

NOVEL MODEL-BASED AND DEEP LEARNING APPROACHES TO
SEGMENTATION AND OBJECT DETECTION IN 3D MICROSCOPY IMAGES

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Camilo Aguilar

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Mary Comer, Chair

School of Electrical and Computer Engineering

Dr. Charles Bouman

School of Electrical and Computer Engineering

Dr. Maggie Zhu

School of Electrical and Computer Engineering

Dr. Jeff Simmons

Air-force Research Lab

Approved by:

Dr. Dimitrios Peroulis

Head of the School Graduate Program

ACKNOWLEDGMENTS

I thank Prof Comer for her guidance, patience, and understanding throughout these years, I grew professionally and personally thanks to her and I will always be grateful.

I thank Jeff Simmons and Josiane Zerubia for their contributions and constant interest in my work. I could finish this thesis in great part thanks to their theoretical and professional contributions.

I thank Professor Bouman and Professor Zhu for agreeing to be in my committee. During our interactions, they provided me great insight to be a better Ph.D. student.

I thank my parents Sofia and Hugo. During the lowest moment of my Ph.D., they were there to tell me to stop whining and start working. Also, I thank my siblings Paul, Karina and Santiago for being my examples to follow and re-assuring me that I should come back and contribute back to our country like they did years ago.

I thank my closest friends: Thilo, Enrico, Leo, Dani, Mathilde, Laura, Maliha, Kyle, Mario, Omar, and Chino for all the great experiences and jokes we built in West Lafayette and California.

I thank all my friends from the lab, classes, Ecuadorian Association at Purdue, soccer and salsa! They have made this journey very enjoyable and I will always cherish the graduate experience at Purdue!

I thank the faculty and staff at the Electrical and Computer Engineering department at Purdue University. Prof. Aly El Gamal, Prof. Nicolo Michelusi, Prof. Stanley Chan and Prof. Gerhard Klimeck were both mentors and friends during these years!

Finally, I thank Tomasz Konopczyński for sharing his fiber dataset that contributed to validate the methods shown in this thesis.

TABLE OF CONTENTS

	Page
LIST OF FIGURES	viii
SYMBOLS	xi
ABBREVIATIONS	xii
ABSTRACT	xiii
1 INTRODUCTION	1
1.1 Motivation	1
1.2 Model-Based Approaches	3
1.3 Deep Learning	4
1.4 Illustrative Dataset and Experiment	5
1.5 Challenges	6
1.6 Report organization and contributions	7
1.7 Publications of this Work	7
2 MARKED POINT PROCESS AND STOCHASTIC SIMULATION	9
2.1 Point Processes	9
2.2 Marked Point Processes	9
2.3 MPP Density	10
2.4 Simulated Annealing	10
2.5 MCMC Samplers	11
2.6 State of the art MPP	11
3 A MARKED POINT PROCESS INCORPORATING AN ACTIVE CON- TOUR ENERGY	14
3.1 Overview	14
3.2 Parametric Active Contours	14
3.3 Contour Energy	16

	Page
3.3.1 External Energy	16
3.3.2 Boundary Energy Optimization	17
3.4 A Marked Point Process with Active Boundary Energy	18
3.4.1 Marked Point Process Framework	18
3.4.2 Irregular Shape Marked Point Process	19
3.5 Optimization	20
3.6 Results and Experiments	22
3.6.1 Void Detection in Fiber Reinforced Polymer	22
3.6.2 Human Cells:	23
3.7 Chapter Summary	25
4 SEGMENTATION AND DETECTION OF IRREGULARLY-SHAPED RE- GIONS USING INTEGRATED MARKED POINT PROCESS AND LEVEL SETS	26
4.1 Overview	26
4.2 Level Sets Segmentation	28
4.2.1 Level Set Optimization	29
4.3 Marked Point Process with Level Sets	30
4.3.1 Single Object Level Set	32
4.3.2 Multiple Objects Level Set	33
4.3.3 MPP Energy	33
4.3.4 Single Object Potential U_o	34
4.3.5 Prior Potential U_p	36
4.4 Marked Point Process Optimization	36
4.4.1 Multiple Births and Deaths with Strictly Unary Level Set	37
4.4.2 Simulating Multiple Births and Deaths with Splitting Level Sets	37
4.5 Experimental Results	40
4.5.1 Human Red Blood Cells	40
4.5.2 NiCrAl Alloy	42
4.5.3 Fiber Reinforced Polymers	44

	Page
4.6 Towards 3D Results in Fiber Reinforced Polymers	49
4.7 Chapter Summary	49
5 3D VOID DETECTION AND FIBER EXTRACTION FOR STATISTICAL CHARACTERIZATION OF FIBER-REINFORCED POLYMERS	52
5.1 Overview	52
5.2 Related Fiber and Void Segmentation Works	52
5.3 Model Based Methods: Marked Point Process	54
5.3.1 Fibers: Connected Tube Marked Point Process	54
5.3.2 Voids: Hybrid-Level sets and Active Contours	55
5.4 Surrogate approach with Convolutional Neural Networks	55
5.4.1 Encoder-Decoder Embedding Instance Segmentation	58
5.4.2 Network Training	62
5.5 Results	64
5.5.1 Experimental Data: Glass Fiber Reinforced Polymer	66
5.5.2 Labeled Data: Synthetic Low Resolution Fibrous Material	68
5.6 Chapter Summary	73
6 3D FIBER SEGMENTATION WITH DEEP CENTER REGRESSION AND GEOMETRIC CLUSTERING	76
6.1 Overview	76
6.2 Related Work	76
6.2.1 Instance Embedded Learning	76
6.2.2 Joint-Task Learning	78
6.3 Method	78
6.3.1 Geometric Constrained Clustering	81
6.3.2 Volume Tiling and Merging	85
6.4 Experiments	85
6.4.1 Training Information: Synthetic Fibers	86
6.4.2 Low Resolution SFRP Dataset	89
6.4.3 High resolution SFRP: Polypropylene Matrix	90

	Page
6.4.4 Preceramic Polymer in Unidirectional Fiber Beds	94
6.5 Chapter Summary	96
7 Summary and Future Work	99
7.1 Summary	99
7.1.1 Model Based Methods vs Deep Learning	100
7.2 Future Work: Towards Model-Based Deep Learning	101
REFERENCES	102
A Hybrid Level Sets with Shape Prior Curve Evolution	108
VITA	110

LIST OF FIGURES

Figure	Page
1.1 Sample Microscopy Images	2
1.2 Sample artifacts generated during the sample reconstruction.	2
1.3 Illustrative dataset: CT Reconstruction of polypropylene matrix reinforced with glass fibers	6
2.1 Marked Point Process Descriptors	10
2.2 Parallel RJMCMC Sampler	12
2.3 Sample arbitrary shapes MPP	13
3.1 Example of irregularly-shaped objects	15
3.2 Sample results from EMMPM and active contours segmentation	15
3.3 Influence of β at constant $\alpha = 1$ when detecting human stem cells	17
3.4 Void Detection on Fiber Reinforced Polymer	22
3.5 Human Cells	24
4.1 Sample results using different approaches.	27
4.2 Converges for different values of β_{edge}	30
4.3 Components of hybrid energy.	31
4.4 Convergence of evolving level sets at different initializations.	32
4.5 Original marks disk and deformed disk.	33
4.6 Sample of multiple objects level set in satellite image of lakes. Image obtained from Kaggle's Satellite Image Classification Challenge [38].	34
4.7 Quality energy term.	35
4.8 Human red blood cells . Images obtained from Broad Bioimage Benchmark Collection [1].	42
4.9 MPP-LS results on NiCrAl alloy image	44
4.10 Fiber reinforced polymer segmentation evaluation	45

Figure	Page
4.11 Detecting voids in fiber reinforced polymers. Second column: hybrid level sets only. Third column: MPP-LS. Images provided by Dr. Michael Sangid's laboratory, Purdue University	48
4.12 MPP-LS results on fiber reinforced polymers	50
4.13 Procedure to expand results to 3D	51
4.14 3D Void detection from different views and comparison with rough manual segmentations	51
5.1 Sample architecture with residual blocks proposed by [13]	53
5.2 Sample connected-tube MPP results	55
5.3 Results of Model-Based void detection	56
5.4 Sample volume sizes	57
5.5 Proposed surrogate approach to train neural networks from model-based results. The Tube MPP detects fibers and the AC-MPP detects voids. . .	58
5.6 Sample bounding-box proposals for 3D narrow fibers.	59
5.7 Sample watershed segmentation approach for 3D fibers.	59
5.8 Method proposal for inference	60
5.9 Encoder Decoder Network Architecture proposed by [5]	61
5.10 Memory Consumption for different semantic segmentation architectures . .	61
5.11 Embedded learning at different iterations. Images have been reduced to 2 dimensions using t-SNE [58] for visualization purposes	64
5.12 f1 score vs number of embedding	65
5.13 Semantic Segmentation for polypropylene matrix composite. Fibers are represented green, voids are represented blue	67
5.14 Fiber orientation tensors at different locations. Image obtained from the ACME Lab at Purdue University.	69
5.15 Instance Segmentation	72
5.16 Time Difference	74
6.1 Sample results of our method displayed with Fiji [37] software. Each color represents a different instance of an object. Image courtesy of the ACME Lab at Purdue University.	77
6.2 Architecture: we used a modified version of U-Net [5].	79

Figure	Page
6.3 Network Outputs: The networks detect the fiber pixels and their instance centroid μ_c . Each color represents an instance, and pixels are clustered around their centers. Gray fibers are for display purposes. The figures were generated with the Fiji Software.	80
6.4 Cropped image and its birthmap.	82
6.5 Sample cluster proposal and ground truth labels.	83
6.6 Ground truth labels and a sample inference.	83
6.7 Merging Procedure	85
6.8 Training dataset: synthetic fibers generated by Konopczynski [62]	87
6.9 Evaluation of average tile scores vs eps parameter in synthetic data. For the proposed method, we use $\text{eps} \equiv r_{min}$	89
6.10 Evaluation of mean tile scores vs eps parameter in Low Resolution SFRP Dataset.	92
6.11 Low Resolution SFRP Dataset. Each color represents a different fiber instance and white pixels represent unlabeled pixels.	93
6.12 High resolution fiber reinforced polymers: evaluation Scores vs eps parameter	95
6.13 SFRP: Polypropylene Matrix. Each color represents a different instance of a fiber	96
6.14 Object detection in unidirectional fiber beds.	97

SYMBOLS

K	Image lattice
\sim	Neighbor relation
U_p	Prior energy
U_d	Data energy
ω	Marked object
\mathbf{w}	Marked object configuration
Ω	Space of object configurations
\mathbf{x}	Point process realization
$p(\cdot)$	Probability density function
$h(\cdot)$	Probability density function
$\tilde{\omega}$	Evolved contour
ϕ	Level set
$\tilde{\phi}$	Evolved level set
\hat{n}	Norma to a curve
e	Output vector from CNN denoting embedded space

ABBREVIATIONS

AC	Active Contour
ACME	Advanced Computational Materials and Experimental Evaluation
CNN	Convolutional Neural Network
CT	Computed Tomography
DIC	Differential Interference Contrast
FCN	Fully Convolutional Neural Network
GAC	Geodesic Active Contour
GPU	Graphics Processing Unit
HLS	Hybrid Level Set
MBD	Multiple Birth and Death
MC	Monte Carlo
MCMC	Markov Chain Monte Carlo
MPP	Marked Point Process
MRF	Markov Random Field
LS	Level Set
RBC	Red Blood Cell
RJMCMC	Reversible Jump Markov Chain Monte Carlo
RCNN	Regions with Convolutional Neural Networks
PDE	Partial Differential Equation
YOLO	You Only Look Once

ABSTRACT

Aguilar, Camilo G. Ph.D., Purdue University, August 2020. Novel Model-Based and Deep Learning Approaches to Segmentation and Object Detection in 3D Microscopy Images. Major Professor: Mary Comer.

Modeling microscopy images and extracting information from them are important problems in the fields of physics and material science. Model-based methods, such as marked point processes (MPPs), and machine learning approaches, such as convolutional neural networks (CNNs), are powerful tools to perform these tasks. Nevertheless, MPPs present limitations when modeling objects with irregular boundaries. Similarly, machine learning techniques show drawbacks when differentiating clustered objects in volumetric datasets. In this thesis we explore the extension of the MPP framework to detect irregularly shaped objects. In addition, we develop a CNN approach to perform efficient 3D object detection. Finally, we propose a CNN approach together with geometric regularization to provide robustness in object detection across different datasets.

The first part of this thesis explores the addition of boundary energy to the MPP by using active contours energy and level sets energy. Our results show this extension allows the MPP framework to detect material porosity in CT microscopy images and to detect red blood cells in DIC microscopy images.

The second part of this thesis proposes a convolutional neural network approach to perform 3D object detection by regressing objects voxels into clusters. Comparisons with leading methods demonstrate a significant speed-up in 3D fiber and porosity detection in composite polymers while preserving detection accuracy.

The third part of this thesis explores an improvement in the 3D object detection approach by regressing pixels into their instance centers and using geometric

regularization. This improvement demonstrates robustness when comparing 3D fiber detection in several large volumetric datasets.

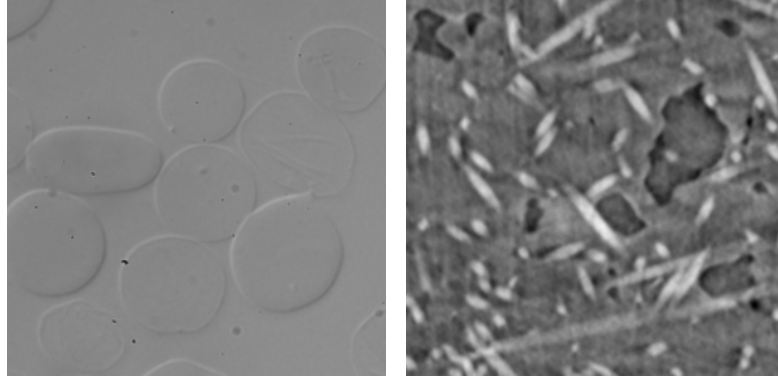
These methods can contribute to fast and correct structural characterization of large volumetric datasets, which could potentially lead to the development of novel materials.

1. INTRODUCTION

1.1 Motivation

Image segmentation and object extraction constitute key elements for characterizing microscopy images. Novel high-throughput imaging techniques such as electron microscopy yield massive amounts of images often comprised of several thousand objects. These images, and the system they represent, contain significant information that contributes to explaining biological or physical phenomena. For instance, the shape and deformability of human red blood cells (RBC) can indicate the existence of pathological conditions; in fact, numerous state-of-the-art works are currently characterizing the biological mechanics involving the cell's boundary deformation [1]. Similarly, numerous works in material science require of segmentation and object detection. For instance, the development of novel lightweight materials with high fatigue resistance requires of detailed microstructural characterization. The arrangement and interaction of the material's microstructural components influence the material's response to external loads before reaching failure [2]. Fig. 1.1(a) shows an example of a differential interference contrast(DIC) image of red blood cells denoting their bi-concave shape and their often-irregular boundary. Fig. 1.1(b) denotes a sample cross section of a fiber reinforced polymer where the direction of fibers can affect the material's response to external loads. Image segmentation and object extraction provide solutions to these tasks relating boundary detection and microstructural characterization.

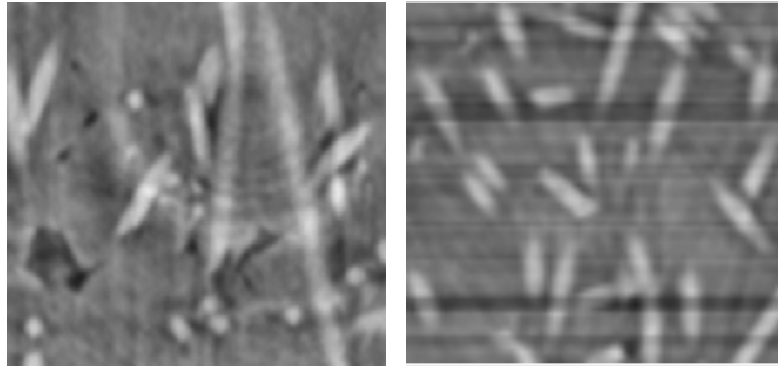
Nevertheless, microscopy images present several challenges to traditional segmentation techniques. These images are subject to distortion originated from diverse sources such as electron scattering, volume averaging, and artifacts generated during the reconstruction procedure. For example, Fig. 1.2(a) shows ring artifacts generated



(a) Human red blood cells [1] (b) Fiber reinforced polymers [2]

Fig. 1.1. Sample Microscopy Images

during the CT image reconstruction of a fiber reinforced polymer sample across the XY plane. Fig. 1.2(b) shows line artifacts generated by the reconstruction software across the ZY plane. Noise sources, combined with the intrinsic ambiguity of microscopy images, poses challenges to compute and to label segmented data, even for domain expert labeling.



(a) Sample ring artifacts (b) Sample line artifacts

Fig. 1.2. Sample artifacts generated during the sample reconstruction.

In recent years, deep learning has gained popularity as a general and accurate segmentation technique. However, deep learning requires of large training datasets, which are not readily available in microscopy imaging. In addition, deep learning often needs of a graphic processing unit (GPU) that limits the amount of memory a deep learning approach can process. This poses a constrain in memory intensive tasks such as 3D segmentation.

In this thesis, we propose model-based approaches to perform 2D segmentation of irregularly shaped objects based on contour boundaries and point processes. Furthermore, we propose neural network approaches adapted to volumetric data to perform segmentation of 3D fiber reinforced polymer data.

1.2 Model-Based Approaches

Model-based image processing techniques draw parallel analogies from pixel-wise interactions to well-known physics properties. For instance, the Ising and Potts models have an equivalent counterpart in the field of statistical mechanics, namely the Boltzmann Distribution. This equivalence implies that we can use Markov Random Fields(MRFs) to model material behaviors. One example of this work is to modify the interaction parameter β to simulate coating interface between different phases in materials [3]. Another example is to use MRFs to model the stationary probability distribution of polycrystalline microstructures and use this model to simulate and characterize the mechanical behavior of these materials [4].

In general, model-based techniques are important to analyze images when we reach physical systems limitations. We are particularly interested in stochastic models because they can express the properties of interest with a probability density function $p(\omega)$. This framework provides a setup to generate and evaluate samples from a distribution, generally using Monte Carlo(MC) sampling techniques.

Among numerous models in the probabilistic framework, we focus our report on the marked point processes. This approach uses a point process to model a distribu-

tion of objects and their interactions with each other. MPPs provide the theoretical framework to employ stochastic samplers in simulated annealing in order to find a likely configuration. In this report, we explore two approaches to add an irregular shape framework in order to consider an object’s boundary properties. For example, surface roughness or curvature. Also, we explore using the balloon method and level sets method to propose samples. These methods allowed us to capture large objects with irregular geometries and use them as part of our object configuration. The results show the characterization of systems with irregularly shaped objects, and the effect of varying the contour parameters. These are the first step towards the characterization of systems comprised of different geometries or irregular shapes.

1.3 Deep Learning

Machine learning techniques aim to approximate a mapping function $y = f(x, \theta)$ to represent the function $f(\cdot, \theta)$ from the input vectors x to the output vectors y , with parameters θ . Convolutional neural networks(CNNs) are a subset of the machine learning techniques that were often employed in the 80s and 90s; however, they presented difficulties training: deep neural networks require of more parameters than classic machine learning techniques and they often over-fitted the training data. The lack of training data and computational power caused neural networks to perform worse than classical machine learning methods. During the recent decade, the increased availability of training data combined with the improved computational resources allowed neural networks to achieve major breakthroughs in areas such as speech recognition, artificial intelligence, and object segmentation and extraction. Since 2006, deep neural networks have received major attention and have achieved unmatched results in segmentation. Popular segmentation architectures involve U-Net [5], Fully Convolutional Neural Networks (FCN) [6], Feature Pyramid Networks [7], Fully Convolutional Dense Nets [8], DeepLabv3 [9] and popular object detection techniques include RCNNs [10, 11], YOLO [12], Embedded learning [13].

In this thesis, we propose a novel deep learning approach to extract objects from 3D data. We propose a memory efficient method inspired in embedded learning and clustering that can capture fibers and voids in a 3D fiber reinforced polymer dataset. In addition, we extend this work to generalize to other datasets by combining concepts from our model-based method and deep learning approach. Our results show improved object detection in several fiber-based reconstructed volumes coming from different datasets.

1.4 Illustrative Dataset and Experiment

We test most of our experiments in a cylindrical specimen comprised of glass fiber in a polypropylene matrix composite. This material was fabricated at Dupont and it was imaged with X-ray tomography at Argonne National Laboratory. The samples were imaged with a 2-BM beamline and have a resolution of $1.3 \mu\text{m}$ per pixel. The reconstruction was performed using the Tomopy software by Aggei et al. and used in several characterization experiments [2, 14, 15].

The sample contains three types of components: matrix (gray voxels), glass fibers (bright voxels), and voids (dark voxels), which present irregular geometries and are mainly described by their edges. Fig. 1.2(b), 1.2(a) 1.1(b) denote sample cross-section views of this material.

The dataset was obtained by the ACME Lab at Purdue University. This experiment proposes to characterize the material's structural and mechanical properties. The structural properties include object location, length, diameter, orientation, volume fraction, or interaction with other objects. Fig. 1.3(a) denotes a sample instance of a reconstructed volume, Fig 1.3(b) denote the segmented fibers and Fig. 1.3(c) denote the segmented voids. The mechanical properties include the tracking of events such as fiber pull out, breakage, or fiber debonding.

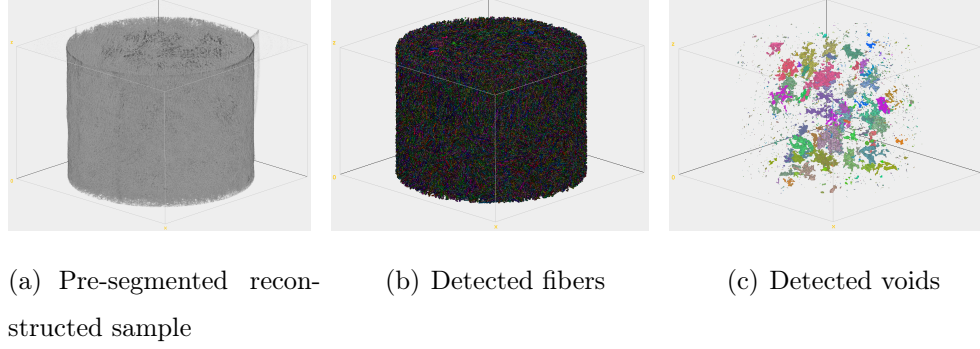


Fig. 1.3. Illustrative dataset: CT Reconstruction of polypropylene matrix reinforced with glass fibers

1.5 Challenges

The challenges we aim to solve with the work proposed in this report are:

- **System Characterization** Numerous experiment's objective is to find the underlying distribution of objects and their effect on the system's overall behavior. This requirement is the motivation to utilize point processes explained in chapter 2 to model objects in systems.
- **Objects with highly irregular boundaries** material science and biological sciences often image and characterize shapes with irregular boundaries or objects without a pre-defined pattern. This property is the main motivation to add boundary energy to the MPP model in chapters 3 and 4.
- **Objects with low contrast** Fig. 1.1(a) and 1.1(b) show that objects such as cells in DIC images and voids in CT images do not present high contrast between their interior and exterior regions. We added the balloon force in chapters 3, 4 and trained CNNs in chapter 5 to deal with the challenge of detecting objects with low-contrast.
- **Big Data** As imaging systems improve, they generate larger amounts of data. For instance, the size of the dataset described in section 1.4 is of $2500 \times 2500 \times 1300$

voxels. Hence we require faster and more efficient algorithms. We propose an approach to propose objects for our sampler in chapter 4 and employ CNNs in chapters 5, 6 to meet this challenge.

1.6 Report organization and contributions

The report and contributions of this thesis are organized as follows:

- **Chapter 1** Provides the motivation for the report and a brief description of model-based and deep learning-based segmentation methods.
- **Chapter 2** Presents a theoretical background of MPPs, the stochastic simulation techniques employed, and the state of the art MPP methods.
- **Chapter 3** Explores the addition of an active contour boundary energy to the MPP framework applied to microscopy images.
- **Chapter 4** Provides an extension to Chapter 3 by exploring the utilization of a level sets framework and proposes a method to guide the MPP sampler.
- **Chapter 5** Proposes a novel CNN approach to tackle microscopy volume segmentation.
- **Chapter 6** Extends the CNN approach to adapt for other datasets by enforcing geometric regularization inspired in the MPP model.

1.7 Publications of this Work

- C. Aguilar and M. Comer, “A Marked Point Process Model Incorporating Active Contours Boundary Energy,” *Electronic Imaging*, vol. 2018, no. 15, pp. 230-12304, 2018

- C. Aguilar and M. Comer, “Void detection and fiber extraction for statistical characterization of fiber-reinforced polymers,” *Electronic Imaging*, vol. 2020, no. 23.
- *C. Aguilar and M. Comer, “Segmentation and Detection of Irregularly-Shaped Regions Using Integrated Marked Point Processes and Level Sets,” in *IEEE Transactions on Image Processing* to be submitted July 2020.
- C. Aguilar and M. Comer, “3D Fiber Segmentation with Deep Center Regression and Geometric Clustering,” in *IEEE Transactions on Image Processing*. To be submitted July 2020.

2. MARKED POINT PROCESS AND STOCHASTIC SIMULATION

2.1 Point Processes

A point process (PP) in a compact lattice $K \subset \mathbb{R}^p$ is a realization of a random configuration of an unordered set of points $\mathbf{x} = \{k_1, k_2, \dots, k_N\}$ where $k_i \in K$ represents the i_{th} point and N is a finite random variable. The most common point process is given by a homogeneous Poisson distribution with intensity $\lambda\mu(K)$, where $\mu(\cdot)$ is proportional to the Lebesgue measure. A sample homogeneous Poisson distribution with intensity $\lambda = 1$ is depicted in Fig. 2.1(a).

2.2 Marked Point Processes

A marked point process is the assignation of a mark $m \in M$ to each point in \mathbf{x} . The mark space M defines the possible geometries and dimensions of the objects, and a single mark $m \in M$ describes a specific object. For example, the mark space for ellipses is defined by $M = [a_{min}, a_{max}] \times [b_{min}, b_{max}] \times [\theta_{min}, \theta_{max}]$, where a, b, θ denote the major and minor axis respectively, and θ represents the ellipse orientation. We describe an object ω by its location and its mark in the form of $\omega = (k, m) \in K \times M$. Fig 2.1(b) denotes a sample ellipse object.

We define a realization of the MPP as collection of objects $\mathbf{w} = \{\omega_1, \omega_2, \dots, \omega_N\} \in \Omega$, where Ω is the space of possible configurations and is defined as:

$$\Omega = \bigcup_{n \in \mathbb{N}} \Omega_n, \quad (2.1)$$

Where Ω_n represents the configuration space containing n objects and $\Omega_0 = \{\emptyset\}$. Figure 2.1(c) shows a sample the realization of a point process denoted in Fig. 2.1(a) with a random mark $m \in M$ assigned to each point $k \in \mathbf{x}$ from Fig. 2.1(a).

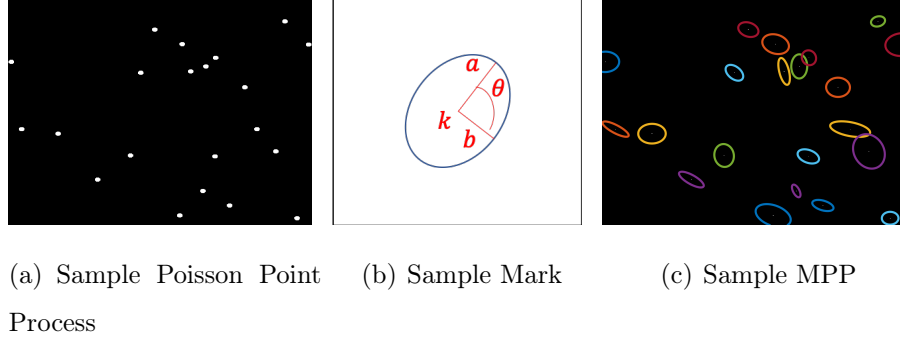


Fig. 2.1. Marked Point Process Descriptors

2.3 MPP Density

In this report we will assume we only work with Markov point process, hence we can express the density of an MPP as $h(\mathbf{w}) \propto \exp \{-U(\mathbf{w})\}$, where $U(\mathbf{w})$ denotes the Gibbs energy and is generally given by:

$$U(\mathbf{w}) = \sum_{\omega_i \in \mathbf{w}} U_d(\omega_i) + \sum_{\substack{\omega_i, \omega_j \in \mathbf{w} \\ \omega_i \sim \omega_j}} U_p(\omega_i, \omega_j) \quad (2.2)$$

Where $U_d(\omega_i)$ describes how well a single object fits the image, $U_p(\omega_i, \omega_j)$ describes an interaction prior, and \sim denotes the symmetric neighbor relation. The data dependent term $U_d(\omega_i)$ usually depends on the contrast between the interior and exterior regions of the object. However, in this report, we modified $U_d(\omega)$ to fit the boundary energy.

2.4 Simulated Annealing

Our aim is to find the most likely configuration $\hat{\mathbf{w}} \in \Omega$ given by:

$$\hat{\mathbf{w}} = \operatorname{argmax} h(\mathbf{w}) \quad (2.3)$$

In order to achieve this goal, we used simulated annealing. This approach repeatedly generates samples on the distribution: $h(\mathbf{w})^{\frac{1}{T_o}}$, where T_o represents the initial system temperature and decreases according to a specified scheme. Generally, the

sampler consists of simulating a Markov Chain $\mathbf{W}_{t \in \mathbb{N}}$ in the configuration space Ω . Initially, at high temperatures, the chain \mathbf{W}_t explores the solution space, but as the temperature approaches zero, \mathbf{W}_t converges to a distribution concentrated around the optimal configuration. While the theory of this approach ensures convergence, in practice, this result is highly influenced by the initial temperature T_o , the selected annealing scheme, and the number of iterations. Our literature review proposed that the optimal choice for T_o can be calculated by running several experiments and calculating twice the standard deviation of the energy at infinite temperature. Hence the optimal value is given by $T_o = 2\sigma(U_{T=\infty})$. Similarly, [16] determined the optimal annealing scheme to be logarithmic, however we chose a geometric decrease in the form of $T_t = T_o \alpha_T^t$, $\alpha \in (0, 1)$ because it was proven to achieve reasonable results in faster time [17].

