



Deep Convolutional Level Set Method for Image Segmentation

Agustinus Kristiadi & Pranowo*

Department of Informatics Engineering, Universitas Atma Jaya Yogyakarta,
Jalan Babarsari 43, Yogyakarta 55281, Indonesia

*E-mail: pran@mail.uajy.ac.id

Abstract. Level Set Method is a popular method for image segmentation. One of the problems in Level Set Method is finding the right initial surface parameter, which implicitly affects the curve evolution and ultimately the segmentation result. By setting the initial curve too far away from the target object, Level Set Method could potentially miss the target altogether, whereas by setting the initial curve as general as possible – i.e. capturing the whole image – makes Level Set Method susceptible to noise. Recently, deep-learning methods, especially Convolutional Neural Network (CNN), have been proven to achieve state-of-the-art performance in many computer vision tasks such as image classification and detection. In this paper, a new method is proposed, called Deep Convolutional Level Set Method (DCLSM). The idea is to use the CNN object detector as a prior for Level Set Method segmentation. Using DCLSM it is possible to significantly improve the segmentation accuracy and precision of the classic Level Set Method. It was also found that the prior used in the proposed method is the lower and upper bound for DCLSM's precision and recall, respectively.

Keywords: *computer vision; convolutional neural network; deep learning; image processing; image segmentation; level set; machine learning; pattern recognition.*

1 Introduction

Image segmentation is a key task in computer vision. It is one of the oldest and most studied problems in this field [1]. Image segmentation plays an important role in understanding visual perception by intelligent systems, as an agent has to be able to localize and recognize entities in the real world. It has been widely implemented in many fields, for example robotics, autonomous vehicles [2], and medical imaging [3].

Image segmentation is still a wide-open and unsolved research area. Many algorithms have been developed to solve image segmentation problems, starting from simple methods, e.g. thresholding, to semantic segmentation using deep learning [4]. A popular method for solving image segmentation is Level Set Method, which is based on the Partial Differential Equation (PDE) that was originally developed as a numerical method for tracking interfaces and shapes

[5]. The idea behind Level Set Method is to represent the zero level set of a higher dimensional function as a curve. The benefit of Level Set Method is that complex curve evolutions, e.g. splitting and merging, can be represented naturally.

Level Set Method has gained a considerable amount of popularity in the computer vision field, especially in image segmentation [6,7] and has been applied successfully in medical image segmentation. However, in more challenging settings, such as natural images, thorough exploration of Level Set Method has not been done yet. One of the reasons is because natural images are much more complex and diverse compared to normalized images such as medical images, which in turn makes it harder to achieve good segmentation results [8].

The recent breakthrough of deep learning in computer vision has opened exciting research possibilities. Deep learning methods, especially Convolutional Neural Networks (CNN), have been successfully applied in image classification, object detection, and caption generation [9-13]. CNN has been so successful that the ImageNet ILSVRC [14] competition has been dominated by CNN submissions in the past years. This phenomenon is not without strong reason: CNN has much better performance compared to classical, shallow networks with handcrafted features. This is apparent in the comparison of the ILSVRC 2012 winner, AlexNet [9], with the second-placed submission.

In this paper, a novel model called the Deep Convolutional Level Set Method (DCLSM) is introduced. The main idea behind DCLSM is to use CNN trained with transfer-learning scheme as a prior for Level Set Method. The hypothesis is that by accurately predicting regions of interest inside the image, Level Set Method will perform better and more accurately. Specifically, the aim was higher segmentation accuracy and a better precision score compared to the classical Level Set Method without prior.

2 Related Work

Level Set Method, originally developed as a method for tracking interfaces [5], is a Partial Differential Equation (PDE) based method. Level Set Method has gained popularity in the computer vision community as a method to approach image segmentation problems [6]. In image segmentation problems, Level Set method is used to track the curve that detects the segmentation edge by evolving a higher-level function and representing the segmentation as the zero-level curve of that function.

Level Set Method for image segmentation has been successfully implemented in medical images such as MRI and CT scans to segment bladder walls [15], white brain matter [16], and lung nodules [17]. In other settings, such as satellite imaging, Level Set Method has been implemented to detect oil spills in the ocean [18]. Study of the application of Level Set Method on natural images has not been thoroughly conducted compared to application on medical images because of the much more challenging nature of natural images [8]. Although TouchCut [19] uses a natural images dataset for its evaluation, it is a semi-automatic method, as it uses an interactive user interface to manually guide Level Set Method to segment the desired object.

