ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

CVL Computer Vision Lab

Semi-supervised learning using Total variation for biomedical image segmentation

Master Thesis

Prateek Purwar
Department of Electrical Engineering and Information Technology

Advisor: Dr. Gregory Paul
Supervisor: Prof. Dr. Orcun Goksel

Abstract

The problem of image segmentation has been tackled using various approaches and now, use of Convolutional Neural Networks(CNN) have set new benchmarks for all problems. With the access to GPUs and huge data, these networks can be trained very fast to give best results. In addition, use of fully convolutional nets have stepped up performance both in terms of accuracy and speed. The only bootleneck that can be thought for use of CNN is availabitlity of data for training. This includes both images and ground truth labels for training.

In this thesis, we try to segment an image of liver tissue acquired using Electron microscope and focus on segmenting vesicles in images. The focus of this thesis is to analyze change in segmentation score with labelling effort. We want to do this analysis because the cellular structure in microscopic images are difficult and time consuming to annotate. Firstly, we make use of fully annotated objects to fine-tune pretrained network, described as OSVOS and observe the trade off between amount of training data required and accuracy. The manual annotations provided by experts are different for same image. This motivates us to provide a method where the experts can make changes in results easily and train interactively. Due to excessive effort required for annotation budget, we decided to use Bayesian approach to solve our segmentation problem. We used simple isotropic total variation as prior and parametrized likelihood using an estimator. We used Random forest to learn likelihood using different cost functions. We compared few cost functions and showed robustness of *(*anti-nll) cost function for varying data and regularization parameter.

Using random forest and total variation, we analysed the amount of annotations required for expected accuracy. We used scribbles as partial annotation to train random forest. We showed that how a given annotation budget can be used to generate best results. We conducted this experiment by dividing manual scribbles into ”easy” and ”hard” class on basis of effort required for annotation. We observed that better results can be obtained if we use our scribbles intelligently. In addition, we observed boost in accuracy due to use of variational method. This also motivates use of prior to compensate for lack of data.

Finally, we used **cross-entropy scribble loss** to use pre-trained deep neural network to learn from scribbles. This was motivated with the emergence of various approached to couple CNN with variational methods. We concluded with stating that layered implementation of variational methods can be coupled with CNN and can be used with ease with CNNs.

Acknowledgements

I would like to acknowledge the support of my Semester Thesis advisor Dr. Gregory Paul, Post-Doc. at Computer Vision Laboratory at ETH. He consistently steered me in the right direction whenever he thought I needed it. I would sincerely thank Denis Samuylov, who helped with implementation and coding at each step of my thesis.

Contents

List of Figures

Chapter 1

Introduction

The task of image segmentation into binary classes is very useful in different cases in biomedical tasks. It can be used for detection of diseases, shape analysis etc. The methods to solve the segmentation problem has evolved among two lines: 1) level of interaction: from semi-interactive to fully automatic, and 2) level of classification: pixels to complete images. Nowadays, with the use of fully-convolutional networks, the segmentation can be obtained for complete image in one forward pass. This helps in using the local as well as contextual information for segmentation. Currently, the benchmark performance in terms of accuracy is achieved by the use of convolutional neural networks (CNN). The neural networks are specialized to learn feature maps from the examples provided and specific to task at hand. These networks require huge amount of training data: images and ground truth i.e. label for each pixel in the input image; to train the network from scratch. In literature, we can find different architectures of neural networks specially designed for task of segmentation, one of the popular architecture is U-Net[cite]. This approach works well for tasks where we can find significant number of images and can train a neural network. However, this poses a difficulty when we are trying to segment objects in microscopic images.

1.1 Electron Microscopy Images

The dataset which we are trying to segment are electron microscopic (EM) images of liver tissue. The dataset consists of a 3D stack of 2D slices of liver tissue as shown in figure 1.1. The dataset can be considered as a single 3D stack or multiple 2D slices for our task. We can observe following traits in EM images:

- High variability between images: The images to be segmented may be entirely different i.e. having fixed objects as in liver tissue or having layers to segments as in neuron tissue. The objects to be segmented may differ completely from being smooth (round vesicles) to branched (neurons). This prohibits use of one dataset to train a network for another dataset and thus, restricting availability of images.

- High variability between objects to segment: The objects to be segmented vary significanly in shape, size and texture in different images.

- High variability between goals: Even for a single image, the goal of the segmentation can be totally different. The images annotated for one object can not be used again for training purpose.

These characteristics of EM images make it very difficult to fully annotate each object of interest and is extremely time consuming for experts. Here for our task, we are interested to segment vesicles, as shown

