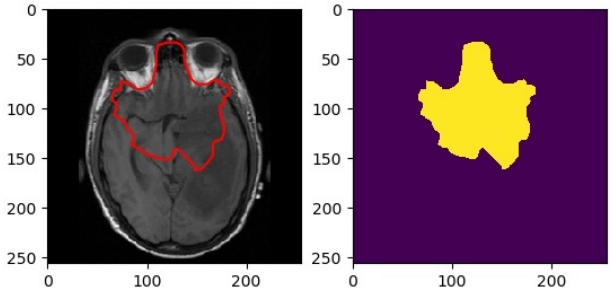


Figure 2: Geodesic contours fails to segment the nerve on right side of the brain.

Given the varying intensity of each images, it is necessary to incorporate deep learning based approach along with the exploitation of deformable algorithms.

## 4 Deep Learning Models

### 4.1 U-Net

Similar to most deep semantic segmentation architectures, U-Net consists of sequential encoder and decoder networks. The convolutional layers in the encoder network reduce spatial information while increasing feature information depth, while the convolutional layers in the decoder network increase spatial information while decreasing feature information, resulting in class probabilities per pixel in the final output layer. Notably, the U-Net architecture also contains skip connections between layers in the encoder and decoder networks, allowing the decoder network to reuse feature information from the encoder network.

### 4.2 TranResUNet

A model which is an improvement on the U-Net [12], TransResUNet[19] is a new fully convolutional encoder-decoder model.This architecture comprises of a pre-trained encoder, a special skip connection and a post-processing module are also included in the proposed architecture. This architecture was

designed and evaluated on lung-segmentation data. The flow of the architecture follows 4 steps primarily:

- Pre-trained Encoder: Given that the weights of higher layers are rather dedicated to specialized feature extraction for the target task, the first 7 layers of VGG16 are selected. Additionally an image (grayscale) is converted into color followed by convolutional layers and maxpooling.
- Skip-Connection: The convolutional blocks take part in skip-connection. In UNet, features originating from the encoder are primitive but decoder are more advanced due to more processing which leads to a semantic gap. This is primarily one of the reasons to use Conv blocks. This skip-connection plays a role in closing the gap and thus enabling the model learn effectively.
- Decoder: Includes 3 decoder blocks with an up-sampling. Features of the last convolution layer in a decoder block are concatenated with the corresponding encoded features from its residual connection. Finally, the last layer of the decoder produces the predicted output mask followed by a convolution and sigmoid activation.
- Postprocessing: Final blocks which helps reduce the noise in images by using the flood fill algorithm.

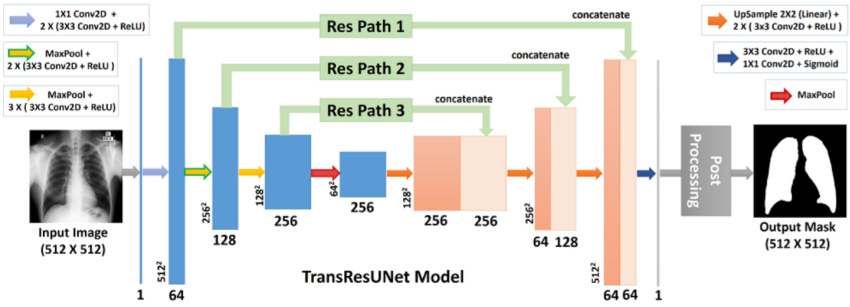


Figure 3: The model architecture of TransResUNet [19].

## 5 Deep Deformable Models

### 5.1 Deformable U-Net

Deformable U-Net is based on original U-Net architecture while replacing standard convolution with deformable convolution, including deformable convolution v1 [20] and v2 [21]. For pooling operations, we replace fixed kernel average/max pooling with learned deformable kernels with offsets. The reason for replacing standard convolution and pooling with deformable is some drawbacks of CNNs: 1). the geometric transformations are assumed fixed and known. Such prior knowledge is used to augment the data and design the features and algorithms. This assumption prevents generalization to new tasks possessing unknown geometric transformations, which are not properly modeled; 2). handcrafted design of invariant features and algorithms could be difficult or infeasible for overly complex transformations, even when they are known. Therefore, we introduce these two new modules that greatly enhance CNNs' capability of modeling geometric transformations. The core idea of deformable convolution and deformable pooling is adding offsets to get adaptive part localization for objects with different shapes. More importantly, both modules are lightweight, which adds a small number of parameters and computation for offset learning. They can easily be embedded in the existing models (such as CNNs) without additional supervision signals and are suitable for end-to-end learning.