Accommodating prior knowledge into Level Set Method has been studied before. For instance, shape priors have been used to guide the segmentation process in [20]. Deep-learning based priors have also been studied. Ngo and Carneiro [3] used Deep Belief Network in combination with shape priors to initialize the parameters of Distance Regularized Level Set Evolution for left-ventricle segmentation. Cha, *et al.* [21] used CNN to estimate the likelihood of regions of interest inside a bladder. By using thresholding and hole filling, an initial contour is generated and further refined using Level Set Method. However, those methods were implemented for medical images, not natural images.

The idea of combining learning algorithms and classical methods such as Level Set Method has also been studied before. Li, *et al.* [22] proposed the use of Variational Level Set Method in conjunction with SVM for medical image segmentation. The Variational Level Set Method is used in the feature extraction pipeline to remove highly uncertain regions. Pawar and Talbar [23] also used Variational Level Set Method as a feature extractor before feeding the image into a feed forward neural network for classification. These previous works used Level Set Method as a prior for the learning method. By contrast, in the present work Deep Neural Network was used as prior for Level Set Method.

3 Deep Convolutional Level Set Method

Our segmentation model, called Deep Convolutional Level Set Method, which from this point on we shall address as DCLSM for the sake of brevity, is composed of two modules. The first module is a CNN, which we call Deep Convolutional Prior (DCP), which classifies and localizes the target object to be segmented. The output of this CNN is the prior of the next module. The second module is the Level Set Method (LSM) segmenter, where its initial parameter is conditioned to the prior. The overall architecture of DCLSM can be seen in Figure 1.

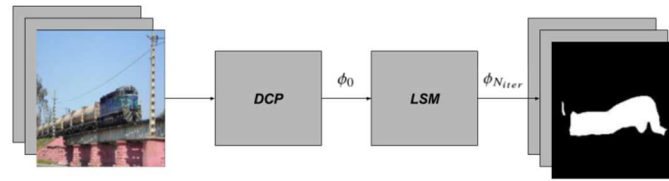


Figure 1 DCLSM overall architecture. The Deep Convolutional Prior (DCP) module is used to initialize the initial segmentation contour ϕ_0 for the Level Set Method (LSM) module. The segmentation mask derived from the latest segmentation contour $\phi_{N_{iter}}$ is the output of DCLSM.

3.1 Deep Convolutional Prior

Our first module, Deep Convolutional Prior (DCP), is a CNN-based prior. We approach our CNN as a network with two output branches: a classification branch and a regression branch. The classification branch is used to recognize the objects that are present in the image. The regression branch is used to predict the location of each object in terms of its bounding box.

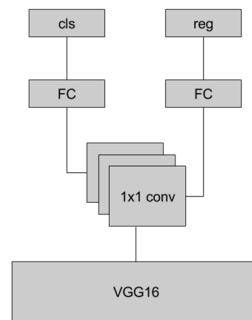


Figure 2 Deep Convolutional Prior (DCP) module architecture.

With recent studies showing the effectiveness of CNN in a transfer learning scheme, even without finetuning, we decided to use VGG16 [10] architecture as our base model, pre-trained on the ImageNet [14] dataset. This model won the classification and localization task in ILSVRC 2014. VGG16 consists of 13 convolutional layers and three fully connected (FC) layers. In our DCP, all of the FC layers are removed and only the convolutional layers are used. In other words, the VGG16 model is used as an offline feature extractor.

On top of VGG16 several layers are attached, as shown in Figure 2. Firstly, as the last convolutional layer of VGG16, it outputs a tensor with dimensions of $M \times 7 \times 7 \times 512$, with M is the minibatch size. A 1×1 convolutional layer is used to reduce the dimensionality to $M \times 7 \times 7 \times 128$, which is then reshaped into a $M \times 6272$ dimensional array.

At this point, the model branches into two FC-networks. The first network, DCP_{cls} , is composed of one FC layer with 256 hidden units and a softmax layer to predict 20 classes of the Pascal VOC dataset [24].

The second network is the localization network, DCP_{reg} . Similar to the classification network, a single FC layer is used, but instead of the softmax layer a regression layer is used on the very top. The regression layer's output is a four-dimensional vector for each dataset in the minibatch. The four-dimensional vector encodes the normalized bounding box of the predicted object, which consists of $B = \{x_{min}, y_{min}, x_{max}, y_{max}\}$, i.e. the location of the upper left and the lower right corner of the bounding box.