2.5 MCMC Samplers

The energy $U(\omega)$ is usually minimized with an RJMCMC sampler [18]. This sampler simulates the MC \mathbf{W}_t and at each iteration proposes a local perturbation according to a kernel function $Q(\mathbf{w} \rightarrow \cdot)$. The choice of kernel functions can include birth/death kernels, translation, rotation, among other local perturbation of the current states. The chain needs to be reversible, irreducible, and aperiodic in order to converge (note that it is not a homogeneous chain because it is temperature dependent). In this report, we based our samplers in the Multiple Birth and Death(MBD) algorithm [19], that aims to parallelize the state transitions of \mathbf{W}_t .

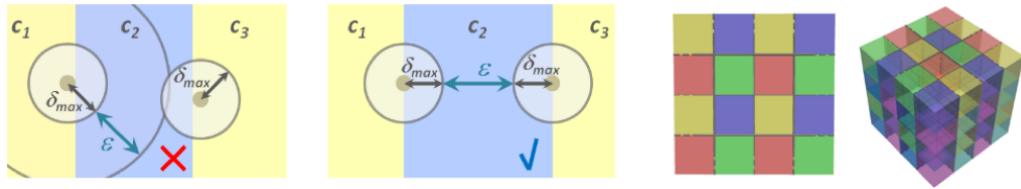
2.6 State of the art MPP

This approach has been used extensively in the last two decades to extract objects of repeated geometries in images. Popular methods involve using ellipses or lines to extract objects in satellite [20] [21] [22] or material images [23], [24]. However, we will list interesting approaches that expanded the capabilities of the MPP method by

improving the sampling techniques, or by proposing adaptations to extend beyond the geometric constraint of the MPP method.

During recent years, the area of instance segmentation experienced a rapid growth manifested in popular object proposal techniques such in mask-rcnn [10], PANet [25], Box2Pix [26]. While these approaches have proven successful in popular benchmark datasets, the instance segmentation paradigm becomes challenging when transitioning to microscopy volumetric datasets: microscopy images tend to have large numbers of clustered objects, the jump in dimensions represents a significant increase in memory requirements, and thin 3D geometries oriented in arbitrary orientations poses challenges for bounding box characterizations.

- **Improved Samplers** Descombe et al. [22] proposed the MBD sampler as to parallelize MC transitions. Verdie et al. [27] proposed a parallel sampler for the RJMCMC by exploiting the Markovian property (Fig. 2.2(a)). Cracium et al. [28] proposed a data driven sampler in spatiotemporal data by using a Kalman filter. Zhu et al. proposed a data-driven sampler [29] by pre-calculating image features.
- **Irregularly shaped objects MPP** Zhao et al. proposal a joint MRF-MPP to impose global and local constraints for segmentation. These results are depicted in Fig. 2.3(b). Descombes et al. [30] proposed an irregular shape shape library



(a) Object independence requirements and image partition in [27]

Fig. 2.2. Parallel RJMCMC Sampler

to capture objects with irregular geometries. These results are shown in Fig. 2.3(d). Kuliakova et al. [31] modeled fireworks by adding active contours with a strong prior, shown in Fig. 2.3(f). Our work extends these results.

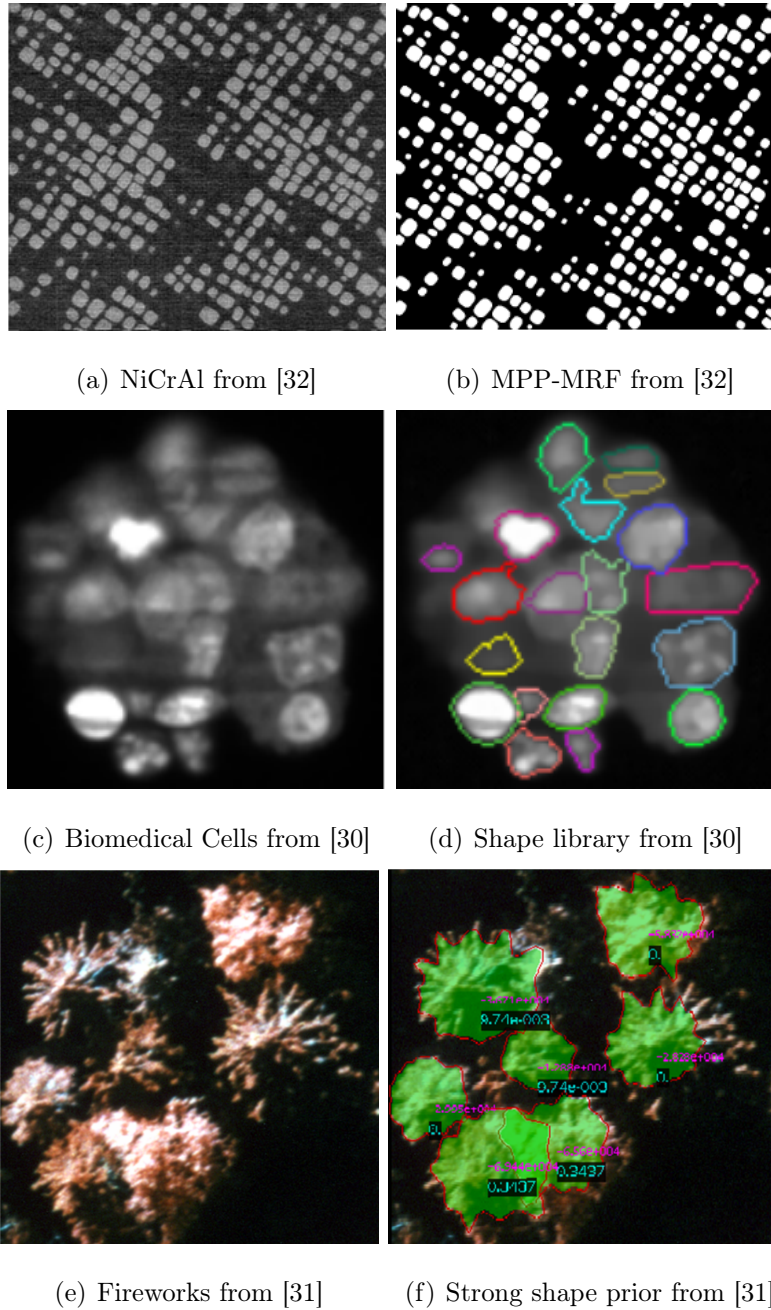


Fig. 2.3. Sample arbitrary shapes MPP

3. A MARKED POINT PROCESS INCORPORATING AN ACTIVE CONTOUR ENERGY

3.1 Overview

The MPP obtained remarkable results but numerous systems require analysis of more complex shapes without a pre-defined geometry. For example, Figure 3.1 shows two images of the fiber reinforced polymer presented in section 1.4. Void extraction represents a challenging task due to two reasons: they often have an irregular geometry, and they do not have a constant pixel intensity. These factors can cause problems in segmentation algorithms that rely on pixel intensities such as MRF based segmentations [33], shown in Figure 3.2(a).

In this chapter, we incorporate a parametric active contour energy(AC) into the MPP framework. The addition of this energy allows the MPP model to detect objects with irregular shapes. This energy accounts for the elasticity and curvature properties of the detected objects. Our method aims to extend the work presented in [34] by incorporating a different optimization technique. We also use different forces such as a pre-segmentation external force obtained from [33]. These changes allow the MPP-Active Contours (MPP-AC) framework to detect a broader range of objects while modeling the object boundary characteristics. We explore the effects of changing the elasticity and viscosity terms, and we demonstrate the results of our model applied to capture irregular shapes in material images.

3.2 Parametric Active Contours

Contour-based detection methods have shown promising results for detecting individual voids. Fig. 3.2(a) denotes the resulting contour from applying the balloon

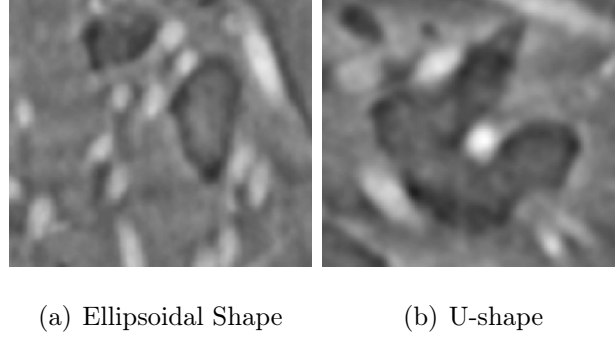


Fig. 3.1. Example of irregularly-shaped objects

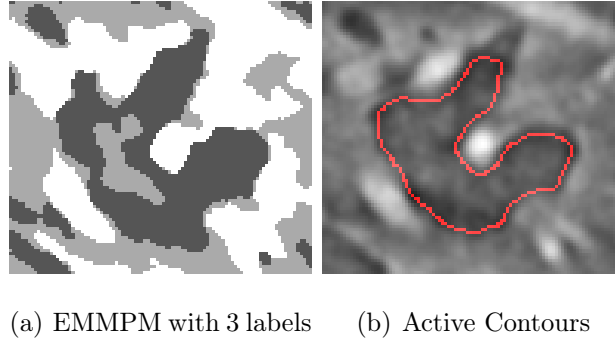


Fig. 3.2. Sample results from EMMPM and active contours segmentation

model proposed by Cohen in [35] on a single void. Kulikova et al. in [34] successfully incorporated the active contours framework into an MPP configuration to model objects with irregular shapes. This method evolved contours based on contour smoothness, image edges and precomputed background and foreground intensities. However, that approach would not yield satisfactory results on images such as Fig. 3.2(b) due to the similar intensities between the object foreground and background.

3.3 Contour Energy

We model an individual object as a closed contour denoted by $C(t) = (x(t), y(t))$, $t \in [0, 2\pi]$. This contour is defined on the image domain and is deformed according to the energy functional $E(C(t))$ given by:

$$E(C(t)) = \int_0^{2\pi} \frac{1}{2}(\alpha \|C'(t)\|^2 + \beta \|C''(t)\|) + E_{ext}(C(t)) dt \quad (3.1)$$

In this equation, $C'(t)$ and $C''(t)$ denote the first and second derivative of the contour with respect to parameter t . These terms model the object elasticity and curvature, respectively, and they are regulated by the positive parameters, α and β . During the methods described in the chapter, we set the α, β parameters experimentally. Fig. 3.3 denotes distinct convergence for different curvature weights. Larger β reduces the object's curvature. The parameter α regularizes the edge smoothness. Hence its value did not have a significant effect on the final contour. Table 3.1 denotes the raw elasticity and curvature energies of each of the converged contours of Fig. 3.3.

3.3.1 External Energy

The external energy defines the fitting of the snake with the image and it is defined by:

$$E_{ext}(C(t)) = E_{Edge}(C(t)) + E_{dark}(C(t)) \quad (3.2)$$

The first term attracts the contour towards image edges and it is given by $E_{Edge} = -\|\nabla(g_\sigma * I(x(t), y(t)))\|^2$, where g_σ is a Gaussian smoothing filter with parameter σ and ∇ is the gradient operator. The second term is a pre-segmentation energy and it is defined as : $E_{dark} = -I_k(x(t), y(t))$, where I_k is zero everywhere except at the output for a pre-segmentation algorithm. For example, in Figure 3.2(a), the class of interest is the void region, therefore we chose class $k = 1$, the one with lowest pixel intensity values. We used the EM/MPM segmentation proposed in [33] as a pre-segmentation algorithm. Finally, We used the method proposed in [35] to ensure the contour will evolve even when it is not subject to an external force. This force contributes to

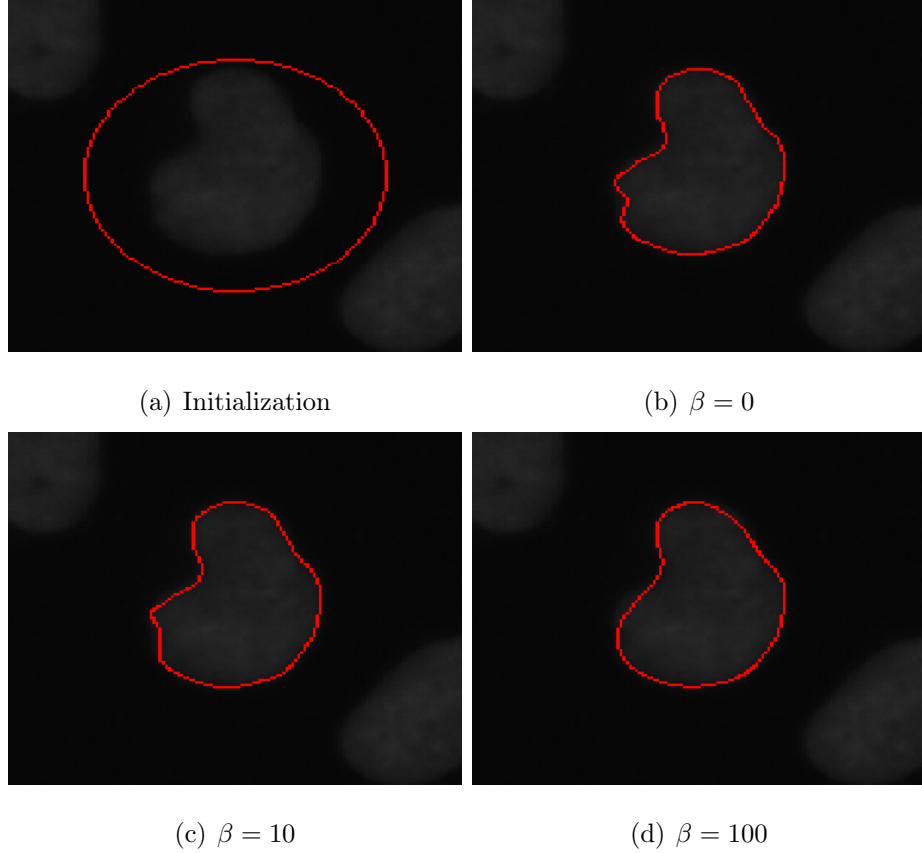


Fig. 3.3. Influence of β at constant $\alpha = 1$ when detecting human stem cells

the characterization of irregular objects that do not have a constant internal pixel intensity. The balloon force is given by: $F_{balloon} = \kappa \hat{n}(t)$, where κ determines the weight and direction and $\hat{n}(t)$ represents the normal vector to the contour.

3.3.2 Boundary Energy Optimization

We employed calculus of variations to find a force-balanced equation. The solution to this method is expressed as the Euler-Lagrange equations given by: $\alpha C'''(t) -$

Table 3.1.
Comparison for internal energy of contours in Fig. 3.3

β	$C''(t)$	$C'''(t)$
0	876.46	44.83
10	818.88	15.77
100	746.83	5.00

$\beta C'''(t) - \nabla E_{ext} = 0$. The numerical approximation for this equation is discussed in [36] and its solution is given by:

$$C^k(t) = (A + \gamma I)^{-1}(\gamma C^{k-1}(t) - \nabla E_{ext}(C^{k-1}(t))) \quad (3.3)$$

Where $C^k(t)$ denotes the contour at iteration k , γ represents the step size or viscosity parameter, Matrix A denotes a pentadiagonal matrix containing the discrete approximations of the first and second derivative coefficients, and ∇E_{ext} denotes the gradient of the external energy.

Note that the external force includes the balloon force to inflate the contour together with three constants $\kappa_1, \kappa_2, \kappa_3$ to weight the effect of each individual force: $\nabla E_{Ext} = \kappa_1 \hat{n} - \kappa_2 \nabla E_{edge} - \kappa_3 \nabla E_{dark}$. In our experiments, we used $\kappa_1 \in [-1, 1]$, $\kappa_2 \in [0, 1]$ and $\kappa_3 \in [0, 1]$, and [35] recommended to choose each κ within the same order of magnitude.

3.4 A Marked Point Process with Active Boundary Energy

3.4.1 Marked Point Process Framework

A Marked Point Process \mathbf{W} defined on $K \times M$ models the observed scene Y as a finite unordered set of random objects. We let Ω_w denote the space of all possible

realizations of \mathbf{W} . The point process can be modeled by a Gibbs density function given by

$$p(\mathbf{w}) = \frac{1}{Z} \exp\{-U(\mathbf{w})\} \quad (3.4)$$

where $U(\mathbf{w})$ describes the energy function and $Z = \int_{\mathbf{w} \in \Omega_w} p(\mathbf{w}) d\mathbf{w}$ is the normalizing constant.

3.4.2 Irregular Shape Marked Point Process

The addition of active contours boundary would require a high dimensional mark space to represent irregular geometries. However, Kulikova presented in [34] an alternative space to represent these configurations based on the disk MPP model and the contour energy functional $E(C(t))$. We let an initial disk ω_i with radius r_i belong to space $\mathbf{W} = K \times M$. We can parametrize a disk ω_i as a contour $\omega_i(t)$ living in a space W_o and define a contour energy functional $E(\omega_i(t))$. We can minimize this functional using the method described in section 3.3.2 to evolve $\omega_i(t)$ into $\tilde{\omega}_i(t) \in W_o$, with $\tilde{\cdot} : W_o \mapsto W_o$ denoting the energy minimization (and following the notation of [34]). Therefore, the new single object space W_o is equivalent to a parametrization of the disks in \mathbf{W} and deformed by $E(\omega(t))$. This space describes the contour initialized by ω_i adapted to the image by a local minimum of $E(\omega(t))$.

We followed [34] to create the extension from a single object to multiple objects, given by the symmetrical set:

$$\Omega_{W_o} = \bigcup_{n=0}^{\infty} [W_o^n / S_n], \quad (3.5)$$

where W_o^n is the space containing n deformed disks and S_n is a symmetry group of n elements on the components of W_o^n . The energy of this MPP model is described as:

$$U(\mathbf{w}) = \sum_{\omega_i \in \mathbf{w}} U_d(\omega_i) + \sum_{\substack{\omega_i, \omega_j \in \mathbf{w} \\ \omega_i \sim \omega_j}} U_p(\omega_i, \omega_j) \quad (3.6)$$

Where U_d denotes the data energy, \sim denotes neighbor relation, and U_p denotes the prior energy.

Data Energy

The data energy describes how well current configuration \mathbf{w} fits with the current image and it is the sum of all the individual data energies for each object ω_i . The data energy for a single object is given by:

$$U_d(\omega_i) = E(\tilde{\omega}_i(t)) \quad (3.7)$$

Prior Energy

This energy $U_p(\mathbf{w})$ accounts for prior knowledge about the system. Numerous MPP models use an overlapping penalizer which discourages spatial overlap between detected objects in \mathbf{w} . Hence, our prior energy depends only on the interaction between objects:

$$U_p(\omega_i, \omega_j) = \left\{ \begin{array}{ll} A(\tilde{\omega}_i, \tilde{\omega}_j) & \text{if } A(\tilde{\omega}_i, \tilde{\omega}_j) \leq T_{overlap} \\ \infty & \text{otherwise} \end{array} \right\} \quad (3.8)$$

Where $A(\tilde{\omega}_i, \tilde{\omega}_j)$ denotes the overlapping ratio between $\tilde{\omega}_i$ and $\tilde{\omega}_j$. For this section, we chose $T_{overlap} = 5\%$ experimentally in order to prevent from multiple contours converging to the same local minimum.

3.5 Optimization

Our objective is to obtain the most likely configuration $\hat{\mathbf{w}} = \operatorname{argmax}(p(\mathbf{w}))$. We can achieve this goal by minimizing the energy function given in equation (3.6). This energy function is not convex and also it is numerically infeasible to calculate the normalizing constant $Z = \int_{\mathbf{w} \in \Omega_w} p(\mathbf{w}) d\mathbf{w}$. Hence we resort to stochastic optimization embedded in a simulated annealing scheme. We simulate a MC from Ω_W and use the multiple births and death dynamics presented by [19]. This algorithm uses a pre-computed birthmap to favor certain regions during the birth phase and gives birth to multiple objects in each iteration. Then it sorts the detected objects by decreasing

energy to "kill" the less likely objects first. Finally it decreases the temperature T and the process intensity σ to ensure convergence to a strong local minimum. This optimization method is summarized in Algorithm 1. This algorithm ensures convergence to the global minimum of eq. (3.6) given the necessary conditions are met.

Algorithm 1 Multiple Birth and Death Algorithm

```

1: procedure MPP ENERGY MINIMIZATION
2:   Initialization:
3:   Create birthmap  $b_o$ 
4:   Initialize  $b_{rate} = b_o$ ,  $T = T_o$ ,  $\sigma = \sigma_o$ .
5:   Birth Step:
6:   Visit pixels in raster order
7:    $\omega' \leftarrow$  draw a sample from space  $W$ 
8:   Add  $\omega'$  to configuration  $\mathbf{w}$  with probability  $\sigma b_{rate}$ 
9:   Evolve  $\omega'$  using the method described in section 3.3 to  $\tilde{\omega}'$ 
10:  Death Step:
11:  Sort all elements of  $\mathbf{w}$  by decreasing energy.
12:  For every object  $\omega_i$  in  $\mathbf{w}$  calculate:
13:   $d_{rate}(\omega_i) = \frac{\sigma^{(k)} \exp \frac{U(\mathbf{w}|Y) - U(\mathbf{w} - \omega_i|Y)}{T^k}}{1 + \sigma^{(k)} \exp \frac{U(\mathbf{w}|Y) - U(\mathbf{w} - \omega_i|Y)}{T^k}};$ 
14:  Delete  $\omega_i$  with probability  $d_{rate}(\omega_i)$ 
15:  Convergence Test:
16:  if all the elements born during the birth step are killed during the death step
      then
17:    terminate process
18:  else
19:    Update parameters:  $T^{k+1} \leftarrow T^k \times \alpha$ ,  $\sigma^{k+1} \leftarrow \sigma^k \times \alpha$   $\alpha \in (0, 1)$ 
20:    goto Birth Step
21:  end if
22: end procedure

```

3.6 Results and Experiments

We tested our model on two different datasets: a fiber reinforced polymer and human cells, and we compared it with the software provided by [37]. This software is a popular tool to use image processing techniques to extract information from images.

3.6.1 Void Detection in Fiber Reinforced Polymer

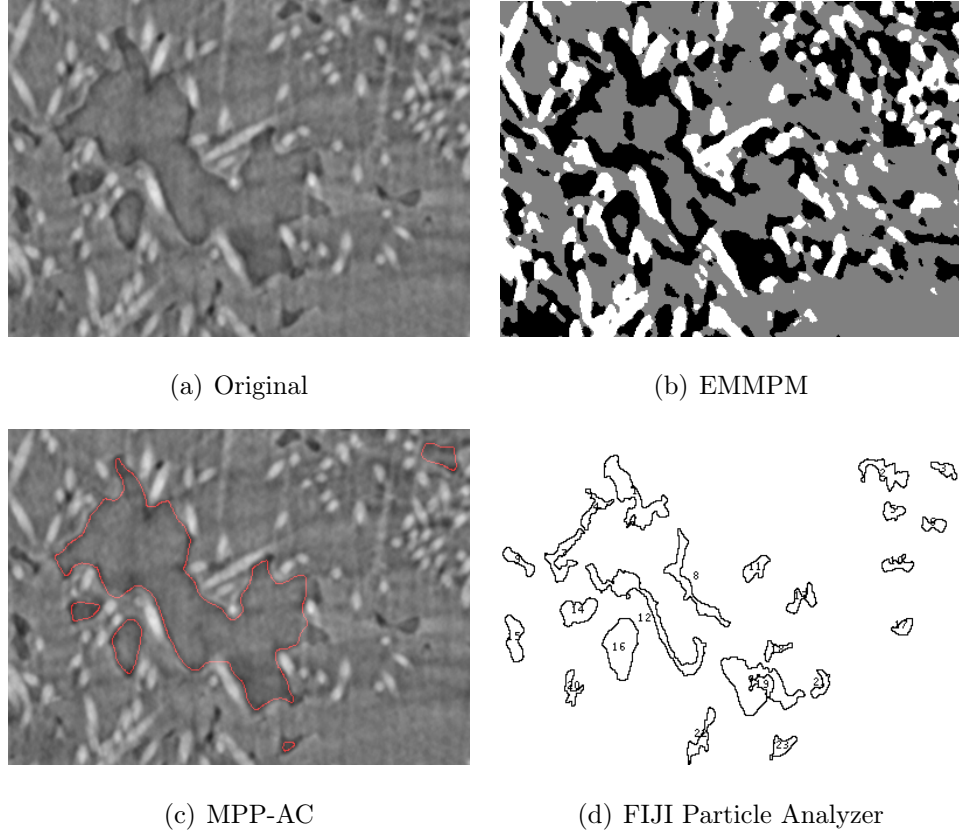


Fig. 3.4. Void Detection on Fiber Reinforced Polymer

Fig. 3.4 represents the fiber reinforced polymer described in [14]. Figure 3.4(c) shows that our algorithm can effectively characterize the largest voids despite their irregular internal pixel intensity. The birthmap used for this data was a dilated image

of the 10% of darkest pixels. We exploited E_{dark} to ensure that contours with dark edges have low energy. Figure 3.4(b) shows the EM/MPM results using 3 classes. This algorithm can detect the contour of the void but the interior region is labeled as background. Figure 3.4(d) shows the results of using FIJI's [37] particle analyzer tool. Our method is particularly better at detecting the large void. This occurs due to the balloon force acting on contours initialized in the interior region, and the energy proposed $U(\mathbf{w})$ in eq. (3.6) used to calculate the likelihood of the object. The caveat of the balloon force is that despite its contributions to our results, we cannot guarantee to achieve the global minimum of $U(\mathbf{w})$ since the MC employed is not irreducible anymore.

3.6.2 Human Cells:

Fig. 3.5(a) shows human stem cells obtained from Kaggle data science bowl competition [38]. This image comes from a set of biomedical images provided to improve segmentation algorithms and contribute to cell characterization.

We chose the birthmap to be the thresholded image with threshold=20. In this image, we set the balloon force $\kappa_1 = 0$, and $\kappa_3 = 0$ since E_{dark} is not necessary for this image. We filtered the image with a Gaussian filter with $\sigma = 2$, and we set the parameter $\kappa_2 = 0.4$. Fig. 3.5(b) denotes the marks for each object and Figs. 3.5(d) and 3.5(f) show the evolution for different parameters of β . Larger values of β tend to preserve an ellipsoidal shape while smaller values allow more deformation. For example, the cell in the center left resembles more the ground truth with $\beta = 10$ than with $\beta = 100$. Similarly, the cell in the top left preserves the curvature better with $\beta = 10$. For all these experiments, we kept the parameter $\alpha = 1$. We varied this parameter but its effect was negligible due to the pre-smoothing cause by the Gaussian filter. The viscosity parameter γ significantly affects the evolution of the contours. A large viscosity value prevented the contour from expanding correctly.

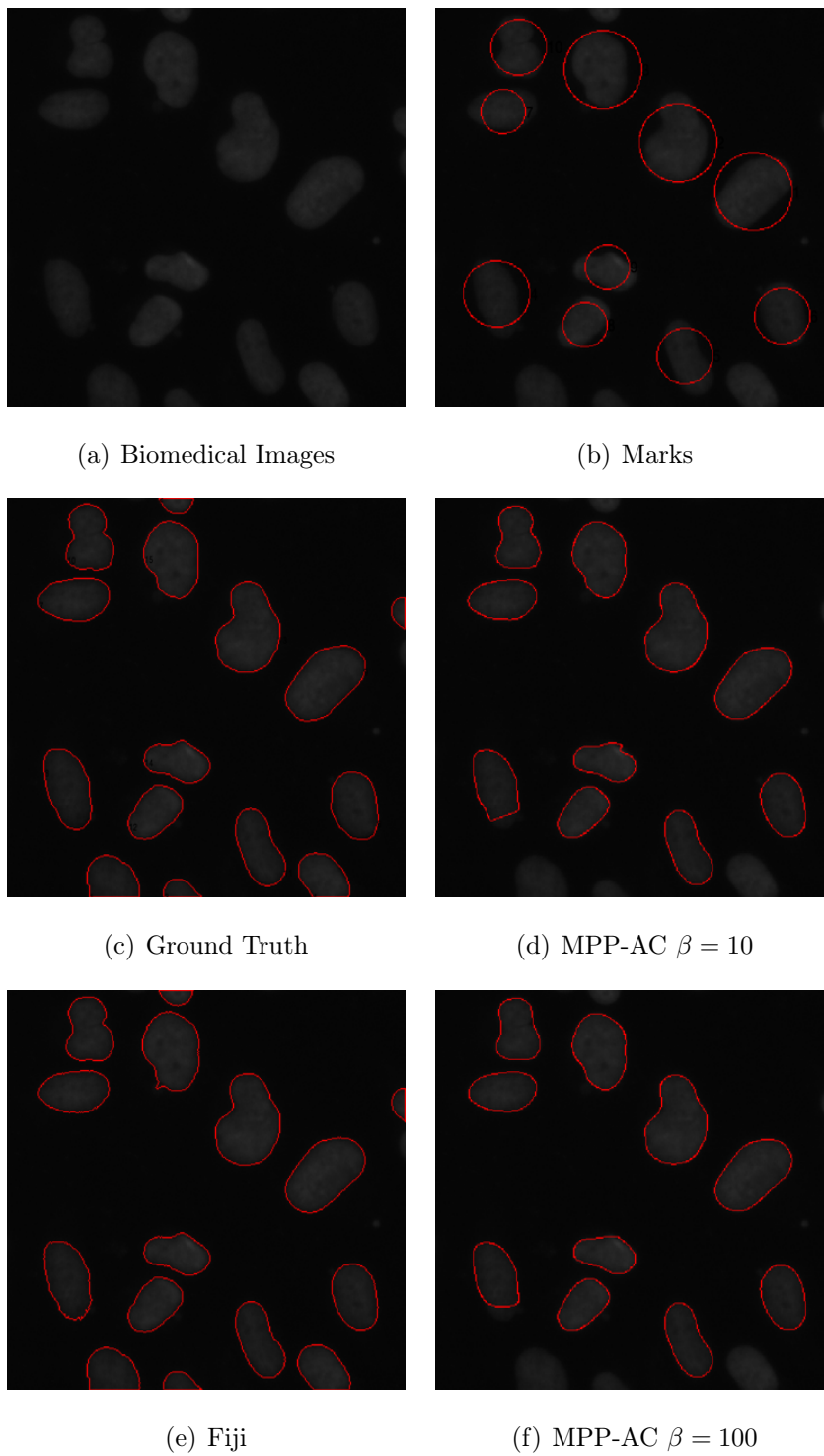


Fig. 3.5. Human Cells

Finally, the objects in the borders were not detected due to the instabilities caused in the parametrization of the contours.

3.7 Chapter Summary

In this section we explored a method to add the parametric active contours as a marked object. We covered the theory to combine both frameworks and we presented results on 2D data. This combination captures shapes with irregular boundaries and allows to modify the boundary properties such as smoothness and curvature for objects in a 2D space. We tested the balloon force to detect voids in material images with the caveat of not guaranteeing to achieve the optimal solution. In addition, the dimension jump to 3D demands for more complex parametrization. One example is to use a mesh of polygons and use the marching cubes algorithm to approximate this mesh. However, this volumetric parametrization involves significant numerical instabilities. Another approach is to look into non-parametric level sets as they provide flexibility to when increasing the number of dimensions. This approach is covered in Chapter 4.

