UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY
ADVANCED PROGRAM IN COMPUTER SCIENCE

DIỆP GIA HÂN
TRẦN NGUYỄN SƠN THANH

ACTIVE CONTOUR UNET WITH TRANSFER LEARNING FOR MEDICAL IMAGE SEGMENTATION

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

HO CHI MINH CITY, 2020

UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY
ADVANCED PROGRAM IN COMPUTER SCIENCE

DIỆP GIA HÂN - 1651077

TRẦN NGUYỄN SƠN THANH - 1651072

ACTIVE CONTOUR UNET WITH TRANSFER LEARNING FOR MEDICAL IMAGE SEGMENTATION

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

THESIS ADVISORS
ASSIST. PROF. LÊ THỊ HOÀNG NGÂN - ASSOC.PROF. TRẦN MINH TRIẾT

HO CHI MINH CITY, 2020

ACKNOWLEDGEMENT

Firstly, we would like to wholeheartedly thank you our research internship supervisor as well as thesis advisor, Prof. Trần Minh Triết, without whose continuous support and expert advice, we could not have complete this thesis. With his patience and friendly guidance, we have acquired valuable fundamental knowledge for our chosen field, and especially the enthusiasm for doing researches. We would also like to thanks our Computer Graphic and Computer Vision lecturer, as well as our thesis reviewer, Dr. Trần Thái Sơn for knowledge in these fields together with the informative feedbacks on our ideas and reports. Additionally, our appreciations earnestly fall for Prof. Lê Thị Hoàng Ngân, our thesis advisor, for the extended lectures and discussions, valuable suggestions and unwavering supports which have contributed primely to our thesis.

Besides, we would like to express our gratitude to all of our lecturers in the Faculty of Information and Technology, University of Science, VNUHCM, for supporting us in building foundation knowledge in Computer Science, especially Deep learning and Computer Vision. This provided us fundamental knowledge to study and acquire deeper knowledge for our thesis.

We are thankful for our generous colleagues: Nguyễn Hải Đăng, Vũ Lê Thế Anh, Hoàng Trung Hiếu, Trương Thành Đạt (and appreciably more) who are always available whenever we need helps, either relating or not to academic problems.

Last but not least, our deepest gratitude goes to all of our parents. It would not be possible to finish this thesis without their immeasurable support, caring and encouragement throughout our education.

Authors

Diệp Gia Hân
Trần Nguyễn Sơn Thanh

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Biomedical imaging is the technique and process that involves a very broad field. It covers data acquiring, image processing, structure visualizing to medical diagnosis based on features extracted from images. Medical image segmentation is one of the most challenging tasks in medical image analysis and widely developed for many clinical applications. While deep learning-based approaches have achieved impressive performance in semantic segmentation, they are limited to pixel-wise settings with **less annotation, imperfect data, imbalanced-class data problems and weak boundary** object segmentation in medical images.

This project tackles the aforementioned limitations by proposing a 3D deep neural network, which inherits the merits from Active Contour model and is guided by a Cycle-Consistent Adversarial Networks (CycleGAN). **CycleGAN aims at transferring data from source domain to target domain to address the problem of less annotations**. In addition to CycleGAN, which helps to transfer the image appearance between the two time-points, we employ the segmentation features to enforce the generator network to guarantee the tissue segmentation consistency, results in more realistic synthetic images. In order effectively process MRI images, which is shown as volumetric data, we improve 2D CycleGAN to 3D CycleGAN. Furthermore we address the problem of imbalanced-data and weak boundary object by proposing a **two-branch UNet-like architecture i.e. Active Contour Unet**. Our proposed Active Contour Unet network takes both higher level feature and intermediate level and lower level feature into account. Our network contains two branches: (i) the first branch extracts higher level feature as region information by a common encoder-decoder network structure such as Unet, FCN; (ii) the second branch focuses on both intermediate level feature as support information around boundary and lower level feature on the boundary/surface. All two branches processes in parallel into

an end-to-end framework. In the second branches named **Narrow Band Active Contour (NB-AC) attention model**, the object contour/surface plays the role of a hyperplane and all data inside a narrow band as support information that influences the position and orientation of the hyperplane. The **proposed network loss contains two fitting terms**: (i) high level features (i.e. region) fitting term from the first branch; (ii) lower level features (i.e. contour) fitting term from the second branch including the (ii_1) length of object contour and (ii_2) regional energy functional formed by the homogeneity criterion of both inner band and outer band neighboring the evolving curve or surface. In order to develop the proposed network regardless medical image modalities of 2D or 3D volumetric, we have improved CycleGAN which was original developed on 2D to 3D volumetric and our two-branch UNet-like architecture has been implemented under 3D network.

The proposed network has been evaluated on different challenging medical image datasets including **iSeg19, MRBrainS18 and Brats18**. The experimental results have shown that the proposed Active Contour Unet with our NB-AC loss o**utperforms other mainstream loss functions such as Cross Entropy, Dice, Focal** on the common segmentation frameworks such as FCN and Unet. Our 3D network which is built upon the proposed NB-AC loss and 3D-Unet framework with the guidance from CycleGAN archives the state-of-the-art results on multiple volumetric datasets.

CHAPTER 1
INTRODUCTION

The development of biomedical imaging techniques, which provides detailed cross-sectional anatomies, leads the way for advanced deep learning approaches beneficial to medical analysis or early diagnosis [1, 2]. For instance, segmentation – the most prerequisite task in medical image processing, as it extracts the region of interest (ROI) and defines the specific boundaries between divided areas of the image. Several clustering or segmenting strategies based only on the global characteristic of the image can also acquire requested results, though proved not very efficient for involved multi- modality inputs [3], specifically Magnetic Resonance Images (MRI). MRI modality can provide complementary information depending on variable acquisition parameters, such as T1, T1c, T2, Flair and it has been widely used in biomedical imaging. Recently, deep learning-based approaches have obtained the state-of-the-art performance in multiple task including image segmentation in both computer vision and medical imaging. There are two main categories Based on the dimensions of convolutional kernel and input size, approaches for volumetric segmentation can be categorized into two: (i) 2D approaches and (ii) 3D approaches. The former approaches take 2D image slice as input, and the feature map of a full volume is formed by feature map of individual slice. In these approaches, the 2D convolutional kernels are able to leverage context across the height and width of the slice to make predictions; however, they inherently fail to leverage context from adjacent slices. The 2D approaches can efficiently reduce the computational cost for training but the performance is limited compared to the 3D approaches. The 3D approaches take the 3D image as input and apply the 3D convolution kernel to exploit the spatial contextual information of the image. Since these approaches can utilize the information from adjacent slices for extracting prediction map, they have archived the state-of-the-art results in volumetric segmentation Densenet [4, 5, 6], Unet

[7, 8], Vnet [9], DeepMedic [10]. Besides, so as to leverage the available data, it is crucial to preprocess medical images: normalize, de-noising, contrast enhancement, cranium removal or bias field correction and augmentation; These can be operated in diverse ways: traditional method (flipping, centering, etc.) or the trending tactics relating to generative adversarial neural networks (GAN).

Most deep learning (DL)-based segmentation networks have made use of common loss functions e.g., Cross-Entropy(CE), Dice [11], and the recent Focal [12]. These losses are based on summations over the segmentation regions and are restricted to pixel-wise settings. Not only pixel-wise sensitivity, these losses are unfavorable to small structures, do not take geometrical information into account as well as limited to imbalanced-class data and weak boundary objects problems. Furthermore, these losses are working on higher level features of region information and none of them intentionally are designed on lower level features such as edge/boundary which plays an important role in medical imaging. We have some observations on medical images as follows: (i) Boundary information plays a significant role in many medical analysis tasks such as shape-based cancer analysis, size-based volume measure. (ii) Medical images contain **weak boundaries** which make segmentation tasks much more challenging due to low intensity contrast between tissues, and intensity inhomogeneity. For example, the myelination and maturation process of the infant brain, the intensity distributions of gray matter(GM) and white matter (WM) have a larger overlapping thus the boundary between GM and WM is very weak, leading to difficulty for segmentation. (iii) In the medical image segmentation problem, **imbalance-class data** is naturally existing. Those two challenges of the imbalanced-class data and the weak boundary object in medical imaging are visualized in Fig. 5.2 and demonstrated in Fig.5.1. Fig.5.1(a) illustrates the imbalanced-class problem in medical images through the statistical class distribution of four different datasets. For each dataset, the number of samples between classes are varied.

Fig.5.1(b) shows statistical values of Mean/Std/Median of pixel intensity in individual class when pixel values are in [0,1]. Within an individual dataset, the difference between classes in term of Mean/Std/Median is very small. Strong correlation between classes makes the problem of distinguishing classes more challenging specially at the boundary as shown in Fig.5.2 which is known as weak boundary problem.

To address the aforementioned problems of weak boundary, imbalance data, we make use of the advantages of LS [13] and propose a two-branch deep network which explicitly takes into account both higher level features, i.e object region in the first branch and lower level features, i.e. contour (object shape) and narrow band around the contour in the second branch. The first branch is designed as a classical CNN, i.e. an encoder-decoder network structure whereas the second branch is built as a narrow band active contour (NB-AC) attention model which processes in parallel to the first branch. The proposed loss for our NB-AC attention model contains two fitting terms: (i) the *length of the contour*; (ii) the *narrow band energy* formed by homogeneity criterion in both inner band and outer band neighboring the evolving curve or surface as illustrated in Fig. 5.6. The higher level feature from the first branch is connected to the lower level feature in the second branch through our proposed *transitional gates* and both are designed in an end-to-end architecture. Thus, our loss function not only pays attention to region information but also focus on support information at the two sides of the boundary under a narrow band. In this proposed network, we consider the object contour as a hyperplane whereas information in the inner and outer bands aims play the role of a supporter which influences the position and direction of the hyperplane.

Furthermore, generative adversarial neural networks (GAN) has been utilized in multiple purposes of biomedical images processing such as reconstruction, image synthesis, or anomaly detection. Among which, medical image synthe-

sis, adopting various favorable models like GANs [14], Conditional GAN [15] or cycle-GAN [16, 17, 18], recycle GAN [19] for domain translation, plays both an impressive role in augmentation solving the adversity of lacking detail annotated data or bias datasets (e.g. scarce amount of data for a rare disease) and a promising shift for patient's privacy issues. Alongside with the race of modernistic neural network models, various other researches tackle the problem of bettering image processing via the amendment of loss functions for diverse purposes, distinguished as boundary-based, region-based, distribution-based or compound loss. These recent researches have significantly contributed to the CNN-based methods for multi-modalities medical images segmentation. For instance, varied attractive approaches have been proposed following this novel trend attempting to use GAN to translate inputted images from one domain to another to guide the segmentation results, with interesting uses of the loss functions. This bridges an ambition to study the effectiveness of the idea using domain transferring for guided segmentation. Inspired by the blossoming of GAN, **we tackle the problem of less labeled data in medical imaging by first improving 2D cycle-GAN [16, 17] to 3D cycle-GAN and then integrating the proposed 3D cycle-GAN into our proposed NB-AC Unet under end-to-end transfer learning framework**.

In the following chapters, we will discuss roughly about the background and related work of this problem, our methods as well as experiments either directly or indirectly relating to this problem. In chapter 2, we will note some basic methods for deep learning and deep learning in medical problems. In chapter 3, we will note some previous techniques (to our furthest knowledge) solving relating problems. Chapter 4 and chapter 5 will be about several approaches we tackled the problem as well as the experiment results for a better understanding of relating works. Future works and other possible approaches are provided in chapter 6. The contributions in this work are:

- Study the properties of medical imaging analysis and the challenges in medical images segmentation.
- Study two main areas in deep neural networks, namely, Convolutional Neural Networks (CNN) and Generative Adversarial Networks (GAN).
- Study the active contour theory, specially we focus on zero level set approach, which is based on Mumford -Shah energy minimization and the most successful active contour model for medical image segmentation.
- Utilize the active contour theory, i.e. variational level set framework to address the weak boundary problem in medical images segmentation.
- Utilize the narrow band theory to address the imbalance data problem in medical images segmentation.
- Utilize CycleGAN to address the less labeled data problem in medical images segmentation.
- We have implemented and reproduced the performance of the state-of-the-art work proposed by [6] which utilized CycleGAN to improve segmentation performance on iSeg19.
- Extend the 2D CycleGAN [18], which was designed for 2D data, to 3D CycleGAN to effectively work on volumetric data.
- Our proposed network is able to tackle multiple limitations in medical images segmentation, namely, less annotation, imperfect data, imbalance class problems and weak boundary object segmentation.
- The entire proposed network is implemented under 3D network architecture and end-to-end framework.

CHAPTER 2
BACKGROUND

This chapter includes three main section discussing medical image and its challenges (section 2.1.2), contour-based approaches (section 2.2) and popular deep learning-based methods (section 2.3.2). In the later parts, we provide an overview of different types of medical images (especially magnetic resonant image for brains), an overview of medical imaging methods with their limitations.

2.1 Medical images and its challenges

There are several common problems in image processing which can be applied for detecting and analysing brain tissues, such as classification, detection and segmentation, etc. No matter what, the most basic step is to understand thoroughly the common input data for brain medical imaging problems: MRI sequences, especially our target problem - brain images. Only after that, image processing can be applied to solve the problems.

2.1.1 Understanding MRI sequences

The MRI sequences is the sequences of event happened inside the MRI machines, which give us the MRI images by adding photons energy to the tissues and observing the rate of bounced back energy [1]. There are several common MRI sequences, such as T1, T2, FLAIR (fluid attenuation inversion recovery), GRE, DWI, and so on. Each type of the MRI sequences comes with different attributes and preferred usage. Here, we will only discuss brain MRI images as it directly related to our considered problem.

T1 and T2 brain sequences are similar: water and fat have opposite intensity

[1]The contrast of MRI images shows the differences in relaxation of photons in different brain tissues.

within a sequence and substances in T1 and T2 also have opposite intensity. Table 2.1 shows the intensities of common tissues in T1, T2 and Flair in brain MRI and example in figure 2.1 illustrates the differences[2].

In T2 MRI, lesions (inflammation in table 2.1) are hard to be distinguish from CSF as they are both bright regions, but can easily be done with Flair (Flair is the same as T2 except for CSF regions are flipped back to dark). This can be illustrates with figure 2.2 [3].

For other types of MRI sequences, GRE (gradient echo or also known as T2*), in brief, shows paramagnetic substances in dark intensity, and it is one of the few brain MRI sequences can help to detect hemorrhages. DWI (diffusion weighted) can help to clearly distinguish infractions (figure 2.3). Which can also help to detect lesion or tumor due to the abnormal phenomenon of infarction. Hence, in summary, it is better to predict brain symptoms using multiple MRI sequences (T1 and T2 enhanced version or weighted version may be used instead of T1 or T2).

Table 2.1: Comparison of T1, T2 and Flair in brain MRI [3]

Tissue	T1-Weighted	T2-Weighted	Flair
CSF	Dark	Bright	Dark
White Matter(WM)	Light	Dark Gray	Dark Gray
Cortex	Gray	Light Gray	Light Gray
Fat	Bright	Light	Light
Inflammation (infection, demyelination)	Dark	Bright	Bright

2.1.2 Medical imaging challenges

We have some observations on medical images as follows: (i) Boundary information plays a significant role in many medical analysis tasks such as shape-based

[2]Table 2.1 and figure 2.1 are adapted from Davis C Preston, Case Western Reserve University

[3]Figure 2.2 is adapted from MRI sequences

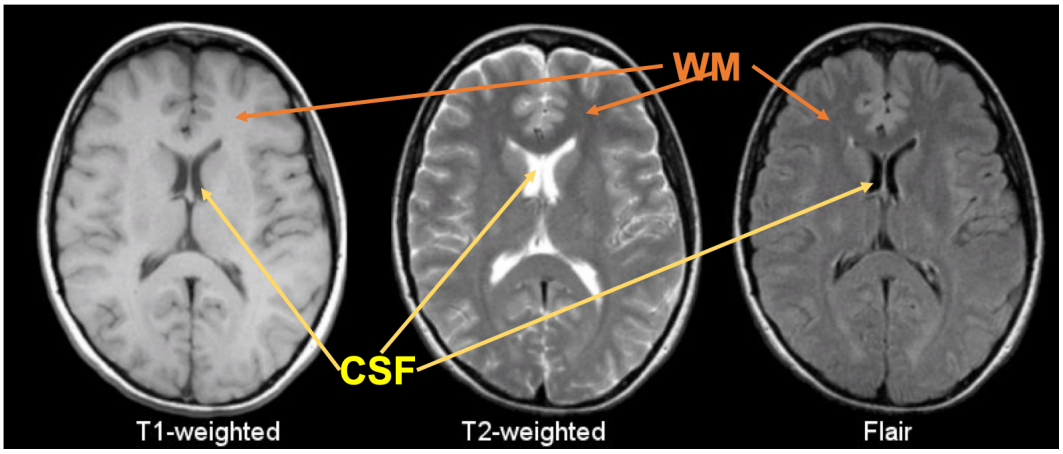


Figure 2.1: Comparison of T1, T2 and Flair in brain MRI [2]

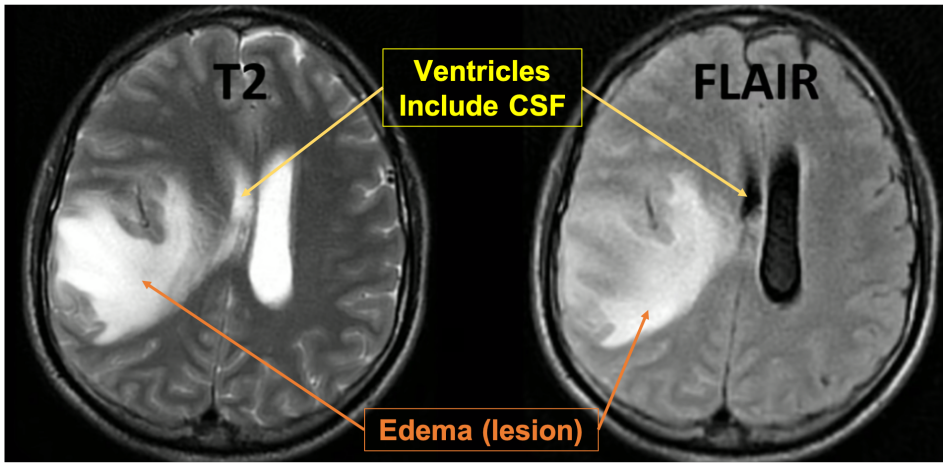


Figure 2.2: T2 versus Flair in detecting edema[3]

cancer analysis, size-based volume measure. (ii) Medical images contain **weak boundaries** which make segmentation tasks much more challenging due to low intensity contrast between tissues, and intensity inhomogeneity. For example, the myelination and maturation process of the infant brain, the intensity distributions of gray matter(GM) and white matter (WM) have a larger overlapping thus the boundary between GM and WM is very weak, leading to difficulty for segmentation. (iii) In the medical image segmentation problem, **imbalance-class data** is naturally existing. Those two challenges of the imbalanced-class data and the weak boundary object in medical imaging are visualized in Fig. 5.2

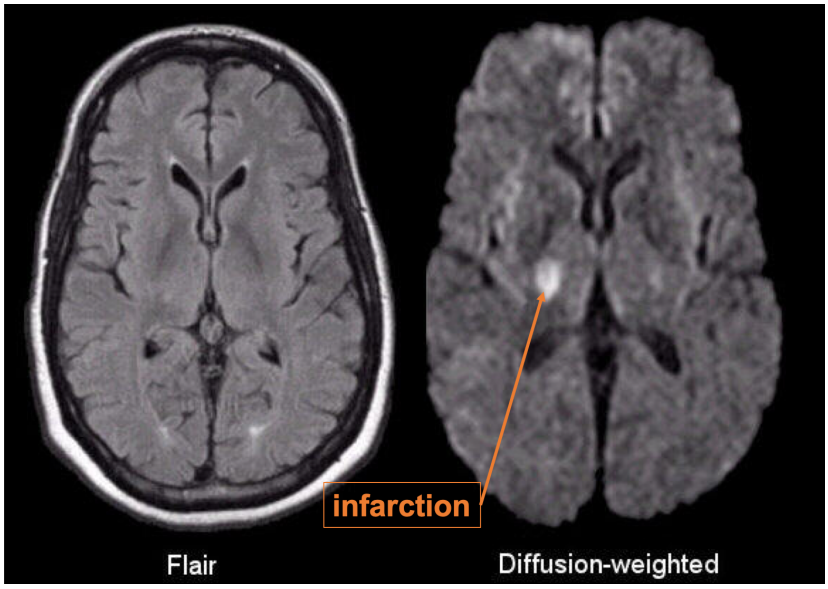


Figure 2.3: T2 versus Flair in detecting infarction[2]

and demonstrated in Fig.5.1. The fig.5.1(a) illustrates the imbalanced-class problem in medical images through the statistical class distribution of three different datasets. For each dataset, the number of samples between classes are varied. Fig.5.1(b) shows statistical values of Mean/Std/Median of pixel intensity in individual class when pixel values are in [0,1]. Within an individual dataset, the difference between classes in term of Mean/Std/Median is very small. Strong correlation between classes makes the problem of distinguishing classes more challenging specially at the boundary as shown in Fig.5.2 which is known as weak boundary problem.

2.2 Active Contour Technique for Image Segmentation

There are two main approaches in active contours: snakes and level sets. Snakes explicitly move predefined snake points based on an energy minimization scheme, while level set approaches move contours implicitly as a particular level of a function.

2.2.1 Classic Snakes

The first model of active contour, named classic snakes or explicit active contour, was proposed by Kass et al.[20]. In this approach, a contour parameterized by arc length s as $C(s)(x(s), y(s)) : 0 \leq s \leq 1$. An energy function $E(C)$ can be defined on the contour such as

$$E(C) = \int_0^1 E_{int} + E_{ext} \tag{2.1}$$

where E_{int} and E_{ext} are the internal energy and external energy functions, respectively. The internal energy function determines the regularity, i.e. smooth shape, of the contour

$$E_{int}(C(s)) = \alpha|C'(s)|^2 + \beta|C''(s)|^2 \tag{2.2}$$

Here α controls the tension of the contour, and β controls the rigidity of the contour while $C'(s)$ makes the spline act like a membrane (like "elasticity") and $C''(s)$ makes it act like a thin-plate (like "rigidity"). The external energy term determines the criteria of contour evolution depending on the image $\mathbf{I}(x, y)$, and can be defined as

$$E_{image} = w_{line}E_{line} + w_{edge}E_{edge} + w_{term}E_{term} \tag{2.3}$$

The first term $E_{line} = \mathbf{I}(x, y)$ depends on the sign of w_{line} which guides the snake towards the lightest or darkest nearby contour. The second term $E_{edge} = -|\bigtriangledown \mathbf{I}(x, y)|^2$ attracts the snake to large intensity gradients. The third term E_{term} attracts the snake toward termination of line segments and corners. E_{term} is defined using curvature of level lines in C: $E_{term} = \frac{\left(\phi_x^2\phi_{yy} - 2\phi_x\phi_y\phi_{xy} + \phi_y^2\phi_{xx}\right)}{|\bigtriangledown\phi|^3}$. Fig.2.4 shows an example of snake with 70 snakes points forming a contour around the moth. Each point moves towards the optimum coordinates, where the energy

function converges to the minimum.

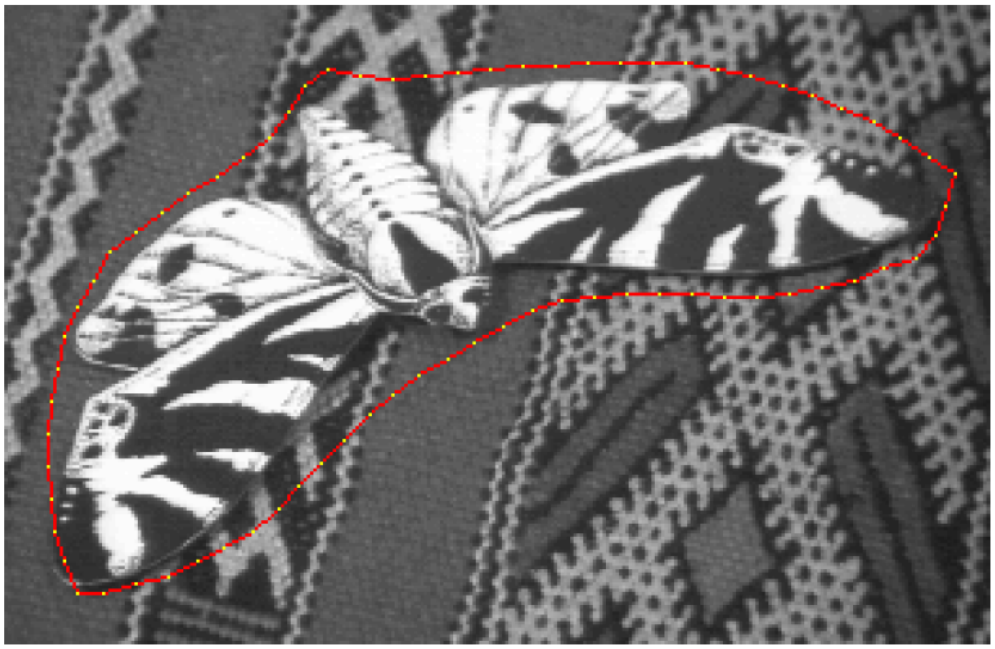


Figure 2.4: An example of classic snakes [21]

The snake provide an accurate location of the edges only if the initial contour is given sufficiently near the desired edges. Moreover, snake cannot detect more than one boundary simultaneously because the snakes maintain the same topology during the evolution stage.

2.2.2 Level Set Method

Level set (LS) based or implicit active contour models have provided more flexibility and convenience for the implementation of active contours, thus, they have been used in a variety of image processing and computer vision tasks. The basic idea of the implicit active contour is to represent the initial curve C implicitly within a higher dimensional function, called level set function $\phi(x, y) : \Omega \rightarrow R$, such as:

$$C = (x, y) : \phi(x, y) = 0, \forall (x, y) \in \Omega \tag{2.4}$$

where Ω denotes the entire image plane. Fig.2.5 (left)shows the evolution of level set function $\phi(x, y)$, and Fig.2.5 (right) shows the propagation of the corresponding contours C.