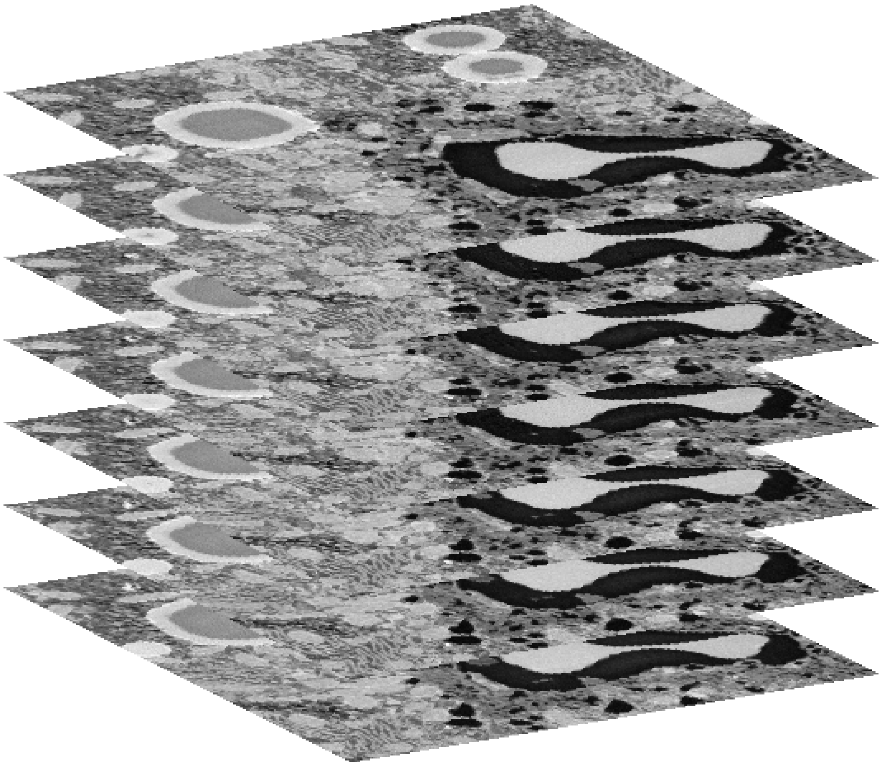


Figure 1.1: A 3D image stack, output from a scanning electron microscope. The stacks contains 458 2D images with a resolution of 1890x1952 pixel each.

in figure 1.2. The difficulty to annotate different vesicles of undefined shapes and sizes can be observed in Figure 1.2. To add to this complexity, the experts are uncertain about existence of vesicles in certain part of images and sometimes, even one expert annotates differently at different times. The difference in annotations can be observed for different experts and also for anotations of same image by same expert, as shown in figure 1.3. For example, we can observe differences in rectangular boxes in figure 1.3. This uncertainity has been analyzed in literature and researchers have tried to come up with different methods to get one ground truth mask from these multiple annotations by experts. We can use STAPLE[cite] algorithm or union or majority voting to derive reference mask. The reference mask derived for one slice using STAPLE and union is shown in figure 1.4

1.2 Focus of this thesis

Nowadays, it is common to train deep neural networks (DNN) using transfer learning to compensate for lack of enough data for training. Recently, Shelmar et al[cite] designed a ”fully convolutional” network that take input of arbitrary size and produce segmented output for complete image. They adapted contemporary classification networks (AlexNet, the VGG net, and GoogLeNet) into fully convolutional networks and transfer their learned representations by fine-tuning to the segmentation task. Similar to this, we use and finetune network explained in Caelles et al[cite]. This paper tackles the task of **semi-supervised** video object segmentation, i.e., segment an object in a video, given fully annotated mask of object in the first frame. This

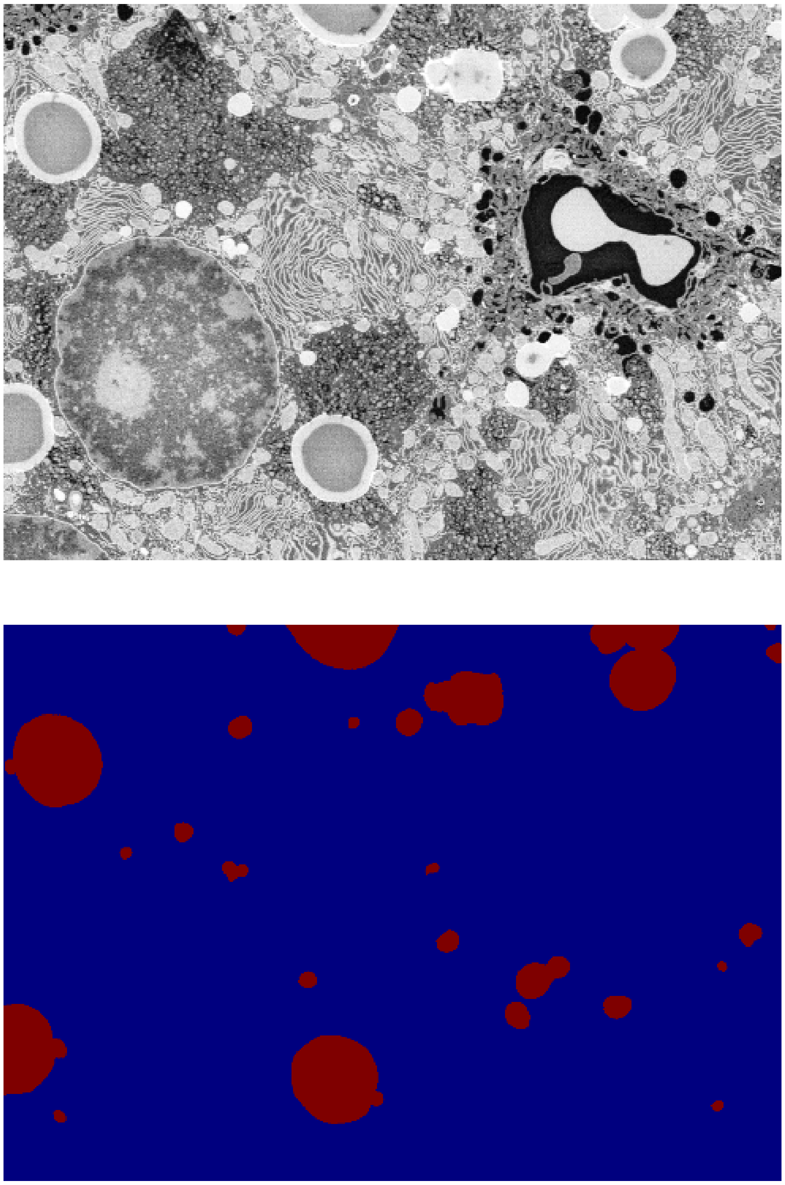


Figure 1.2: Cropped part of slice 15 and its ground truth annotation by an expert.