#### 5.1.1 Deformable Convolution

Following [20]'s work, deformable convolution consists of two steps: 1) sampling using a regular grid $R$ over the input feature map $x$, which defines the receptive field size and dilation, such as,

$\mathcal{R} = \{(-1,-1),(-1,0),\ldots,(0,1),(1,1)\}$; 2) summation of sampled values weighted by $w$. As shown in formula 1, for each location $p_0$ on the output feature map $y$, we have $y(p_0)$, where $p_n$ enumerates the locations in $R$. Compared to standard convolution, deformable convolution adds offsets $\{\Delta\mathbf{p}_n \mid n = 1,\ldots,N\}$ ($N = |\mathcal{R}|$) to the regular grid sampling locations, which makes free form deformation of the sampling grid. Therefore, formula 1 transforms formula 2. The offsets $\Delta p_n$ are learned from the preceding feature maps, via additional convolutional layers. Thus, the deformation is conditioned on the input features in a local, dense, and adaptive manner. During training, both the convolutional kernels for generating the output features and the offsets are learned simultaneously. To learn the offsets, the gradients are backpropagated through the bilinear operations.

$$\mathbf{y}\left(\mathbf{p}_0\right) = \sum_{\mathbf{p}_n \in \mathcal{R}} \mathbf{w}\left(\mathbf{p}_n\right) \cdot \mathbf{x}\left(\mathbf{p}_0 + \mathbf{p}_n\right) \tag{1}$$

$$\mathbf{y}\left(\mathbf{p}_0\right) = \sum_{\mathbf{p}_n \in \mathcal{R}} \mathbf{w}\left(\mathbf{p}_n\right) \cdot \mathbf{x}\left(\mathbf{p}_0 + \mathbf{p}_n + \Delta\mathbf{p}_n\right) \tag{2}$$

Based on deformable convolution v1, [21] proposed a new version called deformable convolution v2 to boost the network's modeling power and to help it take advantage of this increased capability, mainly including three changes: 1). stacking more deformable Conv layers; 2). modulated deformable modules; 3). R-CNN feature mimicking. In this paper, we mainly utilized the first two improvements.

#### 5.1.2 Deformable Pooling

Similar to deformable convolution, deformable pooling also adds an offset to each bin position in the regular bin partition of the original RoI pooling [22], which are learned from the preceding feature maps and the RoIs, enabling adaptive part localization for objects with different shapes. As shown in formula 3 and 4, for $(i,j)$-th bin $(0 \le i,j < k)$, we get $y(i,j)$, where $n_{ij}$ is the number of pixels in the bin, and offsets $\{\Delta\mathbf{p}_{ij} \mid 0 \le i,j < k\}$ are added to the spatial binning positions by bilinear interpolation.

$$\mathbf{y}(i,j) = \sum_{\mathbf{p} \in \mathrm{bin}(i,j)} \mathbf{x}\left(\mathbf{p}_0 + \mathbf{p}\right) / n_{ij} \tag{3}$$

$$\mathbf{y}(i,j) = \sum_{\mathbf{p} \in \mathrm{bin}(i,j)} \mathbf{x}\left(\mathbf{p}_0 + \mathbf{p} + \Delta\mathbf{p}_{ij}\right) / n_{ij} \tag{4}$$

### 5.2 Improvements of Loss Functions

Loss function plays a significant role in training a neural network. In the task of image segmentation, Cross Entropy and Dice Coefficient are two widely used loss functions. However, both of them are pixel-wised loss functions and do not take geometric information into consideration. To remedy this defect, [23] proposes the Active Contour Loss, which is inspired by the general idea of classical active contour models. Here, we will briefly introduce these loss functions. In the following equations, the ground truth image and the prediction are denoted as $T, P \in [0,1]^{m \times n}$ respectively, where $m$ denotes the number of classes and $n$ denotes the index of pixel in image spatial space $N$.