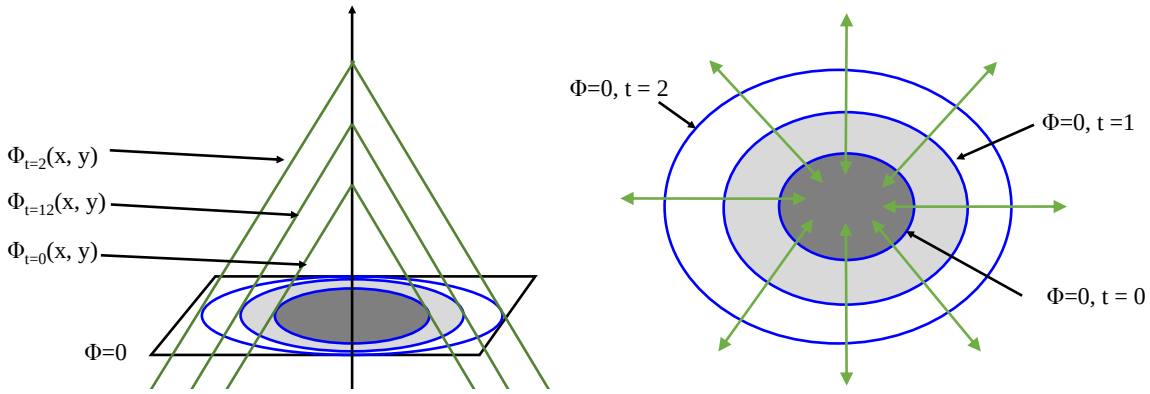


Figure 2.5: Level set evolution and the corresponding contour propagation: (a) topological view of level set $\phi(x, y)$ evolution, (b) the changes on the zero level set $C = (x, y) : \phi(x, y) = 0$

The evolution of the contour is equivalent to the evolution of the level set function, i.e. $\frac{\partial C}{\partial t} = \frac{\partial \phi(x,y)}{\partial t}$. One of the advantages of using the zero level set is that a contour can be defined as the border between a positive area and a negative area, so the contours can be identified by signed distance function as follows:

$$\phi(x) = \begin{cases} d(x, C)) & \text{if } x \text{ is inside } C \\ 0 & \text{if } x \text{ is on } C \\ -d(x, C)) & \text{if } x \text{ is outside } C \end{cases} \tag{2.5}$$

where $d(x, C)$ denotes the distance from an arbitrary position to the curve.

The level set evolution can be written in the form as follows:

$$\frac{\partial \phi}{\partial t} + F\,|\nabla \phi| = 0 \tag{2.6}$$

where F is a speed function. In some particular cases, F is defined as mean

curvature, $F = div\left(\frac{\nabla\phi}{||\nabla\phi||}\right)$

An outstanding characteristic of level set methods is that contours can split or merge as the topology of the level set function changes. Therefore, level set methods can detect more than one boundary simultaneously, and multiple initial contours can be placed as shown in Fig.2.6

Because the computation is performed on the same dimension as the image plane Ω the computational cost of level set methods is high and the the convergence speed is quite slow

2.2.3 Edge-based Active Contours

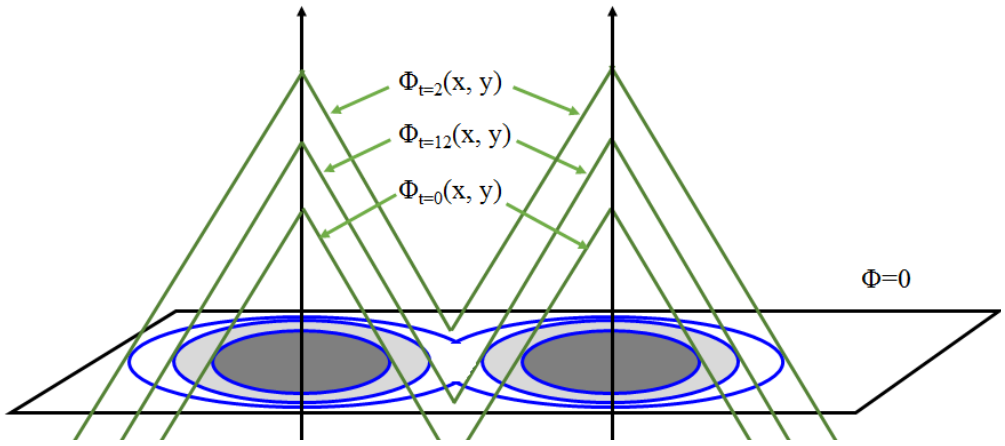
Edge-based active contours are closely related to the edge-based segmentation. Most edge-based AC models consist of two components: regularity and edge detection. The first part determines the shape of contours whereas the second one attracts the contour towards the edges.

Geodesic active contour (GAC) model, one of the most popular methods among the edge-based active contour models, was proposed by Caselles et al.[22]. Given an initial curve C_0, the curve evolution is given as:

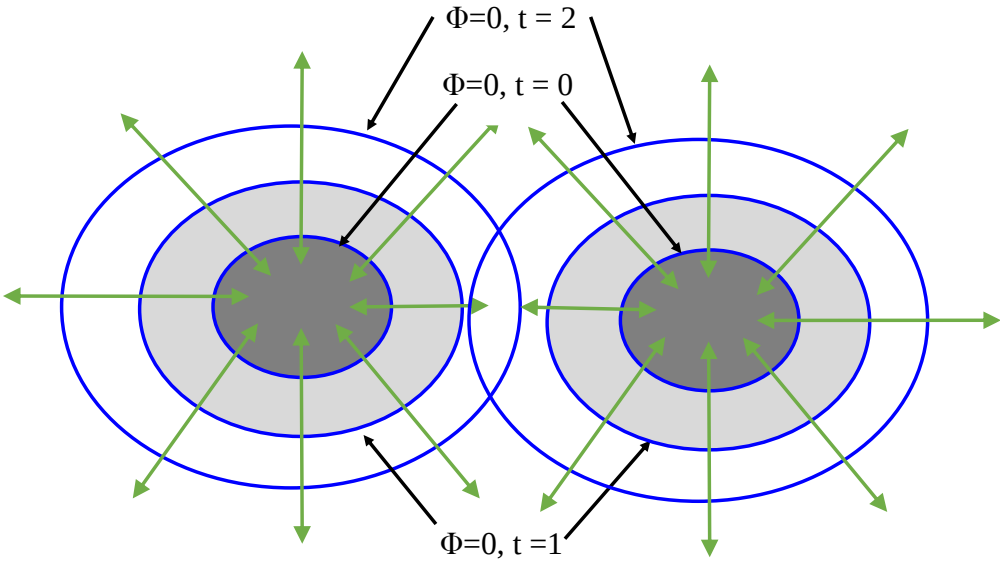
$$\frac{\partial\phi(x,y)}{\partial t} = g(\mathbf{I}(x,y))(\kappa(\phi(x,y)) + F)|\bigtriangledown\phi(x,y)| \tag{2.7}$$

where g denotes a stopping function which is based on an edge indicator scalar function, i.e. $g(\mathbf{I}(x,y)) = \frac{1}{1+|\bigtriangledown\phi(x,y)|}$. The curvature κ maintains the regularity of the contours whereas the constant speed F keeps the contour evolving. In GAC model, the contours move in the normal direction with a speed of $\kappa(\phi(x,y)) + F$ and therefore stops on the edges

Besides inheriting some disadvantages of the edge-based segmentation methods, such as a reliance on the image gradient, omission of blurry boundaries and a



(a) the topological view of level set $\phi(x, y)$ evolution



(b) the changes on the zero level set $C : \phi(x, y) = 0$

Figure 2.6: Topology of level set function changes in the evolution and the propagation of corresponding contours

sensitivity to local minima and noise, edge-based active contour models have a few of their own disadvantages (compared to the region-based active contour models which will be discussed in the next section) that are due to the structure of the speed functions and the stopping functions. It is easy to see that the edge-based active contour models evolve the contour towards only one direction, either inside or outside because of the constant speed F. Thus, an initial contour should be placed completely inside or outside the ROI, so some level of a prior knowledge is still required. Later, Paragios [23] proposed Gradient vector flow fast geodesic active contour by replacing the edge detection with a gradient vector field.

2.2.4 Region-based Active Contours

Most region-based active contour models consist of two components: regularity and energy minimization. The first part is to determine the smooth shape of contours whereas the second part searches for uniformity of a desired feature within a subset.

One of the most popular region based active contour models is proposed by Chan-Vese (CV) [13]. In this model the boundaries are not defined by gradients and the curve evolution is based on the general Mumford-Shah (MS) [24] formulation of image segmentation as shown in Eq.2.8.

$$E = \int_{\Omega} |\mathbf{I} - u|^2 dxdy + \int_{\Omega/C} |\nabla u|^2 dxdy + \nu Length(C) \tag{2.8}$$

CV's model is an alternative form of MS's model which restricts the solution to piecewise constant intensities and it has successfully segmented an image into two regions, each having a distinct mean of pixel intensity by minimizing the

following energy function:

$$\begin{aligned} E(c_1, c_2, \phi) &= \mu \text{Area}(\omega_1) + \nu \text{Length}(C) \\ +\lambda_1 \int_{\omega_1} |\mathbf{I}(x,y) - c_1|^2 dxdy &+ \lambda_2 \int_{\omega_2} |\mathbf{I}(x,y) - c_2|^2 dxdy \end{aligned} \tag{2.9}$$

where c_1 and c_2 are two constants. The parameters $\mu, \nu, \lambda_1, \lambda_2$ are positive parameters and usually fixing $\lambda_1 = \lambda_2 = 1$ and $\mu = 0$. Thus, we can ignore the first term in Eq. 2.9. Assume that we divide the region Ω into two regions, called ω_1 and ω_2, which are separated by the zero level set ϕ. Mathematically,

$$\begin{aligned} \omega_1 &= \{x, y : \phi(x,y) > 0\} : \text{inside } \phi \\ \omega_2 &= \{x, y : \phi(x,y) < 0\} : \text{outside } \phi \\ C &= \{x, y : \phi(x,y) = 0\} : \text{on } \phi \\ \Omega &= \omega_1 \cup \omega_2 \cup C \end{aligned}$$

In the Eq.2.9, the length and the area of zero level set are defined as:

$$\begin{aligned} \text{Length}(C) &= \int_\Omega \delta(\phi(x,y)) |\nabla \phi(x,y)| dxdy \\ \text{Area}(\omega_1) &= \int_\Omega H(\phi(x,y)) dxdy \end{aligned}$$

Where $\delta(z)$ be a Dirac delta function. Thus the energy function is rewritten as follows:

$$\begin{aligned} E(c_1, c_2, \phi) = \mu \int_\Omega H(\phi(x,y)) dxdy + \nu \int_\Omega \delta(\phi(x,y)) |\nabla \phi(x,y)| dxdy \\ + \lambda_1 \int_{\omega_1} |\mathbf{I}(x,y) - c_1|^2 dxdy + \lambda_2 \int_{\omega_2} |\mathbf{I}(x,y) - c_2|^2 dxdy \end{aligned} \tag{2.10}$$

For numerical approximations, the δ function needs a regularizing term for smoothing. In most cases, the Heaviside function H and Dirac delta function

δ are defined as in (2.11) and (2.12), respectively.

$$H_\epsilon(x) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{x}{\epsilon}\right)\right) \tag{2.11}$$

$$\delta_\epsilon(x) = H'(x) = \frac{1}{\pi}\frac{\epsilon}{\epsilon^2 + x^2} \tag{2.12}$$

As $\epsilon \to 0$, $\delta_\epsilon \to \delta$, and $H_\epsilon \to H$. Using Heaviside function H, the Eq.2.10 becomes

$$\begin{aligned} E(c_1, c_2, \phi) = \mu \int_\Omega H(\phi(x,y))dxdy + \nu \int_\Omega \delta(\phi(x,y))|\nabla\phi(x,y)|dxdy \\ + \lambda_1 \int_\Omega |\mathbf{I}(x,y) - c_1|^2 H(\phi(x,y))dxdy + \lambda_2 \int_\Omega |\mathbf{I}(x,y) - c_2|^2 (1 - H(\phi(x,y)))dxdy \end{aligned} \tag{2.13}$$

In the implementation, they choose $\epsilon = 1$. For fixed c_1 and c_2, gradient descent equation with respect to ϕ is

$$\frac{\partial\phi(x,y)}{\partial t} = \delta_\epsilon(\phi(x,y)[\nu\kappa(\phi(x,y) - \mu - \lambda_1((\mathbf{I}(x,y) - c_1)^2 + \lambda_2((\mathbf{I}(x,y) - c_2)^2] \tag{2.14}$$

where δ_ϵ is a regularized form of Dirac delta function and c_1, c_2 are the mean of inside the contour ω_{in} and the mean of the outside of the contour ω_{out}, respectively. The curvature κ is given by

$$\kappa(\phi(x,y)) = -div\left(\frac{\triangle\phi}{|\triangle\phi|}\right) = -\frac{\phi_{xx}\phi_y^2 - 2\phi_x\phi_y\phi_{xy} + \phi_{yy}\phi_x^2}{\left(\phi_x^2 + \phi_y^2\right)^{1.5}} \tag{2.15}$$

For fixed ϕ, gradient descent equation with respect to c_1 and c_2 are

$$\begin{aligned} c_1 &= \frac{\sum_{x,y}\mathbf{I}(x,y)H(\phi(x,y))}{\sum_{x,y}H(\phi(x,y))} \\ c_2 &= \frac{\sum_{x,y}\mathbf{I}(x,y)(1 - H(\phi(x,y)))}{\sum_{x,y}(1 - H(\phi(x,y)))} \end{aligned} \tag{2.16}$$

The optimal level set function ϕ can be computed by solving the associate Euler-Lagrange equation. Here, assuming c_1 and c_2 are fixed,

$$E = \int_{\Omega} f(\phi, \nabla\phi) dxdy, \tag{2.17}$$

then the Euler-Lagrange equation is given by

$$\frac{\partial f}{\partial \phi} - \nabla \cdot \frac{\partial f}{\partial \nabla \phi} = 0 \tag{2.18}$$

We compute each term in the above equation as follows:

$$\begin{aligned} \frac{\partial f}{\partial \phi} &= \frac{\partial}{\partial \phi}(v\delta(\phi)|\nabla\phi| + \mu H(\phi) + \lambda_1|I - c_1|^2 H(\phi) + \lambda_2|I - c_2|^2(1 - H(\phi))) \\ &= v|\nabla\phi|\frac{\partial \delta(\phi)}{\partial \phi} + \mu\delta(\phi) + \lambda_1|I - c_1|^2\delta(\phi) - \lambda_2|I - c_2|^2\delta(\phi) \end{aligned} \tag{2.19}$$

in which, the first term vanishes since we care about the zero level set ($\phi = 0$).

$$\begin{aligned} |\nabla\phi| &= \sqrt{\phi_x^2 + \phi_y^2} \\ \frac{\partial}{\partial \phi_x}|\nabla\phi| &= \frac{2\phi_x}{2\sqrt{\phi_x^2+\phi_y^2}} = \frac{\phi_x}{|\nabla\phi|} \\ \frac{\partial}{\partial \phi_y}|\nabla\phi| &= \frac{2\phi_y}{2\sqrt{\phi_x^2+\phi_y^2}} = \frac{\phi_y}{|\nabla\phi|}, \end{aligned} \tag{2.20}$$

$$\begin{aligned} \Rightarrow \frac{\partial}{\partial \nabla\phi}|\nabla\phi| &= \frac{\nabla\phi}{|\nabla\phi|} \\ \frac{\partial}{\partial \nabla\phi} &= v\delta(\phi)\frac{\nabla\phi}{|\nabla\phi|}. \end{aligned}$$

Thus,

$$\frac{\partial f}{\partial \phi} - \nabla \cdot \frac{\partial f}{\partial \nabla \phi} = -\delta(\phi)\{v\mathrm{div}(\frac{\nabla\phi}{|\nabla\phi|}) - \mu - \lambda_1|I - c_1|^2 + \lambda_2|I - c_2|^2\} = 0 \tag{2.21}$$

The above equation is valide when ϕ is the optimal solution. Parameterizing the descent direction by an artificial time t 0, we can formulate an iterative update

equation for ϕ:

$$\frac{\partial \phi}{\partial t} = \delta(\phi)\{v\text{div}\frac{\nabla \phi}{|\nabla \phi|} - \mu - \lambda_1|I - c_1|^2 + \lambda_2|I - c_2|^2\}. \tag{2.22}$$

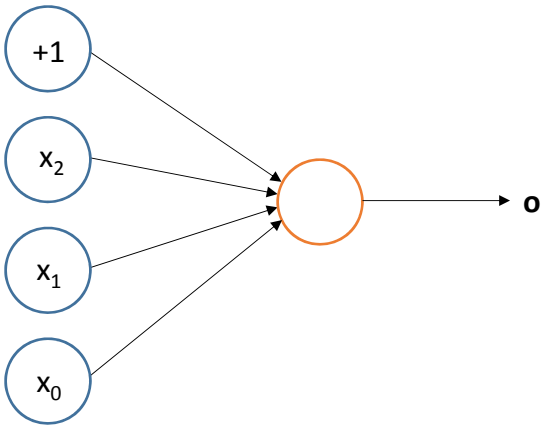
Note that when the time derivative vanishes, ϕ will stop updating.

2.3 Deep learning

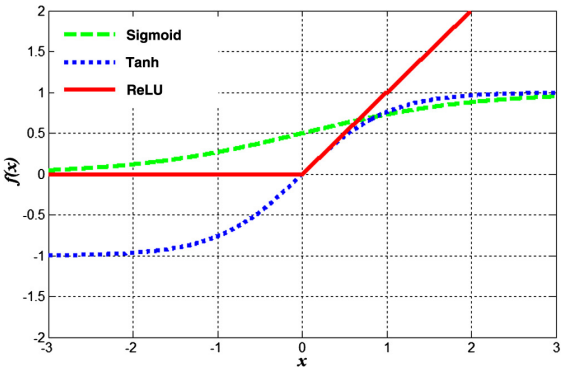
In this section, we will mainly discuss techniques in convolutional neural networks (CNN) and generative adversarial networks (GAN).

2.3.1 Multi-Layer Perceptron (MLP)

Deep learning models, in simple words, are large and deep artificial neural networks. Let us consider the simplest possible neural network which is called "**neuron**" as illustrated in Fig. 2.7. A computational model of a single neuron is called a perceptron which consists of one or more inputs, a processor, and a single output.



(a) An example of one neuron which takes input $\mathbf{x} = [x_1, x_2, x_3]$, the intercept term +1 as bias, and the output **o**.



(b) Plot of different activation functions, i.e. *Sigmoid*, *Tanh* and rectified linear ($ReLU$) functions

Figure 2.7: An example of one neuron and its activation functions

In this example, the neuron is a computational unit that takes $\mathbf{x} = [x_1, x_2, x_3]$ as input, the intercept term $+1$ as bias $\mathbf{b}$, and the output $\mathbf{o}$. The gold of this simple network is to learn a function $f : \mathrm{R}^{\mathrm{N}} \rightarrow \mathrm{R}^{\mathrm{M}}$ where N is the number of dimensions for input $\mathbf{x}$ and M is the number of dimensions for output which is computed as $\mathbf{o} = f(\mathbf{W}, \mathbf{x})$. Mathematically, the output $\mathbf{o}$ of a one output neuron is defined as:

$$\mathbf{o} = f(\mathbf{x}, \theta) = \sigma\left(\sum_{i=1}^{N} w_i x_i + b\right) = \sigma(\mathbf{W}^T\mathbf{x} + b) \tag{2.23}$$

In this equation, σ is the point-wise non-linear activation function. The common non-linear activation function for hidden units are chosen as a hyperbolic tangent ($Tanh$) or logistic sigmoid as shown in Eq. 2.26. A different activation function, the rectified linear ($ReLU$) function, has been proved to be better in practice for deep neural networks. This activation function is different from *Sigmoid* and ($Tanh$) because it is not bounded or continuously differentiable. Furthermore, when the network goes very deep, ReLU activations are popular as they reduce the likelihood of the gradient to vanish. The rectified linear activation ($ReLU$) function is given by Eq. 2.26. These functions are used because they are mathematically convenient and are close to linear near origin while saturating rather quickly when getting away from the origin. This allows neural networks to model well both strongly and mildly nonlinear mappings. Fig. 2.7 is the plot of *Sigmoid*, *Tanh* and rectified linear ($ReLU$) functions.

$$Sigmoid(\mathbf{x}) = \frac{1}{1 + exp^{-x}} \tag{2.24}$$

$$Tanh(\mathbf{x}) = \frac{exp^{2x-1}}{exp^{2x+1}} \tag{2.25}$$

$$ReLU(\mathbf{x}) = max(0, x) \tag{2.26}$$

Notably, the system becomes linear with matrix multiplications if removing the activation function. The *Tanh* activation function is actually a rescaled version of the *sigmoid*, and its output range is [-1,1] instead of [0,1]. The rectified linear function is piece-wise linear and saturates at exactly 0 whenever the input is less than 0.

A neural network is composed of many simple "neurons," so that the output of a neuron can be the input to another. An special case of a neural networks is also called multi-layer perceptron network (MLP) and illustrated in Fig. 2.8.

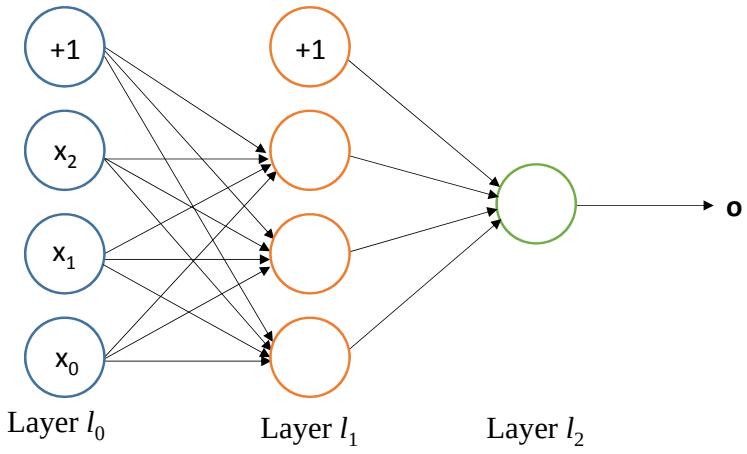


Figure 2.8: An example of multi-layer perceptron network (MLP)

A typical neural network is composed of one input layer, one output layer and many hidden layers. Each layer may contains many units. In this network, $\mathbf{x}$ is the input layer, $\mathbf{o}$ is the output layer. The middle layer is called hidden layer. In the Fig. 2.8, the neural network contains 3 units of input layers, 3 units of hidden layer, and 1 unit of output layer.

In general, we consider a neural network with L hidden layers of units, one layer of input units and one layer of output units. The number of input units is N, output units M, and units in hidden layer l is N^l. The weight of the j^{th} unit in layer l and the i^{th} unit in layer $l+1$ is denoted by w_{ij}^l. The activation of the i^{th} unit in layer l is $\mathbf{h}_i^l$. The input and output of the network are denoted as $\mathbf{x}(n)$,

$\mathbf{o}(n)$, respectively, where n denotes training instance, not time.

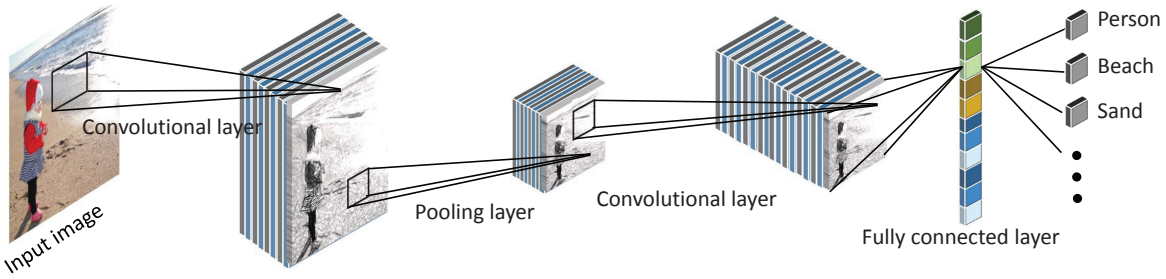


Figure 2.9: Architecture of a typical convolutional network for image classification containing three basic layers: convolution layer, pooling layer and fully connected layer [25]

2.3.2 Convolutional Neural Networks (CNNs)

Neural Networks [26, 27] are a special case of fully connected multi-layer perceptrons that implement weight sharing for processing data that has a known, grid-like topology (e.g. images). CNNs use the spatial correlation of the signal to constrain the architecture in a more sensible way. Their architecture, somewhat inspired by the biological visual system, possesses two key properties that make them extremely useful for image applications: spatially shared weights and spatial pooling. These kind of networks learn features that are shift-invariant, i.e., filters that are useful across the entire image (due to the fact that image statistics are stationary). The pooling layers are responsible for reducing the sensitivity of the output to slight input shift and distortions. Since 2012, one of the most notable results in Deep Learning is the use of convolutional neural networks to obtain a remarkable improvement in object recognition for ImageNet classification challenge [28] [29].

A typical convolutional network is composed of multiple stages, as shown in Fig. 2.9. The output of each stage is made of a set of 2D arrays called feature maps. Each feature map is the outcome of one convolutional (and an optional pooling)

filter applied over the full image. A point-wise non-linear activation function is applied after each convolution. In its more general form, a convolutional network can be written as:

$$\begin{aligned} \mathbf{h}^0 =& \mathbf{x}; \\ \mathbf{h}^l =& pool^l(\sigma_l(\mathbf{w}^l\mathbf{h}^{l-1} + \mathbf{b}^l)), \forall l \in 1, 2, ...L; \\ \mathbf{o} =& \mathbf{h}^L = f(\mathbf{x}, \theta), \end{aligned} \tag{2.27}$$

where $\mathbf{w}^l, \mathbf{b}^l$ are trainable parameters as in MLPs at layer l. $\mathbf{x} \in \mathrm{R}^{c \times h \times w}$ is vectorized from an input image with c is color channels, h is the image height and w is the image width. $\mathbf{o} \in \mathrm{R}^{n \times h' \times w'}$ is vectorized from an array of dimension $h' \times w'$ of output vector (of dimension n). $pool^l$ is a (optional) pooling function at layer l.

CNNs have been applied in *image classification* for a long time [30]. Compared to traditional methods, CNNs achieve better classification accuracy on large scale datasets [28, 31]. With large number of classes, proposing a hierarchy of classifiers is a common strategy for image classification [32]. *Visual tracking* is an another application that turns the CNNs model from a detector into a tracker [33]. As an special case of image segmentation, *saliency detection* is another computer vision application that uses CNNs [34, 35]. In additional to the previous applications, *pose estimation* [36], [37] is another interesting research that uses CNNs to estimate human-body pose. *Action recognition* in both still images and in videos are special case of recognition and are challenging problems. [38] utilizes CNN-based representation of contextual information in which the most representative secondary region within a large number of object proposal regions together the contextual features are used to describe the primary region. CNNs-based action recognition in video sequences are reviewed in [39]. *Text detection and recognition* using CNNs is the next step of optical character recognition

(OCR) [40], word spotting, [41]. Going beyond still images and videos, *speech recognition, speech synthesis* is also an important research field that have been improved by applying CNNs [25, 42]. In short, CNNs have made breakthroughs in many computer vision areas i.e image, video, speech and text.

2.3.3 Generative adversarial nerwork (GAN)

Generative adversarial network (GAN) comprises of two factors: generative (denoted as G) and discriminative (denoted as D) models [43]. The generator G produces images from random vector noises by capturing and mimicking the distribution of images in the training set so as to fool the discriminator D; Discriminator D is to estimate the possibility of any given image's being from the training data. Both G and D could be either a linear mapping or non-linear mapping function such as a multi-layer perceptron [44]. The process of GAN can be considered as a complementary feedback pair. Where the generator striving to provide secured system while the discriminator trying to test the system by cracking it. Noted that these two sub-networks share their results with each other, of whether the system can be cracked.