task can be considered to be similar to segmenting objectsin 3D stack of slices. We try to finetune the network using fully annotated objects in few slices in the stack. We describe the details and observations in Section 2.

The use of pre-trained networks makes it possible to use DNN even with small amount of training data. But still to train the DNN, we need to provide fully annotated masks for objects of interest for all training images and this comes out to be a tedious and difficult task as explained above. In addition, the presence of multiple objects of different shape and sizes makes it even more diffcult and time consuming. Imagine 1000 cells in a 2D slice and possiblity to manually annotate all these cells of undefined shapes!

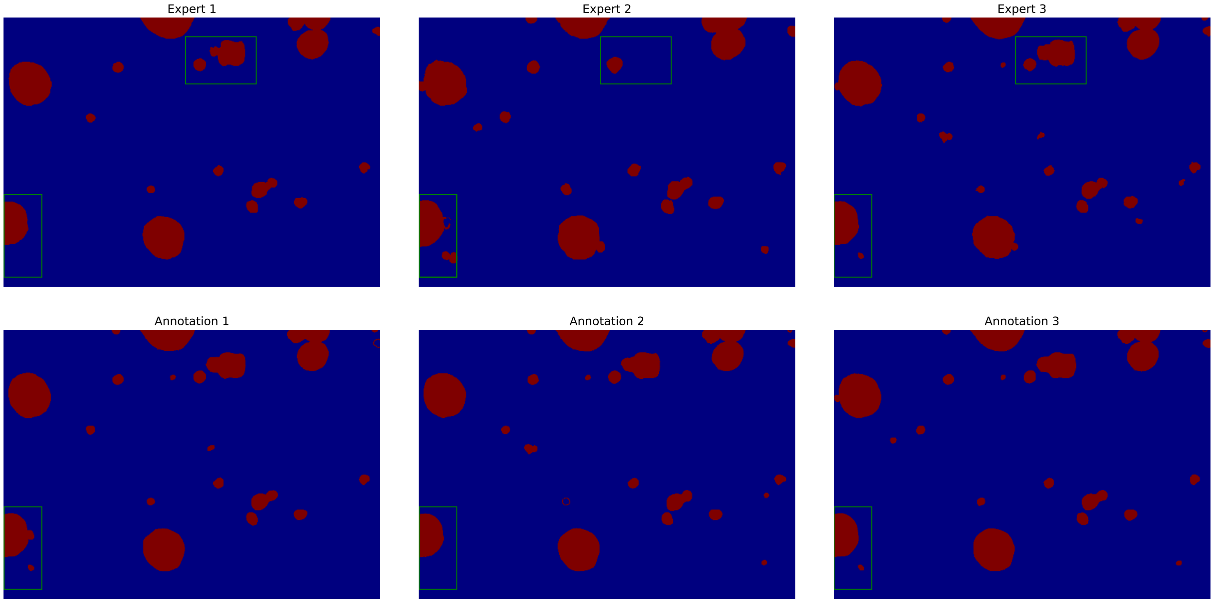


Figure 1.3: Upper row: Annotation of an slice by different experts. Bottom row: Multiple annotations of an slice by same expert. One example of difference can be observed in bounding boxes.

This provide us with option of annotate few objects and train networks using either cropped images or treating rest of image as background. Or we can use semi-supervised learning using partial annotations. In literature, we can find various methods to use these partial annotation to classify each pixel as foreground or background. For example, Santner[cite] describes use of Random forests (RF) for image segmentation using partial annotations. In this thesis, we try to discover the effect of annotation budget i.e. the number of pixels to annotate and the accuracy achieved. We also try to learn which pixels to annotate to use our annotation budget efficiently.

These methods only learn pixel level information and are uncertain for maximum of pixels i.e. the probability of foreground learnt is not binary but lies between 0 and 1. In literature, different approaches can be found to use prior information to compensate for data and for the uncertainity of estimators. The most common is to use Conditional random fields (CRFs) or graph cuts to regularize the probability learnt.