In both of the sub-networks, we use L2 regularizer and Dropout with probability of 0.5. For the activation function, ReLU nonlinearity is used.

As there are two output branches in the model, we have two different loss functions. For the classification branch, cross entropy loss is used Eq. (1) as follow:

$$L_{cls}(p_i, \hat{p}_i) = -\sum_j^C \hat{p}_{ij} \log p_{ij} \quad (1)$$

where C is the number of classes, p_i is the output probability of the softmax function for the i -th data point, and \hat{p}_i are the respective ground truth labels.

For the regression branch, as in [25], Huber loss is used, which is more robust to outliers than squared loss as shown in Eqs. (2) and (3) as follows:

$$L_{reg}(b_i, \hat{b}_i) = \sum_{j \in B} \text{huber}(b_{ij} - \hat{b}_{ij}) \quad (2)$$

where

$$\text{huber}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (3)$$

where b_i is the regression output, \hat{b}_i is the ground truth bounding box for the i -th data point, and B is the set of bounding box coordinates.

Then, the two losses are combined into a multi-task loss:

$$L(p_i, \hat{p}_i, b_i, \hat{b}_i) = \alpha L_{cls}(p_i, \hat{p}_i) + \beta L_{reg}(b_i, \hat{b}_i) \quad (4)$$

That is, the two sub-networks are trained jointly in a single training procedure rather than training them separately. α and β are hyperparameters to control the weight of each loss. In practice, we set α and β to be equal.

3.2 Level Set Method Segmenter

Level Set Method in image segmentation implicitly represents the segmentation contour as a surface. Specifically, the segmentation contour s is the zero level set of surface function ϕ , see Eq. (5) for example:

$$s = \{x \mid \phi(x) = 0\} \quad (5)$$

In particular, in DCLSM, Geodesic Active Contours (GAC) is used [26], which can be solved with the following PDE in Eq. (6):

$$\frac{\partial \phi}{\partial t} = g \|\nabla \phi\| \operatorname{div} \frac{\nabla \phi}{\|\nabla \phi\|} + g \|\nabla \phi\| v + \nabla g \cdot \nabla \phi \quad (6)$$

in which g are the image features, given by Eq. (7) as follow:

$$g(I, \alpha) = \frac{1}{1 + \alpha \|\nabla I\|^2} \quad (7)$$

where I is the smoothed image to be segmented and α is the parameter that controls the strength of the edge.

The GAC formulation above then can be solved with the following finite difference scheme in Eq. (8):

$$\phi_{t+1} = \phi_t + \Delta t \frac{\partial \phi}{\partial t} \quad (8)$$

Therefore, the GAC formulation of Level Set Method depends on several parameters, i.e. the initial surface ϕ_0 , stride parameter Δt , edge strength α , and balloon parameter v .

In this formulation, the choice of initial surface parameter ϕ_0 plays an important role in driving the curve evolution into a correct segmentation, as ϕ_0 implicitly signifies the initial position, size, and shape of the segmentation contour. By setting ϕ_0 randomly in terms of size and position inside the image, the Level Set Method could potentially miss the target object altogether. On the other hand, by setting ϕ_0 as general as possible, i.e. from the borders of the image, the Level Set Method would not be able to solely segment the target object if the image is sufficiently noisy or consists of several other objects.

Driven by that motivation, if we can provide Level Set Method with a prior for the location and size of the object, this could potentially increase the effectiveness of segmentation. Furthermore, by giving a location prior, Level Set Method could target a specific object inside the image, which in turn could reduce the noise of the segmentation result.

Therefore, DCLSM initial surface parameter ϕ_0 now is the following function in Eq. (9):

$$\phi_0^{ij} = \begin{cases} 1 & \text{if } (i, j) \text{ is outside } B \\ -1 & \text{otherwise} \end{cases} \quad (9)$$

where (i, j) is the image coordinate.

The full algorithm for DCLSM is shown in Figure 3.