The generator receives random noise vector and outputs counterfeit images through its black boxes network. The discriminator distinguish whether the inputted images (either generated images or sampled images) are natural (real) or not by estimating the probability of the inputted images being artificial. The principle of loss function used in GANs is to select the parameters for the models which will maximize the likelihood of the training data. This, on the other hand, can be solved using the log likelihood instead, as to reduce the complexity in calculation: the product of all samples will become sum in log likelihood.

$$\theta^* = \underset{\theta}{\text{argmax}} \prod_{i=1}^{m} p_{model}(x^{(i)}; \theta) \tag{2.28}$$

$$= \underset{\theta}{\text{argmax}} \log \prod_{i=1}^{m} p_{model}(x^{(i)}; \theta) \tag{2.29}$$

$$= \underset{\theta}{\text{argmax}} \sum_{i=1}^{m} \log\, p_{model}(x^{(i)}; \theta) \tag{2.30}$$

The use of maximizing the likelihood can also be considered as minimizing the Kullback-Leibler Divergence (KLD), which estimates the distribution distance between the generator and model. By minimizing KLD between generator and model distribution, the resulted group of parameters is expected to be the same as maximizing the log-likelihood of the training set.

$$\theta^* = \underset{\theta}{\text{argmin}}\ D_{KL}(p_{data}(x)||p_{model}(x;\theta)) \tag{2.31}$$

$$\theta^* = \underset{\theta}{\text{argmax}} E_{x\sim P_{data}} \log\, p_{model}(x|\theta) \tag{2.32}$$

The cost used for the discriminator is:

$$J^{(D)}(\theta^{(D)}, \theta^{(G)}) = -\frac{1}{2} E_{x\sim p_{data}} logD(x) - \frac{1}{2} E_2 log(1 - D(G(z))) \tag{2.33}$$

So far we have specified the loss function for only the discriminator. The next step is to do so for the generator as well.

The simplest version of GANs game is a zero-sum game, in which the sum of all player's costs is always zero.

$$J^{(G)} = -J^{(D)} \tag{2.34}$$

The biggest problem facing in GANs that community pays attention to is the issue of non-convergence. Most profound models are trained using optimization algorithms such as SGD, ADAM to figure out a low value of loss function. While numerous problems can interfere with optimization algorithms which creates a stable progress trying to reduce the value of loss function. The training of GANs is to seek for the equilibrium parameters of both generator and discriminator. To gain some intuition for how gradient descent performs, we delivered some experiments of loss we have conducted while training GANs.

Common problems in training GAN. Aforementioned, GAN is a successful method in the image generation, but still challenging to train. It is difficult to achieve the equilibrium point where the ability of generator is competitive compared to discriminator's one: if the discriminator works poorly, the generator does not have the accurate feedback so the loss function can not represent the performance of model and it causes a lot of troubles when we were tracking the loss function to evaluate how well the model works. In brief, GAN loss function can hardly converge. Vanishing gradient is also another disadvantage of GAN, this occurs when the discriminator does a perfect job in recognizing real images. Therefore the loss function $\mathcal{L}$ falls to nearly zero and we end up with no gradient to update the loss during learning iterations.

GAN variations. For different purposes, there exist various popular GAN variances. To directly tackle the problems mentioned above in training GANs, WGAN is created to solve the converge and gradient vanishing problems by using Wasserstein distance to measure the difference between two probability distributions under K-Lipschitz continuous condition [45, 46]. Wasserstein distance is claimed to be better than either Jensen Shannon or Kullback Leibler divergence, for a stable training process using gradient descents as it represents a smooth measure when two distributions are located in lower separate dimensional manifolds. The conditional GANS (cGANs) [47], on the other hand, with

an additional label to better exploit the information from the dataset allows a partially customizable synthesis images. Similary, InfoGAN [48] also utilizes the information from the given dataset, but in this case, for a much more complex dataset, it tries to exploit the similar information obtained from the training dataset and the target latent space. Deeper Convolutional GAN or DCGAN [49], another simple but successful stable training unsupervised GAN model, leverages the batch normalization, convolutional stride, and transposed convolution, with prospective application in image style transferring. For image domain transfering, there are multiple famous models such as CycleGAN (detailed below), Recycle GAN [18, 19], Style GAN [50] for generating gradually higher resolution images by stacking layers training from lower resolution ones, and those for pixel-level like pix2pix [51], etc. With similar idea of generating high quality images from lower ones, StackGAN [52] started with sketch images as low-level resolution given the text description. Or the recent CVPR 2020 SegAttnGAN model [53] also generates high quality images from text via different stages generating lower resolution images as a multi-scale generator (instead of starting from sketch as the lowest resolution image).

Distance metrics. Different distance metrics used in GAN-variation models can highly affect the performance of the models. Hence, in this part we will discuss commonly used distance metrics in GAN (as shown in the original paper of WGAN, Wasserstein-1 outperforms the others).

KL Divergence. The relative entropy or Kullback–Leibler divergence between two probability distributions $P(x)$ and $Q(x)$ that are define.

$$D_{KL}(P||Q) = \sum_{x} P(x) log \frac{P(x)}{Q(x)} \tag{2.35}$$

The relative entropy satisfies Gibbs' inequality.

$$D_{KL}(P||Q) \geq 0 \text{ with equality only if } P = Q. \tag{2.36}$$

Note that in general the relative entropy is not symmetric under interchange of the distributions P and Q: in general $D_{KL}(P||Q) \neq D_{KL}(Q||P)$, so D_{KL}, although it is sometimes called the 'KL distance', is not strictly a distance. The relative entropy is important in pattern recognition and neural networks.

Jensen-Shannon Divergence. The Jensen-Shannon (JS) divergence

$$JS(\mathbb{P}_r, \mathbb{P}_g) = KL(\mathbb{P}_r||\mathbb{P}_m) + KL(\mathbb{P}_g||\mathbb{P}_m) \tag{2.37}$$

where P_m is the mixture $\frac{(P_r+P_g)}{2}$. This divergence is symmetrical and always defined because we can choose $\mu = P_m$.

The Earth-Mover (EM) distance or Wasserstein-1.

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(P_r, P_g)} \mathbb{E}_{(x,y)\sim\gamma}[\Big\|x - y\Big\|] \tag{2.38}$$

where $\Pi(P_r, P_g)$ denotes the set of all joint distributions $\gamma(x, y)$ whose marginals are respectively P_r and P_g. Intuitively, $\gamma(x, y)$ indicates how much "mass" must be transported from x to y in order to transform the distributions P_r into the distribution P_g. The EM distance then is the "cost" of the optimal transport plan.

As this EM distance or Wassertein-1 also constrainted to be under Lipschitz continuous condition searching for the upper bound of expected value distance between two sample which sampled from difference distributions. We shall discuss Lipschitz Condition and its former version - the Picard theorem.

Picard theorem Let $f(x, y)$ and $\partial f/\partial y$ be continuous functions of x and y on

a closed rectangle $\mathbb{R}$ with sides parallel to the axes. If (x_0, y_0) is any interior point of $\mathbb{R}$, then there exists a number $h > 0$ with the property that the initial value problem $y' = f(x, y), y(x_0) = y_0$ has one and only one solution $y = y(x)$ on the interval $|x - x_0| \leq h$.

Lipschitz Condition is an advanced version of Picard Theorem which contributes a solid foundation to optimization especially in finding the optimal point. For instance, our assumption that $\partial f/\partial y$ is continuous on $\mathbb{R}$ which is its hypotheses, is used only to obtain the inequality of Lipschitz statement. The definition is stated below:

Function $f(x, y)$ satisfies a Lipschitz condition in the variable y on a set $A \subset \mathbb{R}^2$ if a constant $k > 0$ exists with $|f(x, y_1) - f(x, y_2)| \leq k * |y_1 - y_2|$, With $(x, y_1), (x, y_2)$ are in A and L is Lipschitz constant.

2.3.3.1 Cycle GAN

Image to image translation is now a trending problem in computer vision, with various applications such as style transferring or novel images generating; However, an attractive question relating to image translation is how to translate images without paired examples. A favored solution for this is the use of Cycle Generative Adversarial Networks (CycleGAN) technique: leveraging the cycle consistency loss to train unsupervised image translation via Generative Adversarial Networks architecture using only the unpaired assemblage of images from the two groups. This report summaries the idea of CycleGAN as well as an overview of its application.

As it is a hindrance to collect paired images for multiple domains, there is a desire for techniques assisting to automatically train style transferring. Hence, the present of Cycle Generative Adversarial Network (CycleGAN) has laid one of the first stones for techniques attempting to translate images between different

domains in the absence of paired examples [18]. The original paper of CycleGAN proposes two key ideas. One is a variation of Generative Adversarial Network (GAN) with forward and inverse mapping. The other huge improvement is modifying the original loss function of GAN with one loss for discriminator and another for generator by two components are: adversarial loss and cycle consistency loss. The idea of cycle consistency loss was to avoid the images, the source distribution, to be mapped to random images from the target distribution. The author suggests using FCN score [2] for evaluating performance of Cycle GAN [47][18].

CycleGAN uses two generators and two discriminators. One is generator G to convert images from distribution X to the distribution Y. The other generator is called F converting images from Y to X. Each generator has their corresponding discriminator to distinguish between its generated images and the real ones.

Generator architecture has three sections: an encoder, a transformer, and a decoder. The input image is fed into the encoder to reduce the representation size of images and feed to Convolutions layers to extract the representation of small region and then is passed to the transformer. After that result goes to the decoder, which use convolutions to enlarge the representation size. The discriminator output the probability of the fixed small region of input images. This small region is called "patch" images. It is more effective in the way that focuses on more texture, which is usually being changed in an image translation task.

The loss function was proposed in the paper has two parts, an adversarial loss and a cycle consistency loss. The adversarial loss attempts to fool the corresponding discriminator. However, adversarial loss alone has limitation to produce "real" images. For example, a generator generates an image Y which has distribution mostly like distribution X, but perceptually looked nothing like x,

[2] to measure the quality of the generated images conditioned on an input segmentation map [54]

the result will output a high adversarial loss; Though that is not what we expect. The proposed loss, cycle consistency loss, solves this issue by relying on the expectation that if an image was converted to the other domain and back again, by successively feeding it through both generators. The output will assure the condition that

$$F(G(x)) \approx x \tag{2.39}$$

$$G(F(y)) \approx y \tag{2.40}$$

This method has huge applications where paired training data does not exist, especially in style transfer, season transfer, photo enhancement, biomedical, et al. Beside these applications, the method remains several limitations such as: not working well with the problems which require understanding of geometric changes on the object.

CHAPTER 3
RELATED WORKS

3.1 CNN-based Medical Image Segmentation

Over the last few years, with the resurgence of deep learning and its application have led to the success of deep convolutional neural networks in the field of segmentation, image segmentation methods based on deep learning have made a large progress in terms of accuracy and efficiency. In particular, CNN-based model gave various successful models to date which can be devided into two categories, one-stage and two-stage segmentation method.

3.1.1 Single-stage segmentation methods

Single-stage segmentation methods which typically have an encoder-decoder structure like Unet. In the encoding part the network try to learn the representation feature of images by sliding convolutional kernel through whole images to learn different variety of edges or features in the images. For the decoding part, deconvolution operators are applied to each pixels which contain the information of instance in order to generate the instance mask.

3.1.1.1 Fully Convolutional Network(FCN)

As an popular approach to segmentation problem, Fully Convolutional Networks have a great influence on image semantic segmentation progress, which is proposed by Long et al. (2015) for pixel-wise labeling by replacing skip layer and bilinear interpolation with fully convolutional networks to expand the application of classification network to dense prediction. The authors proposed applying deconvolution operator to the output activation maps where the pixel-wise result can be calculated. Another important contribution of the authors is fusing the output with shallower layer's output so that preserve the contextual spatial in-

formation of an image as the filtered data progresses go deeper into the network. The architecture of the network is show in the below figure.3.1 The work has

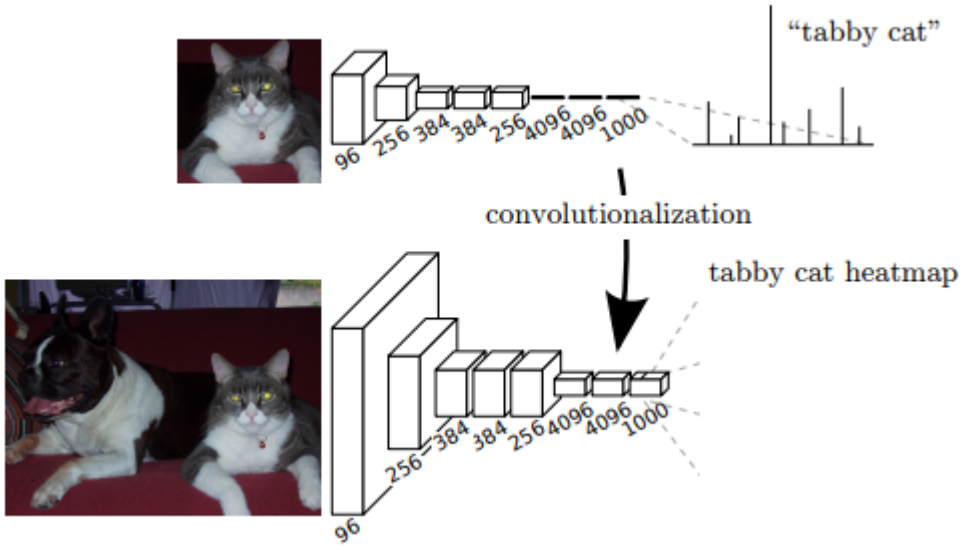


Figure 3.1: FCN architecture [55]

laid an evidently demonstration that deep networks can be trained in end-to-end architecture for image segmentation. Despite its efficiency, FCN can not capture the global context information of an object or volume in efficient way. Therefore many researcher try to overcome this problem by proposing method to improve the performance of FCN. Liu et al [56] proposed Parsenet, to address problems ignoring global context information by using the average feature for a layer to augment the features at each location to add global context to FCN.

Due to the effectiveness of FCNs-based method, a large number of researcher have applied it to medical image segmentation problems. Wang et al [57] leverage FCN-based method to segment multi-modal Magnetic Resonance images with brain tumor. Yuan et al [58] designed 19-layer deep convolutional neural networks to deal that is trained end-to-end to deal with automatic skin lesion segmentation task.

3.1.1.2 Unet-like model

U-Net [7] has been widely used as encoder-decoder Deep Learning based architecture for semantic segmentation that can produce highly reliable results on various metrics like: dice score, surface distance, etc. Unet consists of a downsampling fully convolutional network (FCN) followed by an upsampling FCN known as the network's contractive and expansive paths in the meanwhile the skip connections between the downsampling and upsampling branch are employed to provide local information to the global information while upsampling. Because of these attributes, the networks has a huge amount of feature maps in the upsampling path that can be transformed information from raw images to abstract label. An illustation of Unet is given in Fig. 3.2.

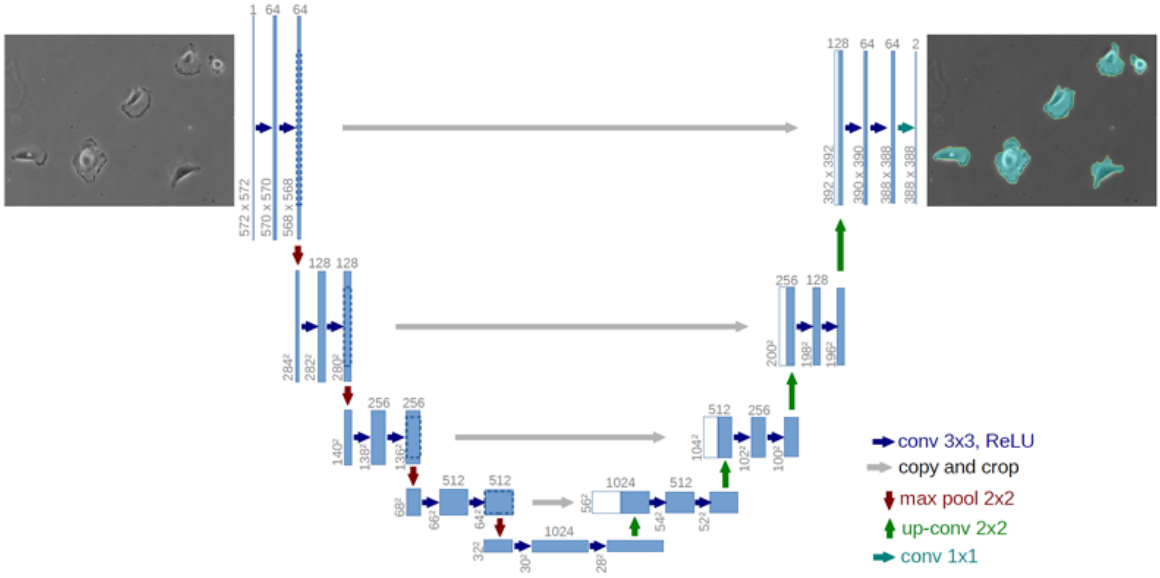


Figure 3.2: Unet architecture [7]

The potential application of Unet in biomedical image segmentation task is demonstrated by its success for being the winner method on the ISBI challenge for segmentation of neuronal structures in electron microscopic stacks, furthermore it has been shown the network work fast, take less than a second on typical

GPU to segment one 512x512 image. The success of Unet in medical image segmentation task has attracted attention of researchers in the world. Dong et al. [59] incorporated Unet-based model into data augmentation technique to solve the problem in brain tumor segmentation on Multimodal Brain Tumor Image Segmentation datasets (BRATS 2015). There are also multiple researches tackling the original Unet architecture trying to enhance the result. Such as Unet++ [60] - adding the skip connections, or in Double-U Unet [61] - adding a constraint of the reconstruction, or in U^2net [62] - leveraging the lower level feature as an attention to the regression loss. There also others focus more on the feature representation of the original Unet model, such as the Hyper-Dense Net [63] leveraging the features extracted from the input using Dense Net. Çiç ek et al.[8] further modified the U-Net architecture by replace 2D convolution operations with 3D convolution ones to create a model that can generalize well on 3D volumes, the Xenopus kidney without full annotation of 3D volumes because of few annotation label data problem in medical images, their work achieved good results on Xenopus kidney and highly variable 3D structure dataset. 3D CNNs with residual connections were also proposed in Deep Medic [64] which is an another successful deep learning approach in medical segmentation i.e. brain tumor segmentation. These 3D U-nets were shown to outperform current 2D medical imaging segmentation models in many 3D medical imaging datasets including prostate, kidney, brain tumor, infant brain segmentation. Thus, we employ 3DUnet as a framework that combine with the other proposed methods to improve the performance in medical image segmentation for our project.

3.1.2 Two-stage segmentation methods

This method for image segmentation consists of two stages: bounding box detection and semantic segmentation within each box. Among different CNN-based semantic segmentation approaches, Fully Convolutional Network and Mask-R-

CNN got enormous of attention which will be discussed.

3.1.2.1 Mask-R-CNN in image segmentation

Mask-R-CNN [65] extends Faster R-CNN to pixel-level image segmentation. Based on architecture of Faster R-CNN, it expanded the method for predicting an object mask in parallel with classification and localization. The mask network is a small- fully-connect network applied to each Region of Interest, predicting a segmentation mask in a pixel-to-pixel label mapping between raw images and mask predictions. In the first stage, the network detects objects and generate object proposals while the second stage is responsible for classifying these proposals to object bounding boxes and then generate masks. Mask-R CNN integrate Feature pyramid network as the backbones to generate region of interest features to increasing the accuracy in object detection phase.

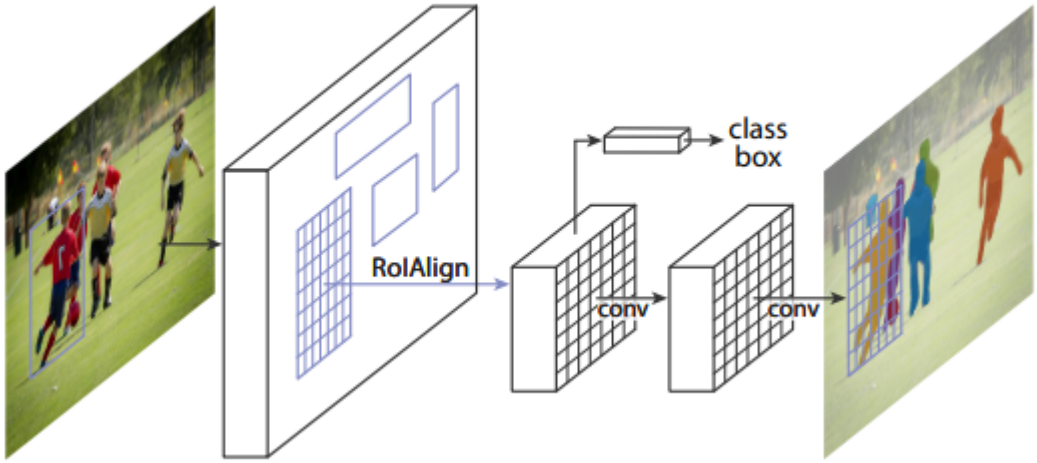


Figure 3.3: Mask-R-CNN architecture [65]

Multi-task loss is employed during training process to calculate the total loss on each sampled RoI as

$$L = L_{cls} + L_{box} + L_{mask} \tag{3.1}$$

respectively to L_{cls} is the classification loss over ground truth and predicting class, L_{box} is the regression loss of bounding boxes when there is an object. L_{mask} is calculated using the average binary cross-entropy loss. The author proposed new kind of loss function L_{mask} to helps the network with generating masks for every class without competition among classes based on the classification branch to predict class label used to select the output mask. This decouples mask and class prediction produced good instance segmentation results compared to FCNs, another two-stage image segmentation method, which uses a per-pixel softmax and a multinomial cross-entropy loss.

Though Mask-R-CNN is a good method for image segmentation, which was practically worked well on common object segmentation real life dataset like: Cityscapes, COCO, it's empirically not good at biomedical images segmentation task due to morphological variant shape of tissue in medical images such as cancer tissue especially brain MRI images.

3.2 GAN-based Medical Image Segmentation

Recently Generative adversarial networks(GANs) catched a lot of attention in biomedical images community because of their ability in data generation without neither explicitly modelling the probability density function or providing causality inference. Gan was proposed by Ian Goodfellow et at., 2015. GAN comprises of two networks which are trained simultaneously, with one called Generator and the other ones called Discriminator. The generator focuses on image generation while Discriminator network was trained to detect fake samples which were generated by the Generator network. An illustration of GAN is shown in the given image [4].

This has proven to be useful in many task such as image-to-image translation,

[4]Figure 3.4 is adopted from A Short Introduction to Generative Adversarial Networks

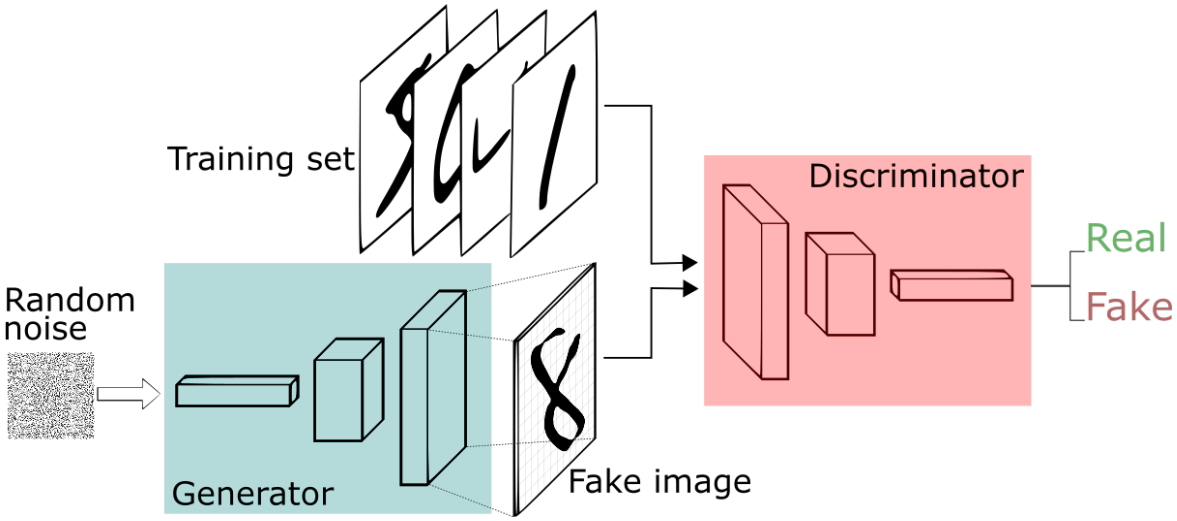


Figure 3.4: GAN architecture [4]

data augmentation relying on the generative aspect of GAN model, which can help in discovering the underlying distribution of objects or volumes in the training data then learning to generate new images. This property makes GANs very potential in dealing with data scarcity for medical image data. The basic framework for image-to-image translation has been proposed by Wolterink et al [66]. By training a CNN jointly with an adversarial CNN, the author aim at improving the CNN's ability to generate images with similar feature to that of reference routine-dose CT images. Chen et al [67] leverage GAN to solve the problem in reconstructing magnetic resonance images (MRI). In image reconstruction of organs, paired training samples are hard to get so Kang et al [68] proposed to use CycleGAN with an identity loss in denoising of cardiac CT. GAN-based models can also be used as an augmentation method, by translating from one type of medical images to others. One of its successful network with various applications (such as transferring PET to CT, correct MR motion, PET denoising, etc.) is the MedGAN, which tries to improve the global consistency using non-adversarial losses with conditional adversarial framework and a CasNET generator [69, 70]. Similarly to the idea of leveraging loss functions, the Perceptual Adversarial

Networks (PAN) [71] proposed a perceptual adversarial loss together with the generative adversarial loss build a novel loss function. Also, recently there exists a novel method named Cycle GAN Segmentation using the dataset from one domain with better segmentation result to translate to and enhance the segmentation result in another domain; the idea used here is trying to force the translated images to have the same segmentation result with it paired original real image [6], which will be detailed in the later sections.

The sucess of GAN and GAN-based method in medical imaging tasks motivate us using the property of generating new images to solve our problem in data scarcity. In particular, because of less annotation data in our dataset, in this project, we proposed to employ CycleGAN to help with guiding the weight initialization process during training time.

3.3 Active Contour-based Medical Segmentation

Though medical image segmentation method can detect true regions really well with deep learning based method, the sensitive noise causes the boundary of extracted region in the volume could be segmented inaccurately. Leveraging this observation in medical imaging, many scientists suggest using boundary refinement as an approach to medical image segmentation. Hadon and Boyce [72] proposed a two-stage method (initialize region segmentation then refine mask) with co-occurrence matrix used as a feature space and clusters within it are the considered regions and boundaries. Sato et al. [73] proposed a technique to obtain an accurate segmentation of 3D medical images for clinical applications by combining the gradients of the boundary and its neighbourhood pixels and then applies the gradient magnitude based on edge detectors such as Sobel detector for boundary improvement. Over the past few years, many efforts [74, 75, 76] have utilized Active Contour and have been proposed to segment the object with weak boundary. Among approaches, active contour (AC) methods

are powerful tools thanks to their ability to adapt their geometry and incorporate prior knowledge about the structure of interest. For instance, Level Set (LS) [13], an implementation of AC using energy functional minimization [24] has been proven to overcome the limitations of uniquely gradient-based models, especially when dealing with data sets suffering from noise and lack of contrast such as weak boundary. Li et al. [77] solved the problem of segmenting images with intensity inhomogeneity by using a local binary fitting energy. By minimizing the unbiased pixel-wise average misclassification probability, Wu et al. [78] formulated an active contour to segment an image without any prior information about the intensity distribution of regions. By realizing curve evolution via simple operations between two linked lists, Shi and Karl [79] achieved a fast level set algorithm for real-time tracking. Also, they incorporated the smoothness regularization with the use of a Gaussian filtering process and proposed the two-cycle fast (TCF) algorithm to speed up the level set evolution.