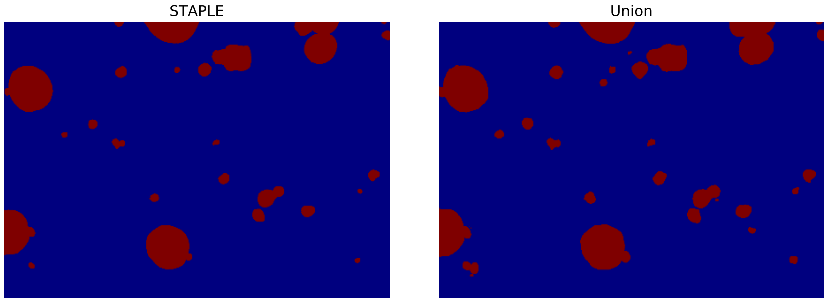


Figure 1.4: Grounth truth mask derived from multiple annotaions using 2 different methods

We solve this problem using a prior in **Bayesian framework**. Santner[cite] uses weighted total variation as prior and Random forests to learn likelihood. Ranftl[cite] uses CNN to learn unary and edge potential and combine this information to get segmentation mask using graph cuts. For our task, we implement the method described in master thesis of Dominic[cite]. In Dominic[cite], they try to learn likelihood using Random forests and prior as isotropic total variation (TV). They use a non-linear cost function to fomulate likelihood from probabilities learnt from Random forests. This is quite different to common approach of using probabilities directly as likelihood to combine with prior. Majority of researchers using CNN use a linear cost function to implement prior with help of CRFs. In this thesis, we analyze and compare these different cost functions. We try to observe the advantage of using these cost functions in different scenarios. For images as 3D stacks, it is obseved to be a difficult task and computationally efficient to encode 3D information in models as CNN or RF for learning likelihood. Also, it is common practice to use prior information in 2D. Thus, we also try to observe benefits of using 3D isotropic total variation in case of 3D stacks.

In summary, we use a Bayesian approach with RF to parametrize likelihood and isotropic TV as prior to predict segmentation mask for a given image. This gives us chance to generate fully annoatated segmentation masks and train CNN to obtain better accuracy. The common problem for use of prior is choice of appropriate scaling to couple likelihood and prior costs. Ranftl[cite] coupled the prior cost function with the likelihood cost function obtained from CNN. They optimized the final loss function to obtain optimal values for network parameters (weigts and biases) and regularization parameter. Riegler[cite] propsed a method to implement TV as specialized layers in CNN and trained the complete model, CNN + TV, together. This motivated us to replace RF with CNN and try to learn pretrained fully convolutional network from partial annotations. We were able to restructure cross-entropy loss to compute loss for partial annotations.

Finally, we also showed advantage of using iterative semi-interactivity for efficient use of annotation budget and also to be able to provide opportunity to experts to improve learning method according to their specific requirements.

1.3 Thesis Organization

The thesis is divided mainly into two sections: segmentation using fully annotated objects and segmentation using partial annotations. The segmentation using full annotations is described in Section 2. The later method is described in Section 3. In section 3.1, we describe improvement in segmentation mask for increased labelling effort. We introduce use of prior and variational methods in section 3.2. In section 3.3, we introduce use CNN to learn from partial annotations. Finally, in section we conclude this thesis and lay out future work that can be done.

Chapter 2

Fully annotated segmentation masks

The deep neural networks have become popular to solve any task in the field of computer vision. To train a network from scratch, the main effort goes into preparing data for training network, and in coming up with best architecture and choosing best training parameters. The data is available on internet and can be extracted and modified for various tasks such as, IMDB database can be used to train network for problem involving faces. This becomes a problem in medical domain where it is very costly to generate images and even more costly and time consuming to prepare it for training. For our problem of image segmentation, we described the problem faced by experts and researchers in generated segmented masks. The lack of data has motivated researchers to use transfer learning. Transfer learning tries to store the knowledge gained from solving one problem and applying it to different but related problem. Thus, in practice, it is very rare to train an entire Convolutional Network from scratch (with random initialization), because it is relatively rare and difficult to have a dataset (images and labels) of sufficient size for training. Instead, it is common to use a pretrained network either to intialize a network or to extract required feature maps. We decided to use pretrained network and fine-tune it for our task.

2.1 One shot Video object segmentation (OSVOS)

Caelles et al[cite] designed an architecture to segment an object in a video sequence using only one frame for training. The network is trained to learn object from only one frame and generate segmentation mask for remaining all frames. The segmentation work well if object remains in relatively similar shape and size. This can be considered similar to our problem of segmenting vesicles in 3D stack of liver tissue. Since, the vesicles are relatively similar in shape and size in different slices, we annotated first few slices to train the network. We generated more data using cropped and flipped slices for training. In literature, we can find that network can overfit on relatively small dataset. The use of augmentation helps in generating more data and avoids network from overfitting.

OSVOS uses pretrained network of VGG-net for initialization. They removed the final fully-connected layers and replaced them with deconvolution layers to generate mask of image size. In addition, the network contains end output, side outputs and main output generated from combination of side outputs. The total loss is calculated for all outputs and used for backpropagation. The details of architecture and initialization parameters can be found in Caelles et al[cite]. As we described the diffculty of annotating full objects in Section 1, we tried to observe accuracy improvement with increase of training data. We trained OSVOS from 1 slice and increased the training data to 10 slices. The initialization parameters were kept same for cases. In figure 2.1, we can observe that OSVOS performs well with with only 2 slices. The segmentation output from CNN trained using 2 slices is also shown in figure 2.1.