Algorithm 1 DCLSM Segmentation

```

1: function DCLSMSEGMENT(Image,  $\Delta t$ ,  $v$ ,  $N_{iter}$ )
2:    $B \leftarrow \text{DCP}(\textit{Image})$ 
3:    $\phi_0 \leftarrow \text{InitializePhi}(\textit{Image}, B)$ 
4:   for  $i = 0$  to  $N_{iter}$  do
5:      $\phi_{i+1} \leftarrow \text{LevelSetEvolution}(\Delta t, v)$ 
6:   return  $\phi_{N_{iter}}$ 

1: function INITIALIZEPHI(Image, BoundingBox)
2:    $M \leftarrow \text{number of row in } \textit{Image}$ 
3:    $N \leftarrow \text{number of column in } \textit{Image}$ 
4:    $\phi \leftarrow 0_{M, N}$ 
5:   for  $i = 0$  to  $M$  do
6:     for  $j = 0$  to  $N$  do
7:       if  $(i, j)$  within BoundingBox then
8:          $\phi^{i, j} = -1$ 
9:       else
10:         $\phi^{i, j} = 1$ 
11:   return  $\phi$ 

```

Figure 3 DCLSM algorithm details.

3.3 Implementation Details

The DCP network was trained with a subset of the Pascal VOC 2012 dataset [24] with a single object in each image. The training set has 4460 images. The images were resized to 224 x 224 x 3 and the ground truth bounding box was normalized to make it scaling invariant.

Adam [27] was used for the optimization with default parameters. During training, the learning rate started at 10^3 and it was lowered by a factor of ten each time the training loss plateaued, for a total of 160 epochs.

The frameworks used were Keras and Theano. The computation platforms used were NVIDIA GTX980 and Intel Core i7 3770K.

4 Experiment Result

4.1 Evaluation Method

DCLSM was evaluated with a subset of the Pascal VOC 2012 dataset, containing 496 images. First, the test set was fed into the DCP network, giving the classification result and the bounding box prior. Then, the bounding box prior was fed into Level Set Method. The zero level curve of the latest evolution of surface parameter ϕ was the final segmentation contour.

To compare the segmentation result of the proposed method with the ground truth segmentation of the test set, the inside of the segmentation contour was filled. Therefore, a segmentation mask was used instead of a contour.

DCLSM was compared with two baseline methods: an uninformative segmentation prior, which initializes the segmentation mask from the borders of the image, and a segmentation mask, derived directly from the bounding box without Level Set segmentation. These methods will be addressed as LSM-baseline and BBox-baseline, respectively.

As our focus was on the initialization of a single parameter in Level Set Method, i.e. ϕ , we chose arbitrary values for the other parameters, i.e. $\nu=1$, $\alpha=10^5$, $\sigma=5$, and $N_{iter}=80$.

The metrics used for the evaluation were: classification accuracy, segmentation accuracy, precision, precision, recall, F1-score. Given \hat{s} and S , the predicted segmentation mask and ground truth segmentation mask, respectively, and \hat{y}

and y , the predicted and ground truth classification label, those metrics are computed in Eqs. (10) to (14) as follows:

$$cls_acc(\hat{y}, y) = \frac{1}{M} \sum_{i=1}^M 1(\hat{y}_i = y_i) \quad (10)$$

$$segm_acc(\hat{s}, s) = \frac{1}{RC} \sum_{i=1}^H \sum_{j=1}^W 1(\hat{s}_{ij} = s_{ij}) \quad (11)$$

$$prec(\hat{s}, s) = \frac{TP(\hat{s}, s)}{TP(\hat{s}, s) + FP(\hat{s}, s)} \quad (12)$$

$$rec(\hat{s}, s) = \frac{TP(\hat{s}, s)}{TP(\hat{s}, s) + FN(\hat{s}, s)} \quad (13)$$

$$F1(\hat{s}, s) = \frac{2 \, prec(\hat{s}, s) \, rec(\hat{s}, s)}{prec(\hat{s}, s) + rec(\hat{s}, s)} \quad (14)$$

where $TP(\hat{s}, s)$ is the number of true positives, $FP(\hat{s}, s)$ is the number of false positives, and $FN(\hat{s}, s)$ is the number of false negatives between \hat{s} and s . $1(\cdot)$ is an indicator function, H and W are the image dimensions.

4.2 Quantitative Analysis

The results of our quantitative evaluation are shown in Table 1.