In addition to methods for multi-region image segmentation, including mean-shift clustering, spectral segmentation, greedy algorithms, learning approaches, level set-based segmentation is another common approach in computer vision. Level set -based multi-region image segmentation approaches either use a discrete labeling problem formulation and solve it using graph-cuts [80] or minimize the segmentation functional using convex relaxation techniques [81].

The traditional level set framework is geared towards binary-phase image segmentation. To overcome this limitation, various methods have been developed, including [82] which associates a level set function with each image region, and evolves these functions in a coupled manner. Later, [83] performs hierarchical segmentation by iteratively splitting previously obtained regions using the conventional level set framework. [84] suggested using a single level set function to perform the level set evolution for multi-region segmentation, it requires managing multiple auxiliary level set functions when evolving the contour, so that

no gaps/overlaps are created. [85] partitions an image into multiple regions by a single, piecewise constant level set function, which is obtained using either augmented Lagrangian optimization, or graph-cuts. Later, Li, et al. [86] proposed an adaptive regularized level-set method to ensure the level-set curve does not pass through weak object boundaries. New approaches [87], [88], [89] have been developed to replace the level set model, which investigate effective optimization schemes [90]. Generally, the level set model minimizes a certain energy function via gradient descent [91], making the segmentation results prone to getting stuck in local minima. To conquer this problem, Chan et al. [92] restated the traditional Mumford-Shah image segmentation model [24] as a convex minimization problem to obtain the global minimum. The above methods have obtained promising performance in segmenting high quality images. However, when attempts are made to segment images with heavy noise, this leads to poor segmentation results. Existing methods assume that pixels in each region are independent when calculating the energy function. This underlying assumption makes the contour motion sensitive to noise. In addition, the implementation of level set methods is complex and time consuming, which limits their application to large scale image databases. To maintain numerical stability, the numerical scheme used in level set methods, such as the upwind scheme or finite difference scheme, must satisfy the Courant-Friedrichs-Lewy (CFL) condition [93], which limits the length of the time step in each iteration and wastes time.

Recently, [94] utilized LS [13] into deep learning framework to improve segmentation performance on medical images. However, the two energy terms corresponding inside energy and outside energy are computed with assumption that the mean values of inside contour and outside contour are constants and set as 1 and 0. Furthermore, [94] applied LS [13] an entire image domain. Different from [94], our proposed network makes use of LS as an attention gate on narrow band around the contour. In addition, the mean values of inside contour and outside

contour in our framework are computed using the deep feature map from the network. Besides the weak boundary object, the unbalanced data problem in medical image segmentation has lately been gotten seriously attention [95]. In [95], a boundary loss was proposed and it is defined as a distance metric on the space of contours (or shapes), not regions, namely, the objective function is defined as a distance between two contours. Furthermore, the boundary loss [95] is implemented as distance between single pixel on the contour, which is high time consumption. Different from boundary loss [95] which considered as the distance between predicted boundary and groundtruth one, our proposed NB-AC loss treats the object contour as a hyperplane and all data inside a narrow band as support information that influences the position and orientation of the hyperplane. Our NB-AC loss with attention mechanism which focuses on on the contour length with the region energy involving a fixed-width band around the curve or surface.

Some limitations of variational level set approaches are observed as follows:

- They are unsupervised approaches and therefore require no learning properties from the training data. Thus, they have difficulty in dealing with noise and occlusions
- There are many parameters which are chosen by empirical results
- The are build off of gradient descent to implement the non-convex energy minimization and can get stuck in undesired local minima and thereby lead to erroneous segmentations
- Most of the level set based approaches are not able to robustly segment images in the wild
- They often give unpredictable segmentation results due to unsupervised

behaviors

- The accuracy of segmenting results strongly depends on the number of iterations which is usually set as a big number

3.4 Class Imbalanced Data

Most of the traditional classifiers assume the input data to be well-behaved in terms of class distributions, balanced size of classes, etc. However, high class imbalance is naturally inherent in many real world applications, robust classification with imbalanced data is an important area of research. Even the recent development of deep learning shows incredible performance in many domains along with its increasing popularity there is still few existing deep learning approaches for class imbalance. Thus, investigating the use of deep neural networks for problems of class imbalance is important and interest. This paper is examine existing deep learning techniques for addressing class imbalanced data.

Class imbalance has been studied thoroughly over the last decades using either traditional machine learning models, i.e. non-deep learning or advanced deep learning. Despite recent advances in deep learning, along with its increasing popularity, very little empirical work in the area of deep learning with class imbalance exists. The previous works using deep leraning to class imbalance can be mainly divided into three groups: data-level methods, algorithm-level methods and hybrid-level methods as follows:

- Data level methods: Those methods aims at altering the training data distribution by either adding more samples into minority class or removing samples from the majority class to compensate for imbalanced distribution between classes.

- Algorithm level methods: Those methods aims at making a modification to

the conventional learning algorithms to reduce bias towards the majority by adjusting misclassification costs.

- Hybrid methods: Those methods are a combination of the merits of both data level and algorithm level strategies

Three categories of solving imbalanced data problem are detailed as follows

3.4.1 Data level methods

This section explores data level methods for addressing imbalanced data with Deep Neural Networks. Most of these methods preprocess a dataset so that the number of labeled examples from both the classes become comparable. There are two approaches in this catergory: (i) under-sampling examples from the majority class ; (ii) over-sampling examples from the minority class.

Anand et al [96] proposed the first work which explores the effects of class imbalance on the backpropagation in a shallow network. The authors show that in the problem of imbalanced data, the majority class usually dominates the network gradient which is responsible for updating the model's weights. With such update, the error of the majority class is quickly reduced while the error of the minority class is increased. This causes the network to get stuck in a slow convergence mode.

Hensman and Masko [97] studied the impact of imbalanced training data on Convolutional Neural Network (CNN) performance in image classification. They showed that imbalanced training data can potentially have a severely negative impact on overall performance in CNN, and that balanced training data yields the best results. They conducted the experiments on The CIFAR-10 [98] benchmark dataset, comprised of 10 classes with 6000 images per class. The dataset is used to generate 10 imbalanced subsets for testing varying class sizes, ranging

between 6% and 15% of the total data set. In addition to varying the class size, the different distributions also vary the number of minority classes. Hensman and Masko chose a variant of the AlexNet [99] as backbone to perform classification task. The baseline performance was defined by training the CNN on all distributions with no data sampling. The over-sampling method being evaluated by randomly duplicating samples from the minority classes until all classes in the training set had an equal number of samples. Their imperial results have shown that over-sampling is a viable way to counter the impact of imbalances in the training data.

Lee et al.[100] incorporated transfer learning into under-sampling method to classify highly-imbalanced data sets of plankton classification on WHOI-Plankton dataset [101]. The data set contains 3.4 million images of over 103 classes where 90% of the images comprised of just five classes (the 5^th largest class makes up just 1.3% of the entire data set and with many classes make up less than 0.1% of the data set). Their approach contains two-phase learning procedure: In the first phase, a deep CNN is pre-trained with thresholded data. The thresholded data sets for pre-training are constructed by randomly under-sampling large classes until they reach a threshold of N examples. In the experimental results, the threshold is chosen as 5000 through preliminary experiments, then all large classes are down-sampled to N samples. In the second phase, the pre-trained model is fine-tuned using all data. Instead of completely removing potentially useful information from the training set as in naive under-sampling approach, the two-phase learning procedure only eliminates samples from the majority group during the pre-training phase. This allows the model to see all of the available data during the fine-tuning phase while helping the minority group to contribute more to the gradient during pre-training. In this work, they conducted the comparison on six methods which are combined with transfer learning and augmentation techniques. The imperial results have shown that under-sampling

aims at increasing the minority class performance while still preserving the majority class performance.

Instead of under-sampling over-sampling, Pouyanfar et al. [102] proposed a dynamic sampling technique in order to perform classification task on imbalanced image data. Their approach is to combine both over-sampling and under-sampling strategies which is to over-sample the low performing classes and under-sample the high performing classes. Their approach contains three core components: real time data augmentation, transfer learning, and a novel dynamic sampling method. The first method, various transformations are applied to select images in each training batch, where Inception-V3 network [103] is used to fined-tune the network which as pre-trained on ImageNet [104] the second method. The third method, dynamic sampling, which is able to self adjust sampling rates, is the main contribution to solve the class imbalance problem.

Recently, Buda et al [105] investigate the effects of class imbalance on classification of different deep learning frameworks under different data-level approaches, namely, over-sampling, under-sampling, two-phase training, and thresholding. Three popular datasets, namely, MNIST[106], CIFAR-10, and ImageNet together different CNN architectures were empirically selected. A improved version of the LeNet-5 [99] and the All-CNN [107] architectures were used as network backbone. From the empirical results, they have conducted that (i) The effect of class imbalanced data on classification performance is detrimental.; (ii) The impact of class imbalance on classification performance increases with the scale of a task.(iii)The influence of class imbalance not only depends on the by the lower total number of training cases but also the sample distribution among classes. In oder to decide which method is used to handle the class imbalanced data problem during deep neural network training, Buda et al suggested (i) Oversampling is the one that outperforms all others with respect to multi-class. (ii) Undersampling is the appropriate method in the case where extreme ratio

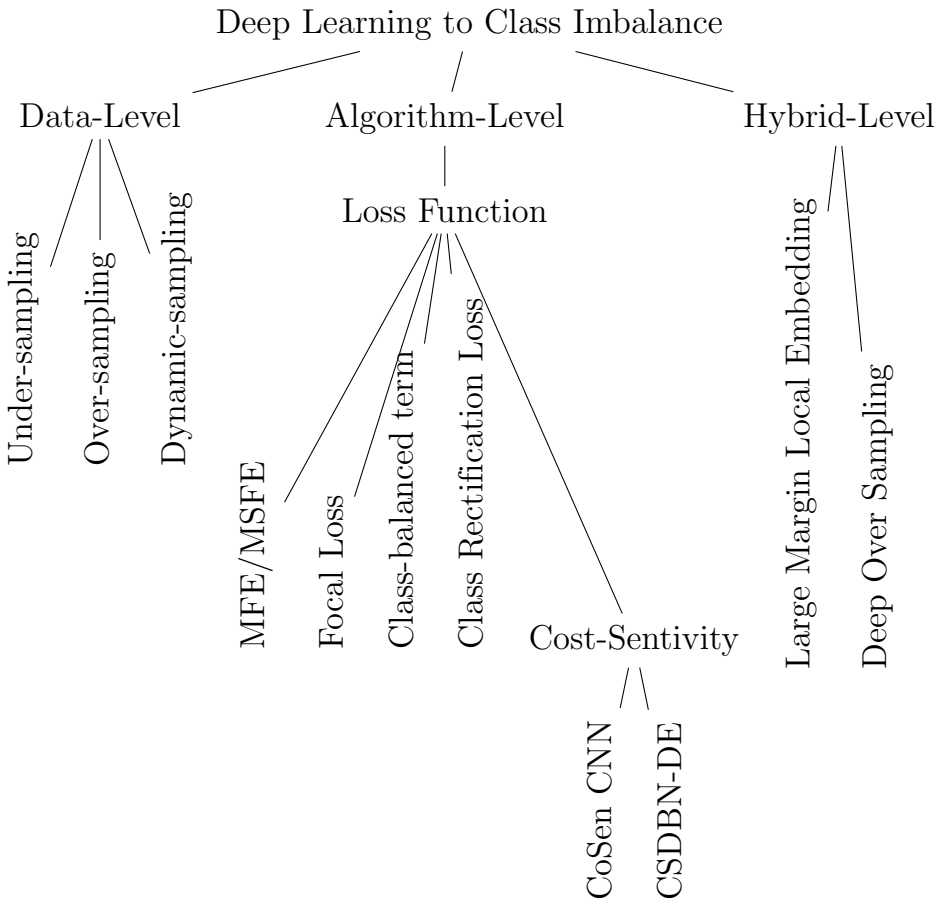


Figure 3.5: Summary of deep learning architectures to class imbalance problem

of imbalance and large portion of classes being minority. (iii) Undersampling is the choice training time is an issue. (iv) Ti achieve better accuracy, thresholding should be applied to compensate for prior class probabilities.

3.4.2 Algorithm level methods

In the context of deep feature representation learning using CNNs, data-level methods may either (i) introduce large amounts of duplicated samples, which slows down the training process and face to over-fitting problem when performing over- sampling, or (ii) discard valuable examples that are important for discriminating when performing under-sampling. Due to these disadvantages of applying under or over sampling for CNN training, the algorith,-level methods

focuses on how to design a better class-balanced loss. Far apart from the previous data-level methods which focus on changing data distribution, algorithm level methods focus on modifying deep learning algorithms. Wang et al. [108] proposed the loss function called mean false error together with its improved version mean squared false error for the training of deep networks on imbalanced data sets. To conduct the experiments, there are eight imbalanced binary datasets, including three image datasets and five text datasets collected. From the empirical results, the authors have shown that the mean squared error (MSE) loss function poorly captures the errors from the minority group in cases of high class imbalance, due to many negative samples dominating the loss function. They then proposed loss functions mean false error (MFE) and its improvement mean squared false error (MSFE) which outperform MSE loss in almost all cases and have prove to be able to handle the errors from the minority class. To effectively address the extreme foreground-background class imbalance encountered during training of dense detectors, Lin, et al.[109] proposed focal loss function which reshapes the cross entropy loss such that it low weights the loss assigned to well-classified examples. RetinaNet is a one-stage focal loss model which is evaluated against several state-of-the-art one-stage and two-stage detectors. In general, RetinaNet model one backbone which is responsible to produce feature maps from the input image, and the two subnetworks which are responsible to object classification and bounding box regression. The authors chose feature pyramid network (FPN) built on top of the ResNet [110] architecture as backbone model and it is pre-trained on ImageNet [99]. RetinaNet is trained on both standard cross entropy loss and the proposed focal loss. The experiments have shown that using standard cross entropy loss quickly fails and diverges due to the extreme imbalance whereas the proposed focal loss is able to outperform existing one-stage and two-stage object detection approaches. Focal loss is then used by Nemoto et al [111] for image classification task. The authors have concluded

that focal loss improves problems related to class imbalance and over-fitting by adjusting the per-class learning speed. In order jointly learns network weight parameters and class misclassification costs during training, Khan et al. [18] introduced an effective cost-sensitive deep learning (CoSen CNN) procedure which has been evaluated on six multi-class data sets. The VGG-16 [112] is used as baseline throughout the experiments. The feature map from VGG-16 is then modified by the cost matrix that is learned by the CoSen CNN which helps to give higher importance to samples with higher cost. The proposed cost is then incorporated into Mean Squared Error loss, Support Vector Machine hinge loss, and Cross Entropy loss. From the experiments, it is shown that the baseline CNN, with no class imbalance modifications, is a close runner-up to the CoSen CNN, outperforming the sampling methods, It is interesting that the baseline CNN, with no class imbalance modifications, is a close runner-up to the CoSen CNN, outperforming the sampling methods, Random Forest classifiers in all cases., and RF classifiers in all cases. Cost-sensitive in deep learning framework is continuelly studied by Zhang et al. [113]. In order to improve the cost matrix and incorporate these learned costs into a deep framework, Zong et al. use a differential evolutionary algorithm. Their proposed cost-sensitive learning approach, CSDBN-DE, has been evaluated against 42 datasets. In their proposed network, the cost matrix is incorporated into the output layer's softmax. Cost matrices are first randomly initialized and then updated by mutation and crossover operations during the training phase. Usually, the class imbalance problem has been evaluated on a small dataset and reach up to CIFAR-10. Zhang et al [114], bought the problem up to larger dataset on CIFAR-100 dataset. They proposed category centers which a combination of transfer learning, deep CNN feature extraction, and a nearest neighbor discriminator to address the class imbalance problem. The proposed approach is based on the observations that (i) the decision boundary made by the final layer of the CNN.(ii) similar im-

ages of the same class tend to cluster well in CNN deep feature space. Thus, they suggest to use high-level features extracted by the CNN to compute the class's centroid in deep feature space. The proposed category center helps to improve the classification performance on CIAFAR-10 but not on CIFAR-100. The proposed method is mainly depends on the category center, the classification boundaries may not be strong enough if the annotated training data is not available to pre-train the network. Focus on the facial action recognition, Ding et al. [115] experimented with very-deep network architectures to determine if deeper networks perform better on imbalanced data. They observe that a larger network contains more local minimum and produce better performance than a smaller network. One of the special case of imbalanced data, called long- tail: a few dominant classes claim most of the examples, while most of the other classes are represented by relatively few examples has been studied in Yin et al.[116]. In this work, they study the effective number of samples and show how to design a class-balanced term to deal with long-tailed training data. From the experiments, they show that adding the proposed class-balanced term to existing commonly used loss functions including softmax cross-entropy, sigmoid cross-entropy and focal loss helps to improve the performance. By considering minority samples as hard samples, Dong et al. [117] proposed Class Rectification Loss to avoid the dominant effect of majority classes by discovering sparsely sampled boundaries of minority classes. Their proposed method is based on batch-wise incremental hard mining of hard-positives and hard-negatives from minority attribute classes alone. Different from most of the other works that work on global clustering of the entire training data, Class Rectification Loss is independent to the overall training data size, therefore very scalable to large scale training data. They conducted the experiments two large scale datasets CelebA [118] and and X-Domain [119] and they conducted the comparisons against 11 different models. Not only in deep learning, the problem of class imbalance is also studied in deep

reinforcement learning by formulating the classification problem as a sequential decision-making process and solve it by deep Q-learning network as proposed in [120]. In their approach, the agent performs a classification action on one sample at each time step, and the environment evaluates the classification action and returns a reward to the agent. The reward from minority class sample is larger so the agent is more sensitive to the minority class. The agent finally finds an optimal classification policy in imbalanced data under the guidance of specific reward function and beneficial learning environment.

3.4.3 Hybrid-level Methods

In order to learn more discriminative deep representations of imbalanced image data, Huang et al. [121] proposed Large Margin Local Embedding method. The proposed methods is based on observation that the minority groups are sparse and typically contain high variability, allowing the local neighborhood of these minority samples to be easily invaded by samples of another class. Their method to enforce the local cluster structure of per class distribution in the deep learning process so that minority classes can better maintain their own structures in the feature space. In their approach, the CNN is trained with instances selected through a new quintuplet sampling scheme and the associated triple-header hinge loss. However, their proposed method has a number of fundamental drawbacks including disjoint feature, quintuplet construction updates and classification optimisation. Ando et al. [122] introduced Deep over-sampling which incorporates over-sampling into the deep feature space produced by CNNs. Their proposed method contains two simultaneous learning procedures: optimizing the lower layer for acquiring the embedding function and upper layer parameters to discriminate between classes using the generated embeddings. Their proposed approach address the effect of class imbalance on both classifier and representation learning by introducing a general re-sampling framework to learn the deep

representation and the classifier jointly in a class-imbalanced setting without substantial modification on its architecture. In this method, the training data is first augmented by assigning multiple synthetic targets to one input sample. Then, process of learning the CNN and updating the targets with the acquired representation enhances the discriminative power of the deep feature.

3.5 Loss function

To train a Deep Neural Network (DNN), the loss function, which is known as cost function, plays a significant role. Loss function is to measure the average (expected) divergence between the output of the network (P) and the actual function (T) being approximated, over the entire domain of the input, sized $m \times n$. We denote i as index of each pixel in an image spatial space $N = m \times n$. The label of each class is written as c in C classes. Herein, we briefly review the some common loss functions.

3.5.1 Cross Entropy (CE) Loss

Cross Entropy loss is a widely used pixel-wise distance to evaluate the performance of classification or segmentation model. In CE loss function, the output from softmax layer (P) is classified and evaluated against the groundtruth (T). For binary segmentation, CE loss is expressed as Binary-CE (CE) loss function as follows:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N}[T_i \ln(P_i) + (1 - T_i)\ln(1 - P_i)] \tag{3.2}$$

The standard CE loss has well-known drawbacks in the context of highly unbalanced problems. It achieves good performance on a large training set with balanced classes. For unbalanced data, it however typically results in unstable training and leads to decision boundaries biased towards the majority classes. To deal with the imbalanced-data problem, two variants of the standard CE loss,

Weighted CE (WCE) loss and Balanced CE (BCE) loss are proposed to assign weights to the different classes.

In medical image segmentation, a common strategy is re-balancing class prior distributions by down-sampling frequent labels [123]. However, this strategy ignore some useful information during training. To deal with the imbalanced-data problem, two variants of the standard CE loss, Weighted CE (WCE) loss and Balanced CE (BCE) loss are proposed to assign weights to the different classes. WCE, BCE losses assign more importance to the rare labels and defined as $WCE(T,P) = -\frac{1}{N}\sum_{i=1}^{N}[\beta T_i \ln(P_i) + \gamma(1-T_i)\ln(1-P_i)]$, where $\beta > 1$ is to decrease the number of false negatives and where $\beta < 1$ is to decrease the number of false positives. In WCE loss, $\gamma = 1$ whereas $\gamma = 1 - \beta$ in BCE.

3.5.2 Dice loss

Dice loss is proposed by [11]. It measures the degree of overlapping between the reference and segmentation. Dice loss comes from Dice score which was used to evaluate the segmentation performance. In general, it is defined as follows:

$$\mathcal{L}_{Dice} = 1 - 2\frac{\sum_i^N T_i P_i}{\sum_i^N T_i + P_i} = 2\frac{T \cap P}{T \cup P} \tag{3.3}$$

Even though Dice loss has been successful in image segmentation, it is still pixel-wise loss and has similar limitations as CE loss. Despite Dice loss improvements over CE loss, Dice loss may undergo difficulties when dealing with very small structures [124] and weak object boundary as missclassifying a few pixels can lead to a large decrease of the coefficient.

3.5.3 Focal Loss

Focal Loss is proposed by [12], Focal loss is a modified version of CE loss. It is to balance between easy and hard samples as follows:

$$\mathcal{L}_{Focal} = \frac{\alpha_i}{N} \sum_{i=1}^{N} \Big((1 - P_i)^{\gamma} T_i \ln(P_i) + P_i^{\gamma} (1 - T_i) \ln(1 - P_i) \Big) \tag{3.4}$$

In Focal loss, the loss for confidently correctly classified labels is scaled down, so that the network focuses more on incorrect and low confidence labels than on increasing its confidence in the already correct labels. The loss focuses more on less accurate labels than the logarithmic loss when $\gamma > 1$.

CHAPTER 4
METHOD

Medical image segmentation is one of the most challenging tasks in medical image analysis and widely developed for many clinical applications. Although deep learning-based approaches have achieved impressive performances in semantic segmentation, their limitations on pixel-wise are extant with **imbalanced-class data problems** and **weak boundary object segmentation** in medical images as well as **less annotation data**. Therefore in order to tackle the weak boundary object problem in medical imaging segmentation we propose the active contour model that focus on the boundary/surface. To address the aforementioned imbalanced-class data problems, our network inherits the advantages of narrow band theory under the zero level set energy minimization. As for the last problem of less annotation data, GAN is employed as a model which helps with data augmentation by transferring images from one domain to another domain. To begin with, we also conduct a motivating experiment in section 5.2 to check whether active contour has a promising result on our considered medical datasets, firstly as a post-processing method on previously trained Unet (guided with GAN).

4.1 Motivation

Considering problems related to boundary in image segmentation, boundary refining methods have always been the well-known solutions. As for medical image segmentation, in general, active contour approaches have been shown to be effective. This motivated us to experiment some of these methods (the active-contour-based methods) to scrutinize their efficiency on our collected medical images. Our collected medical images are 3D images, which allowed us to use them as either 3D inputs or slicing them as 2D inputs.

We trained a 2D deep snake model [125] on the training set of iSeg 2019 dataset and tested with its test set. Via this experiment, this deep snake model has successfully detected the target regions of different brain tissues; but for the segmented boundaries, the outputs are smooth and loosely snapped to the target boundaries. This is why we decided to move to older snake versions, which are expected to provide us better customization on the energy extracted from a given image and a mask.

Originally, the active contour or level set based methods aim to solve the energy minimization equation (section 3.3) to refine the boundary from a **single-label initial boundary** and a **single input image**. This experiment is based on Chanvese 3.3 method using the result of the Cycle GAN Segmentation (section 4.3.1, 5.3.3) model to extract the initial boundary for each labels. For post-processing using active contour based on Chanvese, we tried several approaches for several problems.

Firstly, for **multi-modality input**, we decided to use the difference in intensities of the two T1- and T2- weighted as the input for the Chanvese model (refer to 2 for reasons), as T1-w and T2-w mostly have flipped contrasts (table 2.1). We also tried using concatenation of the two images, the intensity sum of the two images, and resulted in lower segmentation accuracy. The difference in intensities of T1- and T2- weighted, however, cannot successfully represent the information from both these images so that we do not expect the active contour post-processing method to improve the result much. This also motivates us to move further to the later approach, the Narrow Band - Active Contour loss. Briefly understanding, Unet can be considered as a feature extractor to obtain a better representative features for the two inputted images T1- and T2- weighted to successfully calculate the energy of the given "images" and the given contour (from ground truth for the training dataset), with the hope of this energy function will guide the Unet model to "extract" the image so that the actual

boundary regions seems to be boundaries viewing through the active contour model.

Secondly, for **multi-label** segmentation problem, especially for this brain tissues segmentation problems as active contour based methods are often known to be sensitive to large noise regions. Hence, the first step is to reduce the affect of the "noises" (in calculating the energy functions), especially the black background in each slides. As the black background regions is relatively large in comparison to considered parts of our segmented tissues, and especially have higher intensity contrast with the considered tissues (intensities contrast between regions are not as large), keeping this region means adding a force to move the initial boundaries to the background-foreground brain region (where we can understand as skull regions).

Experiments shown that for brain MRI images (iSeg in this case), the tissue regions intensity contrast are not high enough for remarkably moving the boundaries toward the target gradient regions. Please note that the T1-weighted minus T2-weighted does not expected to highly represent the two inputted images and that the initial boundary is extracted from a high result model which also expected not to be changed much, except for the mis-segmented regions. An example for the low intensity contrast between tissues (on the originally low contrast 6-month MRI brain images): the initially segmented white tissues, the inside region intensities does not seem to be much different from the outside region intensities (after removing the background). This is the reason why we later choose narrow band instead of using the whole inside and outside regions to calculate the energy.

As this is brain tissues segmentation, the available regions between tissues are not expected to exist, in other words, no holes between regions as well as no overlapping regions. Hence, we also added another constraint for the above post-

processing method, which is the constraint for boundary movement not to move too much into other tissues (in comparison to the initial segmented masks).

In brief, because this is used as a post-processing techniques, we decided to use the output from previous method based on cycle GAN segmentation as the initial boundaries. This initial masks have already acquire an acceptable result, considering DSC; So we would like to restrict the evolution process of this post-process method to be evolve only in some regions. The regions is defined by the initial masks. For more details, please visit section 5.3.1. Given the result obtained by this post-processing experiment, we attempted to build an end-to-end model as in section 4.2.