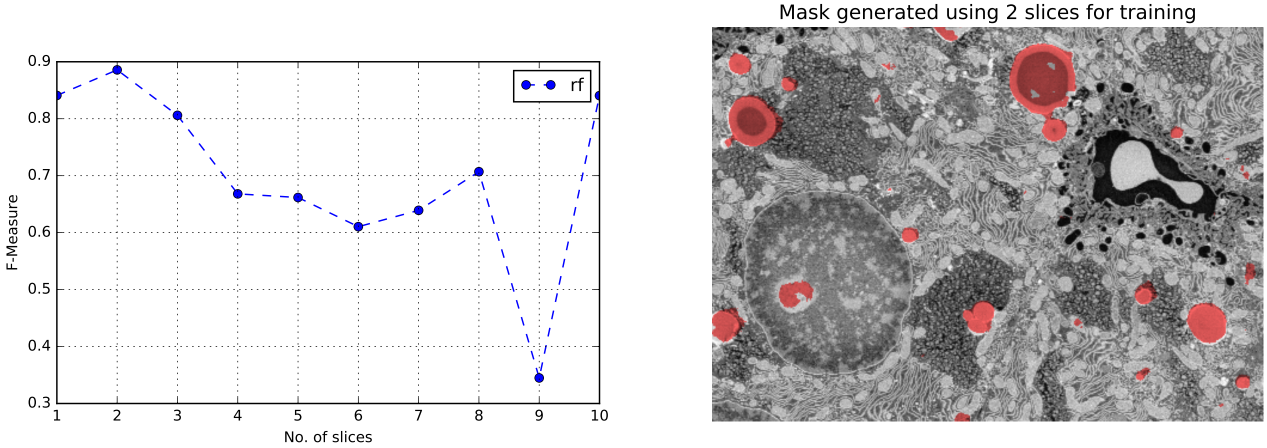


Figure 2.1: Left image:- F-measure computed for different amount of training data; Right image: Predicted mask for one slice

We expected increase in performance with increase in amount of data. This does not happen as CNN is not able to converge equivalently for all cases. It is important to remember here that annotating one slice is not same as one object. We can observe these multiple objects in figure 1.2. Also, change in annotation will force us to train network again. These difficulties motivated us to try semi-supervised learning for our task of segmentation.

Chapter 3

Semi-supervised image segmentation

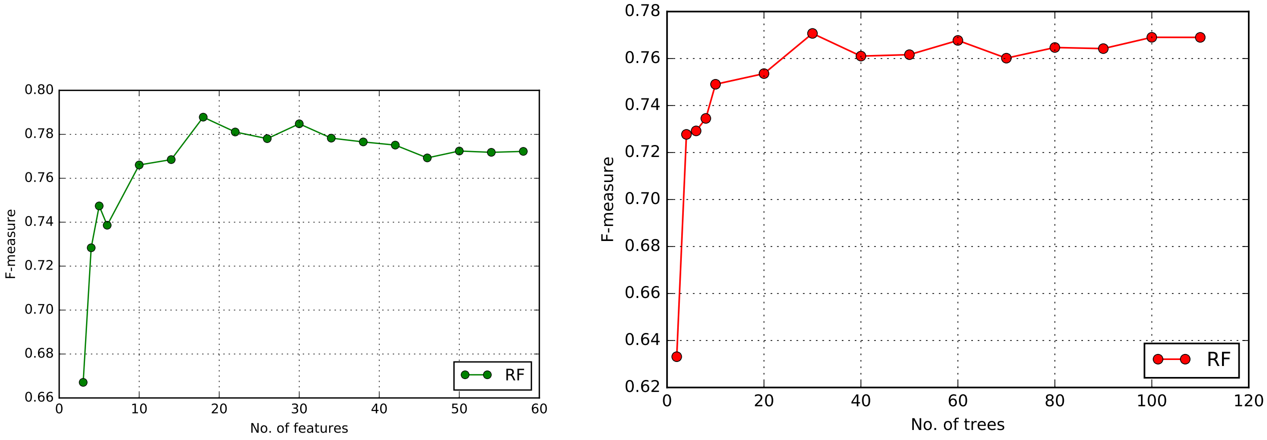
The philosophy behind semi-supervised learning is to propagate label information from labelled to unlabelled data. Image segmentation can be seen as a classification problem which consists of assigning a class label to each pixel. For our task of binary segmentation, this means classifying each pixel as foreground or background. For our task of image segementaion, we make use of partial annotations as *scribbles*. Scribbles are pixels in image annotated by experts as foreground or background. We use example-based methods to learn from these scribbles annotated by experts. In contrast to having different images for training and testing, we use same image for training and testing as the samples used for training are pixels and not images.

3.1 Random Forest

In this section, we make use of randon forest (RF) as semi-supervised learning algorithm. The advantages and details of using RF can be found in Dominic[cite]. For training RF, we compute set of features in Python. We compute different features ranging from simple Sobel edge detectors to higher level Gabor filters. The choice of features was made according to WEKA[cite] toolset of FIJI[cite] plugin. These are set of 2D features and perform well for medical images. We compute different type of features for a range of sigmas, which gives 69 feature maps for a single image. The details of features computed can be found in Appendix[cite]. In thesis by Dominic, we can find details and effect of feature selection for training Random forests. As shown in figure 3.1, we can observe that for given annotation budget, the segmentation measure does not change significantly for more than 30 trees and for more than 20 features. Therefore, for all experiments using RF, we use 20 best features and 30 trees for training. In this thesis, we focus on how to get best results for given annotation budget and thus, use fixed number of features and trees to generate masks. We try to answer the question of where to scribble and how to make best use of our annotation budget and time.