4. SEGMENTATION AND DETECTION OF IRREGULARLY-SHAPED REGIONS USING INTEGRATED MARKED POINT PROCESS AND LEVEL SETS

4.1 Overview

In Chapter 3 we presented the idea of incorporating AC terms to the MPP energy. However, while the AC method can detect irregular shapes, it relies on evolving parametric contours with complex numerical approximations and can present difficulty adapting to topological changes. In this chapter, we extend this idea to incorporate the level sets (LS) framework into the MPP. The LS approach uses a zeroth-level set to model non-parametric contours while considering contour properties such as curvature and stiffness. This framework has been used as a common extension to the parametric active contours and accounts for topological changes, contour merging/splitting, and reduced sensitivity to contour initialization. However, the LS method alone presents drawbacks in multi-object systems: it merges nearby objects, it depends on a correct initialization, and it tends to capture multiple false positives. Fig. 4.1(c) shows an example of level sets result from using the method proposed by Yan et al. in [39].

In this chapter, we propose to model an image as a realization of an MPP, where each marked point represents a disk deformed with the level sets procedure. The level sets evolve constrained by a geometric prior that represents the object's MPP-model mark. Conversely, the Gibbs energy for the MPP is based on a LS-inspired likelihood as well as a prior energy based on object interaction. Finally, we use birth and death dynamics embedded in simulated annealing to explore the solution space at high temperatures and to converge to strong local maxima as the temperature decreases. We also propose an alternate sampling procedure to use the level sets results as object

proposals for the MPP model. We present results using our model to extract porosity in reinforced materials and evaluate our approach on a public dataset of human red blood cells images.

We discuss the LS model in Section 4.2. We develop the proposed MPP model in Section 4.3. We present the optimization approach in Section 4.4 and we present results in Section 4.5.

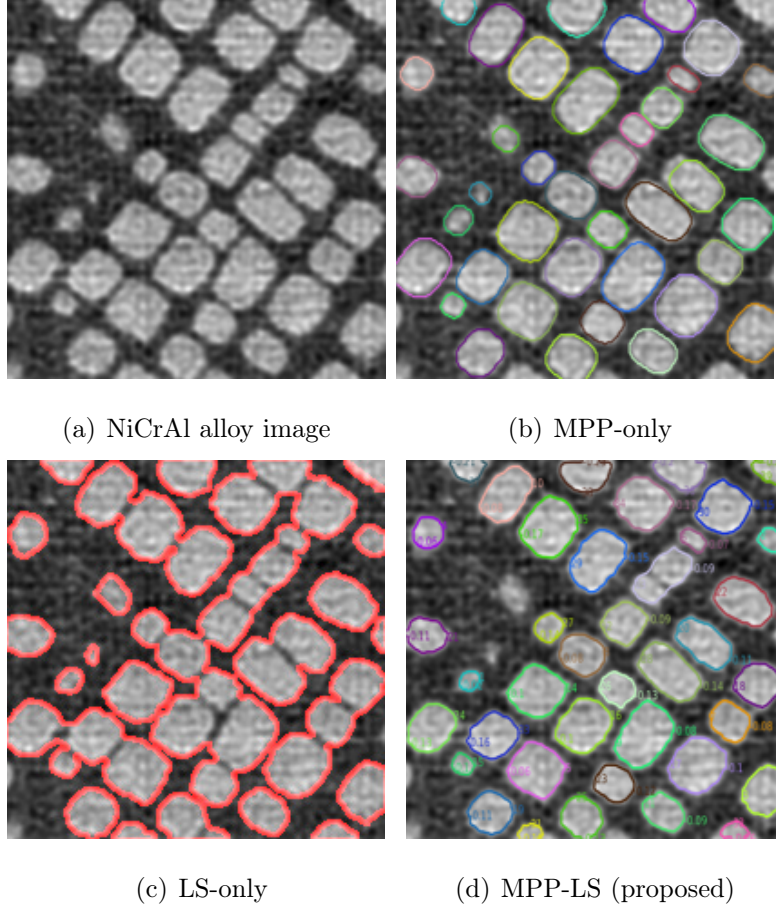


Fig. 4.1. Sample results using different approaches.

4.2 Level Sets Segmentation

The LS paradigm models contours by the zeroth level set of an embedding function ϕ defined on the image lattice $K \subset \mathbb{R}^n$. A contour is defined as $C = \{k | \phi(k) = 0, k \in K\}$, where $\phi(k) > 0$ inside the contour and $\phi(k) < 0$ outside. Many level sets-based methods have been proposed in the last two decades, including the edge-based Geodesic Active Contours(GAC) [40] and the region-based Active Contours without Edges (ACWB) [41]. Popular state-of-the-art methods include Local Image Fitting(LIF) [42], Morphosnakes [43], MCMC contours [44], or multi-object contours [45] [46]. Each method is well designed to tackle specific problems, such as reducing initialization sensitivity, stabilizing numerical optimization, guaranteeing convergence to a strong minimum, or discriminating overlapping objects (given a known number of objects). In this chapter, we used Zhang’s Hybrid Level Sets(HLS) [39] because of its combination of both region and edge terms. Low contrast objects such as material porosity in microscopy images are often described by their edges and a slight intensity difference between the interior and exterior regions of their bodies. The proposed HLS energy depends on a combination of a region term E_{region} and an edge term E_{edge} that can characterize the cross sections of the voids shown in Fig. 4.3(a). In addition, we include a shape prior E_{shape} that regulates the geometry of the level set. The modified level set energy term has the form:

$$E(\phi) = \alpha_r E_{region}(\phi) + \beta_e E_{edge}(\phi) + \gamma_s E_{shape}(\phi) \quad (4.1)$$

If we define I_k to be the image intensity at location k , we let $E_{region}(\phi) = \int_{k \in K} (I_k - \mu) H(\phi_k) dk$. This energy attracts the contour to regions with intensities I_k brighter or darker than the constant $\mu \geq 0$ (depending on the sign of parameter α_r). The function $H(\phi_k)$ is the Heaviside function and is nonzero only in the interior part of the level set, i.e. $\phi_k > 0$ [39]. Fig. 4.3(b) depicts objects of interest in the image shown in 4.3(a), and Fig. 4.3(d) shows that the region energy is lowest near the objects (depicted by bright pixel intensities).

The second term of equation (4.1), $E_{edge}(\phi) = \int_{k \in K} g_\sigma(k) |\nabla H(\phi)| dk$, attracts the contour to edges. The function $g_\sigma(k) = \frac{1}{1 + \alpha_{edge} |f_\sigma * \nabla I_k|^2}$ is minimum at the edges smoothed by a Gaussian filter f_σ with parameter σ . The term α_{edge} works as a threshold parameter to keep only strong edges. Through all the experiments in this chapter, we set the parameter $\sigma = 2$ and vary $\alpha_{edge} \in \{0.1, 1, 10\}$ depending on the quality of the edges in the image. Fig. 4.3(c) denotes the regions with high (dark pixels) and low (bright pixels) energy for the objects shown in Fig. 4.3(b).

The last term $E_{shape}(\phi) = \int_{k \in K} (H(\phi) - H(\phi_m))^2 dk$ was inspired from the LS with shape prior model [47] and it helps to preserve a specific shape in the level set. In this chapter, ϕ_m denotes a disk or ellipse and its angles and dimensions are determined by the mark m of the object in the MPP model. In most of our experiments, we set the parameter $\alpha_{region} \propto \frac{1}{\max(K)}$ to normalize the effect of the region term. We set $\alpha_{shape} \in \{1, 2\}$ to regularize the effect of the two other terms. We used the values of β_{edge} to increase the stiffness and penalize contour splitting. Fig. 4.2 denotes the converge for different values of the geodesic weight β_{edge} . A larger value of β_{edge} penalizes the creation of edges and keeps a smooth shape.

4.2.1 Level Set Optimization

Level sets are commonly evolved according to some velocity ϕ_t and the solution is often approximated with using an Euler Lagrange approximation to reach a stable solution to the partial differential equation $\frac{dE}{d\phi} = \frac{d\phi}{dt}$. We followed the numerical scheme dictated in [39] to evolve the contour in equation (4.1) to its local minimum. This procedure consists in evolving the level set in time according to equation:

$$\phi_t = \alpha_{region}(K - \mu) + \beta_{edge} \operatorname{div}(g_\alpha(K) \nabla \phi) + \gamma_{shape}(H(\phi) - H(\phi_m)) \quad (4.2)$$

The curve evolution involves: re-initializing the contour, updating the contour at iteration k as $\phi_{k+1} = \phi_k + \Delta_t(\alpha_{region}(K - \mu) + \gamma_{shape}(H(\phi_k) - H(\phi_m)))$, and approximating the divergence operator using the additive operator splitting(AOS) approximation. The detailed algorithm for the level set optimization is described in Appendix A. For

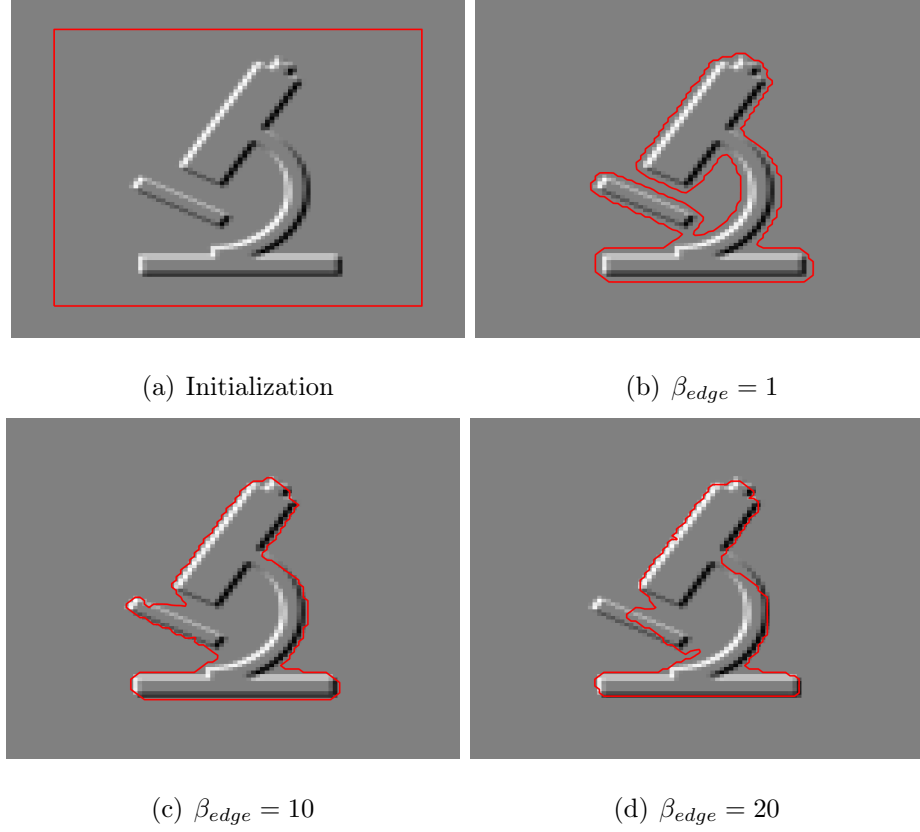


Fig. 4.2. Converges for different values of β_{edge}

all the experiments in this chapter, we chose the step size $\Delta_t = 2$. Fig. 4.4 shows two different initializations and convergences of the hybrid level sets active contours. A common drawback of this method is its sensitivity to initialization and its difficulty to capture nearby elements such as the bottom right corner of Fig. 4.4(d).

4.3 Marked Point Process with Level Sets

The MPP single object with active contours energy was shown in section 3.4. This consisted in proposing a disk ω with marks $m = \{k, r\}$ and deforming it with a parametric active contour energy $E_{AC}(\omega)$, shown in Fig. 4.5. The functional describing the contour was minimized using Euler-Lagrange to achieve a local minimum $E_{AC}(\tilde{\omega})$ as

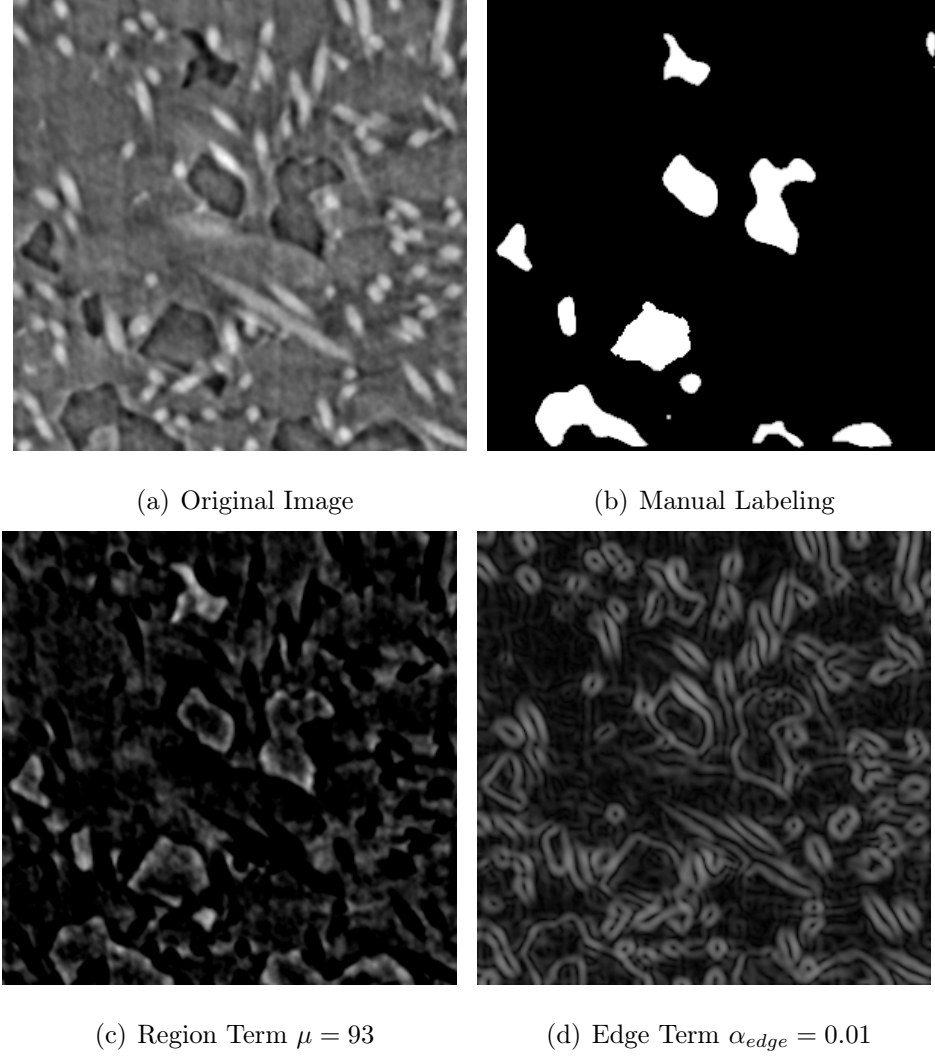


Fig. 4.3. Components of hybrid energy.

shown in Fig. 4.5(b). We defined an object configuration as $\mathbf{w} = \{\omega_1, \omega_2, \dots, \omega_N\} \in \Omega$, where N is a random variable and Ω contains the possible disks configurations, and a space Ω_{Wo} containing the respective deformed disks.

To address the common parametric active contour problems and extend the model to level sets, we investigate two approaches: single object level sets and multiple object level sets.

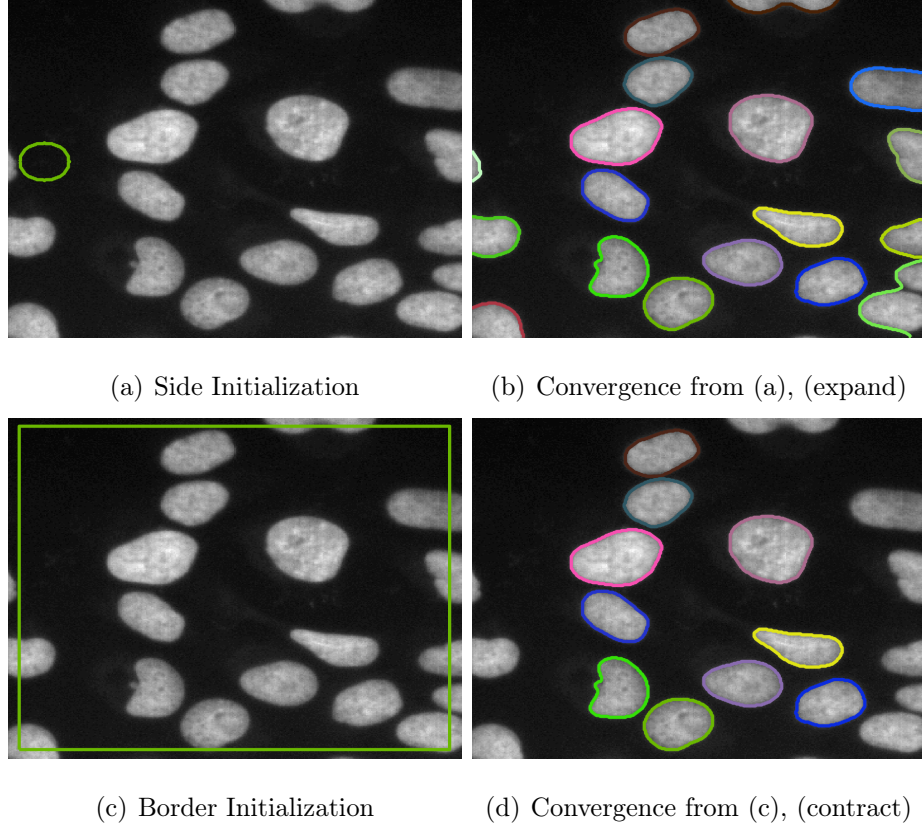


Fig. 4.4. Convergence of evolving level sets at different initializations.

4.3.1 Single Object Level Set

We initialize a level set $\phi(k)$ to an ellipse object $\omega = (k, m)$. We evolve this level set to its local minimum $\tilde{\phi}(k)$ following the procedure in Appendix A, using the shape prior to prevent splitting. Then we parametrize the resulting zeroth level set with a closed contour $\tilde{\omega}$. This would be equivalent to the procedure done in Chapter 3 but using level sets instead of parametric active contours. Fig. 4.5 shows the idea of this approach.

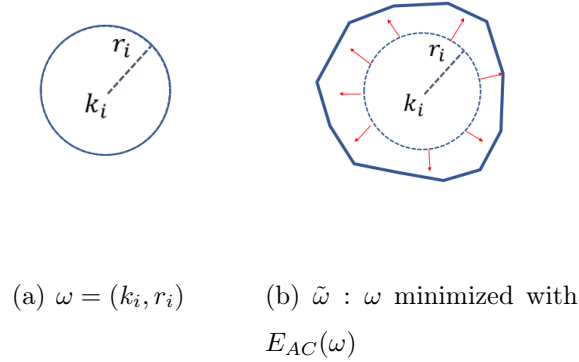


Fig. 4.5. Original marks disk and deformed disk.

4.3.2 Multiple Objects Level Set

This approach consists of initializing random level sets without shape prior ($\gamma_s = 0$) and evolving the LS functional until convergence to $\tilde{\phi}(K)$. This approach allows the zeroth level set to merge and split to adjust to the image. Once the level set $\phi(K)$ is evolved to its local minimum $\tilde{\phi}(K)$, we parametrize all the closed contours created by the zeroth level set by using a best fit approach [48] to assign a location k and mark m to each contour. If a parametrized level set has a mark m outside the mark space M , we automatically discard the object. Fig. 4.6 shows an example of evolving a level set and obtaining multiple marked objects. The caveat of this approach is that we cannot guarantee a global minimum since the markov chain is not irreducible since we cannot visit every state in a finite number of steps.

4.3.3 MPP Energy

We model a set of marked points $W \in \Omega$ with a Gibbs density of the form:

$$p(W) = \frac{1}{z_\omega} \exp(-U(W)), \quad (4.3)$$

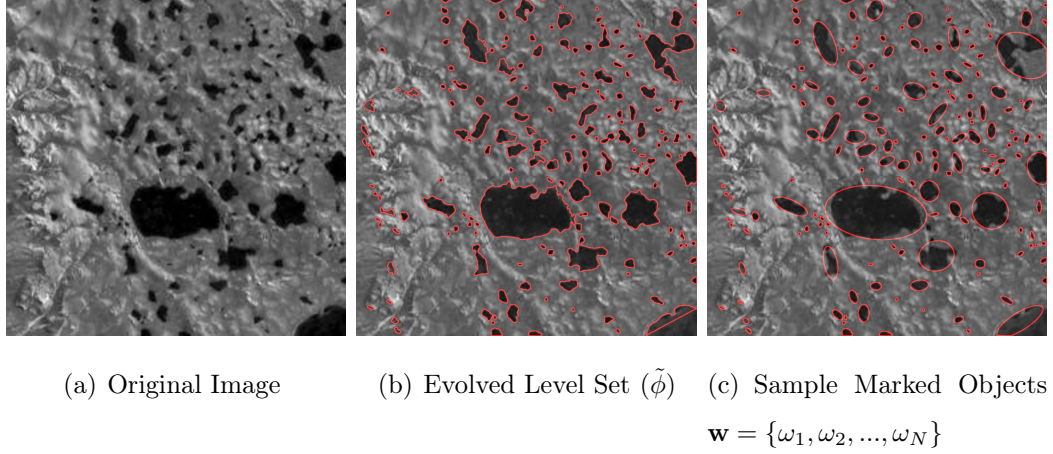


Fig. 4.6. Sample of multiple objects level set in satellite image of lakes. Image obtained from Kaggle’s Satellite Image Classification Challenge [38].

where $U(W)$ is the Gibbs energy and $z_w = \int_{W \in \Omega} \exp(-U(W))$ is the partition function. We define the Gibbs energy of the form

$$U(W) = \sum_{\omega_i \in W} U_d(\omega_i) + \sum_{\substack{\omega_i, \omega_j \in W \\ \omega_i \sim \omega_j}} U_p(\omega_i, \omega_j), \quad (4.4)$$

Where $U_d(\omega_i)$ represents the potential energy for a single object ω_i , and $U_p(\omega_i, \omega_j)$ represents an interaction potential between neighboring elements ω_i and ω_j . In our model, the neighbor relation \sim is defined by object overlapping. The details for $U_d(\omega_i)$ and $U_p(\omega_i, \omega_j)$ will be defined in sections 4.3.5 and 4.3.4 .

4.3.4 Single Object Potential U_o

Since the contour \tilde{w}_i cannot be simply represented by a mark, we parametrize each object $\tilde{\omega}_i$ in the locally minimized level set into $\tilde{\omega}_i(t)$ with n_t points separated by 1 pixel (measured in Euclidean distance) in the boundary of $\tilde{\omega}_i$. We let $D_{\tilde{\omega}_i}$ denote the region occupied by object $\tilde{\omega}_i(t)$ and calculate the single object energy:

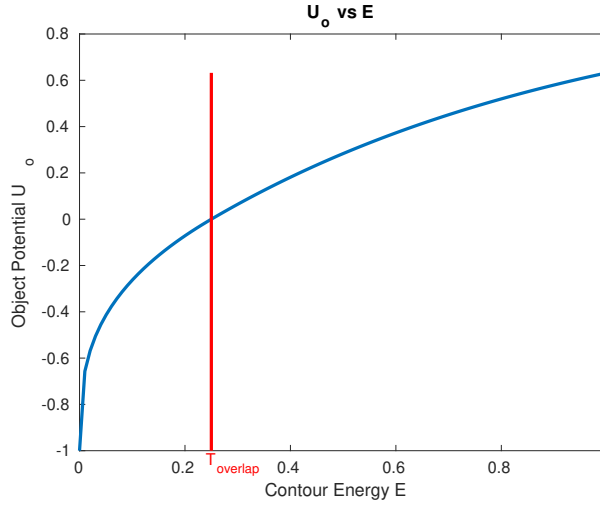
$$E(\tilde{\omega}_i(t)) = \alpha_r E_{region}(\tilde{\omega}_i(t)) + \beta_e E_{edge}(\tilde{\omega}_i(t)) \quad (4.5)$$

where $E_{region}(\tilde{\omega}_i(t)) = \frac{1}{|D_{\tilde{\omega}_i}|} \sum_{s \in D_{\tilde{\omega}}} (I(t) - \mu) ds$ is a normalized parametric representation of $E_{region}(\phi)$ from eq. (4.1). This normalized sum evaluates the percentage of dark pixels inside the parametrized object. The second term $E_{edge}(\tilde{\omega}_i(t)) = \frac{1}{nt} \sum_{t \in \tilde{\omega}_i} g_\sigma(t) dt$ is a normalized parametric representation of $E_{edge}(\phi)$ from eq. (4.1) and it evaluates the percentage of border points located at a strong edge, denoted by the function g_σ (defined in section 4.2). These terms were inspired in the level sets energy given in eq. (4.1), but adapted to a single parametric object.

For the object single potential $U_o(\omega)$, we define a quality term that favors desirable configurations:

$$U_o(\omega) = \begin{cases} 1 - \exp(-\frac{E(\tilde{\omega}) - E_o}{3E_o}), & \text{if } E(\tilde{\omega}) \geq E_o \\ (\frac{E(\tilde{\omega})}{E_o})^{\frac{1}{3}} - 1, & \text{otherwise,} \end{cases} \quad (4.6)$$

where E_o is a hyper-parameter that is image-dependent. This expression results in a value that is between -1 and 1 and ensures that objects with good fit get values close to -1 (favoring) and objects with bad fit get values close to 1 (penalizing). Figure 4.7 shows the energy curve with the parameter E_o set to 0.25.



(a)

Fig. 4.7. Quality energy term.

4.3.5 Prior Potential U_p

For prior potential, we use an overlapping penalizer. This prior potential allows overlap between objects up to a certain threshold $T_{overlap}$. If we let D_ω denote the region occupied by object ω in the image, and $D_{\tilde{\omega}}$ the region occupied by the deformed object $\tilde{\omega}$, the overlap potential is given by:

$$V_p(\tilde{\omega}_i, \tilde{\omega}_j) = \begin{cases} A(\tilde{\omega}_i, \tilde{\omega}_j), & \text{if } A(\tilde{\omega}_i, \tilde{\omega}_j) < T_{overlap} \\ \infty, & \text{otherwise} \end{cases} \quad (4.7)$$

where $T_{overlap} \in [0, 1]$ is a constant indicating the allowed amount of overlap between two distinct objects, $\tilde{\omega}$ is the deformed object from ω , and $A(\tilde{\omega}_i, \tilde{\omega}_j)$ is the overlapping ratio between two distinct regions formed by overlapping objects and is given by:

$$A(\tilde{\omega}_i, \tilde{\omega}_j) = \frac{|D_{\tilde{\omega}_i} \cap D_{\tilde{\omega}_j}|}{\min(|D_{\tilde{\omega}_i}|, |D_{\tilde{\omega}_j}|)} \quad (4.8)$$

This prior potential is often used for satellite images to allow object occlusion for objects at different heights; for all our experiments in section 4.5 we set $T_{overlap} = 0.05$.

4.4 Marked Point Process Optimization

The energy $U(W)$ presented in eq. (4.4) is non-convex. We use stochastic simulation for optimization. Often, MPP models are optimized using a Markov chain Monte Carlo (MCMC) sampling embedded in simulated annealing. This approach is based on starting at a high temperature to explore the configuration space and reducing the temperature to allow the solution to be concentrated around the configuration that is the MAP estimate, ie $\hat{W} = \underset{W}{\operatorname{argmax}}(p(W))$. Popular samplers include the reversible jump Markov chain Monte Carlo (RJMCMC), proposed by Green [18] or Multiple Births and Deaths (MBD) proposed by Descombes [19]. In this chapter, we use the MBD sampler because it was shown to achieve faster convergence than the traditional RJMCMC [30,32]. The choice of the initial temperature T_o and the annealing scheme are essential to ensure the convergence to a global optimum. In our experiments

we determined $\alpha_T = 0.90$ achieved a faster convergence without compromising the quality of detection.

4.4.1 Multiple Births and Deaths with Strictly Unary Level Set

The MBD sampler we use consists of finding a birthmap, normally pre-computed from a segmentation, then iterating over every pixel and giving birth to objects ω with probability p_{birth} with parameters drawn from the distribution of M . The detailed algorithm for this procedure is listed in Algorithm 2. The algorithm reaches convergence if and only if all the objects born during the birth step are killed during the death step. For this procedure, we do not allow merging/splitting of level sets. The algorithm reaches convergence if and only if all the objects born during the birth step are killed during the death step.

4.4.2 Simulating Multiple Births and Deaths with Splitting Level Sets

Stochastic simulation algorithms with simple proposal distributions tend to accept only a fraction of the proposed samples. Because of large image dimensions, traditional MPP sampling techniques can result in a large number of wasted computations, especially in systems with sparse objects and irregular geometries. On the other hand, level sets often evolve with faster deterministic methods but they rely on a correct initialization in order to converge to a meaningful solution. Hence, in order to preserve the probabilistic framework of the MPP and the ability to capture objects from the level sets, we parametrize the objects found with level sets and use them as proposals for the MPP model. In order to do this, we use as reference Algorithm 6 in [30]. This method consists of simulating multiple births in a system by drawing realizations of a Poisson Process. However, instead of drawing realizations from a distribution, we initialize level sets at random locations and evolve the level sets to a local minimum. Then we use all the possible objects as proposals and use the birth

Algorithm 2 Multiple Birth and Death with Stricly Unary Level Set

```

1: procedure MPP ENERGY MINIMIZATION
2:   Initialization:
3:   Create birthmap  $b_o$ 
4:   Initialize  $b_{rate} = b_o$ ,  $T = T_o$ ,  $\sigma = \sigma_o$ .
5:   Birth Step:
6:   Visit pixels in raster order
7:    $\omega' \leftarrow$  draw a sample from space  $W$ 
8:   Add  $\omega'$  to configuration  $\mathbf{w}$  with probability  $\sigma b_{rate}$ 
9:   Evolve  $\omega'$  using the method described in Appendix A to  $\tilde{\omega}'$ 
10:  Death Step:
11:  Sort all elements of  $\mathbf{w}$  by decreasing energy.
12:  For every object  $\omega_i$  in  $\mathbf{w}$  calculate:
13:   $d_{rate}(\omega_i) = \frac{\sigma^{(k)} \exp \frac{U(\mathbf{w}|Y) - U(\mathbf{w} - \omega_i|Y)}{T^k}}{1 + \sigma^{(k)} \exp \frac{U(\mathbf{w}|Y) - U(\mathbf{w} - \omega_i|Y)}{T^k}};$ 
14:  Delete  $\omega_i$  with probability  $d_{rate}(\omega_i)$ 
15:  Convergence Test:
16:  if all the elements born during the birth step are killed during the death step
      then
17:    terminate process
18:  else
19:    Update parameters:  $T^{k+1} \leftarrow T^k \times \alpha$ ,  $\sigma^{k+1} \leftarrow \sigma^k \times \alpha$   $\alpha \in (0, 1)$ 
20:    goto Birth Step
21:  end if
22: end procedure

```

and death dynamics to update our configuration. The details of this approach are given in Algorithm 3.