4.2 Proposed Active Contour Unet

Our proposed Active Contour Unet (AC-Unet) is motivated by the minimization problem of CV's model [13] to efficiently find a contour by minimizing an energy functional. To address the limitations of CV's model, we conduct an attention model to focus on parallel curves of the contour. In the following equations, ground truth and predicted output are denoted as $\mathbf{T}$ and $\mathbf{P}$, where $\mathbf{T}, \mathbf{P} \in [0,1]^{H \times W}$ and H and W are the height and weight of $\mathbf{T}$. Our proposed network loss contains three branch corresponding to higher-level feature , intermediate-level feature and lower-level feature loss as follows. Fig.4.2 shows our proposed segmentation network architecture. Our proposed Active Contour Unet is based on offset curves thoery as follows:

4.2.1 Offset Curves Analysis

The theoretical background of offset curves is based on the theory of parallel curves and surfaces [126, 127]. An illustration of offset curve theory is given in Fig. 5.1. In Fig.5.1(A), the curve Γ, where $\Gamma : \Omega \to \mathbb{R}^2$ is called a parallel curve of

$\Gamma^{\mathcal{B}}$ (either outer curve $\Gamma^{+\mathcal{B}}$ or inner curve $\Gamma^{-\mathcal{B}}$) if its position vector $c^{\mathcal{B}}$ satisfies:

$$c^{\mathcal{B}}(z) = c(z) + \mathcal{B}n(z) \tag{4.1}$$

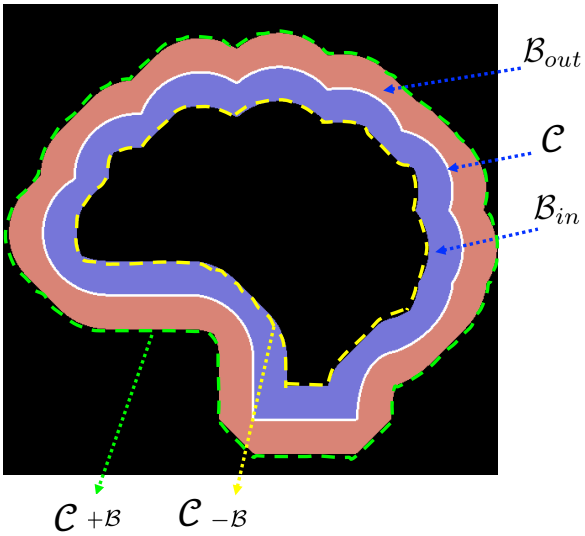
where $z \to c(z) = [x(z), y(z)]$, x and y are continuously differentiable with respect to parameter z and $\Omega \in [0, 1]$. $\mathcal{B}$ is the amount of translation, and n in the inward unit normal of Γ. Based on this equation, the inner band $\mathcal{B}^-$ and outer band $\mathcal{B}^+$ are bounded by parallel curves $\Gamma^{+\mathcal{B}}$ and $\Gamma^{-\mathcal{B}}$. This implies that both curves are continuously differentiable and do not exhibit singularities. Fig.5.1(B) shows a case where band width (translation) B_1 is smaller than the curve's radius of curvature whereas B_2 is larger than the curve's radius of curvature. An important property resulting from the definition of the Eq.4.1 is that the velocity vector of parallel curves depends on the curvature of Γ. That means, the velocity vector of curve $\Gamma^{\mathcal{B}}$ is expressed as a function of the velocity vector, curvature and normal of Γ. Set $n(z) = -\alpha c(z)$, we have

$$c^{\mathcal{B}}(z) = c(z) + \mathcal{B}n(z) = (1 - \alpha\mathcal{B})c(z) \tag{4.2}$$

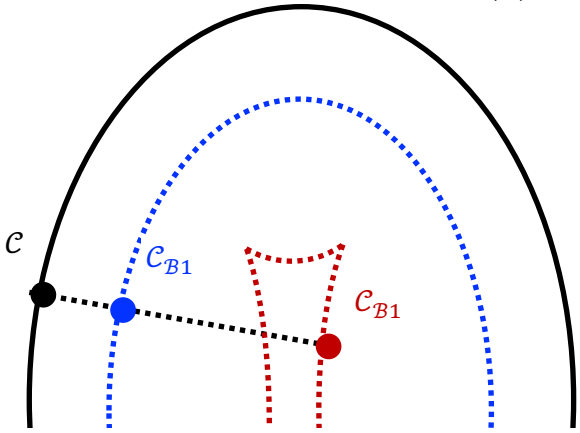
That equation provides the length element of inner parallel curve:

$$l^{\mathcal{B}} = ||c^{\mathcal{B}}(z)|| = l^{\mathcal{B}}(1 - \alpha\mathcal{B}) \tag{4.3}$$

This is also a result in parallel curve theory in [128]. Because the length $l^{\mathcal{B}}$ is also positive, the band width should not exceed the radius of curvature it is expressed as $\frac{-1}{\mathcal{B}} < \alpha < \frac{1}{\mathcal{B}}$. Is this constraint satisfies, the curves $\Gamma^{+\mathcal{B}}$ and $\Gamma^{-\mathcal{B}}$ are simple and regular.



(a) Illustration of inner band $\mathcal{B}_{in}$ and outer band $\mathcal{B}_{out}$ of a contour($\mathcal{C}$)



(b) Main curve $\mathcal{C}$ (black) and two parallel curves: blue curve $\mathcal{C}_{\mathcal{B}1}$ is generated by a small bandwidth of translation; red curve $\mathcal{C}_{\mathcal{B}2}$ is generated by larger bandwidth of translation.

Figure 4.1: Demonstration of offset curve theory

4.2.2 Higher Level Feature Branch

The first branch of the network is a standard segmentation CNN which can utilize any encoder-decoder network such as Unet [7], FCN [55]. Unet [7] has been widely as end-to-end and encoder-decoder framework for semantic segmentation

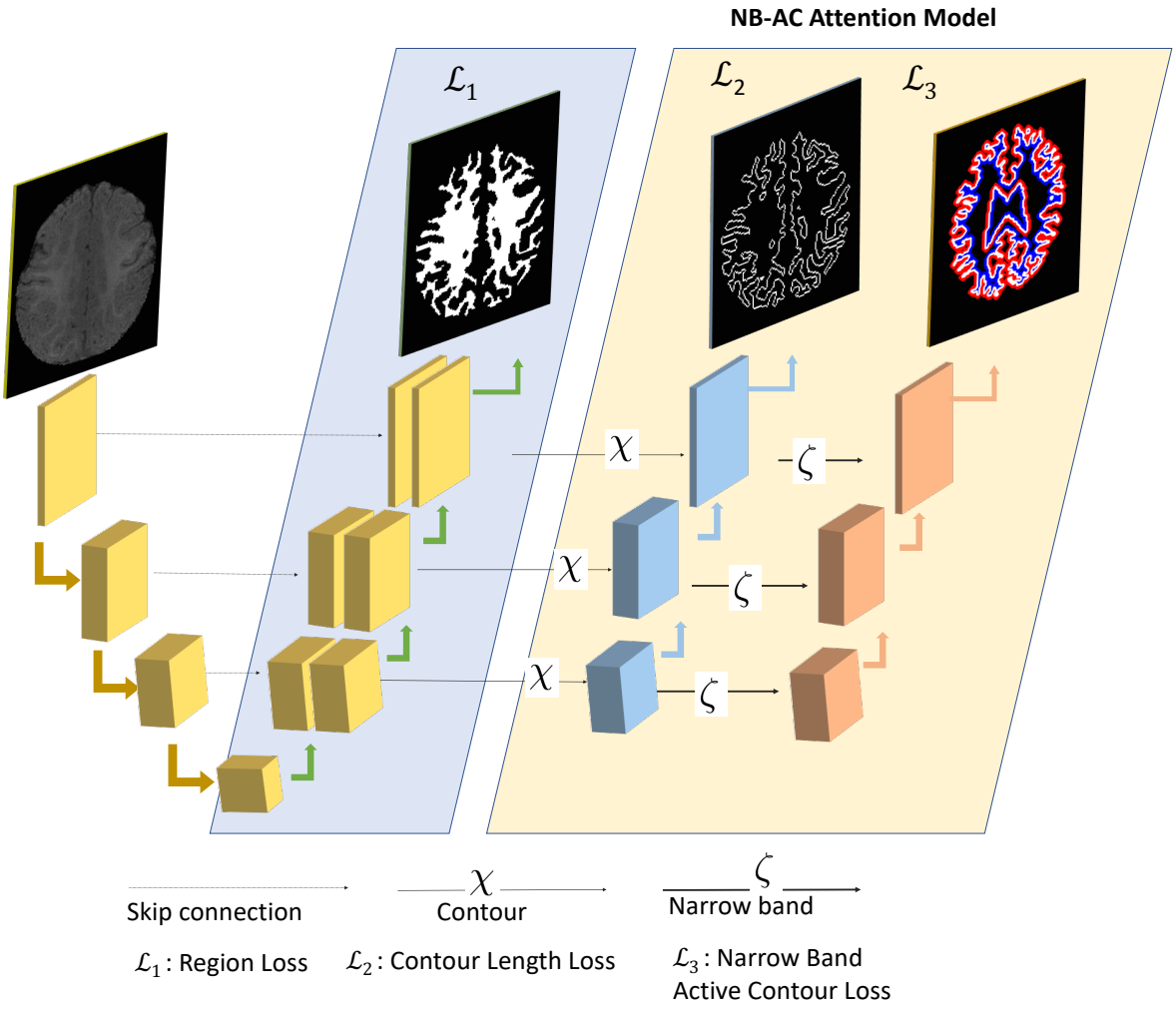


Figure 4.2: Proposed Active Contour Unet architecture

with high precise results. One of the most important building blocks is skipped connections which are designed for forwarding feature maps from down-sampling path to the up-sampling path in order to localize high resolution features. Fully convolutional networks (FCN) [55] also consists of two paths: down-sampling and up-sampling paths. The down-sampling path aims to increase the receptive-field via convolution and pooling layers. In the up-sampling path, the intermediate features are up-sampled to the input resolution by bi-linear operators. Both Unet and FCN network architectures are chosen as the network backbones in our experiments. More formally, for a region segmentation of K classes, the first

branch outputs the categorical distribution and the loss is computed as:

$$\mathcal{L}_1 = -\sum_{c=1}^{K} y_o^c log p_o^c \tag{4.4}$$

where y_o^c is binary indicator (0 or 1) if class label 'c' is the correct classification for observation 'o' and p_o^c is predicted probability observation 'o' is of class 'c'.

4.2.3 Transitional Gate

In semantic segmentation, both object region and object contour are closely related, thus, we present a transitional gate that aims at transferring information from the first branch to the second branch. The transitional gate acts as a filter that focuses on extracting lower level feature and removing irrelevant information from higher level feature. Let denote the output feature representation of the first branch as $F_{\mathcal{H}}$. The output from NB-AC attention model in the second branch is denoted as $F_{\mathcal{L}}^C$ and $F_{\mathcal{L}}^N$ corresponding to contour feature map and narrow band feature map. The contour feature map $F_{\mathcal{L}}^C$ is obtained by applying edge extraction operator χ on the higher level feature map $F_{\mathcal{H}}$ and the narrow band feature map $F_{\mathcal{L}}^N$ is obtained by applying parallel curves operator ζ on $F_{\mathcal{L}}^C$. In our experiments, χ and ζ are chosen as the gradient operator and the dilation operator, respectively. Our NB-AC loss is flexibly incorporated into both 2D and 3D frameworks. In 2D frameworks, the gradient operator (χ) is defined as either 3×3 convolutional layer and dilation operator (ζ) is defined as $\mathcal{B} \times \mathcal{B}$ where $\mathcal{B}$ is the width of narrow band. In 3D frameworks, the gradient operator (χ) is defined as either $3 \times 3 \times 3$ convolutional layer and dilation operator (ζ) is defined as $\mathcal{B} \times \mathcal{B} \times \mathcal{B}$ where $\mathcal{B}$ is the width of narrow band.

$$F_{\mathcal{L}}^C = \chi(F_{\mathcal{H}}) \text{ and } F_{\mathcal{L}}^N = \zeta(F_{\mathcal{L}}^C) \tag{4.5}$$

4.2.4 Lower Level Feature Branch

Our proposed NB-AC attention model in the second branch is motivated by the minimization problem of CV's model [13]. CV's model is to efficiently find a boundary (object contour) by automatically partitioning an image into two regions based on global minimizing active contour energy. The level set function Φ splits the image domain Ω into an inner region $\Omega_I = \Phi > 0$, an outer region $\Omega_O = \Phi < 0$ and on the contour $\Phi = 0$. However, CV's model makes strong assumptions on the intensity distributions and homogeneity criterion, which are usually expressed over regions inside and outside of the contour. Instead of dealing with the entire domains Ω defined by the evolving curve, we only consider the narrow band $\mathcal{B}_{in} \bigcup \mathcal{B}_{out} \bigcup \mathcal{C}$ which is formed by the inner band domain $\mathcal{B}_{in}$, outer band domain $\mathcal{B}_{out}$ from two sides of the curve $\mathcal{C}$ and the curve $\mathcal{C}$ itself (note: $\mathcal{C}$ is presented by $\Phi = 0$), as depicted in Fig.4.1. Our NB-AC loss of the second branch is defined in Eq.4.6:

$$\begin{aligned} \mathcal{L}_2 =& \mu \int_{\omega} |Length(\Phi)| dxdy \\ \mathcal{L}_3 =& \lambda_1 \int_{\mathcal{B}_{in}} |p - b_{in}|^2 dxdy + \lambda_2 \int_{\mathcal{B}_{out}} |p - b_{out}|^2 dxdy \end{aligned} \tag{4.6}$$

where the first term defines smoothness which is equivalent to the length of the contour, the second term defines the inner band energy, the last term defines outer band energy. p is the predicted feature map. By applying the transitional gate (Sec.4.2.3), we can rewrite Eq.4.6 in term of domain Ω as follows:

$$\begin{aligned} \mathcal{L}_2 =& \mu \int_{\Omega} |F_{\mathcal{L}}^{C}(x,y)| dxdy \\ \mathcal{L}_3 =& \lambda_1 \int_{\Omega} |p(x,y) F_{\mathcal{L}}^{N}(x,y) - b_{in}|^2 dxdy + \lambda_2 \int_{\Omega} |p(x,y) F_{\mathcal{L}}^{N}(x,y) - b_{out}|^2 dxdy \end{aligned} \tag{4.7}$$

where b_{in} and b_{out} are intensity descriptors of $\mathcal{B}_{in}$ and $\mathcal{B}_{out}$, respectively.

$$\begin{aligned} b_{in} &= \frac{\int_{\Omega} p(x,y) F^{y}_{\zeta\chi}(x,y) dxdy}{\int_{\Omega} F^{y}_{\zeta\chi}(x,y) dxdy} \text{ and} \\ b_{out} &= \frac{\int_{\Omega} p(x,y)(1 - F^{y}_{\zeta\chi}(x,y)) dxdy}{\int_{\Omega} (1 - F^{y}_{\zeta\chi}(x,y)) dxdy} \end{aligned} \tag{4.8}$$

where $F^{y}_{\zeta\chi}$ is the narrow band of the groundtruth y and is computed by first applying the gradient operator (χ) to extract the gradient and then applying a dilation operator ζ to get the narrow band, namely, $F^{y}_{\zeta\chi} = \zeta(\chi(y))$.

Our proposed NB-AC loss archives good flexibility thanks to the narrow band principle which does not carry a strict homogeneity condition. The theory of our proposed NB-AC attention model comes from the parallel curve also known as "offset curves" [126]. As given in Fig.4.1, the curve $\mathcal{C}_{\mathcal{B}1}$ or $\mathcal{C}_{B2}$ ($\mathcal{C}_{\mathcal{B}}$ in general) is called a parallel curve of $\mathcal{C}$ if its position vector $\mathcal{I}_{\mathcal{B}}$ satisfies:

$$\begin{aligned} &\mathcal{C} : \Omega \to \mathbb{R}^2 \\ &z \to \mathcal{I}(z) = [x(z), y(z)] \\ &\mathcal{I}_{\mathcal{B}}(z) = \mathcal{I}(z) + \mathcal{B}n(z) \end{aligned} \tag{4.9}$$

where x and y are continuously differentiable with respect to parameter z and $\Omega \in [0, 1]$. $\mathcal{B}$ is the amount of translation, and n in the inward unit normal of $\mathcal{C}$. An important property resulting from the definition of Eq.4.9 is that the velocity vector of parallel curves depends on the curvature of $\mathcal{C}$. That means, the velocity vector of curve $\mathcal{C}_{\mathcal{B}}$ is expressed as a function of the velocity vector of $\mathcal{C}$ and its curvature and normal. Set $n(z) = -\kappa\mathcal{I}(z)$, we have:

$$\mathcal{I}_{\mathcal{B}}(z) = \mathcal{I}(z) + \mathcal{B}n(z) = (1 - \kappa\mathcal{B})\mathcal{I}(z) \tag{4.10}$$

Apply Eq.4.10 to the curves in Fig.4.1, we obtain the length element (or velocity)

of outer parallel curve $\mathcal{C}_{+\mathcal{B}}$: $l_{+\mathcal{B}} = ||\mathcal{I}+\mathcal{B}n(z)||$, the length element of inner parallel curve $\mathcal{C}_{-\mathcal{B}}$: $l_{-\mathcal{B}} = ||\mathcal{I}-\mathcal{B}n(z)||$. Based on the above offset curve theory, the inner band $\mathcal{B}_{in}$ and outer band $\mathcal{B}_{out}$ (in Fig.4.1) are bounded by parallel curves $\mathcal{C}_{-\mathcal{B}}$ and $\mathcal{C}_{+\mathcal{B}}$.

In our proposed network architecture, the second branch focuses on only the information around the contour and on the contour itself, i.e. $\mathcal{B}_{in} \bigcup \mathcal{B}_{out} \bigcup \mathcal{C}$ as in Fig.4.1. This aims at addressing not only the problem of weak boundary object segmentation but also the imbalanced data problem. In image segmentation, each pixel is considered as a data sample and needs to be classified. The second branch can be seen as an under-sampling approach where all data samples inside the $\mathcal{C}_{-\mathcal{B}}$ and outside of $+\mathcal{B}$ (i.e. not in the narrow band) are ignored and only data samples between the narrow band formed by $\mathcal{B}_{in} \bigcup \mathcal{B}_{out} \bigcup \mathcal{C}$ are kept for predicting. One can think contour $\mathcal{C}$ plays the role of hyperplane and all data samples inside narrow band play the role of support vectors which influence the position and orientation of the hyperplane.

4.2.5 Network Architecture

The architecture of our proposed two-branch network is illustrated in Fig.4.2 where we choose Unet framework for this demonstration. The first branch is designed as a standard encoder-decoder segmentation network. The second branch is composed of residual blocks interleaved with transitional gates (in subsec.4.2.3) which ensures that the second branch only processes boundary-relevant information (edge and narrow band). Our proposed network is designed as an **end-to-end framework**. The losses from both branches are combined as:

$$\mathcal{L}_{NB-AC} = \gamma_1 \mathcal{L}_1 + \gamma_2 \mathcal{L}_2 + \gamma_3 \mathcal{L}_3 \tag{4.11}$$

where γ_1 and γ_2, and γ_3 are three hyper-parameters that control the weighting between the losses and chosen as $\gamma_1 = 0.6$, $\gamma_2 = 0.2$ and $\gamma_3 = 0.2$ in our experiments.

In this work, we use 2D Unet [7] and 2D FCN [55] architectures as our base segmentation frameworks to evaluate our proposed NB-AC loss function performance in the case of 2D input. Furthermore, we use 3D Unet [8] to evaluate the propoed NB-AC loss function in the case of 3D input. In Unet, feature maps from down-sampling path is forwarded to the up-sampling path by skip connections. Each layer in the down-sampling path consists of two 3×3 convolution layers ($3 \times 3 \times 3$ in 3D Unet), one batch normalization (BN), one rectified linear unit (ReLU) and one max pooling layer. In the up-sampling path, bilinear interpolation is used to up-sample the feature maps. In FCN framework, we choose FCN-32 which produces the segmentation map from conv1, conv3, conv7 by using a bilinear interpolation. At the down-sampling path, each layer in FCN is designed as same as layer in 2D Unet.

4.3 Active Contour Unet with Guided Segmentation

To improve the segmentation result in less annotation data, we also experiment transferring the knowledge obtained by the Cycle GAN guided segmentation model proposed in [6] to a Unet model trained with our proposed loss. Firstly, we trained two independent Unet models separately and respectively on two datasets (6m infant brain and 24m brain images). Then we freeze these two models and train the Cycle GAN segmentation model detailed below in section 4.3.1. Later, we use our trained Unet model with our proposed Narrow Band - Active Contour loss on 3D 6m-infant-brain dataset to learn the knowledge the Cycle GAN segmentation model transferred from 24m-brain to 6m. The result of this method (the Active Contour guided by GAN) is presented in section 5.3.2.

4.3.1 Cycle GAN segmentation

This model, proposed by Toan Duc Bui et al. [6], aims to transfer the 3D 24-month brain images to 6-month brain images which share the same tissues-segmentation result. The figure 4.3 below shows the overview of this method.

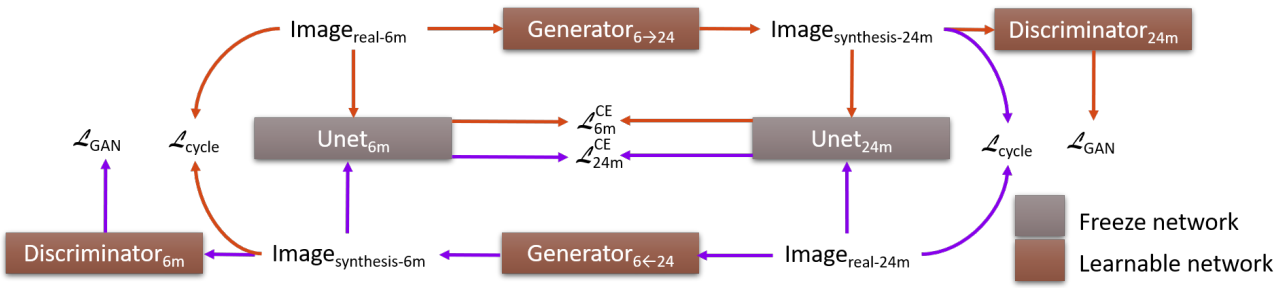


Figure 4.3: Cycle GAN guided segmentation [6]

The first step is to train U-net segmentation for infant and adult brain data independently, respectively called S_X and S_Y for distinguishment. The idea of using (previously-and-independently trained) U-net here is to ensure that the segmentation image of the synthetic images transferred by CycleGAN is estimatedly the same as its origin's, comparing via cross-entropy (CE) loss. As shown in figure 4.3, the real 6-month image will get through the 3D Cycle GAN Segmentation model to generate a synthetic 24-month image and vice versa.

For specific, scrutinizing figure 4.3, the orange line (or similarly, the purple one): real 6-month images will be used to generate synthetic 24-month images. The 24-month discriminator of the the Cycle GAN will judge whether the fabricated images is real or not. Simultaneously, each pair of the original-and-unnatural will get through the Unet models to produce the segmentation results and these two will be compared with segmentation loss (S_X and S_Y respectively output $mask_X$ and $mask_Y$ and CE loss will be calculated from these two). In figure 4.3, the orange line, together with the purple line, yields the basic flows of the

3D-CycleGAN-Seg.

Objective function. As mentioned above, the proposed model is generally based on multiple different constraint such as cycle-consistency for segmented features, discriminator loss for generated images, and feature matching loss for image quality enhancement. Considering all the loss functions above to update weights while training the Cycle GAN Segmentation network gives us a final objective function.

Cycle GAN loss. To transfer the appearance between two times-points of unpaired set X, Y which X is 6-month phase and Y is 24-month phase. The authors use the Cycle GAN network to guide the segmentation. The architecture of Cycle GAN contains two generators and $G = \{G_X,\ G_Y\}$ and two discriminators $D = \{D_X,\ D_Y\}$. The generator generates new image G_Y from G_X in particular it transfers the image appearance from 6-month time-point to the 24-month time-point Y. The discriminator D has the same function with GAN which is providing the feedback for generator to generate more real images.

$$\begin{aligned}\mathcal{L}_{cycleGAN}(G_X, G_Y, D_X, D_Y) = \mathcal{L}_{GAN}(G_Y, D_X) \\ + \mathcal{L}_{GAN}(G_Y, D_Y) \\ + \lambda\mathcal{L}_{cycle}(G_X, G_Y) \\ + \beta\mathcal{L}_{identity}(G_X, G_Y)\end{aligned} \tag{4.12}$$

With $\mathcal{L}_{GAN}$ is the GAN loss (for D networks), $\mathcal{L}_{cycle}$ is the cycle-consistency loss so as to ensure that $G_X(G_Y(x)) \approx x$. And the $\mathcal{L}_{\text{identity}}$ (from Cycle GAN) [16] is to guide the generator in mapping synthesis images domain to the target one faster, with λ & β as parameters.

$$L_{\text{identity}}(G,F) = E_{y\sim p_{data(y)}}[\Big\|G(y)-y\Big\|_1] + E_{x\sim p_{data(x)}}[\Big\|F(x)-x\Big\|_1] \tag{4.13}$$

Segmentation loss. At each time-point, there is a ground-truth label for every sample. The authors propose a 3D Cycle GAN Segmentation network by adding two segmentation networks S_X, S_Y that use ground-truth label to train the segmentation network S_X, S_Y using 3D-Dense-Unet architecture. Pre-tranied weight of the segmentation networks is used to produce the segmentation features of real images to help the generator networks G to generate images that have similar segmentation features to real images. In particular from one sample in 6-month time-point, the network generates another sample that similar to 24-month time-point form and provides the synthetic segmentation result from segmentation networks S_Y which trained on images from domain Y. The network compares that result to the one from segmentation network S_X, which trained on images from domain X to produce cycle loss. The segmentation loss $\mathcal{L}_{seg}$ encourages the synthetic images distribution to move towards distribution of the search space with smallest cross-entropy loss.

$$\begin{aligned}\mathcal{L}_{seg}(G_X, G_Y, S_X, S_Y) = &\sum_{i=1}^{C} T(S_X^i(x))\ \log(S_Y^i(G_X(x))) \\ &+ \sum_{i=1}^{C} T(S_Y^i(y))\ \log(S_X^i(G_Y(y)))\end{aligned} \tag{4.14}$$

Dense loss (Feature matching loss). For further improvement in the image quality, the authors propose contextual loss to measure cosine distance between two segmentation features of real and fake images that extracted from segmentation

networks S_X and S_Y.

$$\begin{aligned}\mathcal{L}_f(G_X, G_Y, S_X, S_Y) = \sum_{l=n}^{m} (&\mathcal{L}_{CX}(f^l_{S_X}(x), f^l_{S_X}(G_X(x)) \\ &+ \mathcal{L}_{CX}(f^l_{S_Y}(y), f^l_{S_Y}(G_Y(y))\end{aligned} \tag{4.15}$$

Objective function. Combining all the loss functions above, the objective function for this model is defined as:

$$\begin{aligned}\mathcal{L}_{cycleGAN\_Seg}(G_X, G_Y, D_X, D_Y, S_X, S_Y) = &\mathcal{L}_{cycleGAN}(G_X, G_Y, D_X, D_Y) \\ &+ \gamma \mathcal{L}_{seg}(G_X, G_Y, S_X, S_Y) \\ &+ \xi \mathcal{L}_f(G_X, G_Y, S_X, S_Y)\end{aligned} \tag{4.16}$$

Which is to add up the Cycle GAN loss, segmentation and feature mapping loss, with γ & ξ as parameters.