3.1.1 Where to scribble?

In general, we believe that the more training data we provide, more we can improve the results. Does this hold for partial annotation such as scribbles? If we go on increasing the pixels annotated arbitatrily, will it improve the segmentation mask or we have to use our labelling effort intelligently to improve results? We conducted an experiment by dividing our set of foreground and background scribbles into 2 classes: easy and hard. We classified scribbles as ”easy” and ”hard” depending on effort required to annotate these pixels. For example, pixels are difficult to annotate near boundary of foreground and background, and we classify



(a) (b)

Figure 3.1: Plot of segmentation measure vs changing complexity of RF: (a) with features, (b) with trees

these pixels as ”hard”, as shown in figure 3.2. We manually scribbled image for both ”easy” and ”hard” subclasses. Then, we trained and tested RF on one image by increasing percentage of scribbles belonging to ”easy” foreground and background class. After, we have used all scribbles belonging to ”easy”class, we added scribbles from ”hard” class for both foreground and background. The increment was done w.r.t. total amount of scribbles we are having and also, for higher percentage of added scribbles, we maintained a ratio between foreground and background pixels. The result can be observed in figure 3.3(a).

In figure 3.3(a), we can observe that after total of 3000 pixels selected from ”easy” foreground and background, the segmentation measure does not change significantly. An improvement can be observed, once we started adding ”hard” scribbles after all ”easy” scribbles were used. This shows that the best results can be obtained by adding ”hard” scribbles after addition of certain percentage of ”easy” scribbles. Looking at the plot, one might think to start adding the ”hard” scribbles after 3000 ”easy” scribbles. We tried this and results can be observed in figure 3.3(b).

In figure 3.3(b), as we started adding ”hard” scribbles on top of 10% (3000 pixels) of scribbles selected from ”easy” scribbles, instead of observing a rise with additional scribbles, we observed a fall in performance (dark blue plot in figure 3.3(b)). This may be due to lack of enough ”easy” scribbles and RF starts training its trees to focus more on ”hard” scribbles. We tried similar experiment with different amount of ”easy” scribbles to start with. We started seeing a significant improvement when we utilized with 70% of all ”easy” scribbles to train RF. We were able to achieve f-measure score of 0.83 in comparison of 0.84 achieved with 100% usage of ”easy” scribbles (See green and pink plot in figure 3.3(b)). Thus, the question arises how to decide the point of addition of ”hard” scribbles.

3.1.2 Iterative semi-interactive approach

In previous section, we showed need of using our annotation budget intelligently to get best performance. But, we observed the problem of deciding on how many ”easy” and ”hard” scribbles are needed to achieve best results. For our problem, we divided the scribbles as ”easy” and ”hard” according to labelling effort, but this division for scribbles may not be same from point of view of Random forest. Apriori, we don’t know which pixels will be difficult for Random forest to classify correctly. The above mentioned two problems can be solved by annotating pixels iteratively to improve results, atleast once to understand which pixels