Algorithm 3 Using Level Sets to simulate Multiple Births

```

1: procedure MPP ENERGY MINIMIZATION IN IMAGE  $I$ 
2:   Initialization:
3:   Initialize  $b_{rate} = b_o$ ,  $T = T_o$ ,  $\sigma = \sigma_o$ ,  $W = \{\}$ 
4:   Birth Step:
5:   Initialize a level set  $\phi(k)$  at a random location
6:   Evolve  $\phi(k)$  using the method described in Appendix A to  $\tilde{\phi}(k)$ 
7:   Parametrize every closed contour  $\tilde{\omega}'$  in  $\tilde{\phi}(k)$ 
8:   Calculate a best fitting marked object  $\omega'$  for for each contour  $\tilde{\omega}'$ . Call  $W' = \{\omega'_1, \omega'_2, \dots, \omega'_n\}$  the new configuration.
9:   Add the configuration to the current configuration  $W \leftarrow W \cup W'$ 
10:  For every object  $\omega$  in  $W$  calculate:
11:   $a_\omega = \exp \left[ \frac{U(W) - U(W \setminus \omega)}{T^k} \right]$ ; draw p form a uniform distribution over  $[0, 1]$ 
12:
13:  if  $p < \frac{a_\omega \delta}{1 + a_\omega \delta}$  then remove  $\omega$  :  $W \leftarrow W \setminus \omega$ 
14:
15:  end if
16:
17:  if  $n < \text{Max Iterations}$  then
18:    Update parameters:  $T^{k+1} \leftarrow T^k \times \alpha$  ,  $\sigma^{k+1} \leftarrow \sigma^k \times \alpha$ ,  $n \leftarrow n + 1$ 
19:
20:  end if
21:  goto Birth Step
22: end if
23: end procedure

```

4.5 Experimental Results

We tested our method on images containing objects of irregular shapes and little or no contrast difference. First, we validated the MPP-LS with human red blood cell data publicly available from the Broad Bioimage Benchmark Collection [1]. We compare our results with the MPP-AC method proposed by Aguilar et al. in [49] and the hybrid-LS approach presented by Yan et al. in [39]. We use the three metrics presented by Broad Bioimage Benchmark Collection to evaluate our method: “background”, “outlines”, and “counts”. The background metric evaluates the quality of pixel-wise segmentation of background/foreground. The outlines metric evaluates the quality of contours detected: a contour pixel is marked as true positive if it lies within two pixels of a ground truth contour pixel. The counts metric evaluates the error between the ratio of detected objects and ground truth objects. In addition, we tested our approach detecting large porosities in fiber reinforced composite images and we compare the results with the hybrid level sets presented by Yan et al. [39]. In this dataset, we only evaluate the segmentation score since the labeled-data is a hand-drawn approximation and can cause ambiguities in the outlines score.

4.5.1 Human Red Blood Cells

This dataset was obtained from image set BBBC009v1 in the Broad Bioimage Benchmark Collection and it consists of five differential intensity contrast (DIC) images of human red blood cells. The dimension of each image is 600×800 pixels and each image contains close to 40 red blood cells. Fig. 4.8(a) denotes a sample DIC image where the cells are characterized by a low contrast body surrounded with an ellipsoidal contour. For these images, we use the birth-death sampling described in Section 4.4 using level sets consisting of a unary contour with an ellipse geometric prior. We define the mark space for the ellipse to be $M = [a_{Min}, a_{Max}] \times [b_{Min}, b_{Max}] \times [\theta_{Min}, \theta_{Max}] = [10, 50] \times [10, 50] \times [0, \pi]$, where a, b represent the ellipse minor and major axis respectively, and θ represents the orientation

with respect to the positive x-axis. In addition, we discard the region term using $\alpha_r = 0$ since there is no contrast between the interior and exterior regions of the cells, we let the edge parameters be $\alpha_{edge} = 1$ and $\beta_e = 7$, and we let the parameter $\gamma_s = 1$. Finally, we let $E_o = 0.5$ so the potential energy would favor objects with 50% or more of its borders located at a strong edge and would penalize the objects otherwise.

Fig. 4.8 denotes sample images with hand annotations, and results for the LS-only method [50], MPP-AC [49], and MPP-LS (proposed) and Table 4.1 presents the numeric results for each metric. The traditional level sets was initialized at the border pixels and contracted until convergence. The LS-only method cannot differentiate between adjacent objects and merges the objects as shown in the third column of Fig. 4.8. On the other hand, the MPP-AC method can differentiate objects near each other but it has the drawback of lower accuracy in edge detection. The parametric contours lead to inaccurate border representations and also present difficulties capturing objects with noisy bodies. Our method, depicted in the fifth column of Fig. 4.8 outperforms the MPP-AC and the traditional LS at both boundary detection, object separation, and segmentation. The improvement of the MPP-LS vs the MPP-AC comes from using non-parametric contours and the improvement from MPP-LS vs LS-only comes from multiple contour initializations and assigning a mark to each object.

The disadvantage of using non-parametric contours is the amount of operations to evolve the curve, since level sets require $O(rows * columns)$ operations while the parametric contour only requires $O(N_t)$, where N_t is the number of points used to model the contour. Also, in both methods there are false positives in empty regions surrounded by cells. This flaw is intrinsic to the MPP model since the deformed model is surrounded by edges and gives a high likelihood to a false positive detection. In addition, our literature review shows that one popular method to segment these images [51] obtained 93.4% segmentation accuracy for image 4.8(a). We obtained 94.35% accuracy for the image in 4.8(a) and 83.94% accuracy for the entire image

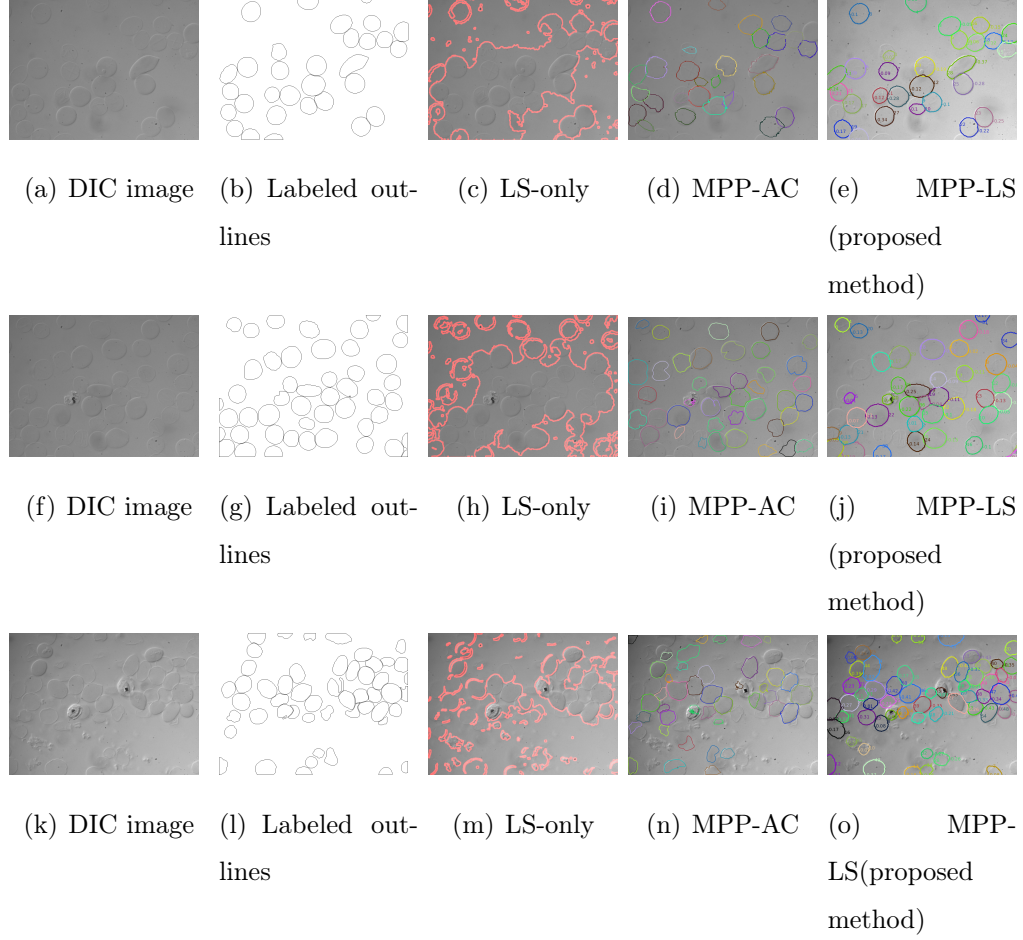


Fig. 4.8. Human red blood cells . Images obtained from Broad Bioimage Benchmark Collection [1].

dataset. Also, unlike the method in [51], our approach yields an individual object-wise segmentation.

4.5.2 NiCrAl Alloy

Fig. 4.9 shows a NiCrAl Alloy microscopy image with dimensions 208×208 . For this image, we chose the level sets parameters $\mu = 90$ based on the average intensity of a particle. We also set $\alpha_{regions} = -1$ since we wanted the contour to evolve into regions with intensities greater than μ . The edge term was set to $\beta_{edge} = 0.01$. In addition, we

Table 4.1.
Comparison f1 scores for human red blood cells

Method	Background	Outlines	Counts
MPP-AC	0.790	0.680	0.897
Hybrid-LS	0.432	0.784	-
MPP-LS (proposed)	0.843	0.820	0.927

used the ellipse model for our mark space $M = [a_{min}, a_{max}] \times [b_{min}, b_{max}] \times [\theta_{min}, \theta_{max}]$. We sampled uniformly for the parameters a and b and we followed [32] to define θ with a Gaussian mixture with means at $\theta = \frac{\pi}{4}$ and $\theta = \frac{3\pi}{4}$, with variance 0.1^2 .

This image was used for testing the MPP method in [32]. The MPP combined with level sets took 40 iterations to detect the particles in Fig. 4.9(b) while the MPP took 200 iterations using similar MPP parameters. The MPP alone works very well for capturing nearby elements, however, it does not provide description of the particles border, nor allows to simulate the boundary properties. On the other hand, the mpp distinguished nearby particles in the middle of the image while the MPP-LS merged both elements together. This is one of the problems with level sets when there is a weak edge between close elements.

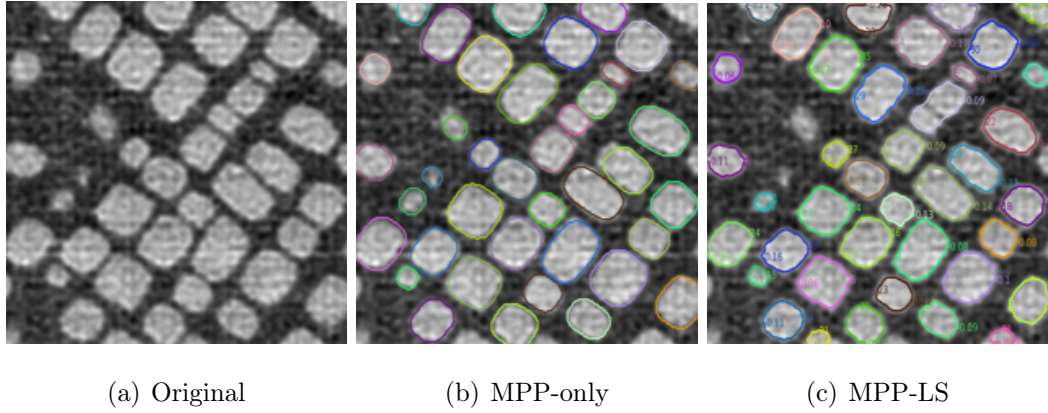
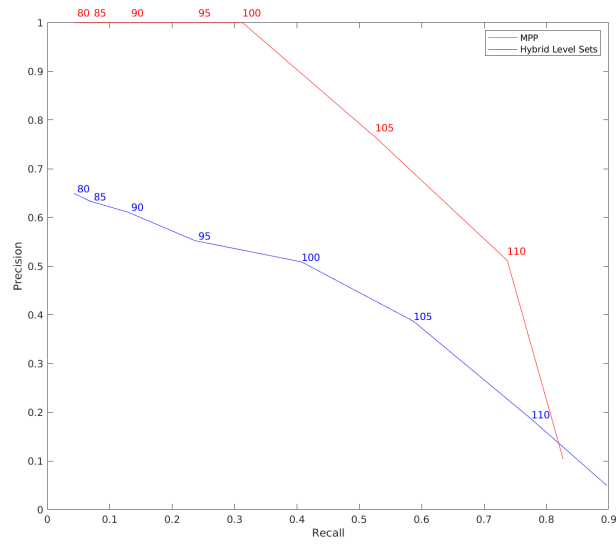


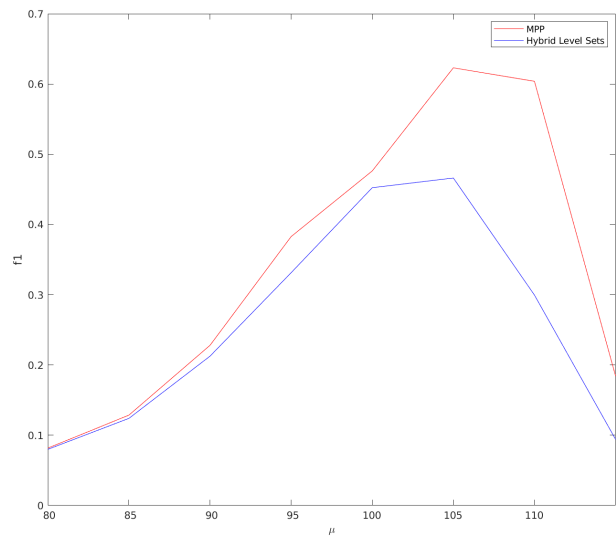
Fig. 4.9. MPP-LS results on NiCrAl alloy image

4.5.3 Fiber Reinforced Polymers

The images shown in the first column of Fig. 4.11 were provided by ACME lab at Purdue University. These images depict slices of a volume reconstructed from CT scans of a fiber reinforced polymer. The volume consists of 1300 slices of 2500×2500 pixels, and the sub-volume used for validation entails 10 images with sizes 950×950 pixels with labeled segmentation. The labeled images were generated by pre-



(a) ROC measurements



(b) f1 scores

Fig. 4.10. Fiber reinforced polymer segmentation evaluation

segmenting the volume with the methods described in [2, 14, 15] and with a post-processing step of hand correction.

The images are comprised of three elements: fibers(bright intensities), voids (dark irregularly shaped objects), and matrix(background). In this chapter, we focused on extracting the voids, which are represented by objects of irregular shapes and low intensity-contrast between the interior and exterior parts. Fig. 4.11 shows different sample images cropped with different dimensions for visualization purposes. The first row has dimensions 950×950 pixels, the second row has 600×650 pixels, and the third row has 300×300 pixels.

We approximated the hyper-parameters with pre-segmentation algorithms and experimentally. The region term $\mu = 95$ was chosen as the average intensity in the 3 darkest regions of a pre-segmentation algorithm [33] with 9 labeled regions. The edge weight was set experimentally to $\alpha_{Edge} = 0.1$ by checking that the regions of interest were captured in the edges. Finally, the weighting parameters $\alpha_{regions} = \frac{1}{15}$ and $\beta_{edge} = 45$ were set to regularize the effects of the region term and to prevent the contours from splitting since we set $\gamma_{shape} = 0$ for this method.

We use the approach presented in section 4.4 to propose objects in the MPP framework based on the results of multiple level sets. We set the geometric prior to zero ($\alpha_s = 0$) due to the irregular shapes and dimensions of voids (shown in Figs. 4.11(b), 4.11(f), 4.11(j)) and due to the ability of the LS-only to detect voids (shown in Figs. 4.11(c), 4.11(g), 4.11(k)). The main drawback of the LS-only method is that the LS depends on correct contour initializations. For instance, Fig. 4.11(c) and Fig. 4.11(g) show the results from initializing the level set at the image borders. Fig. 4.11(c) detected a large false positive that affects significantly the foreground segmentation score and 4.11(k) missed porosity located near the image borders. In addition, the LS-only method does not evaluate the likelihood of each detected object. For example, Fig. 4.11(k) detected multiple false positives generated due to noise and edges adjacent to fibers. We exploit the MPP-LS approach by evolving multiple level sets, obtaining a mark for each object, calculating the potential energy using equation

Table 4.2.
ROC values for fiber reinforced composites

Method	mean f1	ROC
Hybrid-LS	0.559	0.338
MPP-LS (proposed)	0.770	0.592

(4.5), and finding the likelihood for each object. For example, Fig. 4.11(l) preserves only the voids that are surrounded by strong edges and a dark foreground. In addition, 4.11(h) shows a void that was captured by two level sets and was missed in the LS-only method. It is worth mentioning we only compare against the hybrid-level sets method since the MPP-AC presents difficulties adapting to the void topology, and the MPP-AC also presents large computational times due to the large image dimensions.

To improve the comparison across slices, we varied the region intensity parameter μ (from Fig. 4.3(d)) since the image pixel intensity distribution varies depending on the slice number. This issue is generated during the CT reconstruction process and can be solved using histogram matching as pre-processing. The appropriate value for μ ranges between $\mu \in [95, 110]$ depending on the slice of interest. The results of applying our method and the hybrid level sets method at different parameter choices are depicted in Fig.4.10(a). The MPP-LS approach obtains higher precision than the HLS method for values below $\mu = 115$ due to the death step process. However, at higher values than $\mu = 115$, the MPP-LS assigns a high probability to all the proposed objects and therefore, its performance decreases. Fig. 4.10(b) shows the average f1 score across all 10 images. Our method obtains a higher f1 score at all the parameter ranges than the level-sets only approach. In addition Table 4.2 shows the mean f1 scores given the best parameter choice for each image. Our method obtains 0.770 compared to 0.559 using the Hybrid-LS method. It is worth noting the labeling is an approximation to a ground truth due to the labor intensive process of image annotation, and due to the intrinsic ambiguity of microscopy images.

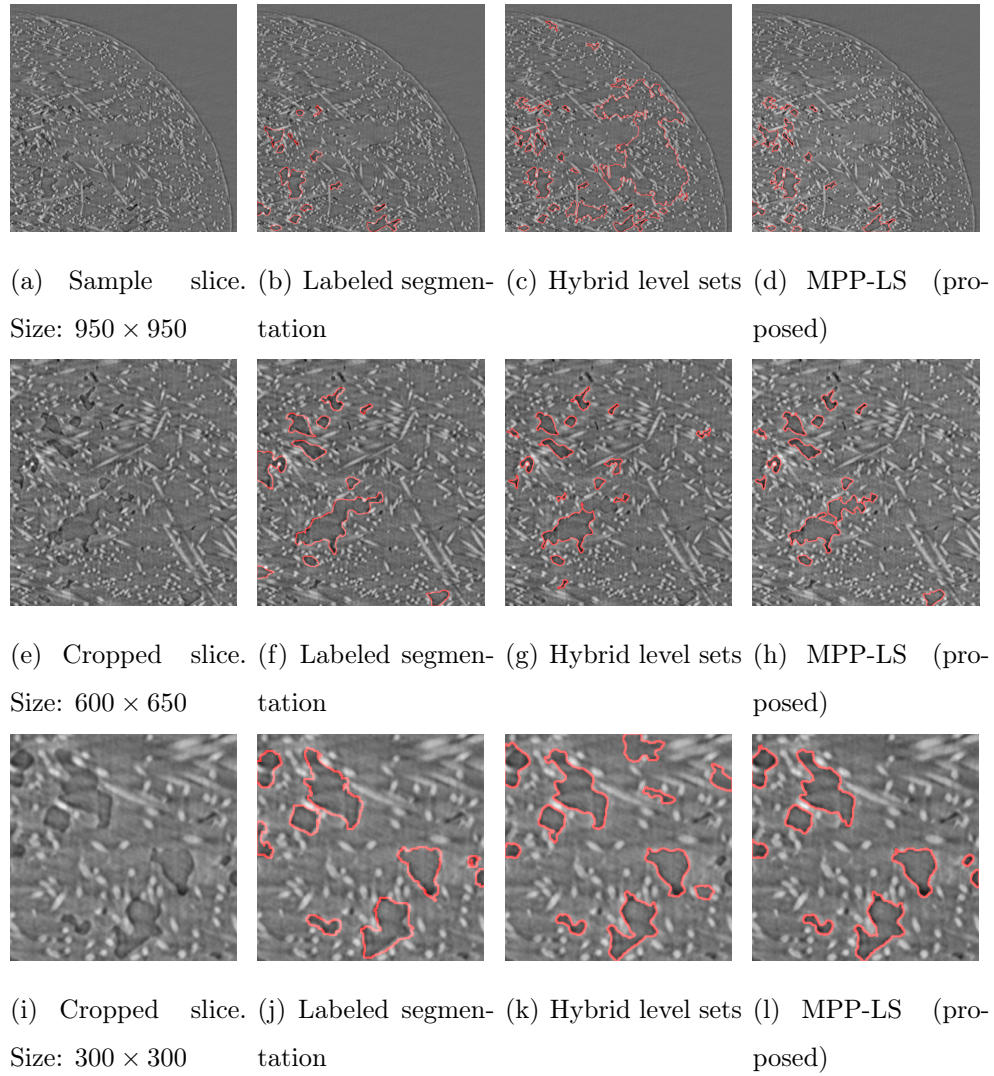


Fig. 4.11. Detecting voids in fiber reinforced polymers. Second column: hybrid level sets only. Third column: MPP-LS. Images provided by Dr. Michael Sangid's laboratory, Purdue University

The HLS method [39] were initialized to the external border and contracted to reach the convergence shown in the middle column of Fig.4.12. We exploited the MPP approach by obtaining a mark for each object, calculating the potential energy of the object using equation (4.5) to find the likelihood. In addition, we explored simulating multiple initializations of the level set to capture different objects, but

in our experiments, few initializations were necessary as the level set converged to similar results. The right column of Fig. 4.12 denotes our results after finding the likelihood for each object and killing the ones with higher energy. Our results show improvement over the HLS approach since our method keeps only the objects of interest. In addition, the bottom left corner of Fig. 4.12(i) shows an object with more accurate boundary characterization than in 4.12(h). However, our approach comes at the expense of more operations as we evolve multiple level sets.

4.6 Towards 3D Results in Fiber Reinforced Polymers

The level sets method provides the setup for 3D sampling. However, our approach requires several level sets initializations and every level set requires $O(n^d)$ operations, where n is the size of a single side and d is the number of dimensions. In addition, CT volumetric images present imaging noise, reconstruction noise, ring artifacts, and intensity variations for individual slices. In our experiments, this data inconsistency created instabilities when running direct 3D models for the contours. Hence, we followed the procedure listed in Fig. 4.13 where we detected 2D objects in each slice, use a 3D Gaussian smoothing filter with parameter $\sigma = 2$, and then used connected components to label each object. This procedure accounted for individual slices that missed contours but had contours detected in the neighboring slices. Fig. 4.14 denotes the comparison of a manual segmentation with the results obtained from the procedure shown in Fig. 4.13.

4.7 Chapter Summary

We explored the addition of level sets framework to the MPP model. We exploit the numerical tools of level sets to find objects and then parametrize the closed contours to treat them as marked objects in the MPP framework. The MPP framework includes the level set energy as well as an overlapping prior. The combination of these models allows modeling objects found in level sets with a probabilistic approach, and

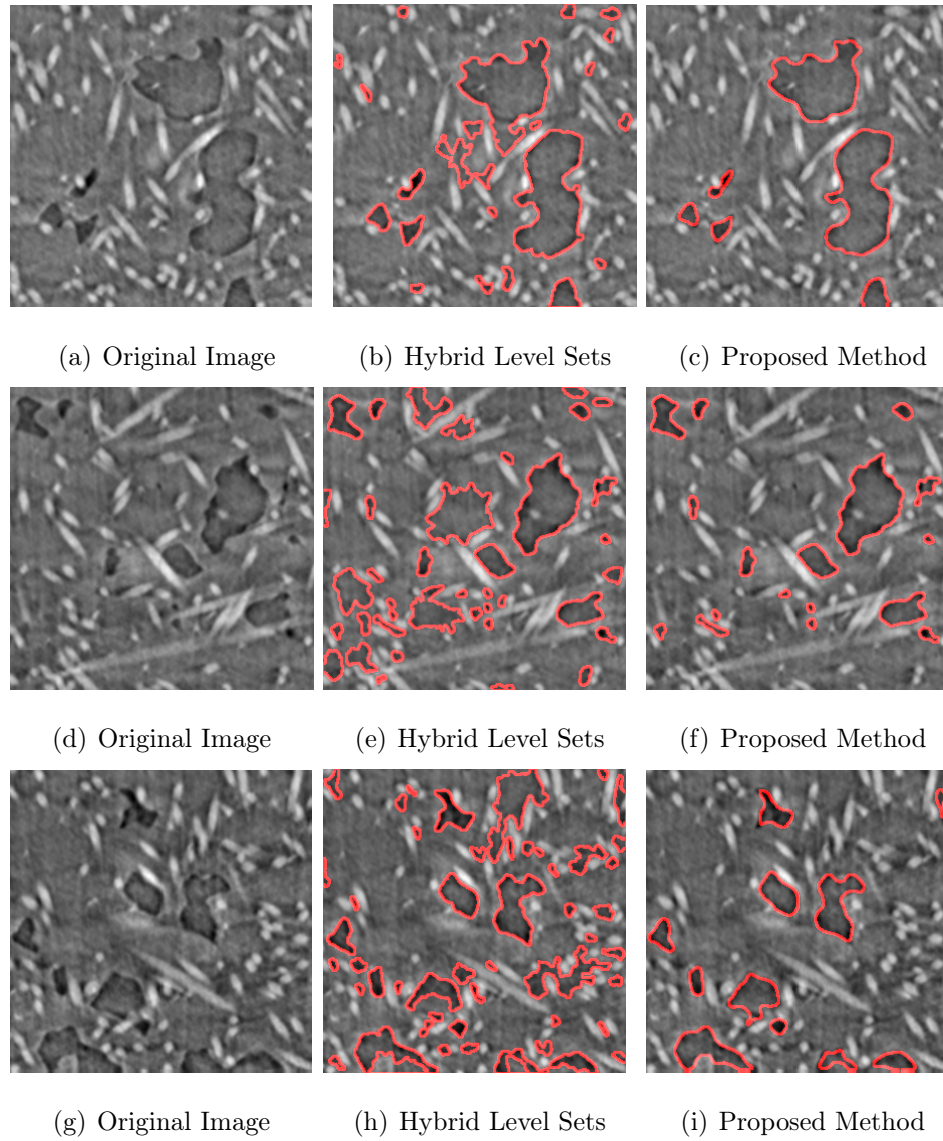


Fig. 4.12. MPP-LS results on fiber reinforced polymers

hence it helps to characterize randomly located irregularly shaped objects. We show results for two different images and compare the results with other methods, showing the advantage of the proposed approach.

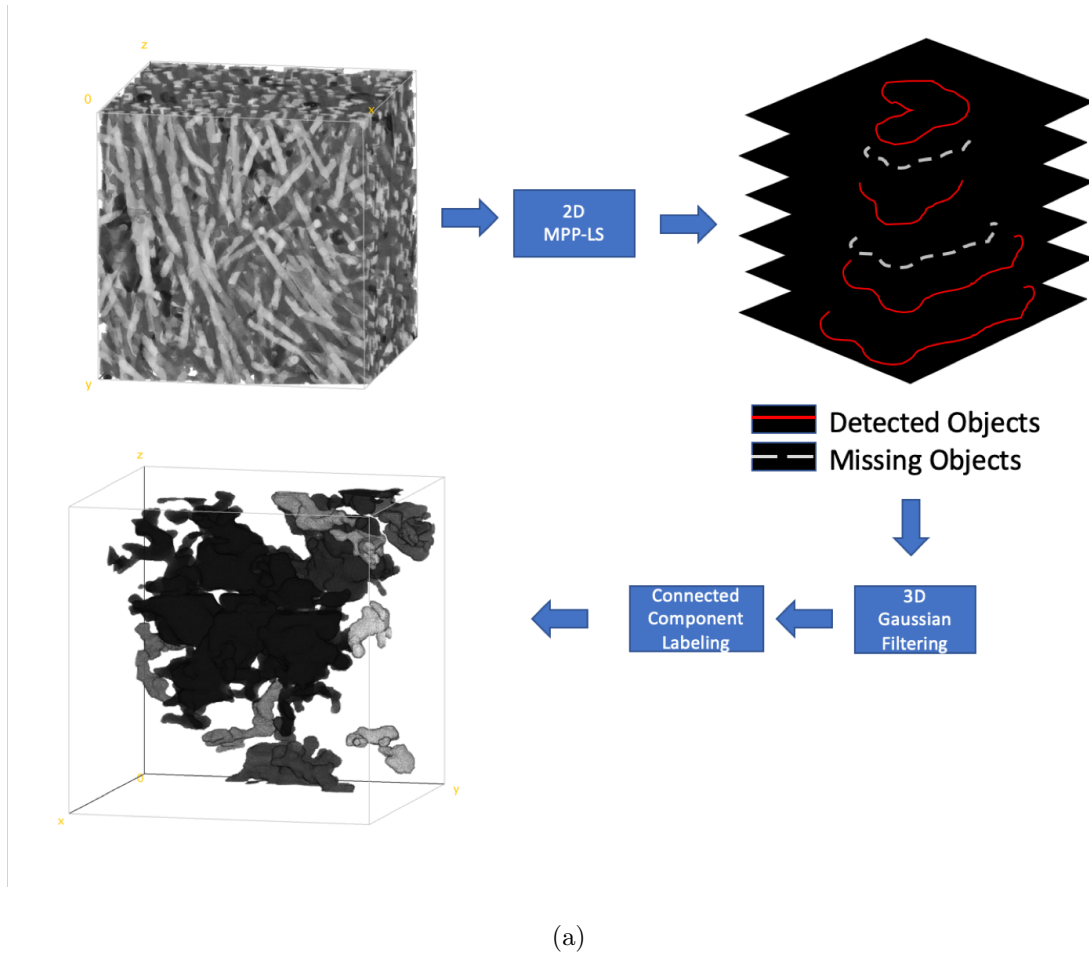
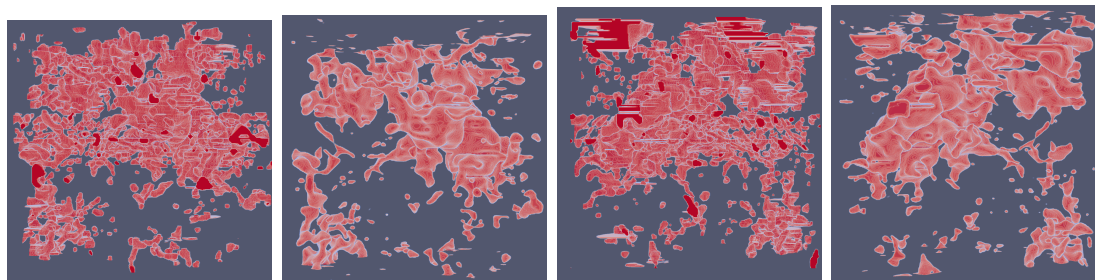


Fig. 4.13. Procedure to expand results to 3D



(a) Manual Labeling (b) Proposed Method (c) Manual Labeling (d) Proposed Method

Fig. 4.14. 3D Void detection from different views and comparison with rough manual segmentations

5. 3D VOID DETECTION AND FIBER EXTRACTION FOR STATISTICAL CHARACTERIZATION OF FIBER-REINFORCED POLYMERS

5.1 Overview

Chapter 4 finishes with an approach to perform volumetric segmentation. However, the combination of multiple geometries such as fibers (tubes) and voids (contours) joint detection increases significantly the model complexity and memory requirements. In addition, the inference time for large volumes can last for several days or up to weeks, presenting a necessity for faster and unified methods. In this chapter, we propose a surrogate approach consisting in training a neural network with the overlapped model-based results. This approach focuses on microscopy images where labeled data is not readily available, but purely model based approaches can be too slow to process entire volumes due to their computational complexity. In addition, we propose an encoder-decoder alternative to a fiber instance segmentation paradigm, showing a speedup in training and inference times without a significant decrease in accuracy with respect to alternative methods. The neural networks approach represent a significant speedup over model based approaches and can correctly capture most fibers and voids in large volumes for further statistical analysis of fiber reinforced polymers.