Guided segmentation. After training the 3D Cycle GAN Segmentation network, the pretrained model is then used to retrain the 3D dense Unet network for segmentation of the infant brains. The objective function (joint segmentation loss) for the 3D dense Unet network is updated to be the sum of the loss for the segmentation of both the natural and unnatural (generated from the pretrained 3D Cycle GANSegmentaiton) images:

$$\begin{aligned}\mathcal{L}_{seg\_join}(S_X) = &\sum_{i=1}^{C} L^i_X \log(S^i_X(x)) \\ &+ \sum_{i=1}^{C} L^i_Y \log(S^i_X(G_Y(y))\end{aligned} \tag{4.17}$$

Evaluation metrics. The proposed method originally uses Dice Similarity Coefficient (DSC) metric for evaluation. The DSC metric is similar to the idea of

F score, defined as:

$$\text{Dice score} = \frac{2 \cdot |A \cap B|}{2 \cdot |A \cap B| + |B \backslash A| + |A \backslash B|} \tag{4.18}$$

To an extent, the authors' work offers an approach using data from one time-point to correct the segmentation errors of another without the need of paired data. Which is to use the Unet added into cycle GAN for Cycle GAN Segmentation network. And then leverage this model to correct the segmentation fault by retraining and modifying the loss function of the 3D dense Unet network.

CHAPTER 5
EXPERIMENT

For the experiment, we start discussing the collected datasets, then our motivation experiment using active contour post-processing on previously trained Unet model detailed in section 5.3.3. Later, we describe our proposed approach, the Narrow Band - Active Contour loss experimented with different networks as in section 5.3. For the result from the experiments, section 5.3.1 noted down both the qualitative and quantitative comparison of this loss function used with different networks as well as another experiment using this proposed loss in combination with 3D Unet guided with GAN (from the Cycle GAN Segmentation model).

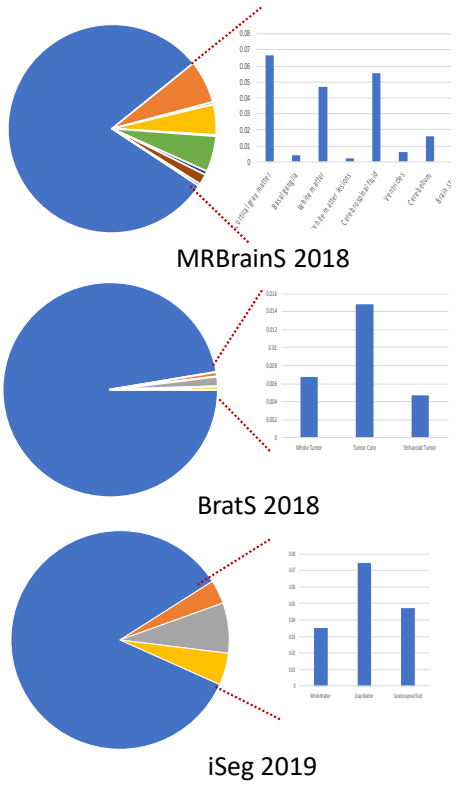
5.1 Dataset

We use four common medical datasets including 2D and 3D images in our experiments as follows:

iSeg: The iSeg19 dataset [129] consists of 10 subjects with ground-truth labels for training and 13 subjects without ground-truth labels for testing. Each subject includes T1 and T2 images with size of $144 \times 192 \times 256$, and image resolution of $1 \times 1 \times 1$ mm^3. In iSeg, there are three classes: white matter (WM), gray matter (GM), and cerebrospinal fluid (CSF).

MRBrainS: The MRBrainS13 dataset contains 6 subjects for training and validation and 15 subjects for testing. The MRBrainS18 dataset [130] contains 7 subjects for training and validation and 23 subjects for testing. For each subject, three modalities are available that includes T1-weighted, T1-weighted inversion recovery and T2-FLAIR with image size of $48 \times 240 \times 240$. Each subject was manually segmented into either 3 or 8 classes by the challenge organizers.

Brats: The Brats18 database [131] contains 210 HGG scans and 75 LGG scans. For each scan, there are 4 available modalities, i.e., T1, T1C, T2, and Flair. Each image is registered to a common space, sampled to an isotropic $1 \times 1 \times 1$ mm^3 resolution by the organizers and has a dimension of $240 \times 240 \times 155$. In Brats18, there are three tumor classes: whole tumor (WT), tumor core (TC) and enhanced tumor (ET).



(a) class distribution of three datasets (blue regions in the pies are backgrounds)

		Mean	Std	Median
MRBrainS	cortical gray matter	0.21	0.04	0.21
	basal ganglia	0.20	0.04	0.20
	white matter	0.20	0.03	0.20
	white matter lesions	0.29	0.06	0.28
	cerebrospinal fluid *	0.12	0.06	0.11
	ventricles	0.06	0.07	0.03
	cerebellum	0.21	0.04	0.21
	brainstem	0.20	0.04	0.20
iSeg	white matter	0.21	0.10	0.19
	grey matter	0.30	0.10	0.26
	cerebrospinal fluid	0.32	0.12	0.28
BRATS	whole tumor	0.51	0.16	0.52
	tumor core	0.51	0.15	0.50
	enhance tumor	0.49	0.14	0.49

*cerebrospinal fluid in the extracerebral space

(b) image contrast shown in Mean/Std/Mean of pixel intensities

Figure 5.1: Statistical information of medical images.

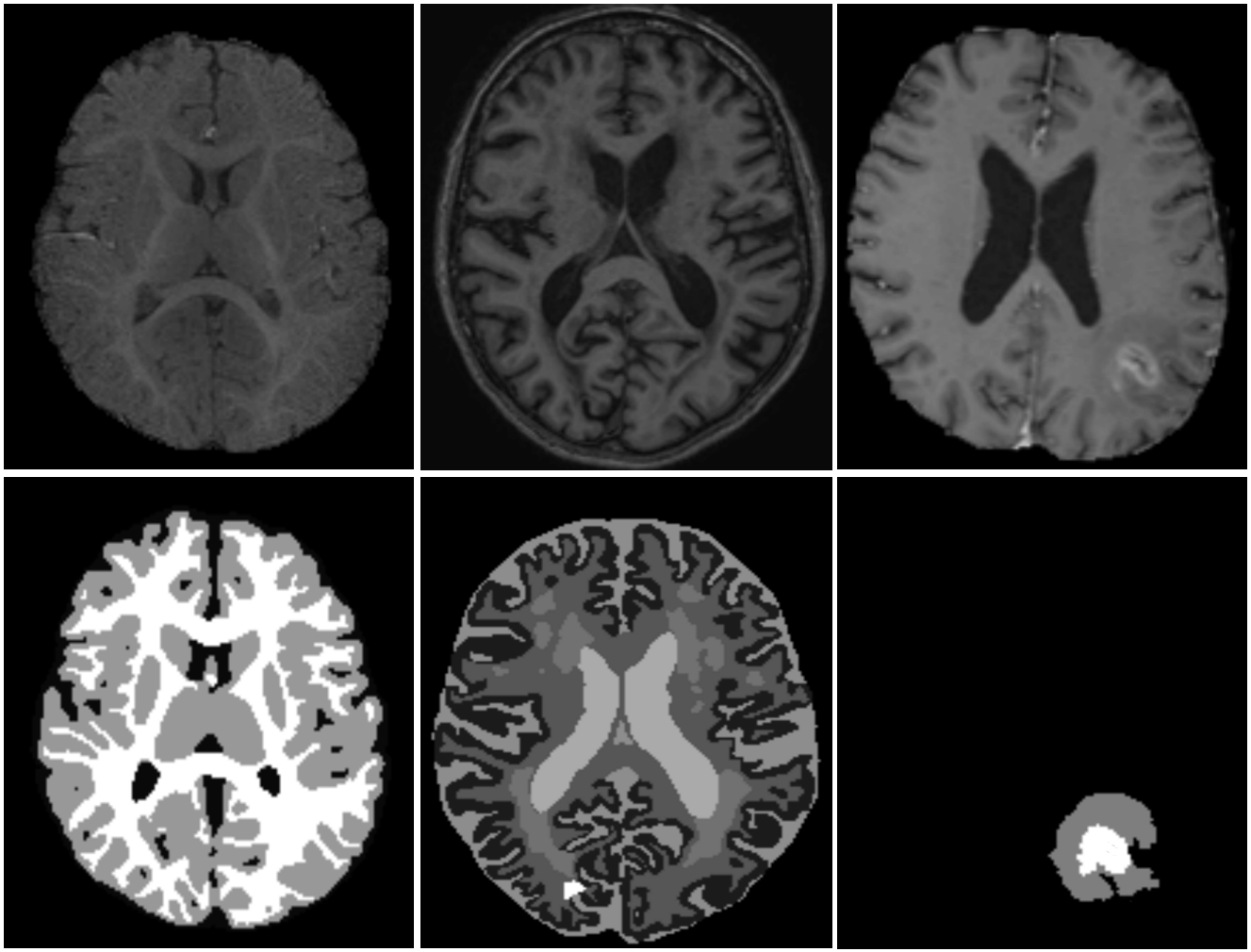


Figure 5.2: Visualization of some medical images from different datasets such as (a) iSeg19, (b) MRBrainS18, (c) Brats18. The first row is raw input images, the second row is labeled images.

Aforementioned in section 2.1.2, and in figure 5.2 and 5.1 respectively shows the examples as well as the statistical information of these three datasets. Sharing the same task of segmenting the target tissues for the given 3D subjects, Brats and MRBrains, however, do not severely affected by the less annotation problem as in iSeg. Hence, for this issue, we also leverage a subset of subjects from The UNC/UMN Baby Connectome Project (BCP) to guide the segmentation 6 and 24 -moth datasets using cyclegan trained on BCP and iSeg 2019. The follows are detailed description of iSeg and BCP dataset as we alloyed these two dataset with different parameters, which also lead to a heavier problem of data pre-processing and normalizing.

Baby Connectome Project (BCP) is an infant brain MRI segmentation dataset which is used for studying abnormal early brain development, the dataset is also utilized for Iseg-2019 Challenge in conjunction with MICCAI 2017 [1]. BCP comprises of infant brain MRI scans with their segmentation labels in 3 types: white matter (WM), gray matter(GM), and cerebrospinal fluid (CSF). The MRI scans is recorded based on standard critical periods in terms of studying both normal and abnormal in early brain development of radiologists or doctors. In the early stage of brain development , there are three importanat phases in the first-year brain MRI, consisting of infantile phase (<= 5 months), isointense phase (6-8 months) and early adult-like phase(>=9 months). Especially, 6-month old and 24-month old record of infant brain images are two critical periods of the problem we study. 6-month old infant brain has respectively low intensity contrast between tissues in comparison to 24-month old, which motivates the use of GAN to guide infant brain segmentation with adult brain dataset. The dataset contains input subject as T1- and T2-weighted MR images of 10 infant subjects in the training set(from ssubject-1 to subject-10). The manual segmentation label for each subject is set as :

[1]http://iseg2019.web.unc.edu/

0: Background (everything outside the brain)

10: Cerebrospinal fluid (CSF)

150: Gray matter (GM)

250: White matter (WM)

In the test set, BCP contains T1- and T2- weighted MR images of 13 infant subjects(from subject-11 to subject-23). Table 5.2 shows the analytical reports of the two dataset we obtained. On the other hand, table 5.1 are parameters for each of the dataset. This table also infers the need of preprocessing and normalizing data.

Table 5.1: Dataset imaging parameters[2]

		TR/TE	Flip angle	Resolution
Iseg	T1-w	1900/4.38 ms	7º	1×1×1 mm3
training	T2-w	7380/119 ms	150º	1.25×1.25×1.95 mm3
BCP	T1-w	2400/2.24 ms	8º	0.8×0.8×0.8 mm3
	T2-w	3200/564 ms	VAR	0.8×0.8×0.8 mm3
Stanford	T1-w	7.6/2.9 ms	11º	0.94×0.94×0.80 mm3
University	T2-w	2502/91.4 ms	90º	1.00×1.00×0.80 mm3
Emory	T1-w	2400/2.19 ms	8º	1×1×1 mm3
University	T2-w	3200/561 ms	120º	1×1×1 mm3

[2]http://iseg2019.web.unc.edu/data/. sagittal: x plane, axial: z plane, coronal: y plane, VAR: for variance

Table 5.2: Dataset description

iseg 2019	**#subjects**	**avg. start slices**	**avg. end slices**	**avg. slices/subject**	**total**
train	10	92.40	192.40	101.00	1010
test	13	87.77	188.46	101.69	1322
	23	89.78	19.17	101.39	2332

(a) iseg 2019

mini-BCP		**#subjects**	**avg. start slices**	**avg. end slices**	**avg. slices/subject**	**total**
6m	train	4	32.50	180.75	150.25	601
	test	2	30.50	183.50	154.00	308
24m	train	4	23.0	187.25	165.25	661
	test	20	21.50	186.00	165.50	3310
		30	23.77	185.30	162.67	4880

(b) a sub-dataset of BCP

Figure 5.3 shows an example of one slices from iSeg 2019 dataset. As noted in the figure, respectively from the left to the right column are T1-w, T2-w and label of this slide. More information about MRI subjects is noted in section 2.1.2.

[2]From iSeg 2019 [129]

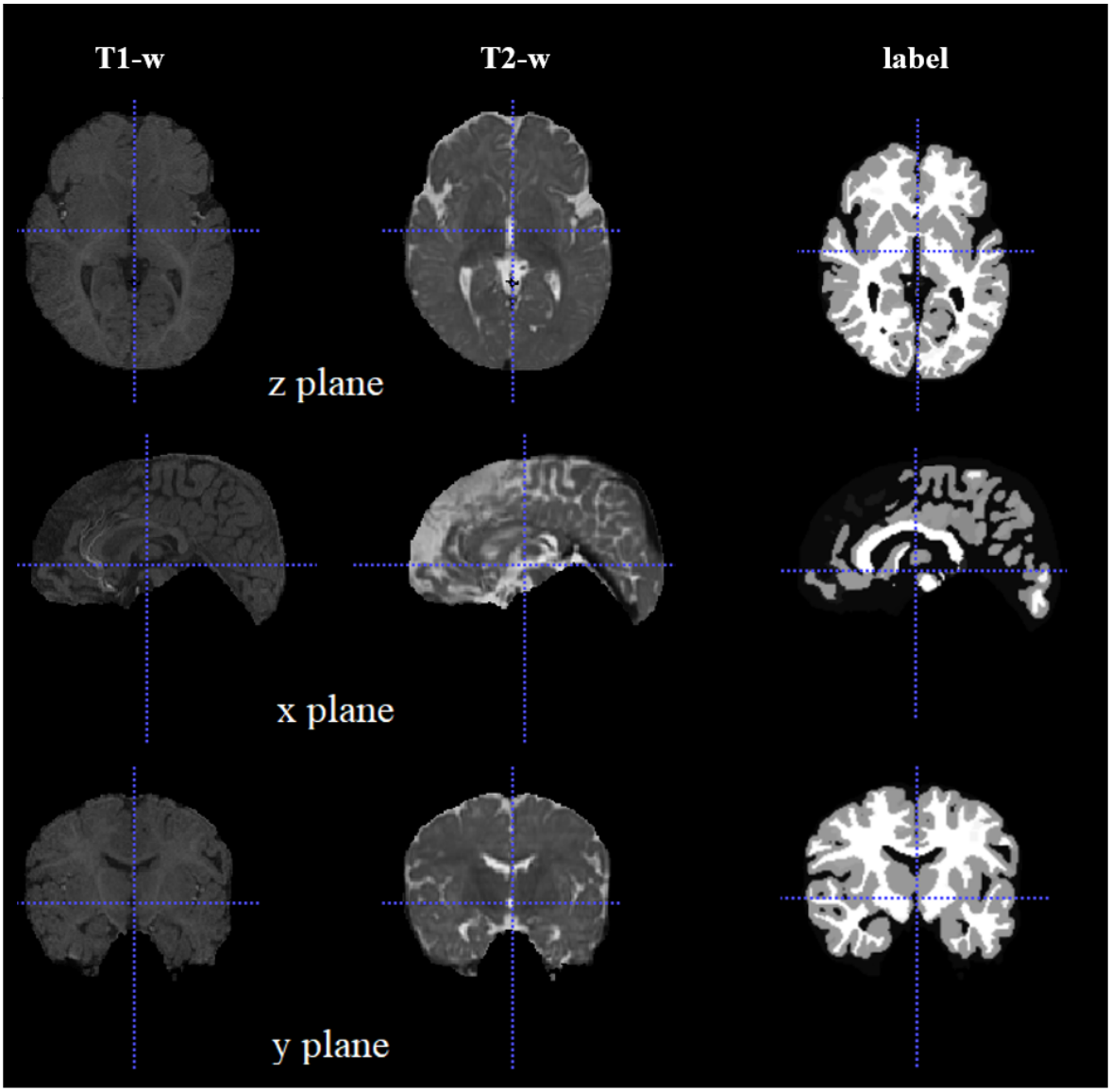


Figure 5.3: Dataset example from iSeg 2019 [129]. The columns are T1-weighted, T2-weighted, and label of the target tissues (from left to right). The rows are the middle slide of a subject viewed in axial/z, sagittal/x, coronal/y plane. For the label of the target tissues: the light gray part is white matter, gray part is for grey matter and the dark grey part is for cerebrospinal fluid (the black part is background).

Figure 5.4: An example of the segmentation result predicted by our model

5.2 Motivation

In this part, we leverage the Cycle GAN Segmentation model to train the 6 and 24 -moth segmentation models (as later detailed in section 5.3.3). Later, we experiment several post-processing using active contour methods to check whether this active contour methods can be used for better medical images segmentation boundaries, as well as whether we should move on to the next part: wrap this active contour method into an end-to-end unet model.

To evaluate appearance transferring efficiency of the 24-month old synthetic images to 6-month old synthetic images vice versa, we compare against state of the art model U-net network for segmentation which were trained with the real-6-month subjects. The result shows that our method has a better performance on Dice Score Coefficient metric. In particular, our method achieves 92.506% on

average with accuracy 94.76% on gray matter, 91.46% on white matter, 91.3% on CSF, while state of the art model of 3D U-Net got 92.50% accuracy on average which was tested on subject-9. Figure 5.4 shows an example of the segmentation result predicted by our model.

Applying Chanvese method with narrow band for [-0.5, 0.5] signed distance map from the inital boundary (retrieved from the previous training inference on Cycle GAN Segmentation model) with an additional restriction that boundaries of a segmented tissue should not move more than 0.5×5 signed distance map (in the initial boundaries) into other tissues with different label.

Figure 5.5 shows the energy of an example subject (each label) before and after applying the active contour post-processing method. The dark regions and the brighter regions respectively are the energy of the whole regions inside and outside the segmented regions. Here, we only show the energy of the whole inside/outside regions instead of narrow band because our chosen narrow band regions are only several pixels wide . The active contour methods strive to move the initial contour to the lower or higher (depending on the sign of the weight of line in equation 2.3 intensity fields of the image where the intensity gradient is larger. The higher contrast between the two inside and outside regions as shown in this figure fails for the higher DSC result (after post-processing) implies that active contour methods might works on this dataset.

Please note that the contrast here might not change much because the initial boundary was extracted from the Cycle Gan Segmentation model, which have already achieve a high result so that the boundary of the object should not change much, except for the regions those are mis-segmented via the above model.

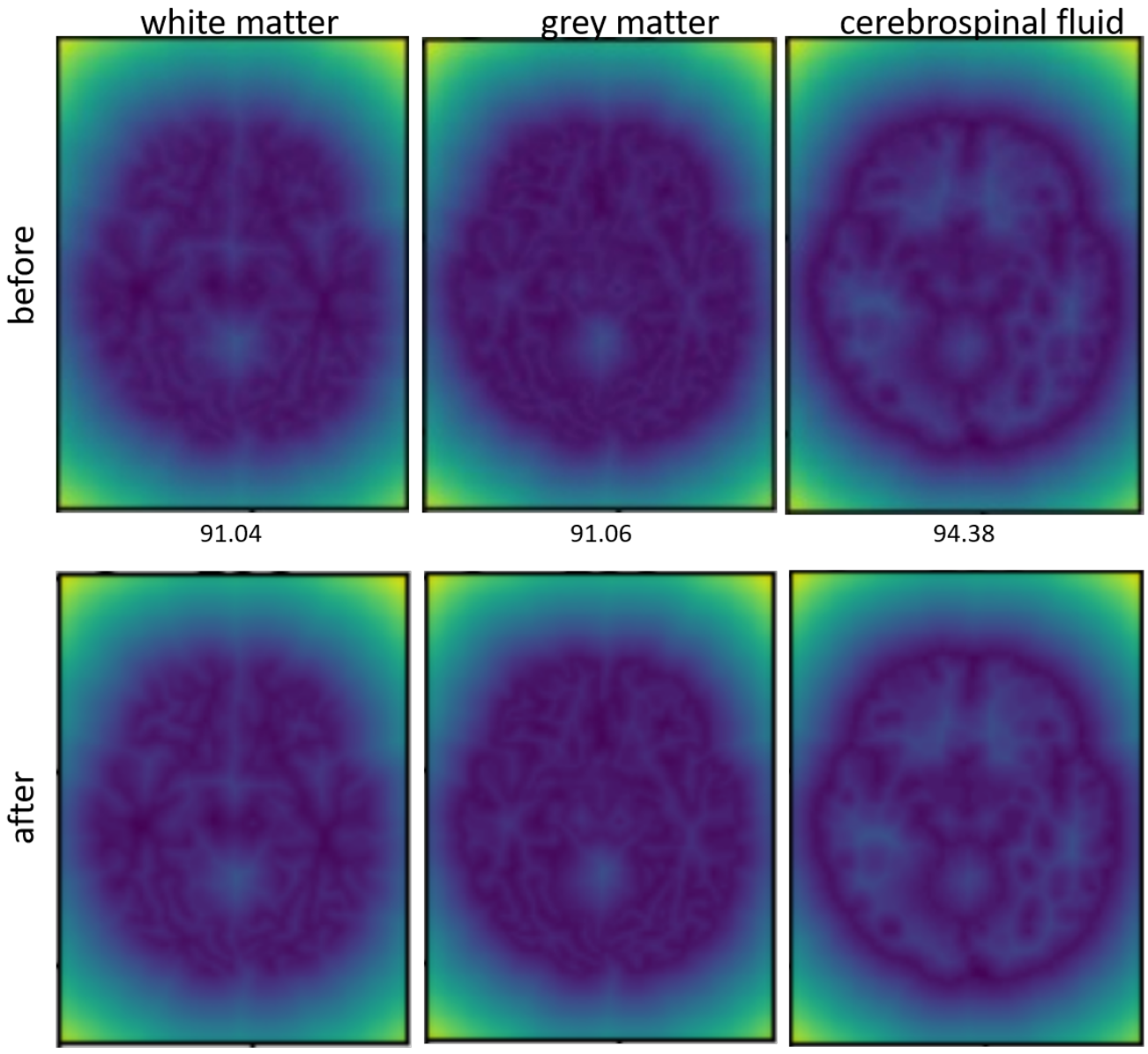


Figure 5.5: Energy of one slice (each label) from an example subject before and after applying the active contour post-processing method with the DSC.

5.3 The proposed approach (NB-AC loss)

In this section, we evaluate the proposed NB-AC loss with different network architectures, such as, Unet [7], 3DUnet [8]. Our performance is compared against other common loss functions i.e. Dice, CE, Focal on the baseline frameworks Unet [7] and compared against other state-of-the-art networks on 3DUnet [8]. Figure 5.6 shows an example of the intermediate outputs of our NB-AC method.

Experiment setting. On 2D images, to train our NB-AC loss on 2D Unet networks we define the input as $N \times C \times H \times W$, where N is the batch size, C is the number of input modalities and H, W are height, width of 2D image. Corresponding to iSeg19, MRBrainS18 and Brats18, we choose the input as $4 \times 2 \times 224 \times 224$, $4 \times 3 \times 224 \times 224$ and $4 \times 4 \times 224 \times 224$, respectively. We employed the Adam optimizer, with a learning rate of 1e-2 with weight decay 1e-4. On 3D volumes, our 3D architecture is built upon 3D-Unet [8] and the input is defined as $N \times C \times H \times W \times D$, where N is batch size, C is the number of input modalities and H, W, D are height, width and depth of volume patch on sagittal, coronal, and axial planes. Corresponding to Brats18, MRBrainS13 and iSeg19, we choose the input as $1 \times 4 \times 96 \times 96 \times 96$, $1 \times 3 \times 96 \times 96 \times 48$, and $2 \times 2 \times 96 \times 96 \times 96$. We implemented our network using PyTorch 1.3.0 and our model is trained until convergence by using the ADAM optimizer. We employed the Adam optimizer, with a learning rate of 2e-4. Our 3D Unet makes use of instance normalization [132] and Leaky reLU. The experiments are conducted using an Intel CPU, and RTX GPU.