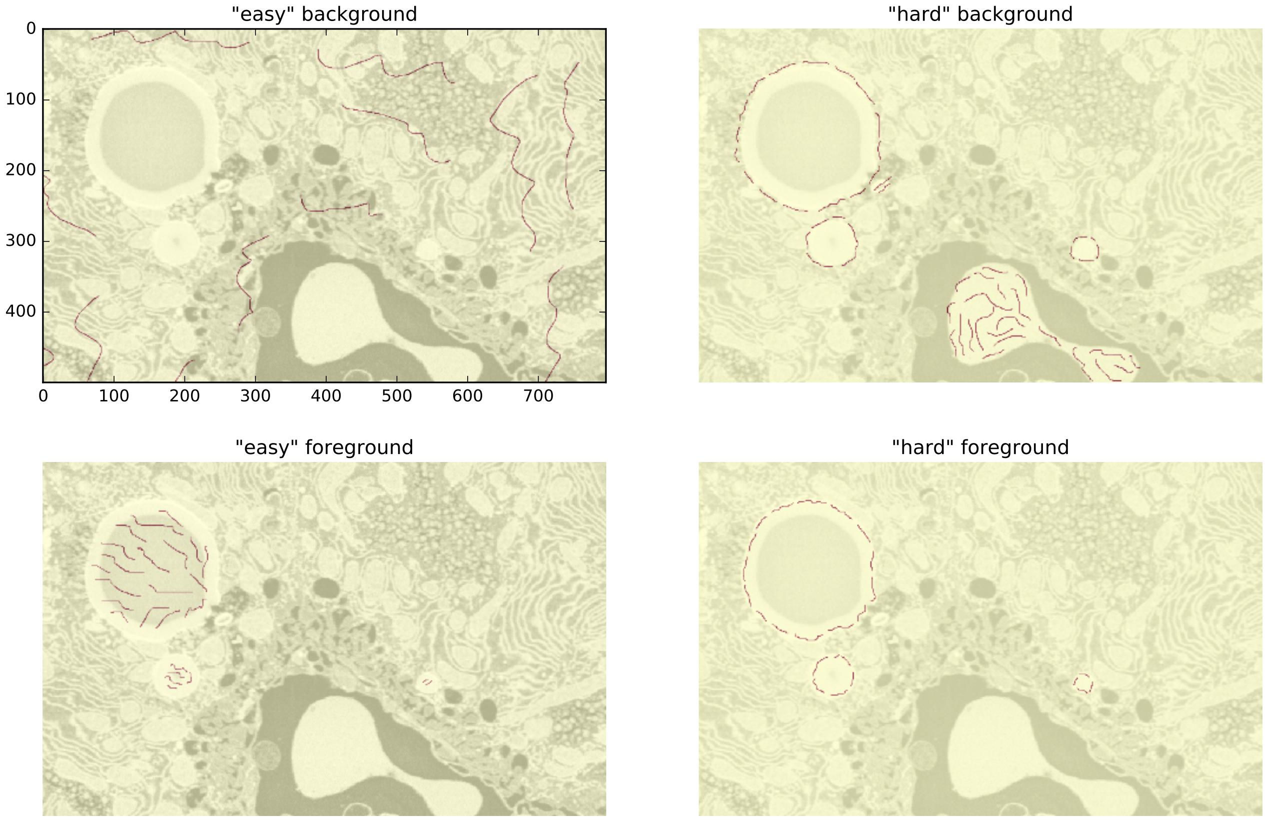
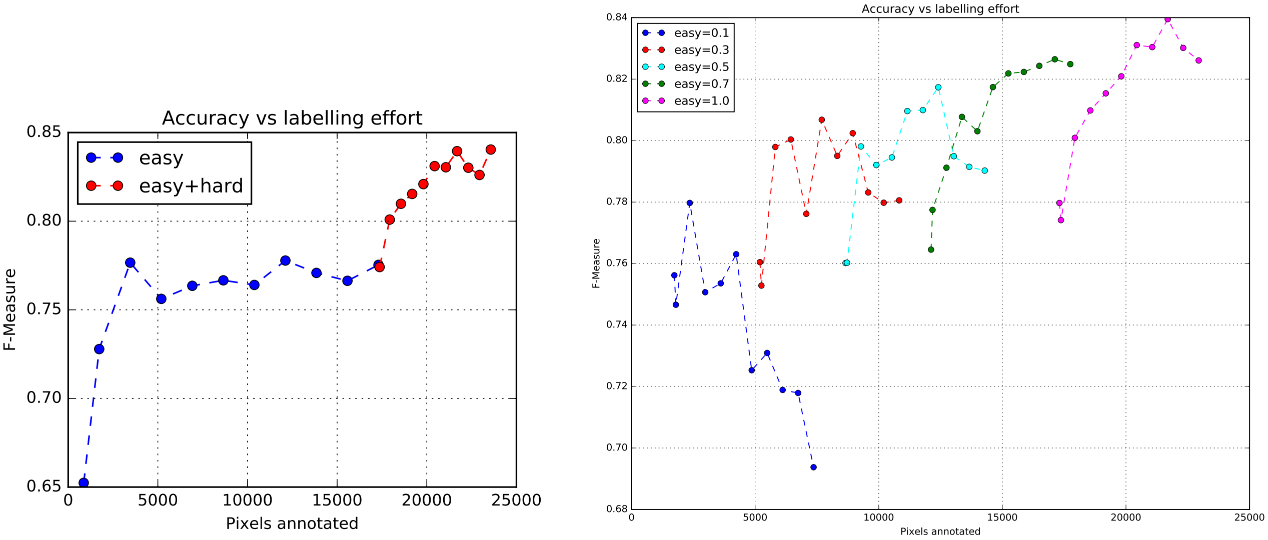


Figure 3.2: Manual scribbles in ”easy” and ”hard” classes

are difficult for RF to classify. We show the improvement in result by doing one iteration in figure 3.4. We can observe that the scribbles are very few to produce a good result. Still, we can observe improvement in f-measure from 0.745 to 0.749 for increasing the annotation budget from 7500 pixels to 10900. Although the increment looks insignificant, but we can observe the difference in large vesicle in left-top corner of the image.

3.1.3 Uncertainity of classifier

The use of iterative semi-interactivity gives best result for given annotation budget, but the output of RF is noisy and uncertain. The uncertainity lies in inability to classify maximum of pixel as foreground and background, as shown in figure 3.5(a). The histogram shows distribution of probability values for complete image. It can be seen that a large number of pixels are not given probability of 0 (background) or 1 (foreground). In figure 3.5(b), we can observe varying results for different threshold applied on output from RF. RF acts as an classifier and classifies each pixel but we need to group these pixels into objects for segmentation. In this thesis, we make use of prior information to compensate for lack of enough annotated data and for uncertainity of classifier.



(a) (b)

Figure 3.3: Plot of segmentation measure vs annotation budget: (a) Change in segmentation measure with increment of scribbles from ”easy” class, and then from ”hard” class. (b) Change in segmentation measure with increment of scribbles from ”hard” class, starting with different fixed amount of ”easy” class

3.2 Bayesian Formulation

To make use of prior, we model our image segmentation problem as a Bayesian inference problem. Let us cosider an observed image, $\boldsymbol{I}$ and labeled or segmented ground truth, $\boldsymbol{M}$, the joint probabilty can be defined as:

$$\boldsymbol{p}(\boldsymbol{I},\boldsymbol{M}) = \boldsymbol{p}(\boldsymbol{M})\boldsymbol{p}(\boldsymbol{I}|\boldsymbol{M}) \quad ,$$

and applying Bayes theorem,

$$\begin{aligned} \boldsymbol{p}(\boldsymbol{M}|\boldsymbol{I}) &= \frac{\boldsymbol{p}(\boldsymbol{M})\boldsymbol{p}(\boldsymbol{I}|\boldsymbol{M})}{\boldsymbol{p}(\boldsymbol{I})} \\ & \alpha\, \boldsymbol{p}(\boldsymbol{M})\boldsymbol{p}(\boldsymbol{I}|\boldsymbol{M}) \end{aligned}$$