Table 1 Evaluation of DCLSM performance compared to baselines.

| Model | Cls. Acc. | Segm. Acc. | Precision | Recall | F1-Score |
|---------------|-----------|---------------|---------------|---------------|---------------|
| LSM-baseline | - | 0.6566 | 0.4117 | 0.8358 | 0.4880 |
| BBox-baseline | 0.6694 | 0.6575 | 0.4128 | 0.6424 | 0.4045 |
| DCLSM | 0.6694 | 0.7598 | 0.5438 | 0.5083 | 0.4240 |

The baseline methods achieved a segmentation accuracy of around 0.65 for both LSM-baseline and BBox-baseline. In contrast, the proposed method, which incorporates a CNN prior into Level Set Method, yielded 0.7598 of segmentation accuracy. This result indicates 15.72% relative improvement compared to the baseline accuracy.

While DCLSM achieved the highest score in accuracy metrics, it achieved the lowest score in segmentation recall, less than LSM-baseline. By initializing ϕ

using an uninformative prior, i.e. always initializing from the borders of the image, LSM-baseline achieved a recall of 0.8358, much higher than any other method. By initializing ϕ at the borders of the image, Level Set Method would segment the image almost globally. Intuitively, by starting the curve evolution from the borders of the image, Level Set Method could trivially mask the whole image. In that case, the recall score would be perfect, as it is the trivial solution for getting a full recall score. This is why DCLSM achieved a lower recall score, as DCLSM is better at pinpointing the object.

As any method can yield a perfect recall score trivially, we focused on the precision measure. Inspecting the precision, LSM-baseline achieved the lowest precision score, 0.4117. This is because, intuitively, by covering many regions of the image, Level Set Method would cover more false positives.

On the other hand, in BBox-baseline, by using informative prior, i.e. the object bounding box but without refining it using Level Set Method, the false positive rate could be reduced, as the segmentation mask will not cover too many regions that could potentially be noisy. Hence, by incorporating the bounding box, the precision is marginally higher than by using an uninformative prior: 0.4128. However, the trade-off is that the recall score is now reduced by 23.14%.

Finally, DCLSM achieved significantly higher precision than LSM-baseline and BBox-baseline. Using DCLSM, 0.5438 for precision and 0.5083 for recall was achieved. In other words, the precision score was improved by 32% relative to LSM-baseline, while only trading off 13.11% from the F1-score. Overall, the proposed method yielded the most balanced results in both precision and recall.

The increased classification accuracy was only achieved by BBox-baseline and DCLSM, not in LSM-baseline, as LSM-baseline does not incorporate the CNN prior. Hence, LSM-baseline – like vanilla Level Set Method – can only segment the image without any assumption on the object that is being segmented.

4.3 Qualitative Analysis

Samples of the segmentation results of the proposed method were qualitatively evaluated and compared with ground truth segmentation labels, LSM-baseline, and BBox-baseline segmentation results. As can be seen in Figure 4, the proposed method consistently yielded fewer false positive pixels compared to both LSM-baseline and BBox-baseline. As LSM-baseline is always initialized from the borders of the image, more regions in the image are included in the segmentation results, hence more false positives are expected. Similarly, for BBox-baseline, as the bounding box shape is constrained, i.e. must be square-

shaped, there are bound to be some false positives in the segmentation result, as real-world objects are not constrained to square shapes. Therefore, this qualitative analysis is consistent with the results shown in Table 1, i.e. the proposed method yields the best segmentation accuracy and precision.



Figure 4 Samples of good segmentation results. From left to right: the original image, ground truth, LSM-baseline, BBox-baseline, and DCLSM segmentation mask.

As the Level Set Method segmenter is conditioned to the DCP prior, i.e. conditioned to the inferred bounding box, the final segmentation result depends on it. Specifically, the LSM module shrinks the bounding box further to get the final segmentation mask.



Figure 5 Samples of failure cases. From left to right: the original image, ground truth, LSM-baseline, BBox-baseline, and DCLSM segmentation mask.

This phenomenon can be seen in Figure 5. Whenever the bounding box prior (BBox-baseline) covers an area smaller than the actual object (low recall situation), the final segmentation result will be even smaller, which means it yields a lower recall score. On the other hand, whenever the bounding box prior covers an area much larger than the actual object (low precision situation), as the LSM module will shrink it further, the precision score of the overall segmentation precision will be increased. Therefore, the precision score of our method is always equal or greater than the bounding box prior, while the recall score of our method is always equal or lower than the bounding box prior. In other words, the DCP prior is the precision lower bound and recall upper bound for DCLSM.