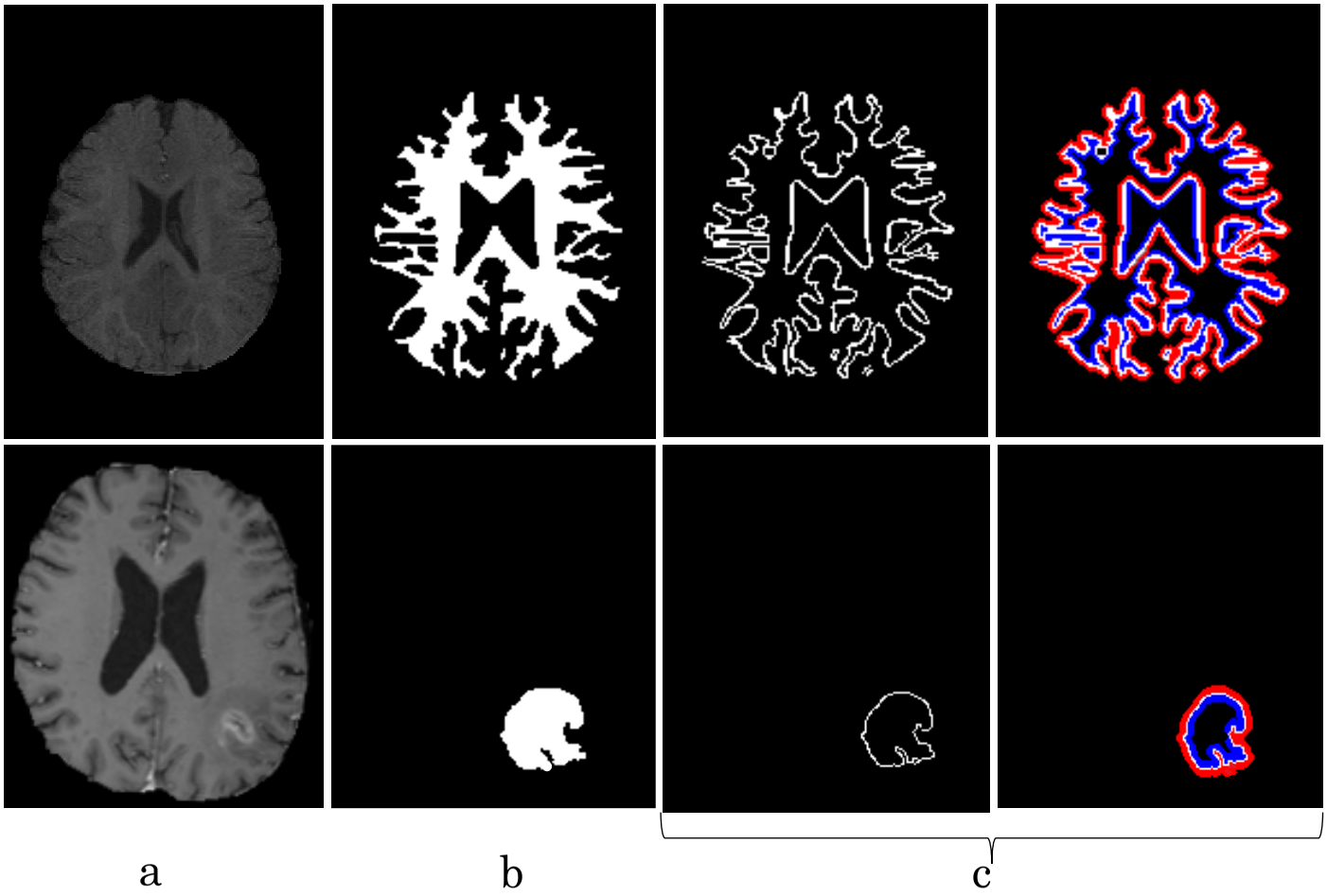


Figure 5.6: The intermediate results from Unet combined with our proposed loss. Given raw data in (a), the proposed NB-AC loss contains both higher level feature loss (b) and lower level feature loss (c) including the length of the contour (left) and narrow band energy from both sides of the contour (right).

5.3.1 Results and Comparison

In the post-processing section, we only test on the iSeg 3D images (and for deep snake we test on 2D slices iSeg). But for the Narrow Band - Active Contour loss, we also check on the other two datasets (both 2D and 3D, for 2D models trained on both FCN and Unet) as a further confirmation for the effectiveness of this proposed loss.

For quantitative assessment of the segmentation, the proposed model is evaluated on different metrics, e.g. Dice score (DSC), Intersection over Union (IoU), Sensitivity (or Recall), Precision (Pre).

The performance of our proposed NB-AC loss is evaluated on both FCN [55] and Unet [7] architectures for 2D input and 3DUnet [8] for 3D input. The comparisons between our proposed loss and other common loss functions: CE, Dice, Focal on challenging datasets MRBrainS18, Brats18 and iSeg19 are given in Tables 5.3. Most yellow-highlighted texts fall for Unet, which implies Unet is generally better than FCN for these three datasets. This is why later in table 5.4 we only compare the 2D or 3D state-of-the-art methods against our proposed losses on 2D or 3D Unet backbones.

Table 5.3: Comparison between our proposed NB-AC loss against other losses CE, Dice and Focal on **MRBrainS18**, **BRATS 2018**, **iSeg 2019** dataset

		Losses	DSC	IoU	Pre	Rec
MRBrainS18	FCN	CE	83.26	74.69	85.0	86.4
		Dice	82.0	73.23	82.67	85.89
		Focal	78.79	70.0	77.78	86.56
		NB-AC	**84.62**	**76.48**	**86.44**	**86.78**
	Unet	CE	83.32	74.73	84.67	**86.56**
		Dice	81.13	71.87	81.78	85.89
		Focal	79.98	70.87	80.22	86.44
		NB-AC	**84.97**	**76.92**	**87.89**	86.11
BRATS 2018	FCN	CE	78.57	73.74	77.33	80.00
		Dice	77.67	72.94	75.00	81.00
		Focal	72.33	68.08	69.00	78.00
		NB-AC	**79.96**	**75.16**	**79.66**	**80.33**
	Unet	CE	79.40	74.59	78.33	81.00
		Dice	78.21	73.44	77.33	78.67
		Focal	76.38	**78.93**	68.00	**87.00**
		NB-AC	**80.38**	75.48	**81.25**	82.19
iSeg 2019	FCN	CE Loss	87.95	83.91	90.25	91.75
		Dice	86.44	82.14	89.5	90.25
		Focal	83.19	78.51	87.25	88.0
		NB-AC	**88.95**	**85.11**	**91.5**	**92.25**
	Unet	CE	88.91	85.06	91.25	**92.25**
		Dice	87.19	83.01	90.03	90.5
		Focal	87.07	82.90	89.75	91.0
		NB-AC	**89.73**	**86.05**	**92.25**	92.0

*blue colored texts denote our results, bold texts denote the highest results for each datasets on each metrices and backbones, yellow-highlighted texts denote the highest results for each datasets on each metrices.

It is clear that the proposed NB-AC loss function outperforms the other common losses under both UNet and FCN frameworks. Take DSC metric on CE loss as an instance, our loss gains 1.36%, 1.39%, 1.0% on MRBrainS18, Brats18, iSeg19 respectively using Unet framework and it gains 1.65%, 0.98%, 0.82% on MRBrainS18, Brats18, iSeg19 respectively using FCN framework.

Fig. 5.7, 5.8 and 5.9 visualize the comparison between our proposed NB-AC loss against other loss functions including Dice, Focal (FC) and Cross Entropy (CE) on Unet framework. These images are randomly select from the testing set of various dataset, namely MRBrainS 2018, BRATS 2018, iSeg 2019. As shown in Fig.5.2, medical images contain poor contrast images where boundary between objects is very unclear and weak. Take iSeg dataset as an instance, due to the myelination and maturation process of the infant brain, the boundary between classes in the infant brain in iSeg is very weak, leading to difficult for segmentation. The segmentation results from different loss functions are visualized in Fig.5.9 (top) with specific differences are highlighted in colored boxes. The infant brain MR images (iseg-2019 dataset) has extremely low tissue contrast between tissues, thus the segmentation results using traditional loss functions (such as CE, Dice, and Focal loss) have a large amounts of topological errors (contain large and complex handles or holes) in the segmentation results, such as WM surface in the Fig.5.9 (bottom) which illustrates a enlarged view of the white matter surface of an infant brain. Fig.5.9 (bottom) demonstrates that the proposed NB-AC loss function produces less topological errors (i.e., holes and handles), indicated by the red arrows, compared against the existing loss functions. In addition to 2D view of brain as in Fig.5.9, 3D view of the entire view white matter surface as in Fig.5.10 demonstrates that the proposed NB-AC loss function produces less topological errors (i.e., holes and handles), indicated by the red arrows, compared against the existing loss functions.

In Fig. 5.9, the weak boundary around gray matter, white matter, CSF is high-

light in colored boxes. In such colored boxes, we can see the boundary is shown in poor contrast in the original image. Far apart from other loss functions which are unable to capture such information, the proposed NB-AC has high capability to work on the case of weak object boundary segmentation. Not only weak object boundary but also imbalanced-class data, figure 5.7, 5.8 contain the performance of middle slide of each image/volume that are from MRBrainS 2018, BRATS 2018 datasets. At each figure, the colored boxes highlight areas corresponding to small class data and weak boundary object (specially the object boundary). Compared against other loss functions, our NB-AC loss obtains closest result to the groundtruth under both cases of weak boundary object, small object.

Clearly, comparing with the common segmentation losses, the proposed NB-AC loss improve the segmenting performance using the same network backbone. Take CE loss function as an example, the proposed NB-AC loss improved from 87.95% to 88.95% segmentation accuracy using FCN architecture and 88.91% to 89.73% using U-Net architecture. Fig. 5.7, 5.8, 5.9 visualizes the comparison between our loss and other loss functions. In these figures, some regions are highlighted to see the difference in segmentation results between loss functions.

The segmentation results from different loss functions are visualized in Fig.5.9 (a) with specific differences are highlighted in colored boxes. Fig. 5.9 (b) illustrates a enlarged view of the white matter surface of an infant brain from the regions highlighted in blue boxes of Fig. 5.9 (a). Fig.5.9 (b) demonstrates that the proposed NB-AC loss function produces less topological errors (i.e., holes and handles), indicated by the red arrows, compared against the existing loss functions. For more detailed visualization, we provide the entire view white matter surface obtained from different loss functions in Fig.5.10.

Table 5.4 shows the comparison against other state-of-the-art methods on three volumetric datasets. Our performance is quite compatible with [133] on MR-

BrainS13 while it outperforms [134] and [4] on BratS18 and iSeg19 with similar network architecture setting up. The result noted in this table for iSeg also leverages the information gained from 24 month dataset (section 5.3.2, without the information from GAN, this Active Contour 3D Unet has already achieved a DSC score of **92.56** on 6000 epoches (to our furthest knowledge, the current SOTA on 3D Unet [4] only scores 92.55 on DSC).

5.3.2 The Active Contour Unet with Guided Segmentation

Transferring the knowledge from 24 month dataset to 6 month dataset in iSeg dataset, we detailed the training state of cycle gan segmentation in section 5.3.3. Using the knowledge learnt by the Cycle GAN Segmentation model from the 24 month brain dataset in BCP, we conduct an experiment striving to transfer this knowledge on to the Active Contour Unet and received a result of **93.07** on the 3D iSeg 2019. Which is an increment of 0.51 from **92.56**, supposedly supports the proposed Narrow Band Active Contour loss.

Table 5.4: Comparison of our proposed loss on 2DUnet and 3DUnet against other 2D and 3D state-of-the-art methods on medical datasets

	Datasets	**DSC**	**Recall**
2D segmentation	Brats18 [[135]/ **Ours**]	77.75 / **80.38**	80.1 / **82.19**
	MRBrainS18 [[136]/ **Ours**]	82.48 / **84.97**	– / **86.11**
	iSeg19 [[137]/ **Ours**]	89.00 / **89.73**	– / **92.00**
3D segmentation	Brats18 [[134] **Ours**]	84.87/ **85.67**	– / **86.47**
	MRBrainS13 [[133]/ **Ours**]	**87.17**/87.02	– / **87.89**
	iSeg19 [[4] **Ours**]	92.55 / **93.07**	92.64 / **93.16**

*blue colored texts denote our results and bold texts denote the highest results

5.3.3 Training inference

We use Baby Connectome Project dataset to experiment our method. Our model was trained and tested on two periods that are 6-month old and 24-month old. We use 4 subjects with labels in each time-points to train the segmentation network S_x and S_y. The original resolution of these images used in our experiments is $0.8 \times 0.8 \times 0.8$mm.

However, to test our result on Iseg 2019 competition we have to resize down to $1.0 \times 1.0 \times 1.0$mm to get the same image resolution to Iseg test set's resolution. Then we use outer interpolation to downsize the resolution of these images we used to test in Iseg-2019 challenge. We linearly interpolate the image in a cubic of $1.0 \times 1.0 \times 1.0$mm.

The pixel value of images was changed after interpolation that make it hard to evaluate our result in the new resolution so we have to render again that value in the boundary region, particularly where the combination two out of three regions white matter, gray matter and cerebrospinal fluid.

The render algorithm we use to take an accurate value for each pixel is k-means clustering, with $k = 3$. We normalized the input image to $[-1, 1]$ to easily compute. However the memory resource we have is limited so we have to randomly cropped a small region with a size of $32 \times 64 \times 64$ (where 32 is number of slice), to test these cubic using our network we apply evaluation metric Dice Similarity Coefficient (DSC) to calculate the percentages of intersection between predicted cubic and label cubic. We use the adam optimizer with a batch size of 8 to train the network, we initialize the learning rate at 0.0002.

The 6-month old U-net segmentation network S_x was train on 16000 epoches and the same number of iterations was applied to train 24-month old U-net segmentation network S_y. The method used U-net with instance normalization as

the generator and a patch-based fully convolutional network as the discriminator. Before we train the Cycle-gan model to convert the input image from 6-month old to 24-month and vice versa in total 6000 epoches.

The balance weights were set as $\lambda = 10$, $\beta = 5$, $\gamma = 3$ and $\zeta = 2$, we freeze the wegihts of the segmentation networks so that it can adjust our segmentation result guided from cycle-gan model.

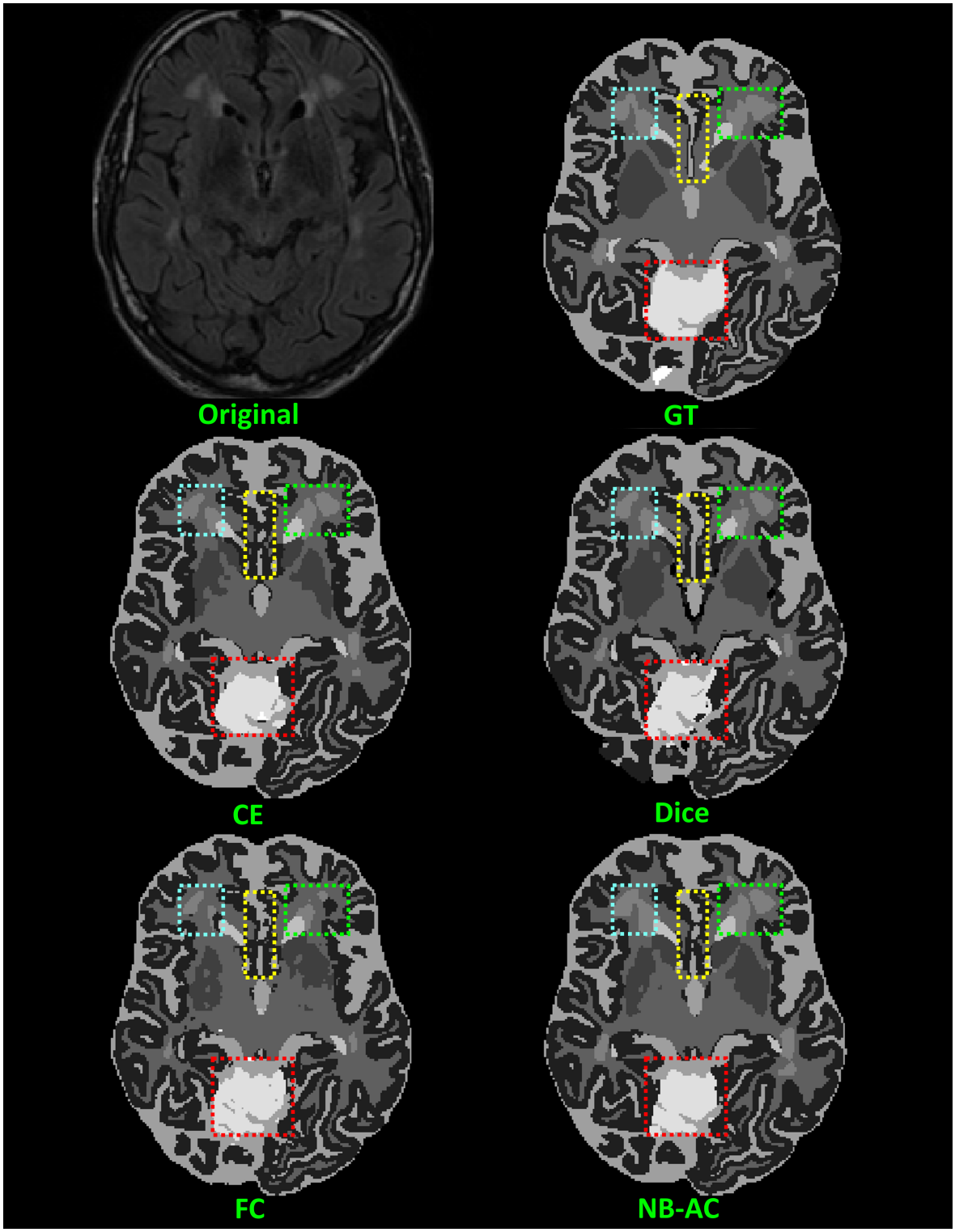


Figure 5.7: Qualitative result on MRBrainS 2018: Comparison between our results against other loss functions on Unet framework where the image is from MRBrainS 2018.

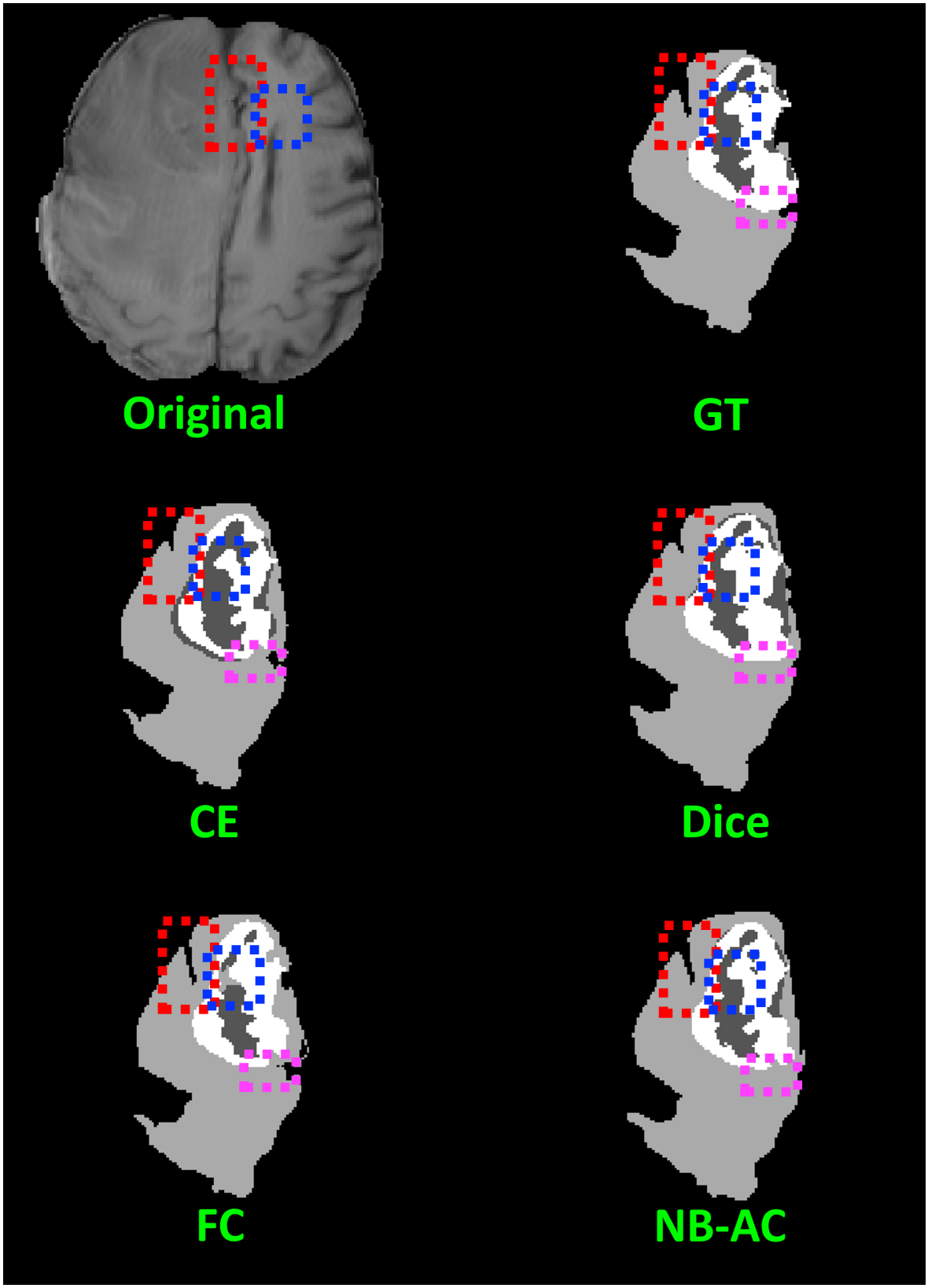


Figure 5.8: Qualitative result on BRATS 2018. Comparison between our results against other loss functions on Unet framework where the image is from BRATS 2018.

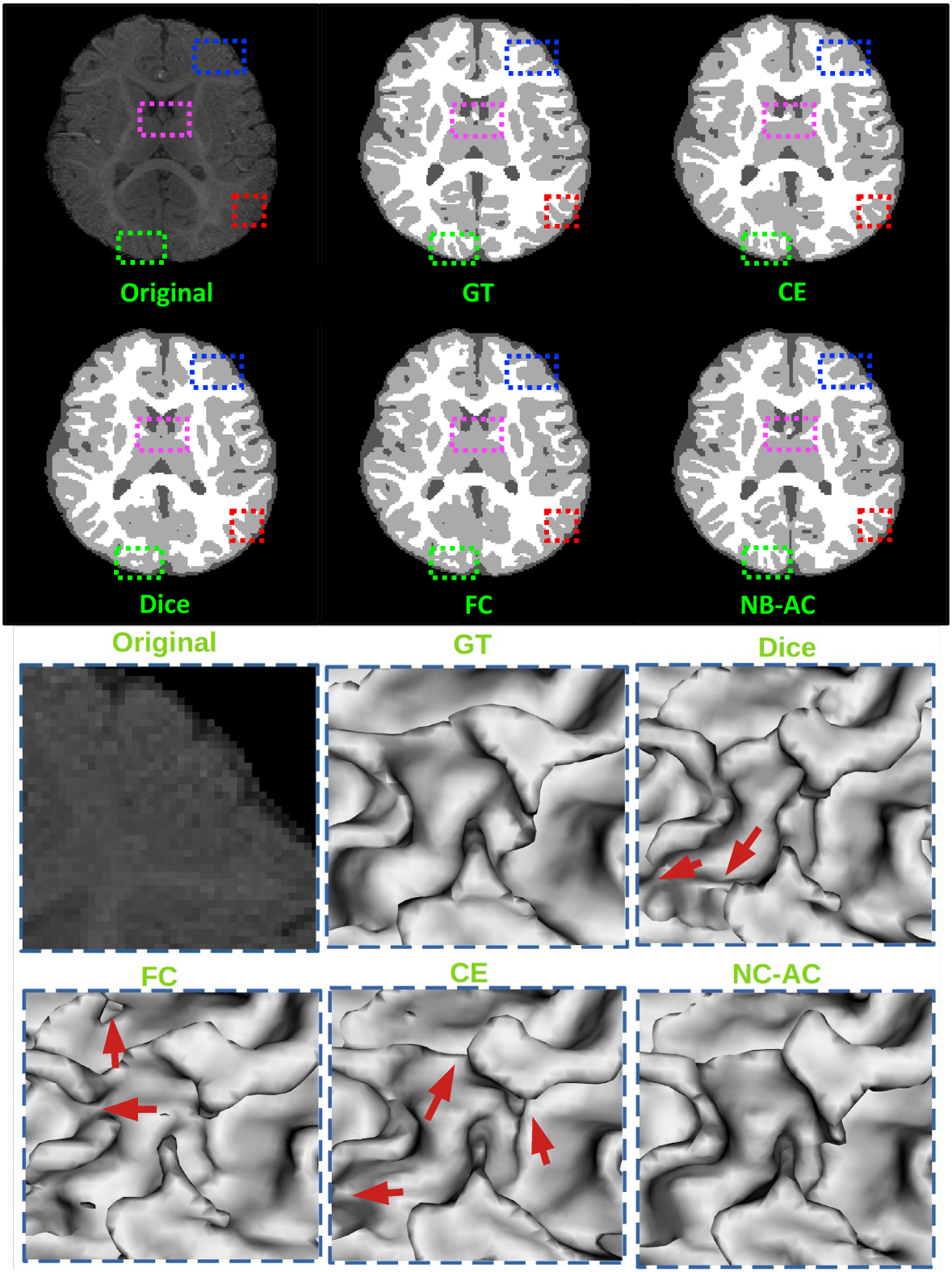


Figure 5.9: Qualitative result on iSeg 2019. (top) Comparison of our proposed NB-AC loss against other loss functions on iSeg19 datasset with colored boxes highlighting specific differences. (bottom) A loser look is also given with the topological errors indicated by red arrows.

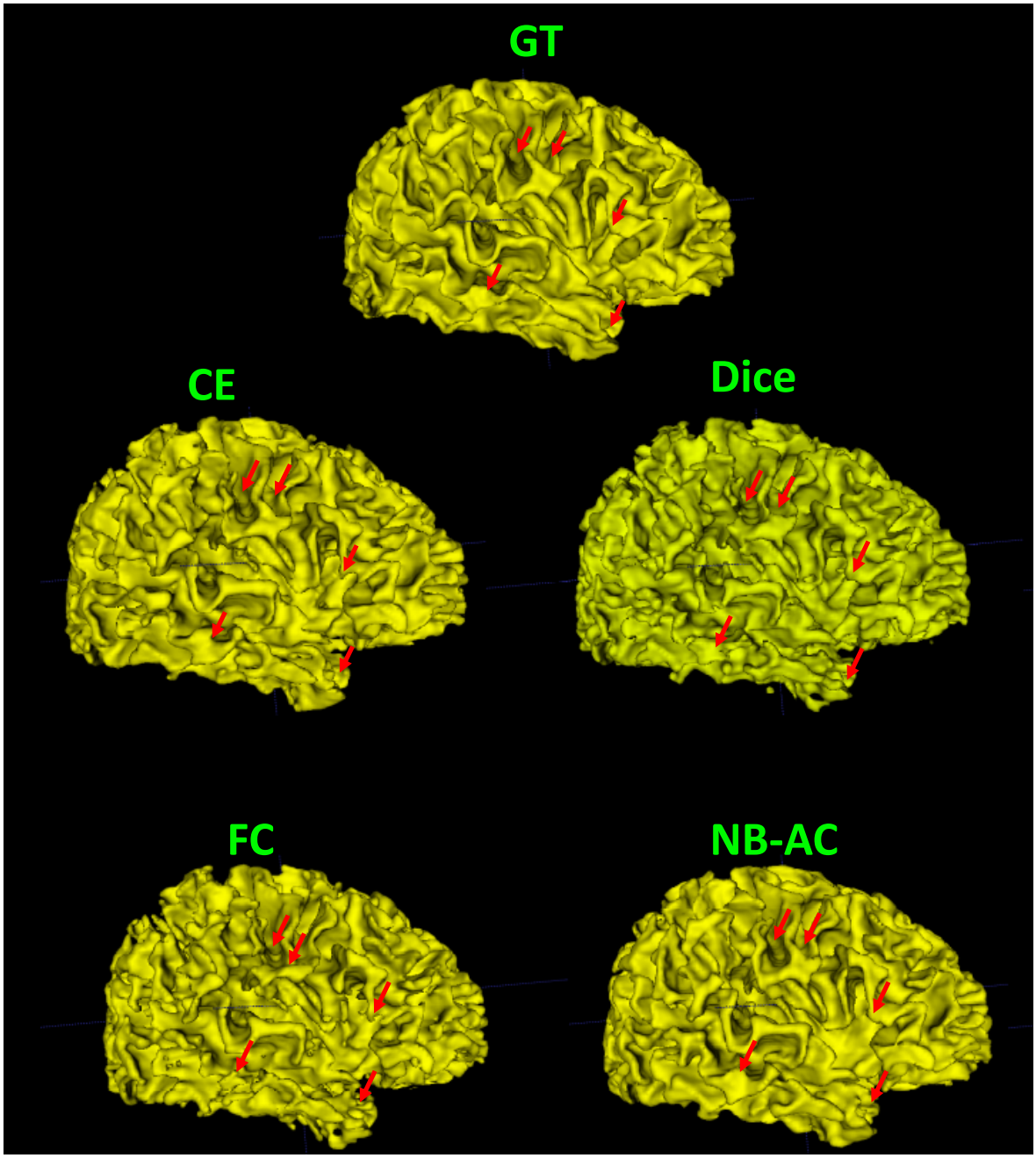


Figure 5.10: Visualization of white matter surface of the existing loss functions on iSeg19 dataset where differences in topology are indicated by red arrows.