**Cross-Entropy (CE) Loss:** CE is a widely used pixel-wise measure to evaluate the performance of the model among various classification tasks. Particularly, for binary classification problems, CE loss can expressed as BCE loss as follows:

$$Loss_{BCE}(T,P) = -\frac{1}{N}\sum_{n=1}^{N}\left[T_n \cdot \log\left(P_n\right) + \left(1 - T_n\right) \cdot \log\left(1 - P_n\right)\right] \tag{5}$$

where $T_n$ denotes the value of $n$-th pixel and the same is with $P_n$. In this sense, CE loss treat each pixel as an independent classification task.

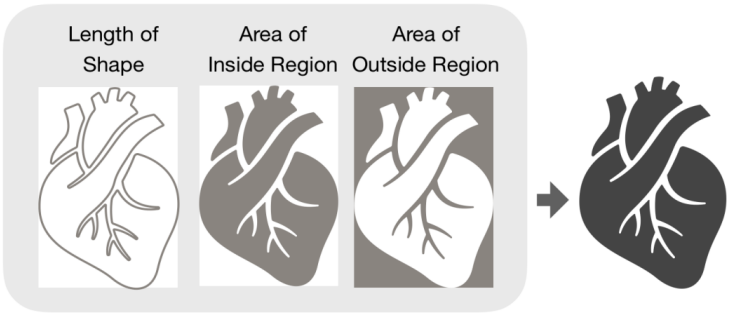


Figure 4: An example from Active Contour Loss [23], which takes the object's area and length of the boundaries in to account.

**Dice Coefficient (DC) Loss:** Previously, DC is used as a metric for the evaluation of the segmentation performance but now has also been demonstrated to be a good loss function [24]. Intuitively, DC measures the degree of overlapping between the ground truth and the prediction. The DC loss is defined as follows:

$$Loss_{DC}(T, P) = 1 - 2 \cdot \frac{\sum_{n=1}^{N} (T_n \times P_n)}{\sum_{n=1}^{N} (T_n + P_n)} \tag{6}$$

From Eq.(6), we can see the second term of DC loss is very similar to the definition of F1-score. Even though CE and DC loss functions have achieved a success in image segmentation, both of them are pixel-wised loss functions to measure the similarity between $T$ and $P$, but the geometrical information are not taken into consideration.

**Active Contour (AC) Loss:** Inspired by the general idea of classical active contour models, [23] proposes the Active Contour Loss, whose objective is to find an active contour which is a global minimization of active contour energy for automated image segmentation. In the following equations, we will denote the ground truth and the prediction as $v, u \in [0, 1]^{m \times n}$ respectively. The AC loss is defined as follows:

$$Loss_{AC} = \lambda \cdot Length + Region \tag{7}$$

where,

$$Length = \int_C |\nabla u| ds \tag{8}$$

$$Region = \int_\Omega \left( (c_1 - v)^2 - (c_2 - v)^2 \right) u dx \tag{9}$$

Since pixels in a image is not continuous, we therefore need to discretize the AC loss. The Eq.(8) and Eq.(9) can be written into pixel-wised way respectively as follows:

$$Length = \sum_{\Omega}^{i=1,j=1} \sqrt{\left| \left( \nabla u_{x_{i,j}} \right)^2 + \left( \nabla u_{y_{i,j}} \right)^2 \right| + \epsilon} \tag{10}$$

where $\epsilon$ is a hyperparameter used to avoid the square root is zero and is set as $10^{-6}$ in practice.

$$Region = \left| \sum_{\Omega}^{i=1,j=1} u_{i,j} \left( c_1 - v_{i,j} \right)^2 \right| + \left| \sum_{\Omega}^{i=1,j=1} \left( \mathbf{1} - u_{i,j} \right) \left( c_2 - v_{i,j} \right)^2 \right| \tag{11}$$

where $c_1$ and $c_2$ represent the energy of foreground and background respectively and, in practice, can be simply defined as $c_1 = 1$ and $c_2 = 0$. In this case, what the $Region$ term essentially calculates is

the Mean Square Error (MSE). However, MSE loss sometimes will cause vanishing gradient problem in the binary classification problem. Therefore, we replace $Region$ term with CE loss and consider $Length$ term as an extra regularization. By introducing the $length$ term, AC loss takes geometrical information into consideration. Figure 4 shows the visualization of AC loss.