5.2 Related Fiber and Void Segmentation Works

Many researchers have proposed solutions to extract structural information from volumetric representations of fiber-reinforced polymers. For example, multiple projects have used morphological filters and watershed to extract fibers Huan et al. [52], yet

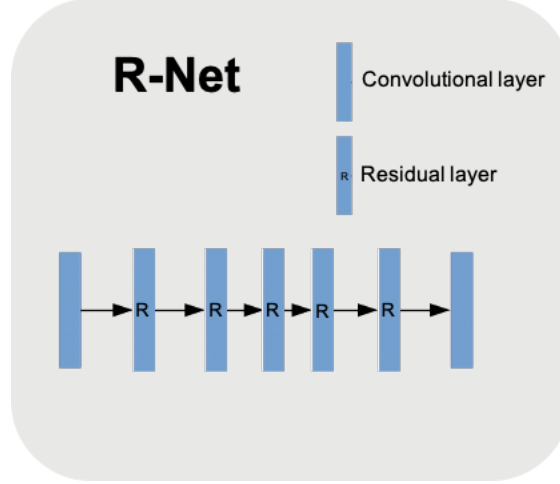


Fig. 5.1. Sample architecture with residual blocks proposed by [13]

this problem presents issues detecting thin fibers. Similarly, Agyei et al. [14] proposed a multi-view fiber fitting based on 2D pre-segmentation and 2D multi-slice ellipse fitting; however this method relies on characterizing fiber cross sections for different axes with 2D ellipses. Also, Li et al. [53] proposed to use a connected tube marked point process model to extract fibers. This approach proved promising but non-scalable to large volumes due to the required time to find each tube. Finally, Konopczynski et al. [13] proposed an embedded learning method to extract fibers from composite materials; however this approach requires pre-labeled data and the large memory requirements of this architecture limits the network to tile the sample into very small subvolumes ($32 \times 32 \times 32$). This architecture is depicted in Fig. 5.1, which consists in a neural network comprised of several convolutional and residual blocks connected in series.

On the other hand, few methods have been proposed to extract voids from tomographic images. This occurs due to the irregular shape and arbitrary sizes that voids can exhibit in 3D. Numerous researchers proposed threshold segmentation, however this approach does not work well for large voids [54] [2]. The work proposed by Nageswara et al. [55] proposed finding voids from phased arrays but in this chap-

ter we only deal with absorption images. In Chapter 3 we proposed void detection using active contour framework embedded in a marked point process model and in Chapter 4 we extended this method to level sets; however, the computational complexity of such a methods present difficulties when extending to a third dimension and extracting objects in very large volumes.

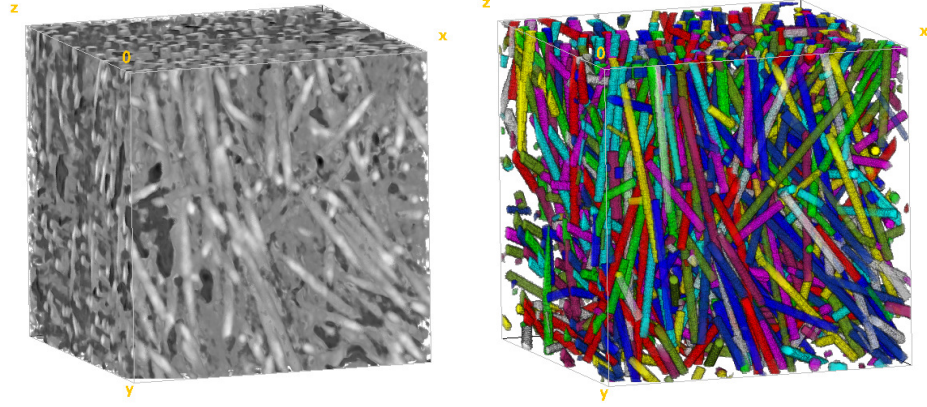
In this chapter, we propose a surrogate approach that incorporates model-based methods on subvolumes and uses their results to train a neural network. This approach takes into account the lack of labeled data but also exploits the inference parallelization of using convolutional neural networks. In addition, we propose an encoder-decoder architecture to perform instance segmentation in order to speed up training and inference times.

5.3 Model Based Methods: Marked Point Process

We used the MPP method explained in Section 2 applied to subvolumetric data to obtain the labeled data. Explicitly, we used the MPP method proposed by Li et al. [53] and the MPP method proposed in Chapter 4. The sample results of fiber detection are denoted in Fig. 5.2(b) and the sample void detection is shown in Fig. 5.3(b).

5.3.1 Fibers: Connected Tube Marked Point Process

We used the 3D extension of the connected tube MPP model proposed in [53] to detect fibers. This approach defines the mark space as $M = [R_{min}, R_{max}] \times [L_{min}, L_{max}] \times [\theta_{min}, \theta_{max}] \times [\phi_{min}, \phi_{max}]$ and proposed a prior model to encourage long tubes, and connections between nearby short tubes. Fig. 5.2(a) and Fig. 5.2(b) shows a volume and 3D results of this approach applied to a glass fiber reinforced polymer. This approach can model fibers by detecting several tubes connected by their end points. We followed the procedure listed in [53] in order to set the model parameters.



(a) Glass fiber reinforced polymer [15]

(b) Connected Tube MPP [53]

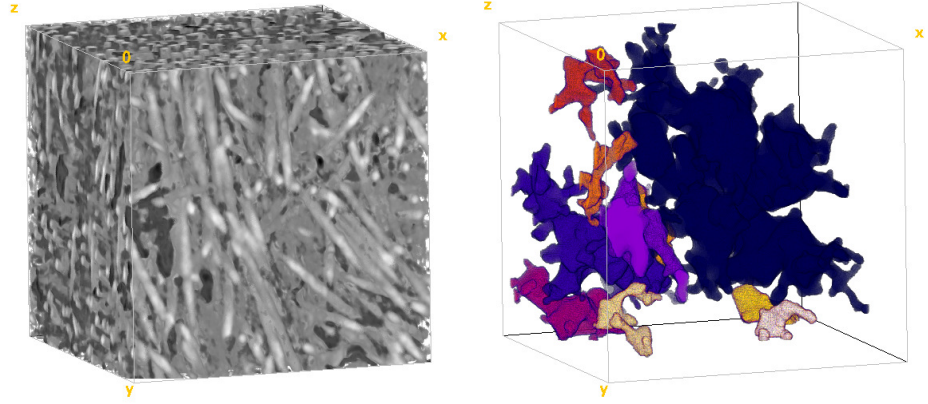
Fig. 5.2. Sample connected-tube MPP results

5.3.2 Voids: Hybrid-Level sets and Active Contours

We detected voids using the method explained in section 4.6. One drawback of combining an MPP with level sets is that the computational requirements used to deform the contours increase exponentially with an increase of dimensions. Therefore, we detected voids in successive 2D slices, and used 3D smoothing filters to merge the results into a 3D structure. This approach is denoted in Fig. 5.3(b).

5.4 Surrogate approach with Convolutional Neural Networks

Despite the promising results, MPP models rely on stochastic sampling for optimization. This approach can result in extensive computational burdens in order to find an optimal object configuration. This issue could make the inference of very large volumetric datasets unfeasible. For example, Table 5.1 shows the running times for a varying size of subvolumes belonging to the fiber reinforced composite shown in Fig. 5.4 . These running times were measured in single subvolumes for the case of $300 \times 300 \times 300$ and $500 \times 500 \times 500$, and they were estimated for the subvolumes of



(a) Glass fiber reinforced polymer [15] (b) Consecutive slices stacked to form 3D volume

Fig. 5.3. Results of Model-Based void detection

Table 5.1.
Time profiling for MPP methods

Voxels	Connected Tube MPP	Level Sets MPP
$300 \times 300 \times 300$	18 mins	3 mins
$500 \times 500 \times 500$	6 hours	20 mins
$2025 \times 2025 \times 1350$	19 days	26 days

sizes $2025 \times 2025 \times 1350$ based on parallel implementations of connected tube MPPs and MPP-LS detection.

In addition, the proposed model based methods detect fibers and voids independently and the results are superimposed; however, both results could yield discrepancies such as classifying a voxel as both fiber and void. In this section, we propose the use of neural networks to tackle both the object detection speed-up and the merging of fiber and void detection into a unified method. We propose to use the combined

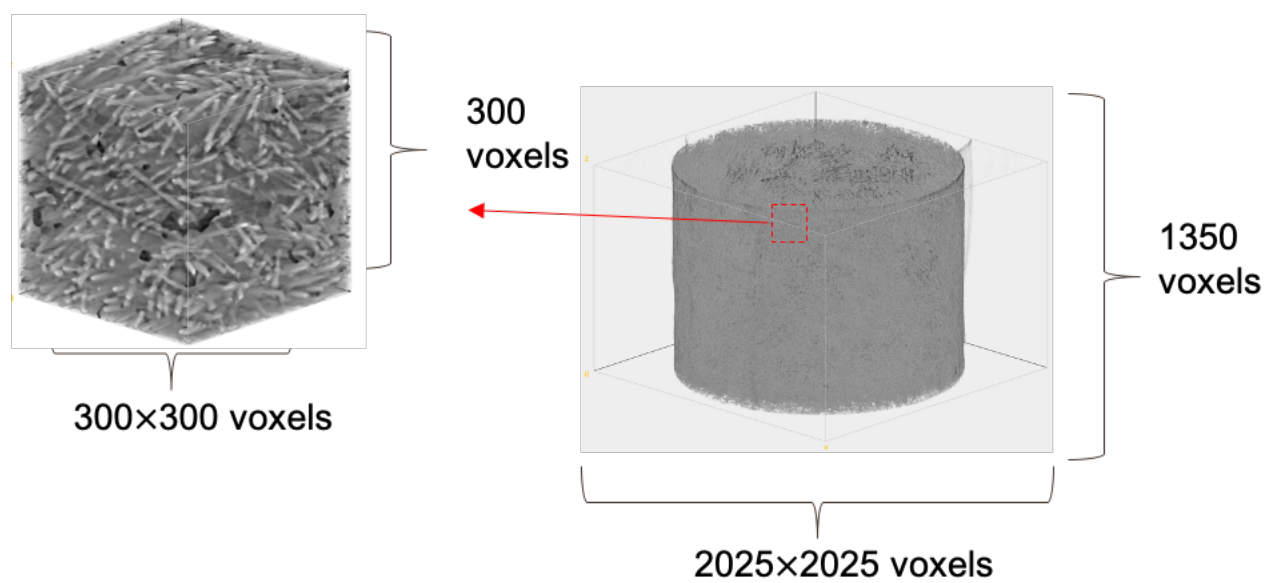


Fig. 5.4. Sample volume sizes

model-based results as labeled data to train neural networks to segment instances of objects in 3D. This approach is depicted in Fig. 5.5.

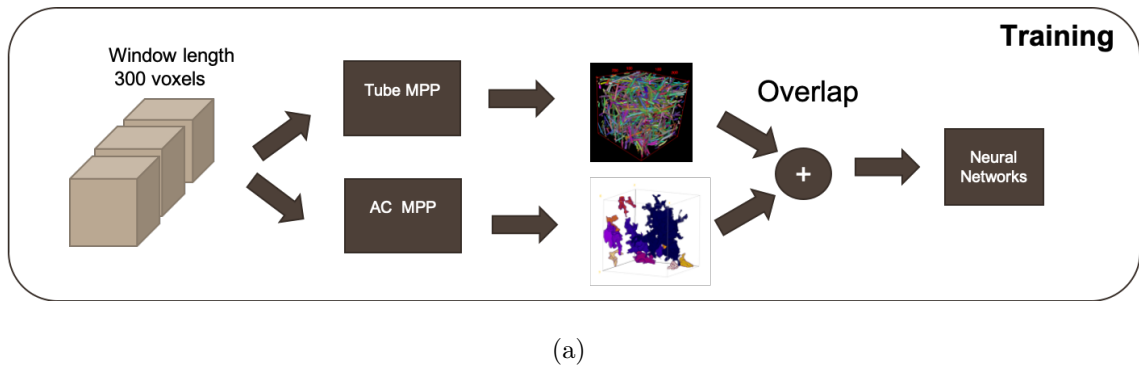


Fig. 5.5. Proposed surrogate approach to train neural networks from model-based results. The Tube MPP detects fibers and the AC-MPP detects voids.

5.4.1 Encoder-Decoder Embedding Instance Segmentation

Multiple projects have proposed instance segmentation approaches such as object proposal based [10] detection. For example Figs. 5.6(a) and 5.6(b) show possible bounding boxes characterization for a system comprised of fibers. These characterizations do not describe the system correctly since fibers are narrow objects oriented in arbitrary directions and the system requires of tight bounding boxes with different orientations. Similarly, a popular approach such as deep watershed [56] could fail to approximate a watershed energy for fibers with small radius. For example, Fig. 5.7(a) shows a sample approximate 3D watershed energy. A neural network would require very high resolution to estimate a correct watershed energy; however, neural networks with very high resolution often are limited to small subvolumes due to their memory requirements. In addition, the watershed approach requires of a correct marker initialization, Fig. 5.7(b) shows the watershed segmentation with an incorrect marker initialization.

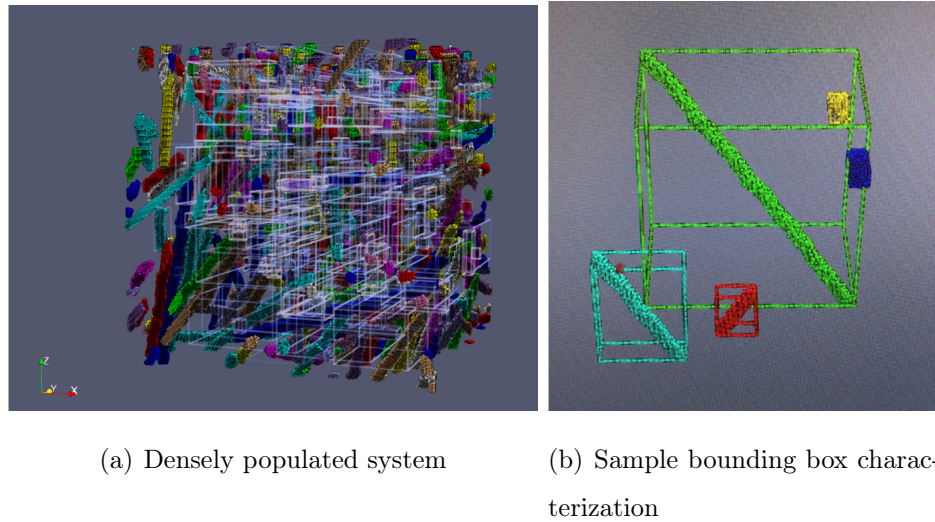


Fig. 5.6. Sample bounding-box proposals for 3D narrow fibers.

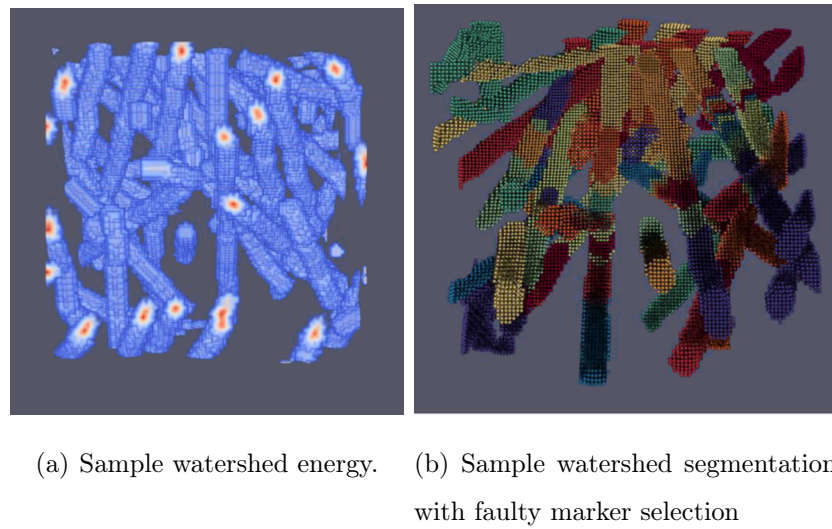


Fig. 5.7. Sample watershed segmentation approach for 3D fibers.

In this chapter, we explore further the work done in fiber embedding segmentation [13]. This method consists of tiling images into cubes of size $32 \times 32 \times 32$ voxels and using a residual network network shown in Fig. 5.1, with 2 output channels to classify pixels into foreground and background. Then, a similar network but with 12 output channels finds a mapping of foreground pixels into distant clusters in an embedded space. This approach is summarized in Fig. 5.8.

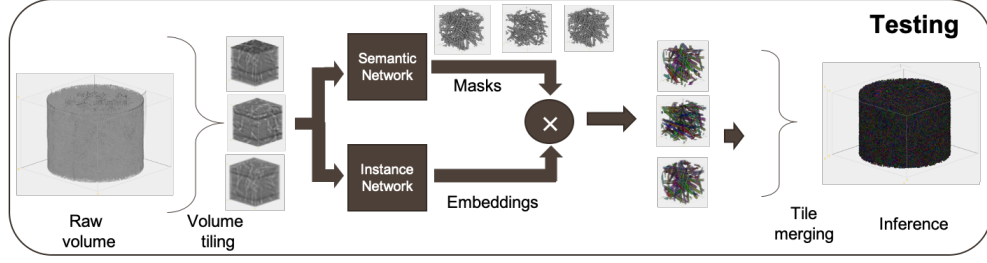
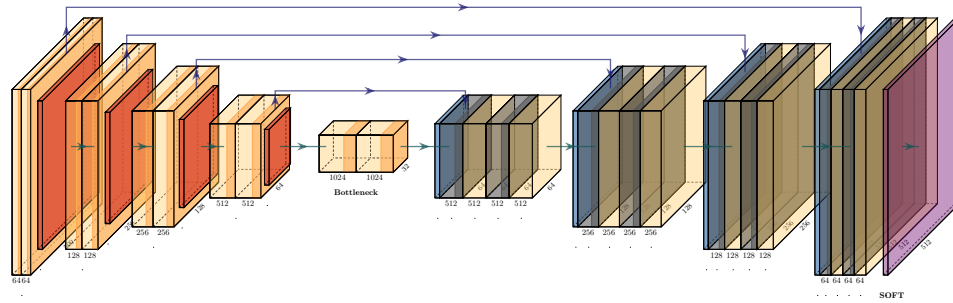
(a) $n=0$

Fig. 5.8. Method proposal for inference

We propose an extension to this work by using a U-Net [5] architecture. This architecture represents both a speed-up in inference times over the residual network and a reduction in memory requirements. This approach allowed us to increase the window size from $32 \times 32 \times 32$ voxels to $128 \times 128 \times 128$ voxels without compromising performance. Fig. 5.9 represents the chosen network architecture, where we followed the architecture proposed by [5] for semantic segmentation, and the same architecture but with 12 output channels for instance segmentation. The required memory for different window sizes is depicted in Fig. 5.10, where we compared the required memory for performing semantic segmentation with a 3D U-Net [5], a 3D DeepLabV3 [9], and the Residual-based Net from [5]. Similarly, Table 5.2 denotes the f1 segmentation scores for detecting fibers and voids for each method evaluated in a manually labeled subvolume from the dataset presented in section 5.5.1. The U-Net shows the best overall performance for segmenting voids and fibers while DeepLabV3 segmented voids better but obtain a 0.420 f1 score for fiber segmentation. In our im-

plementation and parameters, the R-Net had the worst performance for both fibers and voids while requiring the highest amount of memory.



(a)

Fig. 5.9. Encoder Decoder Network Architecture proposed by [5]

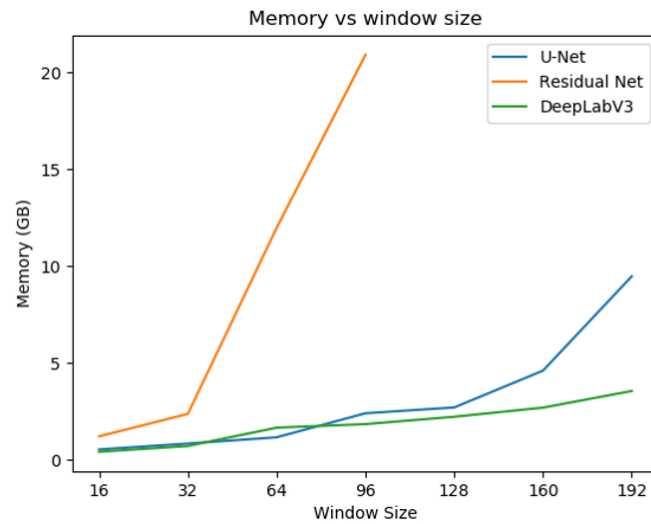


Fig. 5.10. Memory Consumption for different semantic segmentation architectures

Table 5.2.
Semantic segmentation results for different architectures

	f1 fiber segmentation	f1 void segmentation
U-Net [5]	0.809	0.622
R-Net [13]	0.326	0.067
DeepLabV3 [9]	0.420	0.701

5.4.2 Network Training

Semantic Segmentation

We used the weighted cross entropy loss function in order to train the semantic segmentation network. The cross entropy function is defined as:

$$Loss_{Semantic} = - \sum_{i=0}^{L-1} w_i t_i \log(s_i) \quad (5.1)$$

where L is the number of classes, t_i is the ground truth label, s_i is the output score for class i , and w_i is a predefined weight. We trained the network with $L = 3$ for the dataset depicted in Fig. 5.15(a), with $w_0 = 1$ (matrix), $w_1 = 10$ (fiber), $w_2 = 20$ (void) and $L = 2$ for the dataset depicted in Fig. 5.13(a) with $w_0 = 1$, and $w_1 = 5$. These weights were chosen from the voxel ratios between classes available in the labeled data. We used the Adam optimizer with a learning rate $lr = 0.0001$ and we trained the network for 1000 epochs, feeding 30 cropped subvolumes in each epoch.

Instance Segmentation

For instance segmentation, we followed the approach taken by [13]; therefore we used a similar network to our semantic segmentation network, but with 12 output channels. The instance segmentation loss function is defined as follows:

$$Loss_{instance} = \alpha L_{pull} + \beta L_{push} + \gamma L_{regularization} \quad (5.2)$$

The first part of the loss function teaches the network to pull the embedding outputs of an instance towards the center of a cluster and is defined as

$$L_{pull} = \frac{1}{C} \sum_{c=1}^C \sum_{x_i=1, x_i \in c}^{N_c} (||x_i - \mu_c|| - \delta_v)_+^2 \quad (5.3)$$

, where $(x)_+ = \max(0, x)$, C is the number of instances, x_i is the output of the embedding network for the i^{th} voxel that belongs to instance c , and $\mu_c = \frac{1}{N_c} \sum_{i=1}^{N_c} x_i$ is the mean embedding value for one specific instance with N_c voxels. In all our experiments, we set the hyper-parameter $\delta_v = 0.2$. This value could be translated to the euclidean distance parameter in clustering algorithms as the maximum distance to consider two points to belong to the same cluster.

The second term in Eq. (5.2) has the purpose of teaching the network to push the centers of the clusters away from each other. This loss function is defined as $L_{push} = \frac{1}{(C)(C-1)} \sum_{i=1}^C \sum_{j=1, j \neq i}^C (\delta_d - ||\mu_i - \mu_j||)_+^2$. In all our experiments, we set $\delta_d = 5$. This parameter was chosen following the convention $\delta_d \gg \delta_v$.

The third part of the loss function is intended to regularize the mean embeddings and is defined as $L_{regularization} = \frac{1}{C} \sum_{i=1}^C ||\mu_i||$.

This loss function only takes into account the foreground pixels previously classified by the semantic segmentation network. We used the Adam optimizer with learning rate $lr = 0.001$ and we trained the network during 2000 epochs with subvolumes cropped from the labeled data. We followed the parameter setting used by [13] and set the parameters $\alpha = 2$, $\beta = 2$, and $\gamma = 0.001$. Fig. 5.11 shows the learning procedure of the mapping network for different training iterations. Each color represents a distinct instance of a fiber. Finally a clustering algorithm (DBSCAN [57])

assigns each pixel to clusters that represent fiber instances. Fig. 5.12 shows the sample instance segmentation f1 results for the labeled dataset presented in Section 5.5.2 at different number of embeddings. We tested these results in 3 subvolumes of sizes $64 \times 64 \times 64$.

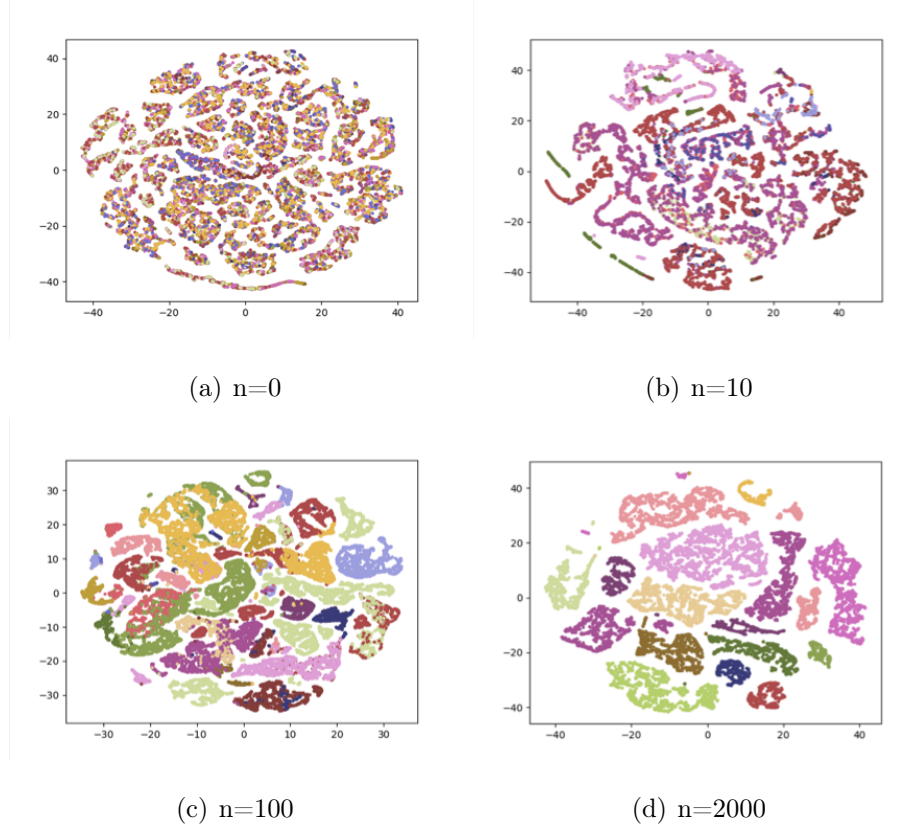


Fig. 5.11. Embedded learning at different iterations. Images have been reduced to 2 dimensions using t-SNE [58] for visualization purposes

5.5 Results

In this section we present results for the two datasets. One experimental dataset where we do not have ground truth; therefore our results are evaluated by comparing sample statistics with the results from Agyei2018 et al. [15] and Hanhan et al. [2]. We focused on comparing statistics of interest such as fiber and void fraction, number

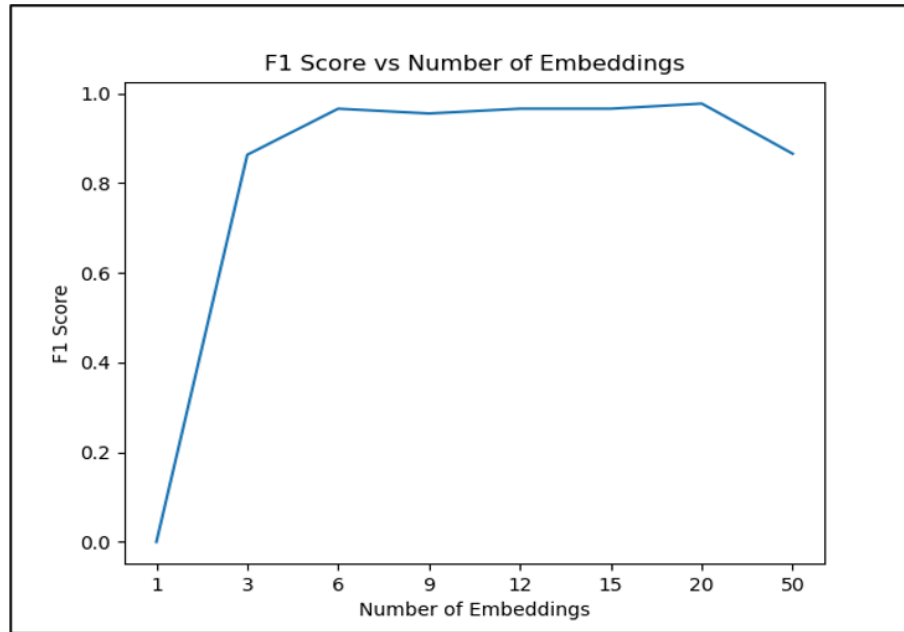
(a) $n=0$

Fig. 5.12. f1 score vs number of embedding

of fibers, fiber length distribution and fiber orientation distributions, and finally, computational time.