The quality of the final segmentation result also depends on the Level Set Method segmenter. As shown in Figure 5, there were several examples where the Level Set Method segmenter would not shrink further or shrink too much. In those cases, the precision and recall of the overall segmentation result would not be improved by a large margin compared to LSM-baseline and BBox-baseline.

4.4 Computation Cost

The computation performance of the proposed method was compared to LSM-baseline and BBox-baseline on multiple implementations:

1. *CPU-Numpy*, which implements DCP on CPU and LSM on Numpy.
2. *CPU-Theano*, which implements DCP on CPU and LSM on CPU with Theano.
3. *Full-GPU*, which implements DCP on GPU and LSM on GPU with Theano.
4. *GPU-CPU*, which combines the GPU implementation of DCP and CPU implementation of LSM using Theano.

Both the CPU and GPU version of DCP are implemented using Keras. The evaluation results can be seen in Table 2.

Table 2 Computation cost relative to CPU-numpy implementation.

| Model | CPU-Numpy | CPU-Theano | Full-GPU | GPU-CPU |
|----------------------|-----------|--------------|--------------|--------------|
| LSM-baseline | 1x | 1.71x | 1.18x | 1.71x |
| BBox-baseline | 1x | 1x | 3596x | 719x |
| DCLSM | 1x | 1.24x | 2.96x | 4.2x |

By replacing Numpy with Theano to implement our LSM module increased the computation performance by 24%, even when using Theano's CPU-mode. This may be attributed to the more optimized computation of Theano compared to Numpy. Moreover, by running our DCP network in GPU instead of CPU, 3596x performance was gained. This effectively eliminates computation bottlenecks in the DCP network, which in turn shifts the bottleneck to the LSM module.

Naturally, the solution of this problem is to run the LSM module on a GPU. This version of implementation gains 296% relative improvement compared to the baseline implementation. Interestingly, however, by combining GPU implementation of the DCP network with Theano CPU implementation of the LSM module, the best performance was achieved: 420% relative improvement compared to CPU-Numpy implementation. We hypothesize that the Theano function that is being used to sequentially evolve the PDE of Level Set Method is not well optimized toward GPUs.

5 Further Improvement

State-of-the-art, very complex models such as Faster R-CNN [25] could be used to improve object detection, which in turn will improve the quality of the prior and ultimately improve the quality of the segmentation results. Careful hyperparameter tuning on the DCP network could also improve the proposed method. As shown in our analysis, DCP is the bound of the overall DCLSM segmentation performance. Hence, a better-quality prior will enhance the overall segmentation result. Different formulations, more complex Level Set Method formulations, such as Distance Regularized Level Set Evolution (DRSLE) [28], could also be experimented with to improve the quality of segmentation.

6 Conclusion

This paper presented a way to improve Level Set Method as automatic natural image segmentation method by incorporating Deep Convolutional Neural Network as a prior. By using the prior knowledge, the proposed method could improve the segmentation result significantly, especially in accuracy and precision compared to Level Set Method, which only incorporate an uninformative prior, even without finetuning any hyperparameters. It was found that the Deep Convolutional Prior (DCP) network in our method is the lower bound and upper bound for the overall precision and recall, respectively.