# 6 Experiments

In this section, we comprehensively compare various image segmentation methods, including classical deformable models, deep learning models, and deep deformable models. Furthermore, we also conduct a comparison with different loss functions that are used commonly in the image segmentation task.

## 6.1 Dataset and Experimental Setting

We mainly use the Brain MRI dataset[1] for our model evaluation. This dataset contains brain MR images together with manual FLAIR abnormality segmentation masks. To goal of the model is to detect the tumor given a brain MRI image. For deep learning based method, we fix the training batch size as 16 and set the learning rate as $3e^{-4}$. We train each model for 30 epochs and select the one that performs best in the validation set for evaluation. Note we resize the image size to $64 \times 64$ to reduce training time.

## 6.2 Evaluation Metrics

**Intersection over Union (IoU):** IoU is one the most commonly used metrics in semantic segmentation. It is defined as the area of intersection between the predicted segmentation map and the ground truth, divided by the area of union between the predicted segmentation map and the ground truth:

$$\text{IoU} = J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{12}$$

where $A$ and $B$ denote the ground truth and the predicted segmentation, in which the value ranges from 0 to 1.

**Dice Coefficient:** Dice is another popular metric for image segmentation, which can be defined as twice the overlap area of the predicted and the ground truth, divided by the total number of pixels in both images:

$$\text{Dice } = \frac{2|A \cap B|}{|A| + |B|} \tag{13}$$

## 6.3 Quantitative Evaluation

As is shown in Table 1, we compare deep learning models (U-Net and TranResUNet) and deep deformable models (Deformable U-Net) with different loss functions (Cross-Entropy, Dice Coefficient, and Active Contour). The reason why we did not compare with the classical deformable models is the limitation of computing resources. It would be very time-consuming to run the classical deformable methods on the complete data set, so we only compared the qualitative results, which will be shown in the next section. From the Table 1, for the deep learning method, the model using the Active Contour loss function has achieved better results, which proves the effectiveness of the combination of deformable loss and deep learning methods. Although Deformable U-Net did not achieve the results we expected, we assume that this is because 30 epochs has not fully converged the model. Because the deformable convolution needs more calculations to compute the offsets, which slows down the training speed. We believe that if we have enough computing resources and time, we can get better results.

[1] https://www.kaggle.com/mateuszbuda/lgg-mri-segmentation

| Model Backbone | Loss | Train Dice | Test Dice | Train IoU | Test IoU |
|---|---|---|---|---|---|
| U-Net | Cross-Entropy | 0.866 | 0.881 | 0.769 | 0.789 |
| | Dice Coefficient | 0.864 | 0.865 | 0.761 | 0.762 |
| | Active Contour | 0.876 | **0.890** | 0.786 | **0.804** |
| TransResUNet | Cross-Entropy | 0.870 | 0.870 | 0.780 | 0.770 |
| | Dice Coefficient | 0.811 | 0.816 | 0.695 | 0.697 |
| | Active Contour | 0.880 | **0.880** | 0.789 | **0.790** |
| Deformable U-Net | Cross-Entropy | 0.706 | **0.734** | 0.546 | **0.580** |
| | Dice Coefficient | 0.678 | 0.695 | 0.513 | 0.532 |
| | Active Contour | 0.644 | 0.722 | 0.498 | 0.569 |

Table 1: Quantitative comparison results of different deep learning based methods with different loss functions.