CHAPTER 6
CONCLUSION

The two main tasks in medical images segmentation are: (1) segment MRI into different areas (e.g. WM, GM, CSF) to get a better understand on brain structure, therefore it is important to keep the topological structure; (2) detect and segment lesion (brain tumor) into different classess with high accuracy. To sum up this work, we attempt to tackle the common problems in medical imaging (directly related to the two main tasks) which are less annotation, imbalance data, and low contrast (or weak boundary). For the less annotation problem, we leverage the Cycle GAN Segmentation model proposed by Toan Duc Bui et al. [6] - using Cycle GAN as an data augmentation method. To address the weak boundary (or low contrast) problem, we propose adding an attention gate on the edge - calculating the dice score on the thick boundary of the segmented mask outputted from Unet. To deal with the imbalance data problem, we focus more on the narrow band around the contour under level set energy minimization, which aims to lessen the effect of large objects on the original segmentation loss where all pixels are treated equally (either inside small or large objects).

Wrapping everything up, we proposed a Narrow Band - Active Contour loss which is the summation of the segmentation loss (CE loss), the attention boundary loss (DICE score on thick boundary), and the narrow band active contour energy on considered mask. To testify the efficiency of the proposed loss function, we provide small tests using the proposed loss on both 2D/3D Unet and Fully Connected Network (FCN) comparing with other losses. And we receive a result of 92.56, 87.02, 85.67 DSC scores relatively on iSeg19, MRBrainS13 and Brats18 using 3D Unet trained with the proposed loss (Active Contour Unet in section 5.3). Adding the aforementioned Cycle GAN Segmentation model with the 3D Unet using this loss function (Active Contour Unet with Guided Seg-

mentation in section 4.3, 5.3.2), we achieve a promising DSC score of 93.07. The dataset we used in this works are MRBrainS 2013, iSeg 2019, Brats 2018, and also a subset of data from BCP used in transfer learning for Unet; The major task in these dataset is segmentation the target tissues.

Future work. There exist various segmentation Unet-like models paying more attention to the object boundary or aiming for better medical image feature representation, as well as multiple methods tackling the CNN oversampling/undersampling problem. In the near future, we shall experiment some of these approaches to build a better segmentation model, also better in terms of time inference and accuracy performance.

REFERENCES

[1] Yanhui Guo and Amira S. Ashour. 11 - neutrosophic sets in dermoscopic medical image segmentation. In Yanhui Guo and Amira S. Ashour, editors, *Neutrosophic Set in Medical Image Analysis*, pages 229 – 243. Academic Press, 2019. ISBN 978-0-12-818148-5. doi: https://doi.org/10.1016/B978-0-12-818148-5.00011-4. URL `http://www.sciencedirect.com/science/article/pii/B9780128181485000114`.

[2] Zhe Guo, Xiang Li, Heng Huang, Ning Guo, and Quanzheng Li. Deep learning-based image segmentation on multimodal medical imaging. *IEEE Transactions on Radiation and Plasma Medical Sciences*, PP:1–1, 03 2019. doi: 10.1109/TRPMS.2018.2890359.

[3] Yousif Abdallah and Tariq Alqahtani. *Research in Medical Imaging Using Image Processing Techniques*. 06 2019. doi: 10.5772/intechopen.84360.

[4] Toan Duc Bui, Jitae Shin, and Taesup Moon. 3d densely convolutional networks for volumetric segmentation. *arXiv preprint arXiv:1709.03199*, 2017.

[5] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016. URL `http://arxiv.org/abs/1608.06993`.

[6] Toan Duc Bui, Li Wang, Weili Lin, and Gang Li. 6-month infant brain mri segmentation guided by 24-month data using cycle-consistent adversarial networks. pages 359–362, 04 2020. doi: 10.1109/ISBI45749.2020.9098515.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. URL `http://arxiv.org/abs/1505.04597`.

[8] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: Learning dense volumetric segmentation from sparse annotation. *CoRR*, abs/1606.06650, 2016. URL `http://arxiv.org/abs/1606.06650`.

[9] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. pages 565–571, 10 2016. doi: 10.1109/3DV.2016.79.

[10] Konstantinos Kamnitsas, Enzo Ferrante, Sarah Parisot, Christian Ledig, Aditya Nori, Antonio Criminisi, Daniel Rueckert, and Ben Glocker. Deepmedic for brain tumor segmentation. pages 138–149, 04 2016. ISBN 978-3-319-55523-2. doi: 10.1007/978-3-319-55524-9_14.

[11] M.Fausto, N.Nassir, and A.Seyed-Ahmad. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *the Fourth International Conference on 3D Vision*, pages 565–571, 2016.

[12] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar. Focal loss for dense object detection. In *ICCV 2017*, pages 2980–2988, 2017.

[13] T. F. Chan and L. A. Vese. Active contours without edges. *TIP*, 10(2): 266–277, February 2001.

[14] Su Wang. Generative adversarial networks (gan): A gentle introduction [updated], 04 2017.

[15] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014. URL `http://arxiv.org/abs/1411.1784`.

[16] Yaniv Taigman, Adam Polyak, and Lior Wolf. Unsupervised cross-domain image generation. *CoRR*, abs/1611.02200, 2016. URL `http://arxiv.org/abs/1611.02200`.

[17] Zhenghua Xu, Chang Qi, and Guizhi Xu. Semi-supervised attention-guided cyclegan for data augmentation on medical images. pages 563–568, 11 2019. doi: 10.1109/BIBM47256.2019.8982932.

[18] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017. URL `http://arxiv.org/abs/1703.10593`.

[19] Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. Recycle-gan: Unsupervised video retargeting. In *ECCV*, 2018.

[20] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, 1988.

[21] Nagaraju Regonda and M Reddy. Review of medical image segmentation with statistical approach -state of the art and analysis. pages 36–55, 06 2017.

[22] Vicent Caselles, Francine Catté, Tomeu Coll, and Françoise Dibos. A geometric model for active contours in image processing. *Numerische Mathematik*, 66(1):1–31, December 1993.

[23] N. Paragios, O. Mellina-Gottardo, and Visvanathan Ramesh. Gradient vector flow fast geometric active contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:402–407, 2004.

[24] D. Mumford and J. Shah. Optimal Approximation by Piecewise Smooth Functions and Associated Variational Problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.

[25] Dong Yu, Wayne Xiong, Jasha Droppo, Andreas Stolcke, Guoli Ye, Jinyu Li, and Geoffrey Zweig. Deep convolutional neural networks with layer-wise context expansion and attention. In *Interspeech*, pages 17–21, 2016.

[26] Yann LeCun, D Touresky, G Hinton, and T Sejnowski. A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school*, pages 21–28. CMU, Pittsburgh, Pa: Morgan Kaufmann, 1988.

[27] Yann LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–50. Springer, 1998.

[28] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[30] Michael Egmont-Petersen, Dick de Ridder, and Heinz Handels. Image processing with neural networksâa review. *Pattern recognition*, 35(10):2279–2301, 2002.

[31] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[32] Anne-Marie Tousch, Stéphane Herbin, and Jean-Yves Audibert. Semantic hierarchies for image annotation: A survey. *Pattern Recognition*, 45(1):333–345, 2012.

[33] Jialue Fan, Wei Xu, Ying Wu, and Yihong Gong. Human tracking using convolutional neural networks. *IEEE Transactions on Neural Networks*, 21 (10):1610–1623, 2010.

[34] Lijun Wang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Deep networks for saliency detection via local estimation and global search. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 3183–3192. IEEE, 2015.

[35] Guanbin Li and Yizhou Yu. Visual saliency based on multiscale deep features. *arXiv preprint arXiv:1503.08663*, 2015.

[36] Massimiliano Patacchiola and Angelo Cangelosi. Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. *Pattern Recognition*, 71:132–143, 2017.

[37] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014.

[38] Georgia Gkioxari, Ross Girshick, and Jitendra Malik. Contextual action recognition with r* cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1080–1088, 2015.

[39] Jing Zhang, Wanqing Li, Philip O Ogunbona, Pichao Wang, and Chang Tang. Rgb-d-based action recognition datasets: A survey. *Pattern Recognition*, 60:86–105, 2016.

[40] Hailiang Xu and Feng Su. Robust seed localization and growing with deep convolutional features for scene text detection. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pages 387–394. ACM, 2015.

[41] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Deep features for text spotting. In *European conference on computer vision*, pages 512–528. Springer, 2014.

[42] Ling-Hui Chen, Tuomo Raitio, Cassia Valentini-Botinhao, Junichi Yamagishi, and Zhen-Hua Ling. Dnn-based stochastic postfilter for hmm-based speech synthesis. In *INTERSPEECH*, pages 1954–1958, 2014.

[43] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. gan. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014. URL `http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf`.

[44] Joachim D. Curto, Irene C. Zarza, Fernando De la Torre, Irwin King, and Michael R. Lyu. High-resolution deep convolutional generative adversarial networks. *CoRR*, abs/1711.06491, 2017. URL `http://arxiv.org/abs/1711.06491`.

[45] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL `http://proceedings.mlr.press/v70/arjovsky17a.html`.

[46] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5767–5777. Curran Associates, Inc., 2017. URL `http://papers.nips.cc/paper/7159-improved-training-of-wasserstein-gans.pdf`.

[47] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-

to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016. URL http://arxiv.org/abs/1611.07004.

[48] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *CoRR*, abs/1606.03657, 2016. URL http://arxiv.org/abs/1606.03657.

[49] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2015. URL http://arxiv.org/abs/1511.06434. cite arxiv:1511.06434Comment: Under review as a conference paper at ICLR 2016.

[50] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *CoRR*, abs/1812.04948, 2018. URL http://arxiv.org/abs/1812.04948.

[51] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016. URL http://arxiv.org/abs/1611.07004.

[52] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, Xiaogang Wang, and Dimitris N. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. *CoRR*, abs/1612.03242, 2016. URL http://arxiv.org/abs/1612.03242.

[53] Yuchuan Gou, Qiancheng Wu, Minghao Li, Bo Gong, and Mei Han. Segattngan: Text to image generation with segmentation attention. *CoRR*, abs/2005.12444, 2020. URL http://dblp.uni-trier.de/db/journals/corr/corr2005.html#abs-2005-12444.

[54] Ali Borji. Pros and cons of GAN evaluation measures. *CoRR*, abs/1802.03446, 2018. URL http://arxiv.org/abs/1802.03446.

[55] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.

[56] Wei Liu, Andrew Rabinovich, and Alexander C. Berg. Parsenet: Looking wider to see better. *CoRR*, abs/1506.04579, 2015. URL `http://arxiv.org/abs/1506.04579`.

[57] Guotai Wang, Wenqi Li, Sébastien Ourselin, and Tom Vercauteren. Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks. *CoRR*, abs/1709.00382, 2017. URL `http://arxiv.org/abs/1709.00382`.

[58] Y. Yuan, M. Chao, and Y. Lo. Automatic skin lesion segmentation using deep fully convolutional networks with jaccard distance. *IEEE Transactions on Medical Imaging*, 36(9):1876–1886, 2017.

[59] Hao Dong, Guang Yang, Fangde Liu, Yuanhan Mo, and Yike Guo. Automatic brain tumor detection and segmentation using u-net based fully convolutional networks. In *MIUA*, volume 723 of *Communications in Computer and Information Science*, pages 506–517. Springer, 2017.

[60] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. 07 2018.

[61] Debesh Jha, Michael Riegler, Dag Johansen, Pål Halvorsen, and Håvard Johansen. Doubleu-net: A deep convolutional neural network for medical image segmentation, 06 2020.

[62] José Ignacio Orlando, Philipp Seeböck, Hrvoje Bogunovic, Sophie Klimscha, Christoph Grechenig, Sebastian M. Waldstein, Bianca S. Gerendas,

and Ursula Schmidt-Erfurth. U2-net: A bayesian u-net model with epistemic uncertainty feedback for photoreceptor layer segmentation in pathological OCT scans. *CoRR*, abs/1901.07929, 2019. URL `http://arxiv.org/abs/1901.07929`.

[63] Jose Dolz, Karthik Gopinath, Jing Yuan, Herve Lombaert, Christian Desrosiers, and Ismail Ben Ayed. Hyperdense-net: A hyper-densely connected cnn for multi-modal image segmentation. *IEEE Transactions on Medical Imaging*, PP, 04 2018. doi: 10.1109/TMI.2018.2878669.

[64] Konstantinos Kamnitsas, Christian Ledig, Virginia F. J. Newcombe, Joanna P. Simpson, Andrew D. Kane, David K. Menon, Daniel Rueckert, and Ben Glocker. Efficient multi-scale 3d CNN with fully connected CRF for accurate brain lesion segmentation. *Medical Image Analysis*, 36:61–78, 2017.

[65] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.

[66] Jelmer M. Wolterink, Tim Leiner, Max A. Viergever, and Ivana Išgum. Generative adversarial networks for noise reduction in low-dose ct. *IEEE transactions on medical imaging*, 36(12):2536–2545, 12 2017. ISSN 0278-0062. doi: 10.1109/TMI.2017.2708987.

[67] Yuhua Chen, Feng Shi, Anthony G. Christodoulou, Zhengwei Zhou, Yibin Xie, and Debiao Li. Efficient and accurate MRI super-resolution using a generative adversarial network and 3d multi-level densely connected network. *CoRR*, abs/1803.01417, 2018. URL `http://arxiv.org/abs/1803.01417`.

[68] Eunhee Kang, Hyun Jung Koo, Dong Hyun Yang, Joon Bum Seo, and Jong Chul Ye. Cycle consistent adversarial denoising network for multiphase

coronary CT angiography. *CoRR*, abs/1806.09748, 2018. URL `http://arxiv.org/abs/1806.09748`.

[69] Karim Armanious, Chenming Yang, Marc Fischer, Thomas Küstner, Konstantin Nikolaou, Sergios Gatidis, and Bin Yang. Medgan: Medical image translation using gans. *CoRR*, abs/1806.06397, 2018. URL `http://arxiv.org/abs/1806.06397`.

[70] Karim Armanious, Chenming Yang, Marc Fischer, Thomas Küstner, Konstantin Nikolaou, Sergios Gatidis, and Bin Yang. Medgan: Medical image translation using gans, 06 2018.

[71] Chaoyue Wang, Chang Xu, Chaohui Wang, and Dacheng Tao. Perceptual adversarial networks for image-to-image transformation. *CoRR*, abs/1706.09138, 2017. URL `http://arxiv.org/abs/1706.09138`.

[72] J. F. Haddon and J. F. Boyce. Image segmentation by unifying region and boundary information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):929–948, 1990.

[73] M. Sato, S. Lakare, M. Wan, A. Kaufman, and M. Nakajima. A gradient magnitude based region growing algorithm for accurate segmentation. *Proceedings 2000 International Conference on Image Processing (Cat. No.00CH37101)*, 3:448–451 vol.3, 2000.

[74] Meng Li, Chuanjiang He, and Yi Zhan. Adaptive regularized level set method for weak boundary object segmentation. 2012.

[75] Julien Mille. Narrow band region-based active contours and surfaces for 2d and 3d segmentation. *Comput. Vis. Image Underst.*, 113(9):946–965, September 2009. ISSN 1077-3142.

[76] Li Xu, Bing Luo, and Zheng Pei. Weak boundary preserved superpixel segmentation based on directed graph clustering. *Signal Processing: Image Communication*, 65:231–239, 2018.

[77] C. Li, C. Kao, J. Gore, and Z. Ding. Implicit active contours driven by local binary fitting energy. In *CVPR*, pages 1–7, 2007.

[78] Hao Wu, Vikram V. Appia, and Anthony J. Yezzi. Numerical conditioning problems and solutions for nonparametric i.i.d. statistical active contours. *TPAMI*, 35(6):1298–1311, 2013.

[79] Y. Shi and W. C. Karl. Real-time tracking using level sets. volume 2, pages 34–41 vol. 2, June 2005.

[80] H. N. Isack A. Delong, A. Osokin and Y. Boykov. Fast approximate energy minimization with label costs. *International Journal Computer Vision*, 96 (1):1–27, 2012.

[81] J. Yuan E. Bae and X.-C. Tai. Global minimization for continuous multiphase partitioning problems using a dual approach. *International Journal Computer Vision*, 92(1):112–129, 2011.

[82] G. Aubert C. Samson, L. Blanc-Feraud and J. Zerubia. A level set model for image classification. *International Journal Computer Vision*, 40(3):187–197, 2000.

[83] T. Brox and J. Weickert. Level set segmentation with multiple regions. *IEEE Transactions on Image Processing*, 15(10):3213–3218, 2006.

[84] M. Kazhdan B. Lucas and R. Taylor. Multi-object spring level sets (muscle). pages 495–503, 2012.

[85] E. Bae and X.-C. Tai. Graph cut optimization for the piecewise constant

level set method applied to multiphase image segmentation. pages 1–13, 2009.

[86] C. Li, R. Huang, Z. Ding, C. Gatenby, D. N. Metaxas, and J. C. Gore. A level set method for image segmentation in the presence of intensity inhomogene ities with application to mri. *IEEE Transactions on Image Processing (TIP)*, 20(7):2007–2016, 2011.

[87] Qinghua Huang, Xiao Bai, Yingguang Li, Lianwen Jin, and Xuelong Li. Optimized graph-based segmentation for ultrasound images. *Neurocomputing*, 129(Complete):216–224, 2014.

[88] Jianbing Shen, Yunfan Du, and Xuelong Li. Interactive segmentation using constrained laplacian optimization. *IEEE Trans. Circuits Syst. Video Techn.*, 24(7):1088–1100, 2014.

[89] Kaihua Zhang, Qingshan Liu, Huihui Song, and Xuelong Li. A variational approach to simultaneous image segmentation and bias correction. *IEEE Trans. Cybernetics*, 45(8):1426–1437, 2015.

[90] Richang Hong, Meng Wang, Yue Gao, Dacheng Tao, Xuelong Li, and Xindong Wu. Image annotation by multiple-instance learning with discriminative feature mapping and selection. *IEEE Trans. Cybernetics*, 44(5):669–680, 2014.

[91] Huiyu Zhou, Xuelong Li, Gerald Schaefer, M. Emre Celebi, and Paul C. Miller. *Computer Vision and Image Understanding*, 117(9):1004–1016, 2013.

[92] Tony F. Chan, Selim Esedoglu, and Mila Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. Technical report, SIAM JOURNAL ON APPLIED MATHEMATICS, 2006.

[93] Joachim Weickert, Bart M. Ter Haar Romeny, and Max A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *TIP*, 7:398–410, 1998.

[94] Xu Chen, Bryan M. Williams, Srinivasa R. Vallabhaneni, Gabriela Czanner, Rachel Williams, and Yalin Zheng. Learning active contour models for medical image segmentation. In *CVPR*, pages 11632–11640, June 2019.

[95] Hoel Kervadec, Jihene Bouchtiba, Christian Desrosiers, Eric Granger, Jose Dolz, and Ismail Ben Ayed. Boundary loss for highly unbalanced segmentation. In *Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning*, volume 102, pages 285–296, 2019.

[96] Rangachari Anand, Kishan G. Mehrotra, Chilukuri K. Mohan, and Sanjay Ranka. An improved algorithm for neural network classification of imbalanced training sets. *IEEE transactions on neural networks*, 4 6:962–9, 1993.

[97] David Masko and Paulina Hensman. The impact of imbalanced training data for convolutional neural networks. 2015.

[98] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). URL `http://www.cs.toronto.edu/~kriz/cifar.html`.

[99] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. 2012.

[100] Hansang Lee, Minseok Park, and Junmo Kim. Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning. pages 3713–3717, 09 2016.

[101] Eric C. Orenstein, Oscar Beijbom, Emily Peacock, and Heidi Sosik. Whoi-plankton- a large scale fine grained visual recognition benchmark dataset for plankton classification. 10 2015.

[102] Samira Pouyanfar, Yudong Tao, Anup Mohan, Haiman Tian, Ahmed S. Kaseb, Kent Gauen, Ryan Dailey, Sarah Aghajanzadeh, Yung Hsiang Lu, Shu Ching Chen, and Mei-Ling Shyu. Dynamic sampling in convolutional neural networks for imbalanced data classification. In *Proceedings - IEEE 1st Conference on Multimedia Information Processing and Retrieval, MIPR 2018*, pages 112–117, 6 2018.

[103] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016.

[104] Jia Deng, Wei Dong, Richard Socher, Li jia Li, Kai Li, and Li Fei-fei. Imagenet: A large-scale hierarchical image database. In *In CVPR*, 2009.

[105] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks : the official journal of the International Neural Network Society*, 106:249–259, 2018.

[106] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[107] J.T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR (workshop track)*, 2015. URL `http://lmb.informatik.uni-freiburg.de/Publications/2015/DB15a`.

[108] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy. Training deep neural networks on imbalanced data sets. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 4368–4374, July 2016.

[109] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, Oct 2017.

[110] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015.

[111] Keisuke Nemoto, Ryuhei Hamaguchi, Tomoyuki Imaizumi, and Shuhei Hikosaka. Classification of rare building change using cnn with multi-class focal loss. pages 4663–4666, 07 2018.

[112] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[113] Chong Zhang, Kay Chen Tan, and Ruoxu Ren. Training cost-sensitive deep belief networks on imbalance data problems. In *2016 International Joint Conference on Neural Networks, IJCNN 2016*, pages 4362–4367, 2016.

[114] Y. Zhang, L. Shuai, Y. Ren, and H. Chen. Image classification with category centers in class imbalance situation. In *2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 359–363, May 2018.

[115] Wan Ding, Dong-Yan Huang, Zhuo Chen, Xinguo Yu, and Weisi Lin. Facial action recognition using very deep networks for highly imbalanced class distribution. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA*, pages 1368–1372, 2017.

[116] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[117] Qi Dong, Shaogang Gong, and Xiatian Zhu. Class rectification hard mining for imbalanced deep learning. pages 1869–1878, 10 2017. doi: 10.1109/ICCV.2017.205.

[118] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pages 3730–3738, 2015. ISBN 978-1-4673-8391-2.

[119] Qiang Chen, Junshi Huang, Rogerio Feris, L Brown, Jian Dong, and Shuicheng Yan. Deep domain adaptation for describing people based on fine-grained clothing attributes. pages 5315–5324, 06 2015. doi: 10.1109/CVPR.2015.7299169.

[120] Enlu Lin, Qiong Chen, and Xiaoming Qi. Deep reinforcement learning for imbalanced classification. *CoRR*, abs/1901.01379, 2019.

[121] C. Huang, Y. Li, C. C. Loy, and X. Tang. Learning deep representation for imbalanced classification. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5375–5384, 2016.

[122] Shin Ando and Chun Yuan Huang. Deep over-sampling framework for classifying imbalanced data. In Michelangelo Ceci, Jaakko Hollmén, Ljupčo Todorovski, Celine Vens, and Sašo Džeroski, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 770–785, Cham, 2017. Springer International Publishing.

[123] M.Havaei, A.Davy, D. Warde-Farley, A. Biard, A. C. Courville, Y. Bengio, C. Pal, P.Jodoin, and H. Larochelle. Brain tumor segmentation with deep neural networks. *Medical Image Analysis*, 35:18–31, 2017.

[124] S.Carole H., L. Wenqi, V.Tom, O.Sebastien, and M. J.Cardoso. Gener-

alised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *DLMI and MLCS*, pages 240–248, 2017.

[125] Sida Peng, Wen Jiang, Huaijin Pi, Xiuli Li, Hujun Bao, and Xiaowei Zhou. Deep snake for real-time instance segmentation. In *CVPR*, 2020.

[126] Alfred Gray, Elsa Abbena, Simon Salamon, et al. Modern differential geometry of curves and surfaces with mathematica. 2006.

[127] R. T. Farouki and C. A. Neff. Analytic properties of plane offset curves. 7(1–4), 1990.

[128] Gershon Elber, In-kwon Lee, and Myung-Soo Kim. Comparing offset curve approximation methods. *Computer Graphics and Applications, IEEE*, 17:62–71, 06 1997. doi: 10.1109/38.586019.

[129] Li Wang, Dong Nie, Guannan Li, Élodie Puybareau, Jose Dolz, Qian Zhang, Fan Wang, Jing Xia, Zhengwang Wu, Jiawei Chen, et al. Benchmark on automatic 6-month-old infant brain segmentation algorithms: The iseg-2017 challenge. *IEEE TMI*, 2019.

[130] Adriënne Mendrik, Koen Vincken, Hugo Kuijf, Marcel Breeuwer, Willem Bouvy, Jeroen de Bresser, Amir Alansary, Marleen de Bruijne, Aaron Carass, Ayman El-Baz, Amod Jog, Ranveer Katyal, Ali Khan, Fedde Lijn, Qaiser Mahmood, Ryan Mukherjee, Annegreet Opbroek, Sahil Paneri, Sérgio Pereira, and Max Viergever. Mrbrains challenge: Online evaluation framework for brain image segmentation in 3t mri scans. *Computational Intelligence and Neuroscience*, 2015:1–16, 01 2015. doi: 10.1155/2015/813696.

[131] Bjoern H Menze, Andras Jakab, Bauer, et al. The multimodal brain tumor image segmentation benchmark (brats). *TMI*, 34(10):1993–2024, 2015.

[132] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *CoRR*, abs/1607.08022, 2016. URL `http://arxiv.org/abs/1607.08022`.

[133] Hao Chen, Qi Dou, Lequan Yu, Jing Qin, and Pheng-Ann Heng. Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images. *NeuroImage*, 170:446 – 455, 2018.

[134] Richard McKinley, Raphael Meier, and Roland Wiest. Ensembles of densely-connected cnns with label-uncertainty for brain tumor segmentation. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, pages 456–465, 2019.

[135] Shengcong Chen, Changxing Ding, and Minfeng Liu. Dual-force convolutional neural networks for accurate brain tumor segmentation. *Pattern Recognition*, 88:90–100, 2019.

[136] Reuben Dorent, Wenqi Li, Jinendra Ekanayake, Sebastien Ourselin, and Tom Vercauteren. Learning joint lesion and tissue segmentation from task-specific hetero-modal datasets. *arXiv preprint arXiv:1907.03327*, 2019.

[137] Kevin Pham, Xiao Yang, Marc Niethammer, Juan C Prieto, and Martin Styner. Multiseg pipeline: automatic tissue segmentation of brain mr images with subject-specific atlases. In *Medical Imaging 2019: Biomedical Applications in Molecular, Structural, and Functional Imaging*, volume 10953, page 109530K. International Society for Optics and Photonics, 2019.

APPENDIX