In addition, we used a set of synthetic datasets for numeric evaluation and we considered three parameters: 1) voxel-wise segmentation accuracy, 2) fiber detection correctness, and 3) inference time. In addition, we perform experiments in an experimental dataset where we do not have ground truth,

We evaluate our method compared to the superimposed model-based results (Connected Tube MPP and Active Contours MPP), the residual network-based approach [13], our proposed encoder-decoder approach using the MPP results as training data and our proposed encoder-decoder approach using the true labels as labeled data (when available).

5.5.1 Experimental Data: Glass Fiber Reinforced Polymer

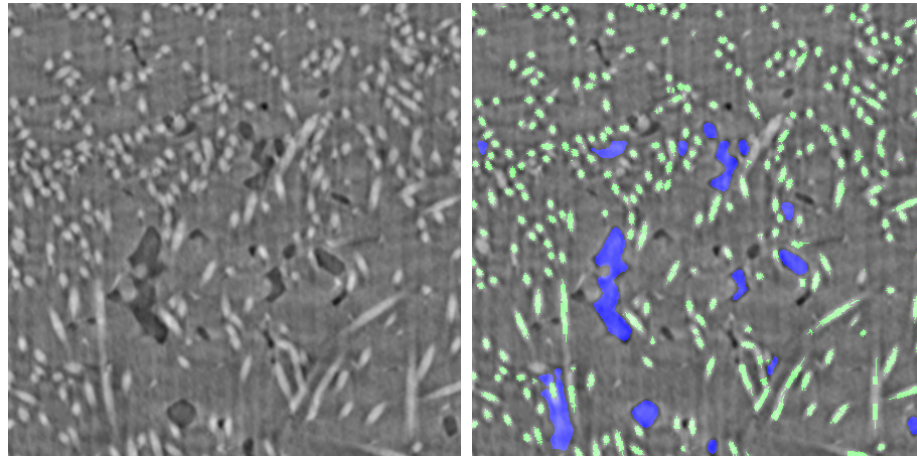
The first dataset was an X-ray micro-computed tomography scan of a glass fiber reinforced polypropylene [15] and represents glass fibers in a polypropylene matrix composite. The sample was imaged at $1.3\mu m$ of resolution and its dimensions are $2400 \times 2400 \times 1300$ voxels. In this chapter, we used a subvolume of $301 \times 301 \times 301$ voxels for display purposes. It is worth noting we used the connected tube MPP and active contours MPP to obtain labeled training data.

Semantic Segmentation

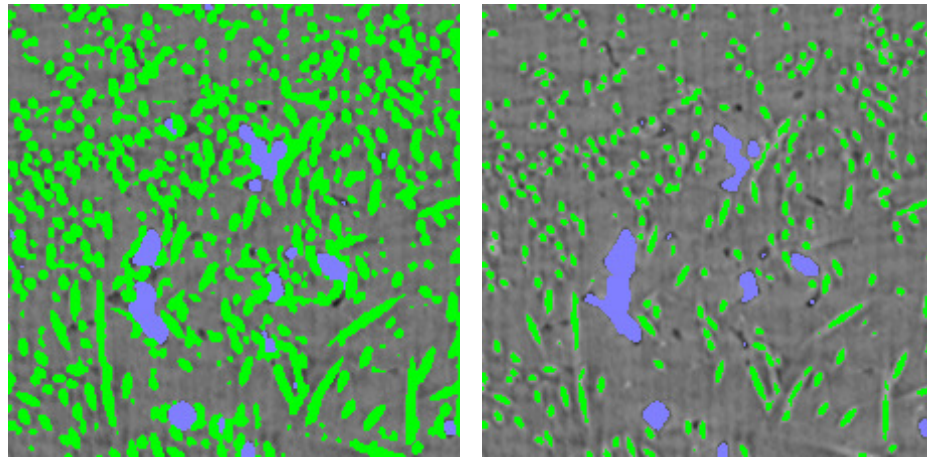
For semantic segmentation, we converted the superimposed MPP results to three classes denoted in Fig.5.13(b). These methods provided a starting point for the semantic segmentation, but they still presented noise in the results. For instance, Fig.5.13(b) shows a false void(blue) at the bottom left part of the image. Similarly, Fig.5.13(b) shows multiple missed fibers(green) near the center and left part of the image. This issue happens due to noise generated during the imaging and sample reconstruction procedure, and due to parameter calibration for both MPP approaches.

We trained both networks using the same training parameters, however we used a window size of $128 \times 128 \times 128$ voxels for our network, and we used a window size of $32 \times 32 \times 32$ voxels for the residual network. Both neural network approaches were able to segment all the voids and reduced the amounts of phantom voids that were detected in the model-based approaches. Additionally, the networks learned to segment fibers that were not segmented originally in the model-based methods; however, Fig. 5.13(c) shows the residual network presented over-segmentation of fibers. This issue leads to difficulty performing instance segmentation.

Fig. 5.13 denotes the results obtained from finding the semantic segmentation for 3 different approaches: Connected Tube MPP [53], Fully Convolutional Residual Network [13], and our method. Fig 5.13(c) represents the output of our implementation of the work proposed by Konopczynski [13] trained with the MPP as labeled data.



(a) Cross Section of Fiber Reinforced Composite (b) Connected Tube MPP and MPP-LS Composite [53]



(c) Residual Network [13]

(d) **Proposed Encoder-Decoder**

Fig. 5.13. Semantic Segmentation for polypropylene matrix composite. Fibers are represented green, voids are represented blue

Instance Segmentation Statistical Comparison

In order to validate the instance segmentation, our method was evaluated and compared in a subvolume of size $950 \times 950 \times 150$ with the results of [2] and [14]. For example, Fig. 5.14 shows the orientation A11, A22, and A33 orientation tensors of fibers detected across different regions of the sample. The plots demonstrate that

Table 5.3.
Statistical Comparisons for the Proposed approach and leading fiber segmentation approaches

Statistic	Hanhan [2] and Agyei [14]	Proposed Approach
Fiber volume fraction	9.47%	9.21%
Void volume fraction	3.63%	2.78%
Number of fibers	4613	4045
Fibers with aspect ratio > 5	45.70%	45.96%
Computation time for full volume	125 hours	30 hours

the proposed method had close values to the one proposed by [15] and [2]. Similarly, Table 5.5.1 shows the results for numerous statistics where the results of the proposed method matched the statistics obtained by leading fiber segmentation methods. The fiber volume fraction has a 0.26% difference, the void volume dafraction has a 0.85% difference, and the difference in percent of fibers detected with aspect ration greater than 5 is 0.26%; however, the proposed method obtained a computation time of 30 hours versus the 125 hours required for the fiber segmentation approach proposed by [14] and [2].

5.5.2 Labeled Data: Synthetic Low Resolution Fibrous Material

We tested our methods on a synthetic dataset [13] for numerical comparison. This dataset provides several real and synthetic fibrous volumes imaged at $3\mu m$ of resolution. We used the datasets R_HR3_1 and R_HR3_2 for training the networks, and we evaluated the results on multiple subvolumes of the dataset S_HR_5.35. We use only fiber data for validation since this dataset does not contain voids and we used subvolumes of size $600 \times 600 \times 600$ pixels for numerical and timing analysis.

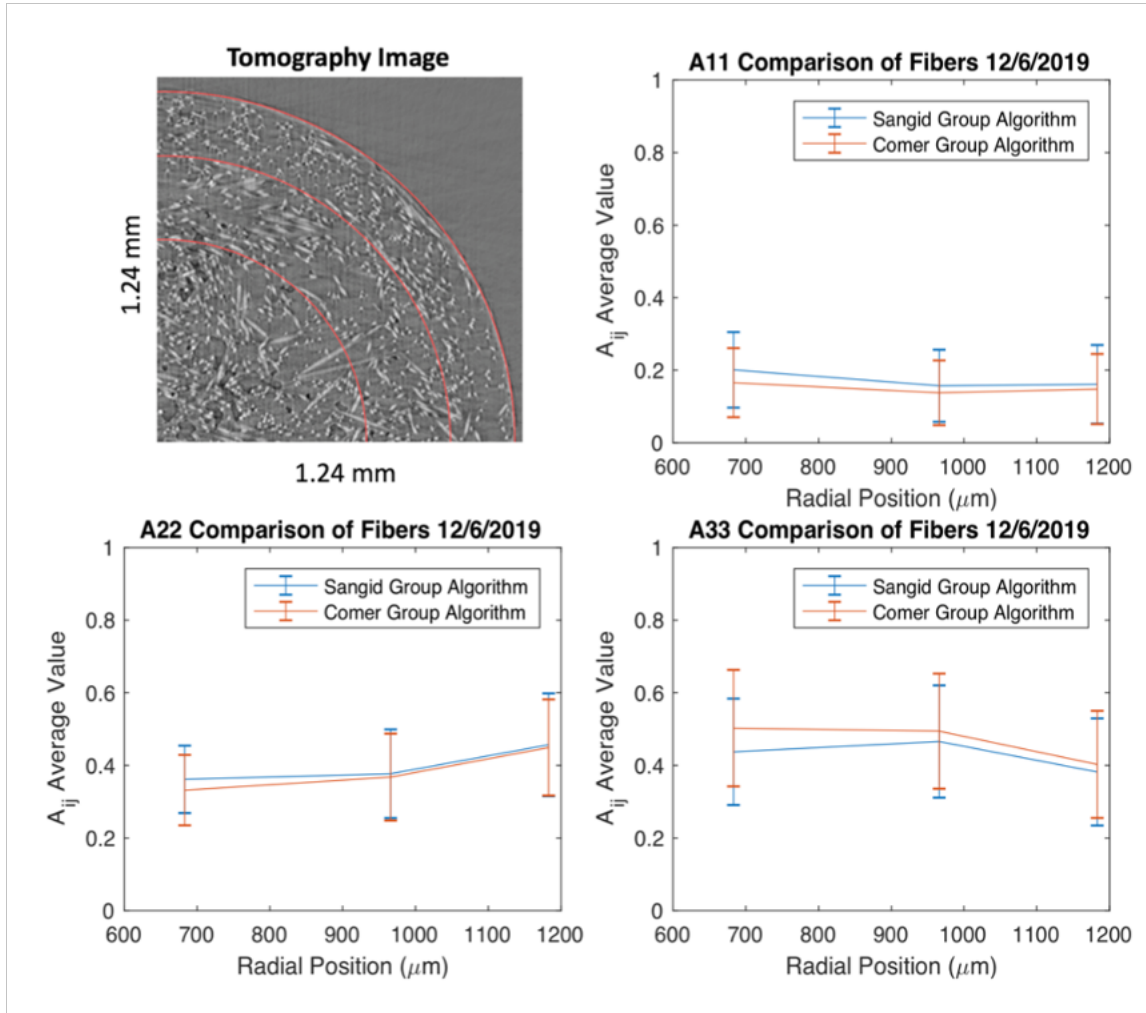


Fig. 5.14. Fiber orientation tensors at different locations. Image obtained from the ACME Lab at Purdue University.

Semantic Segmentation

For evaluation of semantic segmentation, we evaluated precision, recall and the f1 score, which is defined as:

$$f1 = \frac{2 * precision * recall}{precision + recall} \quad (5.4)$$

where $precision = \frac{TP}{TP+FP}$, and $recall = \frac{TP}{TP+FN}$, and TP denote true positives, FP denote false positives, and FN denote false negatives.

The results for the segmentation numerical comparison are depicted in Table 5.4. All the neural networks-based approaches present high recall values but low precision values. This occurs due to the over segmentation occurring at the fiber boundaries. On the other hand, the connected tube MPP has a larger precision but lower recall values. This issue could happen because the MPP models objects with basic geometries(tubes) and does not account for the fibers' surface roughness.

Table 5.4.

Quantitative Results

Method	Precision	Recall	f1
Connected Tube MPP	0.801	0.692	0.743
Embedded Learning [13]	0.547	0.986	0.704
Proposed trained with MPP	0.561	0.976	0.712
Proposed trained with labels	0.662	0.972	0.787

Despite the different approaches proposed for segmentation, the maximum f1 score is 0.787 by the encoder-decoder network. The low scores happen mainly due to the ambiguity at fiber boundaries. However, it should be noted that, due to limitations in spatial resolution of physical imaging systems, it is often impossible for even materials experts to determine exact boundaries between different regions in a microstructure.

We consider any pixel to be within $\sqrt{2}$ voxels from a fiber voxel to be a TP; similarly, we consider any segmented voxel farther than $\sqrt{2}$ voxels from a labeled fiber voxel to be a FP, and we consider any missed fiber voxel farther than $\sqrt{2}$ from a labeled fiber voxel to be a FN. The results with the this "relaxed-boundary" scoring are presented in Table 5.5.

Table 5.5.
Relaxed-Boundary Quantitative Results

Method	Precision	Recall	f1
Connected Tube MPP [53]	0.979	0.966	0.972
Residual Network [13]	0.918	0.999	0.957
Proposed-trained with MPP	0.990	0.911	0.949
Proposed-trained with labels	0.894	0.999	0.944

Instance Segmentation

For instance segmentation, we considered 3 common cases in fiber detection: the fiber is detected correctly (true positive), the fiber is artificially broken (broken pieces are false positives), or the fiber is missed(false negative). We say a fiber is detected correctly if the IoU between the detected object and the ground truth label is greater than 0.5. We consider a fiber to be broken if multiple detected fibers fit into the ground truth fiber, such that the total IoU is greater than 0.5. We say a fiber is missed if either the total IoU of overlapping detected objects and ground truth label is less than 0.5, or if the fiber was merged with a nearby fiber. Fig. 5.15 shows the results for a subvolume of $600 \times 600 \times 600$ voxels using both MPP and Connected Tube MPP. Table 5.6 denotes the fiber detection numerical results using different approaches. In terms of detecting fibers at least partially (including the detected and

broken cases), the approach with best results was the Connected Tube MPP which missed only 0.6% of fibers. The next best result uses the proposed encoder-decoder network using the labeled data for training, which missed 12% of fibers. Finally, the Residual Network approach together with the proposed method trained with MPP labels showed worse performance a 74% and 78% of fibers correctly detected.

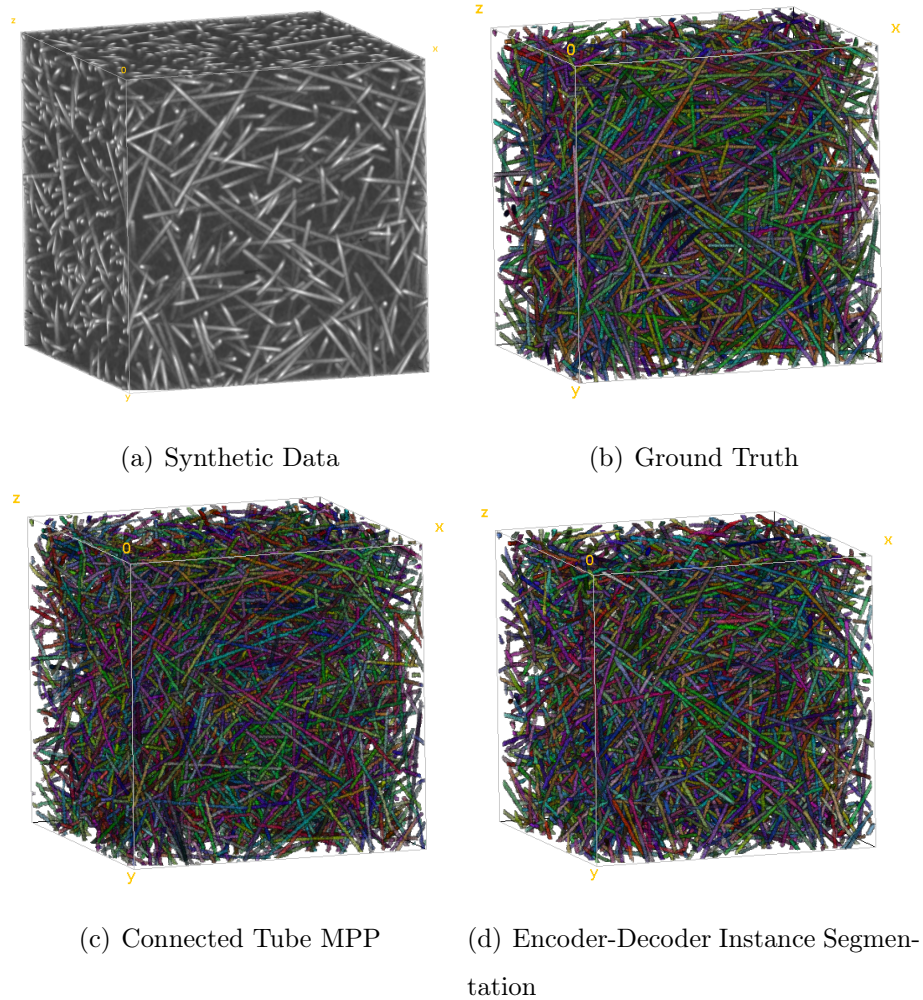


Fig. 5.15. Instance Segmentation

Table 5.6.
Fiber Detection Results

Method	f1
Connected Tube MPP [53]	0.932
Residual Network [13]	0.855
Proposed trained with MPP [53] results	0.880
Proposed trained with labels	0.930

Required Time

We performed timing comparisons for volume inference for volumes of size 100^3 , 200^3 , 400^3 , and 600^3 . Fig. 5.16 denotes the different timing measurements for each method at different subvolumes. The connected-tube MPP model was implemented in C++ programming language and was timed using a single core on an Intel(R) Core(TM) i9-9900X CPU processor running at 3.50GHz. Both neural networks were implemented in pytorch and tested in the same machine with an NVIDIA TITAN RTX GPU. The encoder/decoder architecture represented a 10 times speedup over the connected tube MPP, and a 4 times speedup over the residual network architecture.

5.6 Chapter Summary

We presented an approach to train a neural network with model based approaches in order to detect objects in microscopy images. We validated our method comparing the extracted material statistics with different methods and we also validated our data using f1 semantic and instance segmentation score on a synthetic dataset. Our approach showed several advantages over the model based results and also over common deep learning instance segmentation methods.

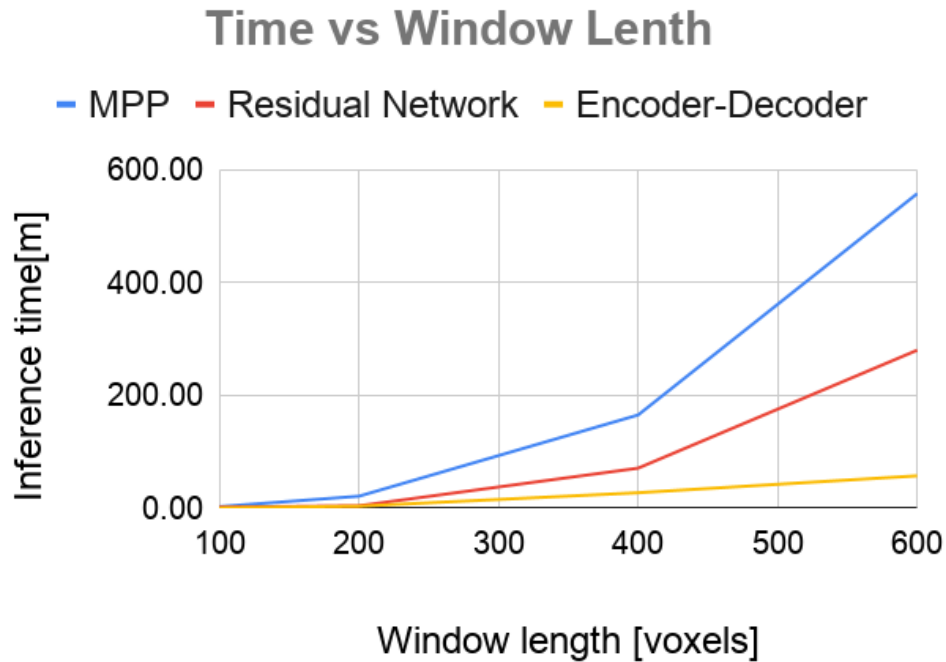


Fig. 5.16. Time Difference

First, we discussed the motivation of using CNNs over model based methods for inference in large datasets. While model-based approaches allow system characterization and simulation, they present drawbacks in the required inference times. For example, Table 5.1 shows that the model-based methods alone would require up to 26 days to obtain results in a $2025 \times 2025 \times 1350$ voxels sample. However, they can be used to segment subvolumes of $300 \times 300 \times 300$ and these results can become training data for a CNN. On the contrary, Table 5.5.1 shows the computation time on the same volume reduced from 26 days from model based and from 125 hours from modern fiber segmentation approaches, to 19 hours when using CNNs.

In addition, we analyze using embedded segmentation over object proposal-based RCNNs and Deep Watershed to detect fibers and voids. Fibers in microscopy images are narrow objects oriented in arbitrary directions and cannot be correctly characterized with a four point bounding box. Similarly, a method such as Deep Watershed

would require very high resolution to correctly estimate the watershed energy for fibers; however, 3D segmentation imposes memory constraints and this method would be constrained to very small tiles in order to correctly estimate a watershed energy.

Also, our results showed that for a dataset consisting of fibers and porosity, a U-Net [5] architecture obtains the best overall f1 scores over two other popular semantic segmentation architectures, namely the residual based convolutional neural network (R-Net [13]), and the encoder-decoder based DeepLabV3 [9] neural network. The U-Net architecture was able to segment voids thanks to its multi-resolution approach and was able to segment fibers thanks to its skipped connections. On the other hand, DeeplabV3 was better at capturing large voids but it did not segment fibers correctly due to its loss of resolution during the encoding procedure. The residual based network performed poorly detecting voids due to its small field of view and was also outperformed by encoder-decoder networks for fiber detection.

Finally, we tested the effects of training the neural networks with the results of model-based approaches. While the model-based approaches obtain reasonable results, they contain noise and can yield sub-optimal segmentations. However, the neural network obtained better semantic segmentation results than the noisy data used for training. It can be inferred that 1) the neural network generalized the 3D data from few examples and that 2) the neural network worked as a denoiser due to its encoder-decoder architecture. Nevertheless, Tables 5.5 and 5.4 show that our method still performs better when trained on real labeled data versus trained with noisy model-based results.

6. 3D FIBER SEGMENTATION WITH DEEP CENTER REGRESSION AND GEOMETRIC CLUSTERING

6.1 Overview

In Chapter 5, we presented a surrogate approach to detect fibers and voids in fiber-reinforced polymers. While this approach represented a speed-up in inference without compromise in accuracy, it is limited to its training dataset and has difficulties adapting to new data. In this chapter, we propose a volumetric object detection approach comprised of fibrous structures by using deep centroid regression and geometric regularization. To this end, we train encoder-decoder networks for segmentation and centroid regression. We use the regression information combined with prior system knowledge to propose cylindrical objects and enforce geometric regularization in the segmentation. We train our networks on synthetic data and test our methods in several experimental datasets. Our approach shows competitive results when tested in the synthetic data and outperforms common 3D segmentation methods across different datasets.

6.2 Related Work

6.2.1 Instance Embedded Learning

This method consists of segmenting and grouping pixels into clusters in an abstract embedded space. The network outputs a vector e of dimensions $N_F \times K$ where N_F is the number of foreground pixels and K is the dimension of the embedded space in which the networks learns to form clusters. The loss function consists of a weighted sum of three parts and is defined as:

$$L_{embedded} = \alpha L_{pull} + \beta L_{push} + \gamma L_{reg} \quad (6.1)$$

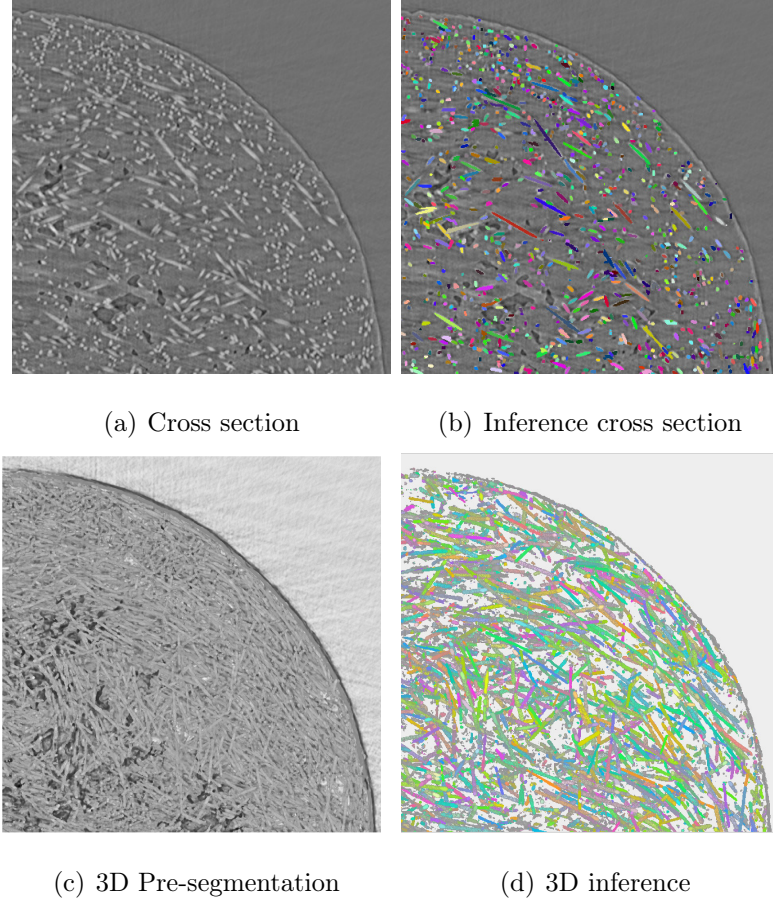


Fig. 6.1. Sample results of our method displayed with Fiji [37] software. Each color represents a different instance of an object. Image courtesy of the ACME Lab at Purdue University.

The L_{pull} term learns to pull each embedding point to the center of the mean embedding of an instance, the L_{push} term learns to separate the cluster means from each other, and the L_{reg} term regularizes the instance means. Finally, a clustering algorithm groups points in the embedding space in order to map pixels to instances.

The embedded learning concept has been shown to be memory efficient and promising; however, this method relies on finding an arbitrary embedded space that does not have a direct interpretation in the image space and the choice of both the clustering algorithm and its parameters influence greatly the segmentation results. A

large *eps* parameter for DBSCAN merges nearby clusters and a small *eps* parameter splits clusters or does not detect them.

6.2.2 Joint-Task Learning

Several papers (for both 2D and 3D) have proposed joint-task approaches to circumvent the foreground/background pixel classification and instance regression. These methods rely on single encoder-multiple decoder networks. For example, Neven et al. [59] trained an encoder-double decoder network to jointly learn instance centers and the cluster bandwidth. However, this method uses a loss function that relies on a weighted sum of a center regression and bandwidth loss. This issue could result in time-consuming parameter tuning when training the networks for several days. Kendall et al. proposed an efficient ensemble training for both tasks [60]. This approach estimates the weight uncertainties for each task to find the optimal weight parameter. The weight uncertainty approach states the multi-task loss function $\mathcal{L}_{\mathcal{MT}}$ as a Gaussian log likelihood :

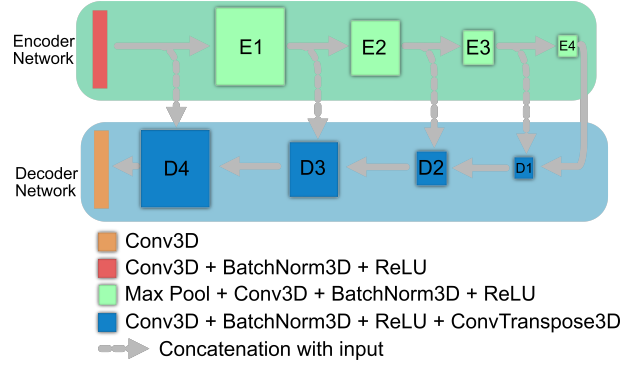
$$\mathcal{L}_{\mathcal{MT}} = \frac{1}{2\sigma_1^2}\mathcal{L}_{Seg}(W) + \frac{1}{2\sigma_2^2}\mathcal{L}_{Reg}(W) + \log \sigma_1\sigma_2, \quad (6.2)$$

where W represents the network weights to be optimized, $\mathcal{L}_{Seg}(W)$ represents the segmentation loss, $\mathcal{L}_{Reg}(W)$ represents the regression loss, and σ_1 and σ_2 represent the weight uncertainties for the segmentation and regression losses respectively. The optimization procedure minimizes with respect to W , and (σ_1, σ_2) . This approach is promising; however, the combination of two decoders and one encoder can worsen the results of each decoder. In fact, in our experiments, the implementation of two separate networks obtained better results across all the tests.

6.3 Method

We propose a two-network approach, one for segmentation, one for centroid regression. In addition, we propose to use the regression vectors to obtain information

about the original properties of each instance, such as the instance’s orientation and length. We used a modification of the architecture presented in [5] due to its effectiveness in microscopy volumes and robustness to scarce training data. The two networks are the semantic segmentation, which has two outputs denoting the probability of each class, and the regression network, which has three outputs denoting a vector pointing to the instance center. Fig. 6.2 denotes the detailed architecture of our networks.



(a)

Fig. 6.2. Architecture: we used a modified version of U-Net [5].

Pixel Classification

For semantic segmentation, we use the dice loss with two classes, foreground to represent fibers and background to represent non-fiber pixels. The dice loss function is expressed as:

$$\mathcal{L}_{Seg}(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|} \quad (6.3)$$

where X and Y are the input and training vectors respectively. The output of this network is depicted in Fig. 6.3(c).