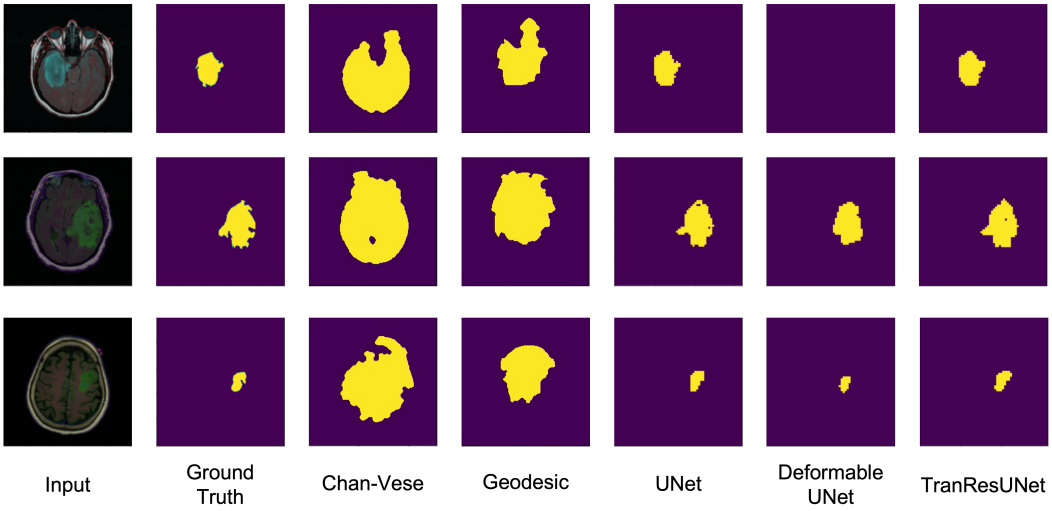


Figure 5: Comparison results of classical deformable models and deep learning based methods. Chan-Vese and Geodesic fail to predict the segmentation of tumor but outline the brain instead. In contrast, supervised neural networks are able to detect the tumor much more accurately.

### 6.4 Qualitative Evaluation

#### 6.4.1 Comparison with baseline methods

Our baseline methods include classical deformable models (Chan-Vese and Geodesic Active Coutours), deep learning models (U-Net and TranResUNet), and deep deformable models (Deformable U-Net). Note we only use Cross Entropy Loss for deep learning methods for this comparison. As is shown in Figure 5, we compared the performance of different models. Since both Chan-Vese and geodesic active coutours predict the segmentation based on the image intensity, they usually simply outline the brain region instead of segmenting the tumor given only the RGB information. In this case, the neural network is a much better choice. After acquiring enough knowledge through the training process, the network has the ability to tell where is the tumor. According to visualized results, U-Net and TranResUNet outperform other methods while the predicted result of Deformable U-Net is not so accurate and sometimes it even fails to detect the tumor. Note the boundary of predicted results of UNet family looks not smooth because we resize the input image from $256 \times 256$ to $64 \times 64$.

#### 6.4.2 Comparison with different loss functions

We then conduct comparison study of loss functions, including Cross Entropy (CE) Loss, Dice Coefficient (Dice) Loss as well as Active Contour (AC) Loss. As we can see in Figure 6, there are some holes inside the segmentation result of DC loss. Similarly, in the predicted map of CE loss, there are also some areas with low degree of confidence. This can be attributed to the drawback of


Figure 6: Predicted results of UNet when using different training loss functions: Cross Entropy (CE) Loss, Dice Coefficient (DC) loss and Active Contour (AC) Loss.

pixel-wise loss functions. On the contrary, AC loss, which takes the geometrical information into account, does not have such problem.