References

- [1] Szeliski, R., *Computer Vision: Algorithms and Applications*, Springer Science & Business Media, 2010.
- [2] Ros, G., Ramos, S., Granados, M., Bakhtiary, A., Vazquez, D. & Lopez, A.M., *Vision-based Offline-online Perception Paradigm for Autonomous driving*, in 2015 IEEE Winter Conference on Applications of Computer Vision, pp. 231-238, 2015.
- [3] Ngo, T.A. & Carneiro, G., *Left Ventricle Segmentation from Cardiac MRI Combining Level Set Methods with Deep Belief Networks*, IEEE International Conference on Image Processing, pp. 695-699, 2013.
- [4] Long, J., Shelhamer, E. & Darrell, T., *Fully Convolutional Networks for Semantic Segmentation*, IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431-3440, 2015.
- [5] Sethian, J.A., *A Fast Marching Level Set Method for Monotonically Advancing Fronts*, Proceedings of the National Academy of Sciences **93**(4), pp. 1591-1595, 1996.
- [6] Chan, T.F. & Vese, L.A., *Active Contours without Edges*, IEEE Transactions on image processing **10**(2), pp. 266-277, 2001.
- [7] Samson, C., Blanc-Feraud, L., Aubert, G. & Zerubia, J., *A Variational Model for Image Classification & Restoration*, IEEE Transactions on Pattern Analysis & Machine Intelligence **22**(5), pp. 460-472, 2000.
- [8] Pinto, N., Cox, D.D. & J. J. DiCarlo, *Why is Real-world Visual Object Recognition Hard?*, PLoS Comput Biol **4**(1), pp. e27, 2008.
- [9] Krizhevsky, A., Sutskever, I. & Hinton, G.E., *Imagenet Classification with Deep Convolutional Neural Networks*, in Advances in neural information processing systems, pp. 1097-1105, 2012.
- [10] Simonyan, K. & Zisserman, A., *Very deep convolutional networks for large-scale image recognition*, arXiv preprint arXiv:1409.1556, 2014.
- [11] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. & Rabinovich, A., *Going Deeper with Convolutions*, IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-9, 2015.
- [12] He, K., Zhang, X., Ren, S. & Sun J., *Deep Residual Learning for Image Recognition*, arXiv preprint arXiv:1512.03385, 2015.
- [13] Karpathy, A. & Fei-Fei, L., *Deep Visual-semantic Alignments for Generating Image Descriptions*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3128-3137, 2015.
- [14] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C. & Fei-Fei, L., *ImageNet Large Scale Visual Recognition Challenge*,

- International Journal of Computer Vision (IJCV) **115**(3), pp. 211-252, 2015.
- [15] Duan, C., Liang, Z., Bao, S., Zhu, H., Wang, S., Zhang, G., Chen, J.J. & Lu, H., *A Coupled Level Set Framework for Bladder Wall Segmentation with Application to MR Cystography*, IEEE transactions on medical imaging **29**(3), pp. 903-915, 2010.
- [16] Vese, L.A. & Chan, T.F., *A Multiphase Level Set Framework for Image Segmentation using the Mumford and Shah Model*, International journal of computer vision **50**(3), pp. 271-293, 2002.
- [17] Liu, S. & Li, J., *Automatic Medical Image Segmentation using Gradient and Intensity Combined Level Set Method*, in Engineering in Medicine and Biology Society, pp. 3118-3121, 2006.
- [18] Airouche, M., Bentabet, L. & Zelmat, M., *Image Segmentation using Active Contour Model and Level Set Method Applied to Detect Oil Spills*, in Proceedings of the World Congress on Engineering, Vol. 1, pp. 1-3, 2009.
- [19] Wang, T., Han, B. & Collomosse, J., *Touchcut: Fast Image and Video Segmentation using Single-touch Interaction*, Computer Vision and Image Understanding **120**, pp. 14-30, 2014.
- [20] Rousson, M. & Paragios, N., *Shape Priors for Level Set Representations*, European Conference on Computer Vision, pp. 78-92, 2002.
- [21] Cha, K.H., Hadjiiski, L., Samala, R.K., Chan, H.P., Caoli, E.M. & Cohan, R.H., *Urinary Bladder Segmentation in CT Urography using Deep-learning Convolutional Neural Network and Level Sets*, Medical physics **43**(4), pp. 1882-1896, 2016.
- [22] Li, S., Fevens, T., Krzyzak, A. & Li, S., *Automatic Clinical Image Segmentation using Pathological Modeling, PCA and SVM*, Engineering Applications of Artificial Intelligence **19**(4), pp. 403-410, 2006.
- [23] Pawar, V. & Talbar, S., *Object Recognition Technique based on Level Set Method and Neural Network*, idea **40**(2), 2012.
- [24] Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J. & Zisserman, A., *The pascal visual object classes (voc) challenge*, International Journal of Computer Vision **88**, pp. 303-338, Jun. 2010.
- [25] Ren, S., He, K., Girshick, R. & Sun, J., *Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks*, in Advances in neural information processing systems, pp. 91-99, 2015.
- [26] Caselles, V., Kimmel, R. & Sapiro, G., *Geodesic Active Contours*, International journal of computer vision **22**(1), pp. 61-79, 1997.
- [27] Kingma, D. & Ba, J., *Adam: A Method for Stochastic Optimization*, arXiv preprint arXiv:1412.6980, 2014.
- [28] Li, C., Xu, C., Gui, C. & Fox, M.D., *Distance Regularized Level Set Evolution and Its Application to Image Segmentation*, IEEE Transactions on Image Processing **19**(12), pp. 3243-3254, 2010.