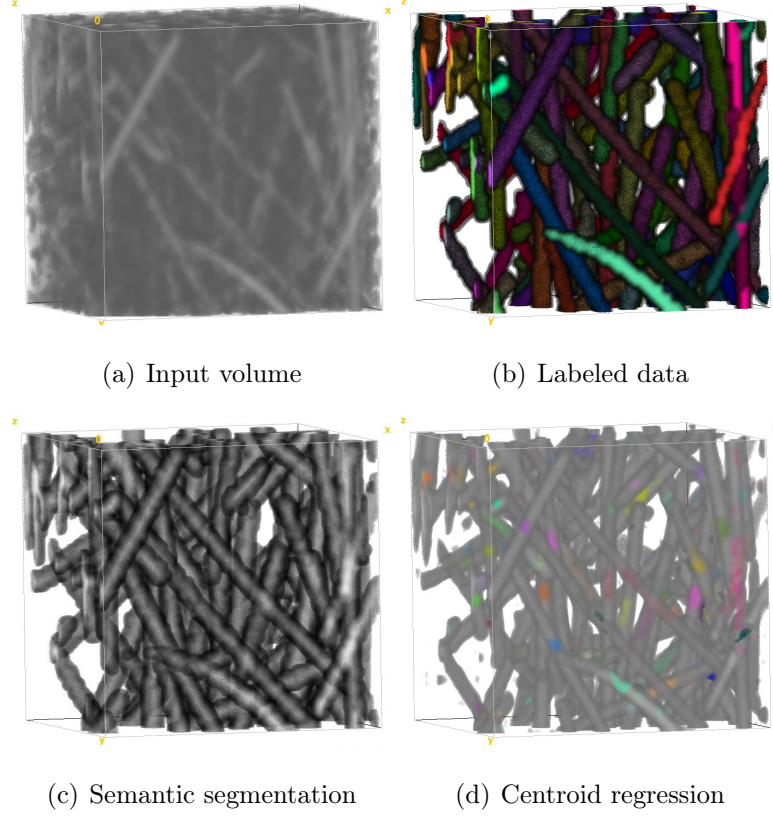


Fig. 6.3. Network Outputs: The networks detect the fiber pixels and their instance centroid μ_c . Each color represents an instance, and pixels are clustered around their centers. Gray fibers are for display purposes. The figures were generated with the Fiji Software.

Centroid Regression

The second network learns to cluster pixels around their instance centroid μ_c . The network outputs a vector $v_i \in \mathbb{R}^{k \times k}$ for each foreground pixel's coordinate $s_i \in S_f$ such that $v_i = s_i - \mu_c$. The loss function for centroid regression is defined as:

$$\mathcal{L}_{Inst}(S_f) = \sum_{c=1}^C \sum_{s_i \in S_c} (\|s_i - \mu_c\| - \delta_v)_+^2, \quad (6.4)$$

where S_f is the set of foreground pixels coordinates, $S_c \subset S_f$ is the subset of pixel coordinates belonging to object c and C is the total number of objects obtained from the labeled data. The term $(a)_+ = \max(a, 0)$ is inspired by the Hinge Loss

function, with δ_v as a hyper-parameter representing the maximum distance between neighboring points. In all our experiments, we set this hyperparameter $\delta_v = 1$. We use the result from this clustering network to guide our geometric clustering. The results of applying the offset output by this network are depicted in Fig. 6.3(d).

6.3.1 Geometric Constrained Clustering

In this section, we aim to find a set of clusters $\mathbf{w} = \{\omega_1, \omega_2, \dots, \omega_{\hat{C}}\}$ that represent a set of objects in the volumetric space, where \hat{C} is the final estimate of the unknown number of objects. Fig. 6.3(d) shows that the clusters preserve the cylindrical shape of their original objects. We follow the convention used in the connected tube marked point process [53] to describe objects with marks describing their properties. We use cylindrical clusters with the marks $m_c = (\mu_c, r_c, l_c, \theta_c, \phi_c)$ where μ_c denotes the center coordinate of the cylinder, $r_c \in [r_{min}, r_{max}]$ denotes the cylinder radius and r_{min}, r_{max} denote the minimum and maximum possible radii respectively. The parameter $l_c \in [l_{min}, l_{max}]$ denotes the fiber length, and l_{min}, l_{max} denote the minimum and maximum possible fiber lengths. The parameters $\theta_c \in [\theta_{min}, \theta_{max}]$ and $\phi_c \in [\phi_{min}, \phi_{max}]$ denote the cylinder orientation with respect to the positive xy axis and with respect to the positive z axis, with their respective possible minimum and maximum values.

Birthmap Computation

We use the centroid regression vectors v_i to shift all the foreground pixels $s_i \in S_f$ to their estimated instance center o_i , i.e. $o_i = s_i - v_i$, hence generating a set of offset pixels O_f . The offset pixels tend to be concentrated around the fibers' center, thus, we generate a heat map near the true instance center by counting the number of offset pixels that lie in each coordinate. This heat map is shown in Fig. 6.4(b) and is used as a birthmap for further computation.

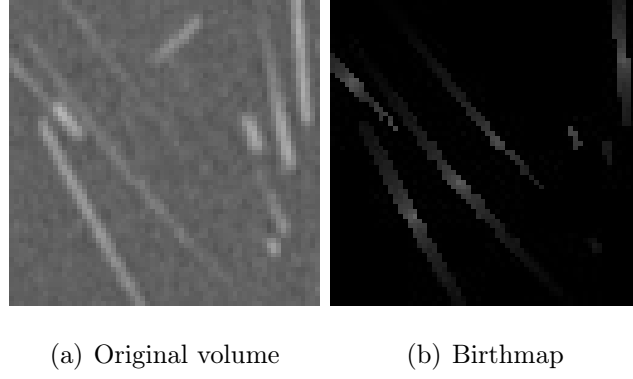


Fig. 6.4. Cropped image and its birthmap.

Cluster Proposal

We pick the unlabeled offset pixels o_i sorted by the number of votes from the birthmap to propose a new estimated cluster center $\hat{\mu}_c$. We gather the offset pixels that are within a distance eps of the estimated center, $\hat{w}_c = \{o_i \in O_f : ||o_i - \hat{\mu}_c|| < eps\}$. These pixels contain their original coordinate information and hence generate a cloud of points that can be fitted with a cylinder. We use a GPU adaptation of [61] to estimate the marks \hat{m}_c : radius \hat{r}_c , length \hat{l}_c , and orientation $\hat{\theta}_c, \hat{\phi}_c$ of the cloud of points to propose a cylindrical cluster. Finally we add to the proposed cluster set \hat{w}_c all the offset pixels that lie inside the proposed cylinder. Fig. 6.5(b) shows a sample cluster proposal result.

Cluster Evaluation

Finally, we evaluate the proposed cylinder in the original volume space by calculating the volume percent of segmented pixels covered by the cylinder V_d and the volume percent of overlap with other cylinders V_p . We set two threshold parameters for volume percent of segmented pixels T_d and for volume percent of overlap T_p . If $V_d > T_d$ and $V_p < T_p$ we accept the cylinder, otherwise we leave the set of pixels as unlabeled. We choose $T_d = 0.5$ to keep consistent with the IoU metrics explained

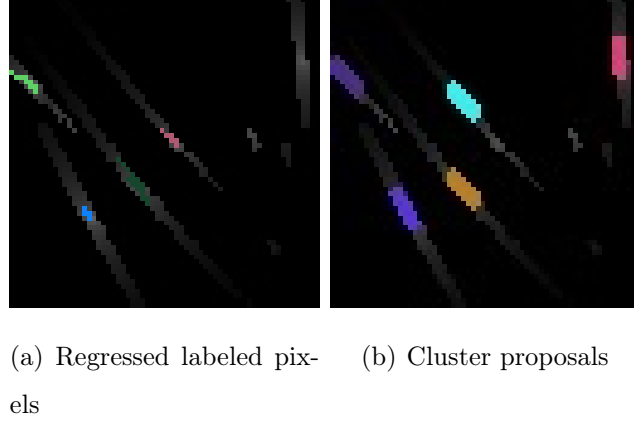


Fig. 6.5. Sample cluster proposal and ground truth labels.

in Section 6.4 and $T_p = 0.2$ to allow the proposed cluster 20% of overlap with other clusters. The volume percent V_p ensures the proposed object fits the semantic segmentation and the overlap percent V_p prevents from proposing multiple clusters for the same fiber. These evaluations were inspired by the success of the overlap prior from the marked point process model [53]. The detailed procedure of the clustering algorithm is listed in Algorithm 4.

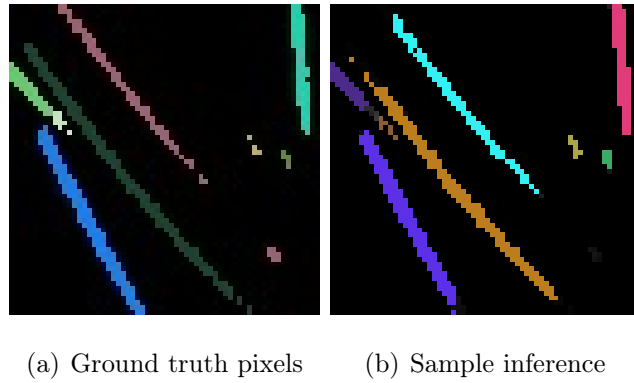


Fig. 6.6. Ground truth labels and a sample inference.

Algorithm 4 Gemetric Constrained Clustering

```

1: procedure OBTAIN SET OF CLUSTERS  $\mathbf{w} = \{\omega_1, \omega_2, \dots, \omega_{\hat{C}}\}$ 
2:   Initialization:
3:      $\mathbf{w} = \{\}$ 
4:      $S_f \leftarrow$  original foreground pixels coordinates
5:      $O_f \leftarrow$  center regressed foreground pixels coordinates
6:     for each pixel  $s$  in birthmap do
7:       if  $s \in w_i : w_i \in \mathbf{w}$  then
8:         continue (pixel has a label)
9:       else
10:        Cluster Proposal
11:         $\hat{\mu}_k \leftarrow s$ 
12:         $\hat{w}_k \leftarrow \{s_k \subset O_f : \|s_f - \hat{\mu}_k\| < eps\}$ 
13:         $\hat{m}_k \leftarrow \text{marks}(\hat{w}_k, S_f, O_f)$ 
14:        Evaluate Cluster
15:         $V_d \leftarrow$  percent of segmented pixels inside  $\hat{m}_k$ 
16:         $V_p \leftarrow$  percent of overlapping pixels of  $\hat{m}_k$  and other cylinders
17:        if  $V_d > T_d$  and  $V_p < T_p$  then
18:           $\mathbf{w} \leftarrow \mathbf{w} \cup w_k$ 
19:           $k \leftarrow k + 1$ 
20:        end if
21:      end if
22:    end for
23: end procedure

```

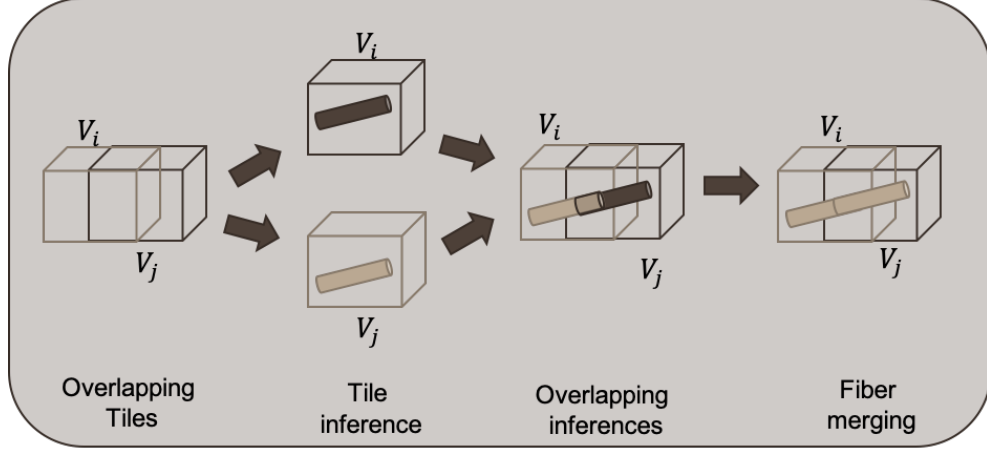


Fig. 6.7. Merging Procedure

6.3.2 Volume Tiling and Merging

We tile the initial volumetric data into overlapping subvolumes of size $64 \times 64 \times 64$ in order to circumvent large memory demands from some datasets. We detect fibers locally and we implement a merging procedure to detect objects in the full volume. We extend the approach presented by Konopczynski [13], which consists of merging nearby fibers in overlapping tiles. However, this procedure does not consider the fiber properties and can merge nearby local fibers even if they are perpendicular to each other. We implement an additional constraint based on the orientation information provided by the cluster properties. If two fibers in overlapping tiles are nearby, we merge them if the angle between them is less than a threshold $T_{\Delta\theta, \Delta\phi}$. The value of this threshold is set based on prior knowledge of the system, such as the fiber expected curvature. A sample representation of this procedure is denoted in Fig. 6.7.

6.4 Experiments

We use two metrics to evaluate the instance segmentation results. First, we follow the guideline provided in [62] and use the Adjusted Rand Index [63] (R_a) metric

for evaluation. This metric evaluates the agreement between two sets of clusters $C = \{c_1, c_2, \dots, c_k\}$, and $C' = \{c'_1, c'_2, \dots, c'_l\}$ and ranges from 0 to 1 depending on the degree of similarity between C and C' . The R_a criteria is given by:

$$R_a(C, C') = \frac{\sum_i^k \sum_j^l \binom{m_{ij}}{2} - t_3}{\frac{1}{2}(t_1 + t_2) - t_3} \quad (6.5)$$

Where $m_{ij} = |c_i \cap c'_j|$, $t_1 = \sum_i^k \binom{|c_i|}{2}$, $t_2 = \sum_j^l \binom{|c'_j|}{2}$, and $t_3 = \frac{2t_1 t_2}{n(n-1)}$.

In addition, we use the intersection over union evaluation (IoU) where we say a fiber is detected correctly if its IoU with the equivalent ground truth fiber is greater than 0.50. We describe an equivalent ground truth fiber as the object in the ground data that overlaps the most with the estimated fiber. We label broken fibers whose IoU is less than 0.5 as false positives (fiber segments). We label as false negative all the ground truth fibers that were not captured by the initial segmentation or that had an IoU less than 0.5.

We average the results obtained from multiple tiles of size $64 \times 64 \times 64$ voxels and compared our method with our implementation of the method proposed by Kendall et. al [60], with the method proposed in Chapter 5, and with our method but using center regression and DBSCAN for clustering. We do not use the angle criteria for fiber merging when we use the DBSCAN algorithm since we do not have fiber orientation information.

We implemented all the networks and trained with the same training parameters as described in Section 6.4. We followed the parameters denoted in each work and chose $\delta_v = 0.2$ and $eps = 0.4$ for the embedding learning parameters shown in Chapter 5.

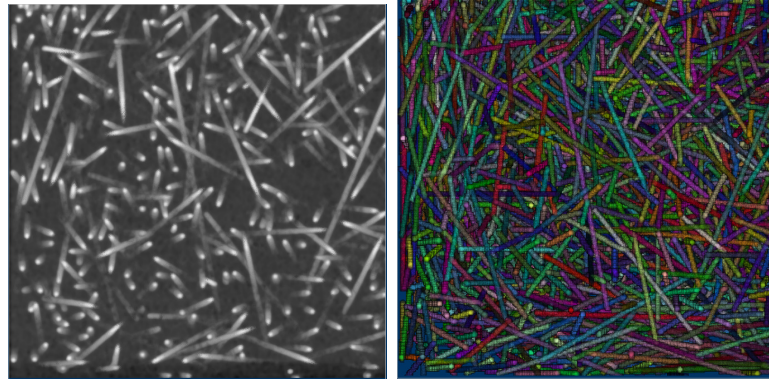
6.4.1 Training Information: Synthetic Fibers

This dataset was generated from a computational model by Konopczynski et al. [62] and it is a simulation of short glass fibers embedded in a reinforced polymer. The fibers have a radius of $6.5 \mu\text{m}$ and a mean length of $500 \mu\text{m}$ with a deviation of

100 μm and are oriented in arbitrary directions. We used the dataset named “2016-S-HR-5.35p” depicted in Fig. 6.8 for training the network and the dataset named “2016-S-HR-5.38p” for testing.

We trained all the networks with the Adam [64] optimizer with a learning rate of $\text{lr}=0.001$ over 2000 epochs. We normalized the data to unit variance and zero mean and we trained cropping subvolumes of sizes $64 \times 64 \times 64$ cropped at uniformly random locations of the full volume, and we performed volume rotations for data augmentation. All the models were trained on an NVIDIA-Titan RTX GPU with 25GB of memory with a training time of approximately two days.

Fig. 6.9 shows the average results over all tiles when varying the eps parameters for different methods and Table 6.1 shows the best scores obtained for each method over the tested eps values. The implementation that performs only centroid regression with DBSCAN achieved the best scores with 0.993 f1 score and 0.767 R_a score. Our approach obtained 0.973 for the object-wise f1 score and 0.719 for the R_a score.



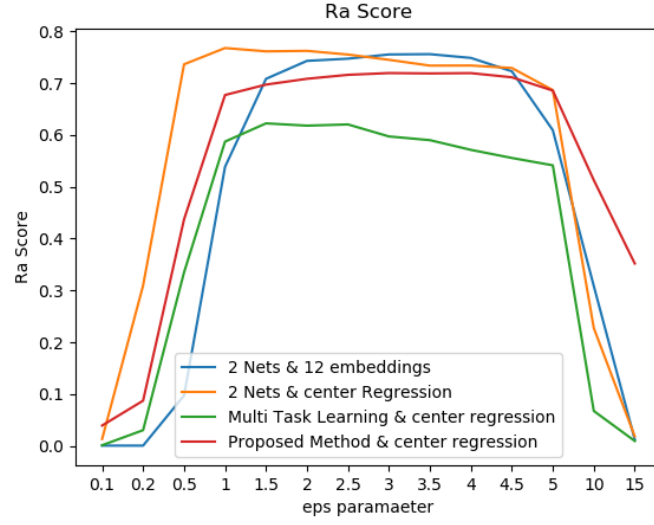
(a) Synthetic data

(b) Synthetic labels

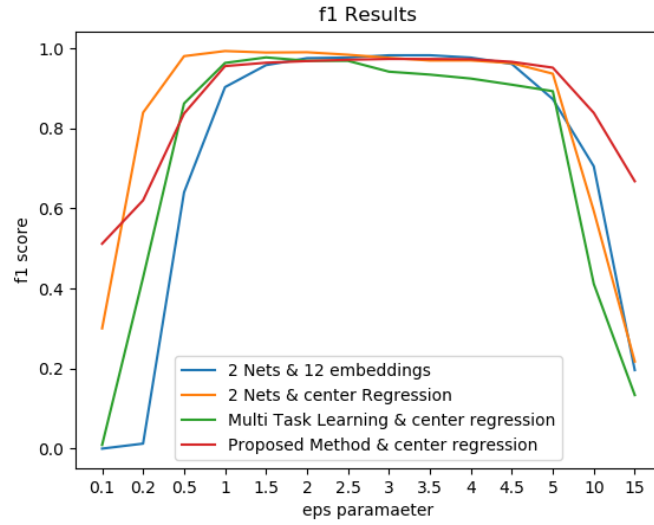
Fig. 6.8. Training dataset: synthetic fibers generated by Konopczynski [62]

Table 6.1.
f1 and R_a results for synthetic dataset

Method	R_a	f1
Embedding Learning, Chapter 5	0.756	0.983
Multitask Learning [60]	0.622	0.977
Centroid Regression	0.767	0.993
Proposed	0.719	0.973



(a) Ra score



(b) f1 score

Fig. 6.9. Evaluation of average tile scores vs eps parameter in synthetic data. For the proposed method, we use $\text{eps} \equiv r_{min}$

6.4.2 Low Resolution SFRP Dataset

This dataset was provided by Konopczynski et al. [62] and it represents two samples of a commercial polybutylene terephthalate PBT reinforced with short glass

fibers. The fibers have a diameter of 10-14 μm and 1.1mm of length. The samples were imaged with X-ray CT with an isotropic resolution of 3.9 μm and the volumetric dimensions of $200 \times 260 \times 260$ voxels. We used the dataset named “Real MR2” for testing our approach. It is worth noting that the labeled fibers were pre-segmented, hand labeled, and refined with the watershed algorithm. Therefore, the segmentation labels and instance segmentation labels can be biased to the pre-segmentation algorithm followed by the watershed algorithm. The first column of Fig. 6.11 shows a tiled cross section of sizes 64×64 pixels and each row shows results for different approaches. The third and fourth row show that the DBSCAN approaches encounter difficulties clustering fibers. For example, Fig. 6.11(c) shows numerous groups of white pixels (unlabeled pixels) that are near two different fibers. Similarly, Fig. 6.11(d) shows merging of that are close to each other. These issues are translated in the volumetric results as shown in Figs. 6.11(h) and Figs. 6.11(i) where fibers are artificially broken or nearby fibers are merged. Our method, shown in Fig. 6.11(e) can discriminate between nearby fibers and has also the ability to segment curved fibers when using volume merging as shown in Fig. 6.11(j).

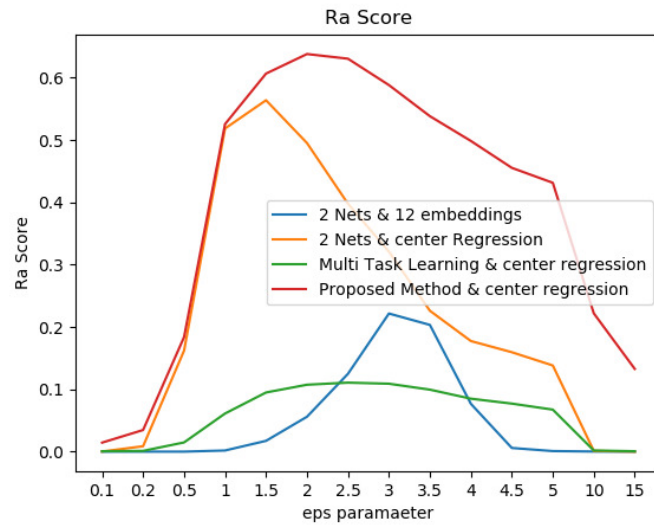
The numerical evaluation in Table 6.2 shows that our method obtains significantly better scores for both metrics with an R_a value of 0.638 and a $f1$ score of 0.917. Fig. 6.11 shows that we obtain the highest score when the eps parameter is equivalent to the true fiber radius. However, Fig. 6.10 shows that our method is more robust to parameter variations than the rest of proposed approaches.

6.4.3 High resolution SFRP: Polypropylene Matrix

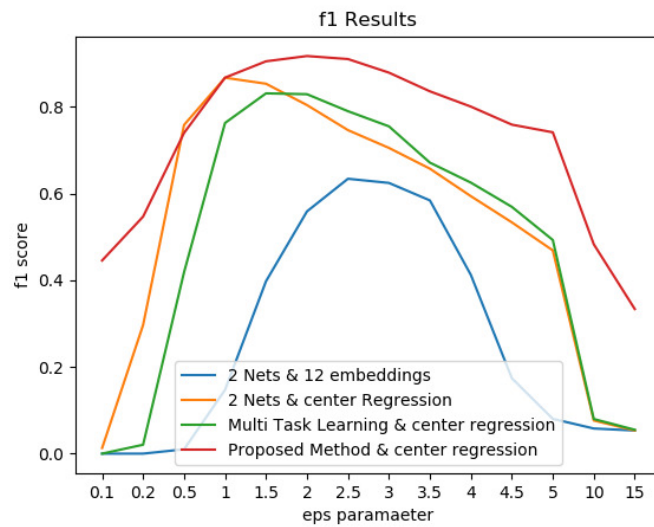
This dataset was provided by the ACME Laboratory at Purdue University. The sample consists of a polypropylene material reinforced with glass fibers, imaged at 1.3 μm resolution. The reconstructed volume has dimensions $2300 \times 2300 \times 1300$ voxels and we used a sub-volume comprised of $950 \times 950 \times 150$ voxels (shown in Fig.

Table 6.2.
f1 and R_a results for low resolution SFRP dataset.

Method	R_a	f1
Embedding Learning, Chapter 5	0.222	0.634
Multitask Learning [60]	0.111	0.831
Centroid Regression	0.563	0.831
Proposed	0.638	0.917



(a) Ra scores



(b) f1 scores

Fig. 6.10. Evaluation of mean tile scores vs eps parameter in Low Resolution SFRP Dataset.

6.13(a)). The ground truth is from the results of Agyei et al. [15] and is shown in Fig. 6.13(b).

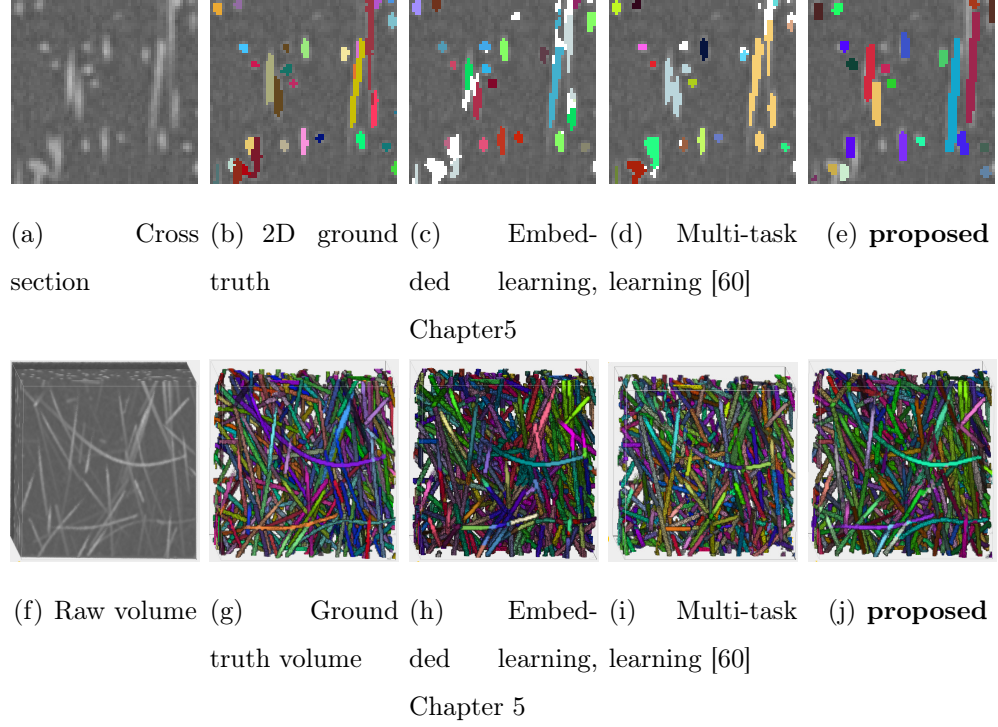


Fig. 6.11. Low Resolution SFRP Dataset. Each color represents a different fiber instance and white pixels represent unlabeled pixels.

The first row of Fig. 6.13 shows the XY-axis cross section no the testing sample, the second row shows a cropped volume of size $256 \times 256 \times 150$, and the third row shows a YZ-axis cross section of the cropped volume. This dataset is significantly different from the training dataset, and hence both the embedding learning and multi-task method merge multiple nearby fibers. These results can be noticed in Figs. 6.13(h), 6.13(m). and Figs. , 6.13(i) and 6.13(n). Our results, shown in Figs. 6.13(j) and 6.13(o) show that our method does not merge nearby fibers thanks to the regularization imposed by cylindrical geometry regularization. We should also point out that our method detects fibers that were not detected in the ground truth dataset. Our method also shows promising results for the merged volume in Fig. 6.13(e) compared to the merged results shown in Figs.6.13(c) and 6.13(d). Our method shows

Table 6.3.
f1 and R_a results for high resolution SFRP: Polypropylene Matrix.

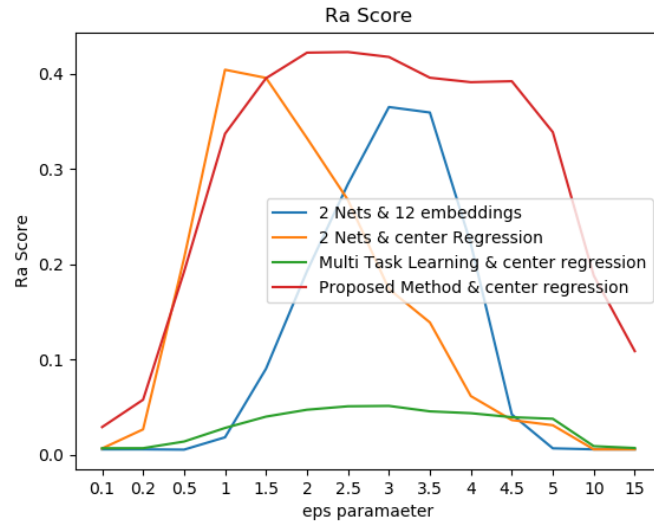
Method	R_a	f1
Embedding Learning, Chapter 5	0.365	0.604
Multitask Learning [60]	0.051	0.733
Centroid Regression	0.134	0.767
Proposed	0.422	0.855

improvement over the provided ground truth in detecting long fibers shown in Fig. 6.13(b).

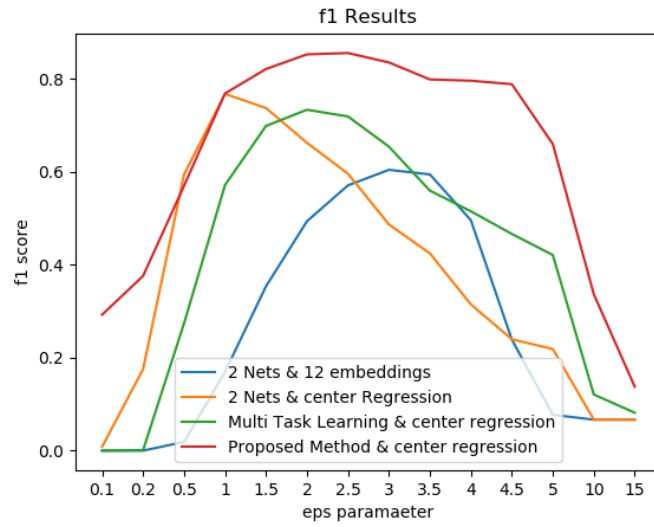
Table 6.3 shows the R_a and f1 results for each approach. The R_a score is relatively low because it also depends on the segmentation and our method has an over segmentation compared to the provided ground truth, and also because the provided ground truth is not perfect. Fig. 6.13 shows several visual examples where we believe our approach captured fibers that were not captured in the ground truth. Fig. 6.12 shows the scores across varying parameters denoting our proposed approach is more robust to parameter variation.