$$x = x + 5 * \frac{x^2}{ec} \tag{14}$$

$$replylikelihood = \frac{x}{x+5} \tag{15}$$

$$aggr = abs(upvote) + contro * upvote \tag{16}$$

$$aggr = aggr + 10 * \frac{x^2}{ec} \tag{17}$$

$$behavioral = \frac{aggr}{aggr + 10} \tag{18}$$

$$EC = abs(upvotes) + controversy + replycount \tag{19}$$

$$emotionaleng = abs(sentiments - 0.5)^2 \tag{20}$$

$$score = w1 * re + w2 * ae + w3 * be + w4 * ee \tag{21}$$

$$attention = min(\frac{100 * edit + longest}{300}, 1.0) \tag{22}$$

## 7 Conclusion and Future Work

We have demonstrated that deep deformable models perform decently when compared to state-of-the-art Deep Learning models. While hyperparameter tuning can boost the model performance, increasing the training samples can in turn produce better results. We also observe that in our experiments, Vanilla networks generally perform better than the transformed deformable networks for the same number of epochs and training data. This can be explained by the fact that training deformable convolutional layers is computationally more expensive than convolutional layers, we believe that

more training epochs and larger image size could yield better results for deformable UNet-like models. Due to our computational restraints, we limit our experiments for now but leave the exploration for future. For further future work, Medical Imaging still remains one of the most important fields where the applications of vision models are widely used. Given the models that have been implemented over the decades being CNN primarily, transformer-based models have evolved and proven instrumental in not only in text but vision-based tasks as well. A few models which have proven effective in image segmentation/registration tasks include ViT-V-Net and voxelmorph [25].

## References


[1] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag, Berlin, Heidelberg, 1st edition, 2010.

[2] Andrew Blake and Michael Isard. *Active contours: the application of techniques from graphics, vision, control theory and statistics to visual tracking of shapes in motion*. Springer Science & Business Media, 2012.

[3] Emmanuel Montagnon, Milena Cerny, Alexandre Cadrin-Chênevert, Vincent Hamilton, Thomas Derennes, André Ilinca, Franck Vandenbroucke-Menu, Simon Turcotte, Samuel Kadoury, and An Tang. Deep learning workflow in radiology: a primer. *Insights into Imaging*, 11(1):22, February 2020.

[4] Tim McInerney and Demetri Terzopoulos. Deformable models in medical image analysis: a survey. *Medical image analysis*, 1(2):91–108, 1996.

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

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

[7] Tim McInerney and Demetri Terzopoulos. T-snakes: Topology adaptive snakes. *Medical image analysis*, 4(2):73–91, 2000.

[8] William A Barrett and Eric N Mortensen. Interactive live-wire boundary extraction. *Medical image analysis*, 1(4):331–341, 1997.

[9] Alexandre X Falcao, Jayaram K Udupa, Supun Samarasekera, Shoba Sharma, Bruce Elliot Hirsch, and Roberto de A Lotufo. User-steered image segmentation paradigms: Live wire and live lane. *Graphical models and image processing*, 60(4):233–260, 1998.

[10] Pablo Márquez-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.

[11] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Proceedings of IEEE International Conference on Computer Vision*, pages 694–699, 1995.

[12] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation, 2014.

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018.

[15] 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. *CoRR*, abs/1606.00915, 2016.

[16] Ali Hatamizadeh, Assaf Hoogi, Debleena Sengupta, Wuyue Lu, Brian Wilcox, Daniel Rubin, and Demetri Terzopoulos. Deep active lesion segmentation. In Heung-Il Suk, Mingxia Liu, Pingkun Yan, and Chunfeng Lian, editors, *Machine Learning in Medical Imaging*, pages 98–105, Cham, 2019. Springer International Publishing.

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

[18] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Proceedings of IEEE International Conference on Computer Vision*, pages 694–699, 1995.

[19] Sakib Reza, Ohida Binte Amin, and M.M.A. Hashem. Transresunet: Improving u-net architecture for robust lungs segmentation in chest x-rays. In *2020 IEEE Region 10 Symposium (TENSYMP)*, pages 1592–1595, 2020.

[20] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[21] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019.

[22] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[23] Xu Chen, Bryan M Williams, Srinivasa R Vallabhaneni, Gabriela Czanner, Rachel Williams, and Yalin Zheng. Learning active contour models for medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11632–11640, 2019.

[24] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016.

[25] Guha Balakrishnan, Amy Zhao, Mert R. Sabuncu, John Guttag, and Adrian V. Dalca. Voxelmorph: A learning framework for deformable medical image registration. *IEEE Transactions on Medical Imaging*, 38(8):1788–1800, 2019.