6.4.4 Preceramic Polymer in Unidirectional Fiber Beds

This dataset was obtained from Globus [65, 66] and has been used in [66–68]. The aim of the experiment is to characterize the microstructure evolution during the impregnation of a polymer into fiber beds. The volume represents a reconstructed X-ray CT of size $2000 \times 2000 \times 2000$ voxels and contains several time instances obtained during the polymer impregnation. The specimen consists of SiC fibers with an average fiber radius of $6.4 \mu m$ and was imaged at high resolution to yield 7-10 pixels of fiber radius. We used a cropped volume of $500 \times 500 \times 500$ pixels and enforced a minimum



(a) Ra Scores



(b) f1 Scores

Fig. 6.12. High resolution fiber reinforced polymers: evaluation Scores vs eps parameter

fiber radius of 7 pixels. Fig. 6.14(a) shows a cross section and Fig. 6.14(b) shows a cropped pre-segmented volume. The second row shows the results from using an

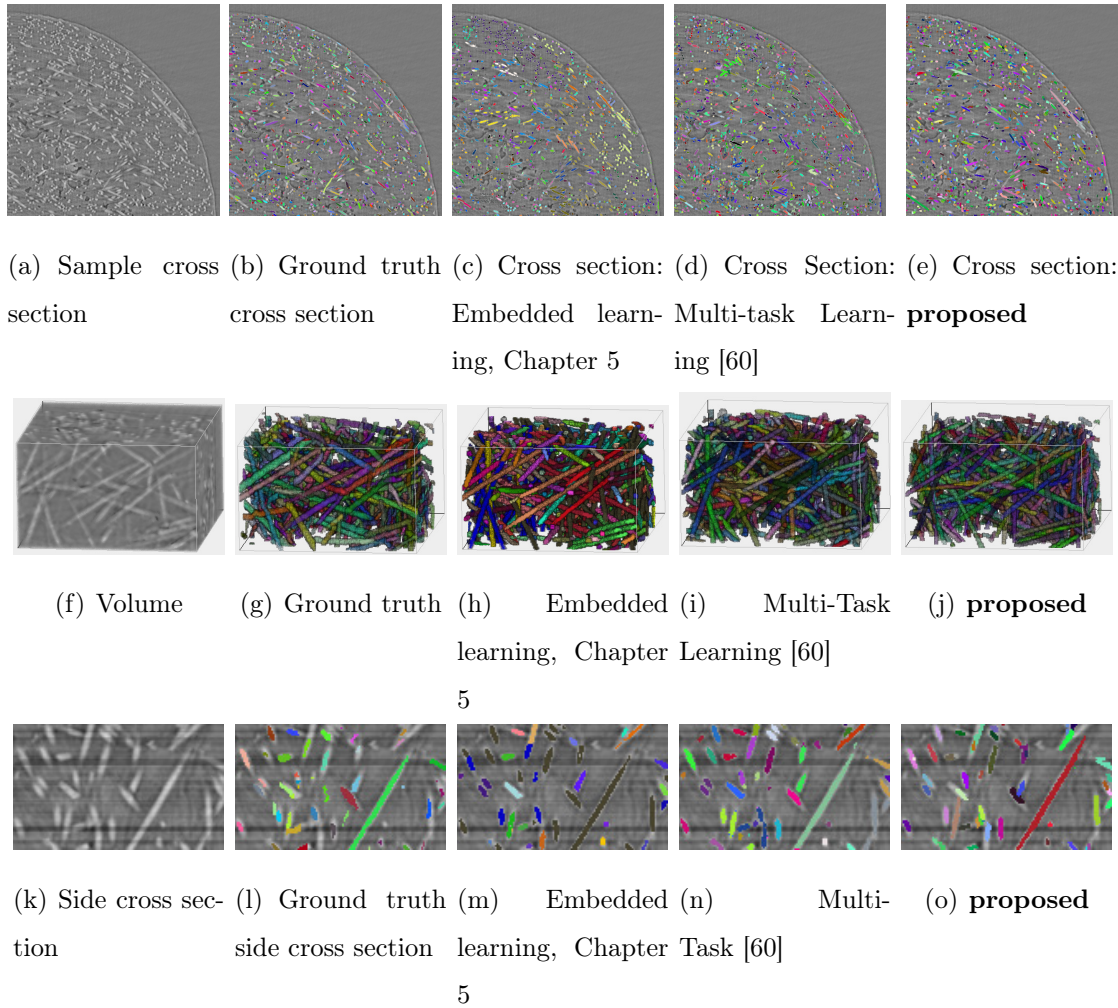


Fig. 6.13. SFRP: Polypropylene Matrix. Each color represents a different instance of a fiber

embedded learning approach and the third row shows the results of using centroid regression but with DBSCAN. Our proposed method can segment a large number of fibers although it is constrained by the quality of the segmentation.

6.5 Chapter Summary

We presented a neural network approach to detect fibers in large volumetric datasets by first segmenting and then regressing a vector pointing from each fore-

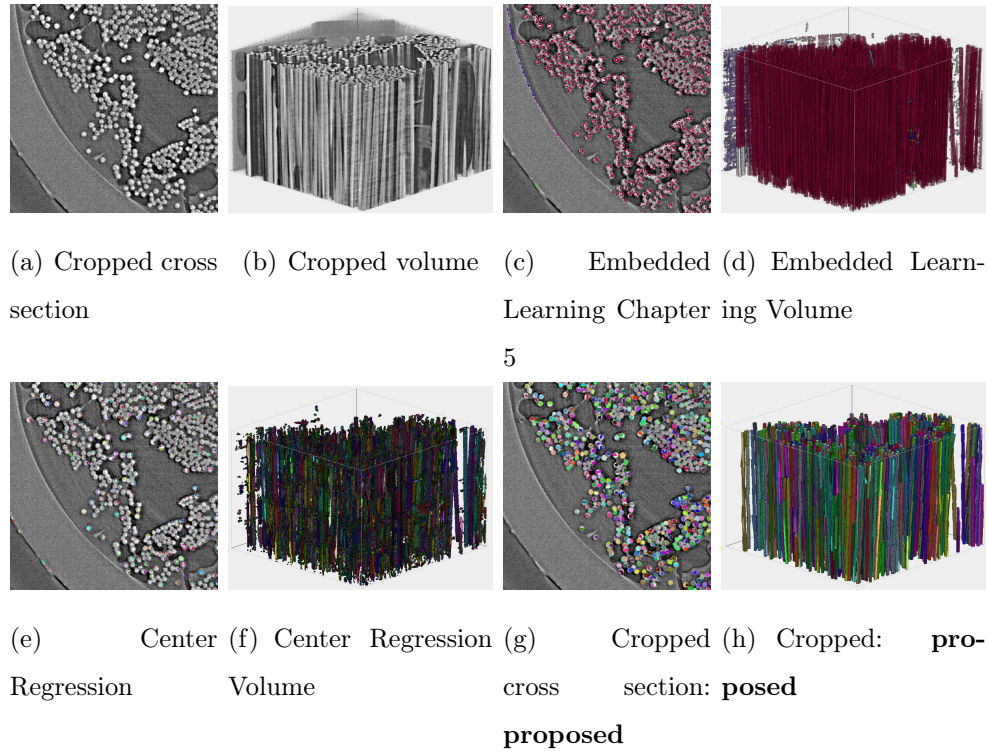


Fig. 6.14. Object detection in unidirectional fiber beds.

ground pixel to its instance centroid. Our approach showed robustness across several datasets thanks to the geometric constrained clustering and also allows to constrain the proposed objects with prior image knowledge. Unlike common instance segmentation and clustering techniques, we proposed a clustering technique that relies on finding objects of specific shapes. We showed that our approach outperforms 3D object detection in several datasets and we believe that it can contribute to improve fiber-reinforced materials characterization.

In this chapter, our results demonstrated that neural networks that learn to regress objects' centers adapt better to different datasets than their training dataset. For example, Tables 6.2 and 6.3 show results of multiple approaches trained on synthetic data and tested in real datasets. The methods that perform center regression obtain the best f1-performance among the possible approaches.

In addition, in this chapter we explored the implementation of an encoder-multiple decoder network, such as a 3D extension of the multi-task learning approach proposed by Kendall [60]. This method obtained competitive results when tested on synthetic data; however, its performance decreased significantly when testing the network in a dataset composed of real fibers. In this work, we showed that while a multi-task learning approach presents an elegant solution to a two stage semantic/instance segmentation; however, this method is outperform by implementing two different networks to perform semantic and instance segmentation individually.

Finally, our results showed that when we used a regularized clustering across different parameters, we obtained the highest scores when the *eps* parameter resembled the fiber's real radius. For example, Figs. 6.10 and 6.12 show the R_a and $f1$ scores at different parameter choices. Our method obtained the highest performance when $eps \in [2, 3]$ in Fig. 6.10, this is translated to the real fiber radius $r \in [2.5, 3.5]$ voxels. Similarly, Fig. 6.12 shows that our clustering obtains the best results when $eps = 2$ while the mean radius of the fibers (given our resolution) is $r = 2.08$.

7. SUMMARY AND FUTURE WORK

7.1 Summary

In this report we explored methods to incorporate the boundary energy to the MPP framework. In chapter 2 we provided an overview of the theory and samplers used in our method.

In chapter 3, we talked about the MPP with parametric active contours and balloon force. The inclusion of a parametric model helped to capture multiple shapes of irregular boundaries. The balloon method contributed to detect objects with low contrast such as large voids, and the addition of a pre-segmentation energy contributed to keep only the objects of interest.

In chapter 4, we explored the addition of the level sets framework to the MPP. The results show an improvement over the original MPP-AC method. This method handles better changes in topology is less sensitive to initialization. The addition of the method into the MPP-LS framework contributed to sample objects and also keep only the ones of interest. In addition, we proposed the combination of level sets results to simulate the MBD sampler in order to converge faster to a solution.

In chapter 5, we explored a deep learning approach based on learning embeddings and performing clustering. This method denoted a significant speed up in volume object detection over traditional model-based approaches.

Finally, in chapter 6, we extended our approach from chapter 5 by performing centroid regression and using this information to propose objects. Our approach proved robustness across multiple datasets when compared to other embedding learning and center regression approaches.

7.1.1 Model Based Methods vs Deep Learning

In this report, we proposed both model-based approaches and deep learning methods to detect fibers and porosity in material images. Our model-based procedures incorporate real world variables into the segmentation such as the contour smoothness and rigidity. For example, setting the parameter $\beta_{Edge} = 100$ in the active contour model in Chapter 3 enforces very smooth boundaries while setting $\beta_{Edge} = 1$ allows sharp edges. These variables can be tuned to modify the segmentation to match desired physical properties. Additionally, model based approaches allow to propose energy functions to match hand desired features. For example, E_{edge} and E_{region} in Chapter 4 favor void configurations around edges and dark(or bright) regions, and E_{shape} enforces an ellipsoidal shapes. Finally, model based methods and probabilistic approaches allow to simulate systems based on a pre-defined probability function. For example, we used birth and death simulations in Chapters 3 and 4.

On the other hand, deep learning techniques learn patterns in the data and adjust their parameters to perform classification or regression. While common deep learning approaches do not model physical properties, they often match or beat the segmentation results obtained from model-based methods. For instance, in this report we show a deep embedding learning matched the f1 scores from the connected tube MPP to segment fibers in a synthetic dataset; however the deep embedding learning approach took 1 hour while the MPP required of 10 hours. Similarly, in Chapter 5, we show that networks trained on noisy MPP results were able to correct for noise and obtained better results than their training data. Similarly, in Chapter 5, we show that our proposed deep embedded learning obtained the similar results to leading fiber and void segmentation techniques, but our results required of 30 hours versus 125 hours required by other methods.

7.2 Future Work: Towards Model-Based Deep Learning

In this report we explored the combination of the MPP framework and the active contours approach. However, MCMC techniques require extensive sampling which makes this technique unpractical for very large datasets. Perhaps this computational cost is one of the reasons the MPP has not been widely adopted by industry and researchers. Similarly, deep learning methods obtain competitive results but requires large amounts of training data. Particularly 3D microscopy labels are expensive to obtain and can be ambiguous due to imaging and reconstruction noise. It would be an interesting approach to explore further into the combination of MPP-based methods with deep learning techniques.

During the last two decades, multiple MCMC samplers have borrowed concepts from the data-driven MCMC (DDMCMC) proposed by Tu et al. [29]. This algorithm considers classic image processing algorithms such as active contours, edge detection, and clustering algorithms, to guide the RJMCMC samplers. He used parzen windows $G(\theta - \theta_i)$ with pre-computed parameters θ_i and pre-computed weights to propose the next RJMCMC steps. Some MPP techniques have followed this approach and improved their samplers [69] [27]. Similarly, Jampani et al. built on this idea to create the informed sampler [70] and Kortylewski et al. proposed the informed samplers using neural networks [71].

Chapter 6 sets a base to combine a Tube MPP with deep learning. Particularly, we used deep centroid regression to create a birthmap for MPP objects, and to propose objects with a mark that resembles the MPP mark. Future work could involve using the MPP with an initial classic MCMC sampler to train a center regression network, and inform the MCMC sampler with the network results as the network learns to regress centroid pixels.

REFERENCES

REFERENCES

- [1] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, “Annotated high-throughput microscopy image sets for validation,” *Nature Methods*, vol. 9, no. 7, p. 637, 2012. [Online]. Available: <https://doi.org/10.1038/nmeth.2083>
- [2] I. Hanhan and M. D. Sangid, “ModLayer: A MATLAB GUI Drawing Segmentation Tool for Visualizing and Classifying 3D Data,” *Integrating Materials and Manufacturing Innovation*, vol. 8, no. 4, pp. 468–475, 2019. [Online]. Available: <https://doi.org/10.1007/s40192-019-00160-5>
- [3] J. Simmons, C. Przybyla, S. Bricker, D. W. Kim, and M. Comer, “Physics of MRF regularization for segmentation of materials microstructure images,” *2014 IEEE International Conference on Image Processing, ICIP 2014*, pp. 4882–4886, 2014.
- [4] A. Kumar, L. Nguyen, M. Degraef, and V. Sundararaghavan, “A Markov random field approach for microstructure synthesis,” *Modelling and Simulation in Materials Science and Engineering*, vol. 24, no. 3, pp. 0–16, 2016.
- [5] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9351, pp. 234–241, 2015.
- [6] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *CoRR*, vol. abs/1411.4038, 2014. [Online]. Available: <http://arxiv.org/abs/1411.4038>
- [7] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, “Feature pyramid networks for object detection,” *CoRR*, vol. abs/1612.03144, 2016. [Online]. Available: <http://arxiv.org/abs/1612.03144>
- [8] S. Jégou, M. Drozdal, D. Vázquez, A. Romero, and Y. Bengio, “The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation,” *CoRR*, vol. abs/1611.09326, 2016. [Online]. Available: <http://arxiv.org/abs/1611.09326>
- [9] L. Chen, G. Papandreou, F. Schroff, and H. Adam, “Rethinking atrous convolution for semantic image segmentation,” *CoRR*, vol. abs/1706.05587, 2017. [Online]. Available: <http://arxiv.org/abs/1706.05587>
- [10] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 91–99. [Online]. Available: <http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks.pdf>

- [11] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.
- [12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
- [13] T. K. Konopczynski, T. Kröger, L. Zheng, and J. Hesser, "Instance segmentation of fibers from low resolution ct scans via 3d deep embedding learning," in *BMVC*, 2018.
- [14] R. F. Agyei, B. Sharma, and M. Sangid, "Investigating Sub-surface Microstructure in Fiber Reinforced Polymer Composites via X-Ray Tomography Characterization," *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, no. January, pp. 1–6, 2016. [Online]. Available: <http://arc.aiaa.org/doi/10.2514/6.2016-0409>
- [15] R. F. Agyei and M. D. Sangid, "A supervised iterative approach to 3D microstructure reconstruction from acquired tomographic data of heterogeneous fibrous systems," *Composite Structures*, vol. 206, no. January, pp. 234–246, 2018. [Online]. Available: <https://doi.org/10.1016/j.compstruct.2018.08.029>
- [16] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, pp. 721–741, Nov 1984.
- [17] A. J. Baddeley and M. N. Van Lieshout, "Stochastic geometry models in high-level vision," *Journal of Applied Statistics*, vol. 20, no. 5-6, pp. 231–256, 1993.
- [18] P. J. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 12 1995. [Online]. Available: <https://doi.org/10.1093/biomet/82.4.711>
- [19] X. Descombes, R. Minlos, and E. Zhizhina, "Object extraction using a stochastic birth-and-death dynamics in continuum," *Journal of Mathematical Imaging and Vision*, vol. 33, no. 3, pp. 347–359, Mar 2009. [Online]. Available: <https://doi.org/10.1007/s10851-008-0117-y>
- [20] R. S. Stoica, X. Descombes, and J. Zerubia, "A Gibbs Point Process for Road Extraction from Remotely Sensed Images," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 121–136, 2004. [Online]. Available: <http://www.springerlink.com/openurl.asp?id=doi:10.1023/B:VISI.0000013086.45688.5d>
- [21] C. Lacoste, X. Descombes, and J. Zerubia, "Point processes for unsupervised line network extraction in remote sensing," *Pami*, vol. 27, no. 10, pp. 1568–1579, 2005.
- [22] X. Descombes and J. Zerubia, "Marked Point Process in Image Analysis," *IEEE Signal Processing Magazine*, vol. 19, no. 5, pp. 77–84, 2002. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1028354
- [23] H. Zhao and M. Comer, "A hybrid markov random field/marked point process model for analysis of materials images," *IEEE Transactions on Computational Imaging*, vol. 2, no. 4, pp. 395–407, Dec 2016.

- [24] D. W. Kim, C. Aguilar, H. Zhao, and M. L. Comer, "Narrow Gap Detection in Microscope Images Using Marked Point Process Modeling," *IEEE Transactions on Image Processing*, vol. PP, no. c, pp. 1–1, 2019.
- [25] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8759–8768, 2018.
- [26] J. Uhrig, E. Rehder, B. Fröhlich, U. Franke, and T. Brox, "Box2pix: Single-shot instance segmentation by assigning pixels to object boxes," in *IEEE Intelligent Vehicles Symposium (IV)*, 2018. [Online]. Available: <http://lmb.informatik.uni-freiburg.de/Publications/2018/UB18>
- [27] Y. Verdi, "Efficient Monte Carlo Sampler for Detecting Parametric Objects in Large Scenes," *ECCV*, pp. 539–552, 2012.
- [28] P. Craciun, "Stochastic Geometry for Automatic Multiple Object Detection and Tracking in Remotely Sensed High Resolution Image Sequences," Ph.D. dissertation, Universite Nice Sophia Antipolis, 2016.
- [29] Z. Tu and S.-C. Zhu, "Image segmentation by data-driven Markov chain Monte Carlo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 657–673, 2002. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1000239>
- [30] X. Descombes, "Multiple objects detection in biological images using a marked point process framework," *Methods*, 2016. [Online]. Available: <https://hal.inria.fr/hal-01383165>
- [31] M. Kulikova, I. Jermyn, X. Descombes, E. Zhizhina, and J. Zerubia, "Extraction of arbitrarily-shaped objects using stochastic multiple birth-and-death dynamics and active contours," *Electronic Imaging*, vol. 7533, 2010. [Online]. Available: <https://doi.org/10.1117/12.839191>
- [32] H. Zhao, "Combining Markov Random field and Marked Point Process for Microscopy Image Modeling," Ph.D. dissertation, Purdue University, 2016.
- [33] M. L. Comer and E. J. Delp, "The EM/MPM algorithm for segmentation of textured images: Analysis and further experimental results," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1731–1744, 2000.
- [34] M. Kulikova, I. Jermyn, X. Descombes, J. Zerubia, and E. Zhizhina, "A marked point process model with strong prior shape information for the extraction of multiple, arbitrarily-shaped objects," *Proceedings - 5th International Conference on Signal Image Technology and Internet Based Systems, SITIS 2009*, no. November 2016, pp. 180–186, 2010.
- [35] L. D. Cohen and I. Cohen, "Finite Element Methods for Active Contour Models and Balloons for 2D and 3D Images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. November, pp. 1131–1147, 1993.
- [36] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *INTERNATIONAL JOURNAL OF COMPUTER VISION*, vol. 1, no. 4, pp. 321–331, 1988.

- [37] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.-Y. Tinevez, D. J. White, V. Hartenstein, K. Eliceiri, P. Tomancak, and A. Cardona, *Fiji: an open-source platform for biological-image analysis*, 06 2012. [Online]. Available: <http://dx.doi.org/10.1038/nmeth.2019>
- [38] B. A. Hamilton, “Kaggle data science bowl: Find the nuclei in divergent images to advance medical discovery,” 2018, data retrieved from Kaggle Website, <https://www.kaggle.com/c/data-science-bowl-2018>.
- [39] Z. Yan, B. J. Matuszewski, L. K. Shark, and C. J. Moore, “Medical image segmentation using new hybrid level-set method,” *Proceedings - 5th International Conference BioMedical Visualization, Information Visualization in Medical and Biomedical Informatics, MediVis 2008*, no. 1, pp. 71–76, 2008.
- [40] V. Caselles, R. Kimmel, and G. Sapiro, “Geodesic active contours,” *International Journal of Computer Vision*, vol. 22, no. 1, pp. 61–79, Feb 1997. [Online]. Available: <https://doi.org/10.1023/A:1007979827043>
- [41] T. F. Chan and L. A. Vese, “Active contours without edges,” *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.
- [42] K. Zhang, H. Song, and L. Zhang, “Active contours driven by local image fitting energy,” *Pattern Recognition*, vol. 43, no. 4, pp. 1199–1206, 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2009.10.010>
- [43] P. Marquez-Neila, L. Baumela, and L. Alvarez, “A morphological approach to curvature-based evolution of curves and surfaces,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 2–17, 2014.
- [44] A. C. Fan, J. W. Fisher, W. M. Wells, J. J. Levitt, and A. S. Willsky, “MCMC Curve Sampling for Image Segmentation,” *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2007*, pp. 477–485, 2007.
- [45] M. S. Nosrati and G. Hamarneh, “Segmentation of overlapping cervical cells: A variational method with star-shape prior,” *Proceedings - International Symposium on Biomedical Imaging*, vol. 2015-July, pp. 186–189, 2015.
- [46] B. C. Lucas, M. Kazhdan, and R. H. Taylor, “Multi-Object Geodesic Active Contours (MOGAC),” *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 404–412, 2012.
- [47] T. Chan and Wei Zhu, “Level set based shape prior segmentation,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 2, June 2005, pp. 1164–1170 vol. 2.
- [48] O. Gal, “fit ellipse: Matlab software for finding best fit, version 1.0,” [https://www.mathworks.com/matlabcentral/fileexchange/3215-fit ellipse](https://www.mathworks.com/matlabcentral/fileexchange/3215-fit-ellipse), Oct. 2003.
- [49] C. Aguilar and M. Comer, “A Marked Point Process Model Incorporating Active Contours Boundary Energy,” *Electronic Imaging*, vol. 2018, no. 15, pp. 230–1–2304, 2018.

- [50] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher Discrimination Dictionary Learning for sparse representation," in *2011 International Conference on Computer Vision*, 2011, pp. 543–550. [Online]. Available: <http://ieeexplore.ieee.org/document/6126286/>
- [51] S. Huh, D. F. E. Ker, R. Bise, M. Chen, and T. Kanade, "Automated mitosis detection of stem cell populations in phase-contrast microscopy images," *IEEE Transactions on Medical Imaging*, vol. 30, no. 3, pp. 586–596, March 2011.
- [52] X. Huang, D. Wen, Y. Zhao, Q. Wang, W. Zhou, and D. Deng, "Skeleton-based tracing of curved fibers from 3d x-ray microtomographic imaging," *Results in Physics*, vol. 6, pp. 170 – 177, 2016.
- [53] T. Li, M. Comer, and J. Zerubia, "A Connected-Tube MPP Model for Object Detection with Application to Materials and Remotely-Sensed Images," *Proceedings - International Conference on Image Processing, ICIP*, pp. 1323–1327, 2018.
- [54] A. Madra, N. E. Hajj, and M. Benzeggagh, "X-ray microtomography applications for quantitative and qualitative analysis of porosity in woven glass fiber reinforced thermoplastic," *Composites Science and Technology*, vol. 95, pp. 50 – 58, 2014.
- [55] C. Nageswaran, C. Bird, and R. Takahashi, "Phased array scanning of artificial and impact damage in carbon fibre reinforced plastic (cfrp)," *Insight*, vol. 48, pp. 155–159, 03 2006.
- [56] M. Bai and R. Urtasun, "Deep watershed transform for instance segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 2858–2866.
- [57] M. Hahsler, M. Piekenbrock, and D. Doran, "dbscan: Fast density-based clustering with R," *Journal of Statistical Software*, vol. 91, no. 1, pp. 1–30, 2019.
- [58] L. van der Maaten and G. Hinton, "Visualizing high-dimensional data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [59] D. Neven, B. D. Brabandere, M. Proesmans, and L. V. Gool, "Instance segmentation by jointly optimizing spatial embeddings and clustering bandwidth," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- [60] A. Kendall, Y. Gal, and R. Cipolla, "Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics arXiv : 1705 . 07115v1 [cs . CV] 19 May 2017," *Cvpr*, pp. 7482–7491, 2018. [Online]. Available: http://openaccess.thecvf.com/content_cvpr_2018/papers/Kendall_Multi-Task_Learning_Using_CVPR_2018_paper.pdf
- [61] X. Pan, "Fit a set of 3d points to a cylinder surface," mar 2017.
- [62] T. K. Konopczynski, J. Rathore, T. Kröger, L. Zheng, C. S. Garbe, S. Carmignato, and J. Hesser, "Reference setup for quantitative comparison of segmentation techniques for short glass fiber ct data," *ArXiv*, vol. abs/1901.01210, 2019.
- [63] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, no. 1, pp. 193–218, 1985. [Online]. Available: <https://EconPapers.repec.org/RePEc:spr:jclass:v:2:y:1985:i:1:p:193-218>

- [64] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014.
- [65] I. Foster, “Globus online: Accelerating and democratizing science through cloud-based services,” *Internet Computing, IEEE*, vol. 15, pp. 70 – 73, 07 2011.
- [66] B. Allen, J. Bresnahan, L. Childers, I. Foster, G. Kandaswamy, R. Kettimuthu, J. Kordas, M. Link, S. Martin, K. Pickett, and S. Tuecke, “Software as a service for data scientists,” *Commun. ACM*, vol. 55, no. 2, p. 81–88, Feb. 2012. [Online]. Available: <https://doi.org/10.1145/2076450.2076468>
- [67] N. M. Larson and F. W. Zok, “Insights from in-situ x-ray computed tomography during axial impregnation of unidirectional fiber beds,” *Composites. Part A, Applied Science and Manufacturing*, vol. 107, no. 5, 12 2017.
- [68] N. M. Larson, C. Cuellar, and F. W. Zok, “X-ray computed tomography of microstructure evolution during matrix impregnation and curing in unidirectional fiber beds,” *Composites Part A: Applied Science and Manufacturing*, vol. 117, pp. 243 – 259, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1359835X18304603>
- [69] F. Lafarge, G. Gimel, and X. Descombes, “by a Multimarked Point Process,” *Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1597–1609, 2010.
- [70] V. Jampani, S. Nowozin, M. Loper, and P. V. Gehler, “The informed sampler : A discriminative approach to Bayesian inference in generative computer vision models,” *Computer Vision and Image Understanding*, vol. 136, pp. 32–44, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.cviu.2015.03.002>
- [71] A. Kortylewski, A. Morel-forster, A. Wiecezorek, S. Parbhoo, V. Roth, and T. Vetter, “Informed MCMC with Bayesian Neural Networks for Facial Image Analysis,” *Third workshop on Bayesian Deep Learning*, no. 1, pp. 1–4, 2018.

APPENDIX

A. HYBRID LEVEL SETS WITH SHAPE PRIOR CURVE EVOLUTION

Given an image lattice $K \subset \mathbf{R}^d$, we let a contour be represented as a set of connected points along the zeroth level set $C_t = \{k | \phi(k) = 0\}$. We used a shape prior ϕ_o to modify the original version of the functional in [39] to:

$$E(\phi) = \alpha \int_{k \in K} (k - \mu) H(\phi) dk + \beta \int_{k \in K} g(k) |\nabla H(\phi)| dk + \gamma \int_{k \in K} (H(\phi) - H(\phi_o))^2 dk \quad (\text{A.1})$$

Where ϕ_o is negative inside a fixed shape and positive outside, and $H(\phi)$ denotes the Heaviside function:

$$H(\phi) = \begin{cases} 0, & \text{if } \phi < 0 \\ 1, & \text{if } \phi \geq 0 \end{cases}$$

and $g(K)$ is a function that is minimum at strong edges and is denoted as:

$$g(K) = \frac{1}{1 + \alpha_{edge} |\nabla(f_\sigma * K)|^2}$$

Where f_σ is a Gaussian smoothing filter with parameter σ and α_{edge} affects the slope. The associated PDE for this functional is given by:

$$\phi_t = \delta(\phi) \left[\alpha(K - \mu) + \beta \operatorname{div} \left(g(K) \frac{\nabla \phi}{|\nabla \phi|} \right) + 2\gamma (H(\phi) - H(\phi_o)) \right] \quad (\text{A.2})$$

Where $\delta(\phi)$ represents the diract function and it is the derivative of the Heavyside function. We can let $\delta(\phi) = |\nabla(\phi)|$ [41], and we let $|\nabla(\phi)| = 1$ since we work only with signed distance functions [41]. Hence the curve evolution equation looks as follows:

$$\phi_t = \alpha(K - \mu) + \beta \operatorname{div} \left(g(K) \nabla \phi \right) + 2\gamma (H(\phi) - H(\phi_o)) \quad (\text{A.3})$$

We followed the indications dictated in [39] evolve the contour. However, we added the shape prior term. The method with the shape prior consist in:

Algorithm 5 Evolve Hybrid Level Sets with Shape Prior

```

1: procedure HYBRID LEVEL SETS ENERGY MINIMIZATION
2:   Initialization:
3:   Initialize level set  $\phi^0$ , shape prior  $\phi_o$ ,  $\Delta_t$ , and iteration number  $t_o$ 
4:    $t \leftarrow 0$ 
5:   Repeat while  $t < t_o$ :
6:     Reinitialize level set  $\phi^t$  to be distance function  $|\phi^t| = 1$ .
7:      $\phi^t \leftarrow \phi^{t-1} + \Delta_t \left[ \alpha(K - \mu) + \gamma \left( H(\phi^{t-1}) - H(\phi_o) \right) \right]$ 
8:     Reinitialize  $\phi^t$ 
9:      $\phi^t \leftarrow \beta \operatorname{div} (g(K) \nabla \phi^t)$  using additive operator splitting (AOS)
10:     $t \leftarrow t + 1$ 
11: end procedure

```

VITA

VITA

Camilo Aguilar comes from Quito, Ecuador, where he completed his high school education. He received his BS in Electrical Engineering with specialization in Signal Processing at the University of California Irvine in 2015. He obtained his Doctorate degree in Electrical and Computer Engineering at Purdue University in 2020, where he worked under the supervision of Prof. Mary Comer and specialized in the area of image processing.