The left hand side is the probability of obtaining segmentation mask, $\boldsymbol{M}$ given the image $\boldsymbol{I}$, is called the posterior probability. $\boldsymbol{p}(\boldsymbol{M})$ is the prior probability of mask, $\boldsymbol{M}$. The Maximum a posteriori (MAP) estimate, $\boldsymbol{M}^*$ can be calculated as follow:

$$\boldsymbol{M}^* = \arg\max_{\boldsymbol{M}}(\boldsymbol{p}(\boldsymbol{M})\boldsymbol{p}(\boldsymbol{I}|\boldsymbol{M})) \quad . \tag{3.1}$$

The above problem can as well be stated as an energy minimization problem by writing Equation 3.1 in terms of energy by taking negative log-likelihood:

$$\begin{aligned} E(\boldsymbol{M}) &= -\log(\boldsymbol{p}(\boldsymbol{I},\boldsymbol{M})) \\ &= -\log(\boldsymbol{p}(\boldsymbol{I}|\boldsymbol{M})) - \log(\boldsymbol{p}(\boldsymbol{M})) \\ &= E_d(\boldsymbol{I},\boldsymbol{M}) + E_r(\boldsymbol{M}) \end{aligned}$$

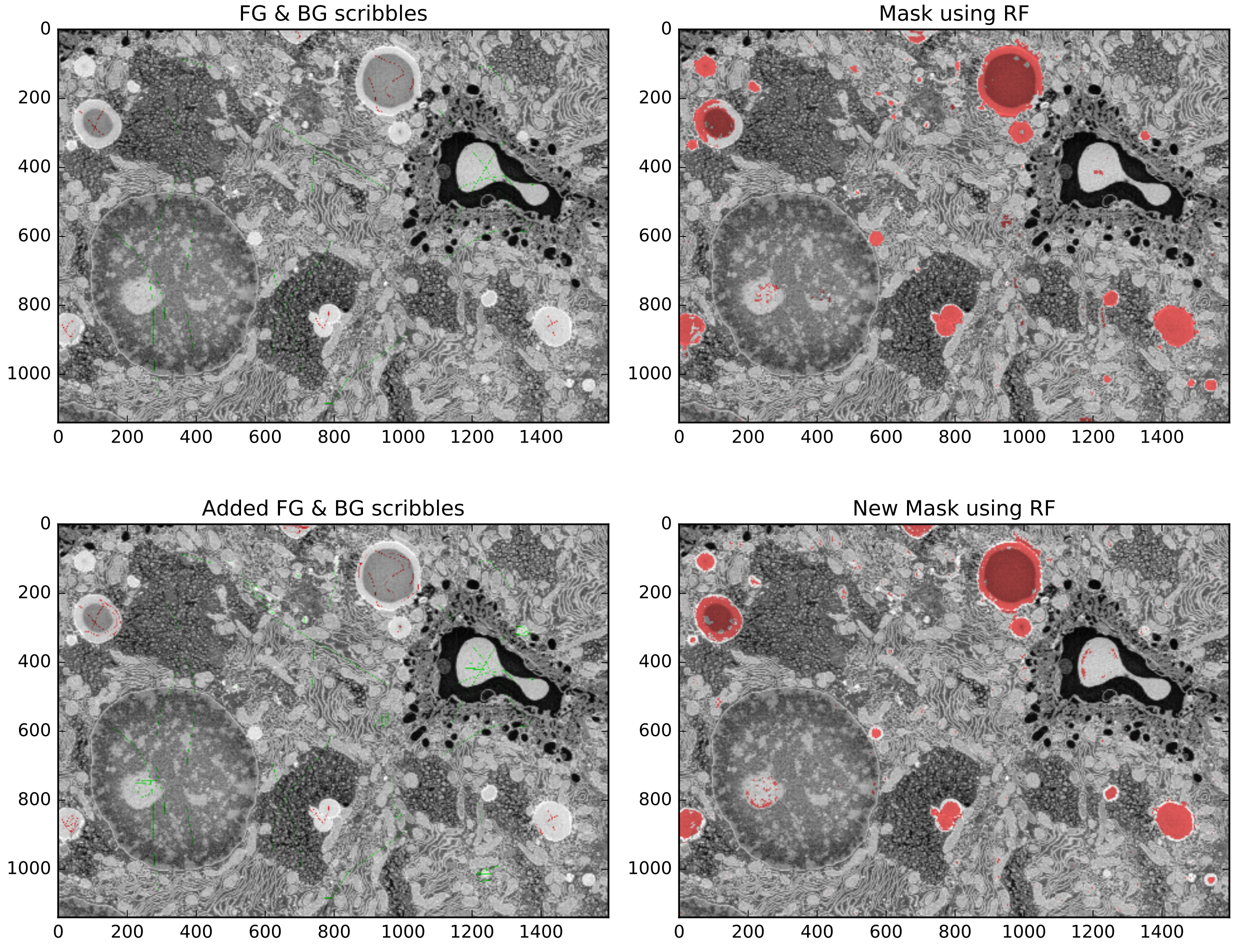


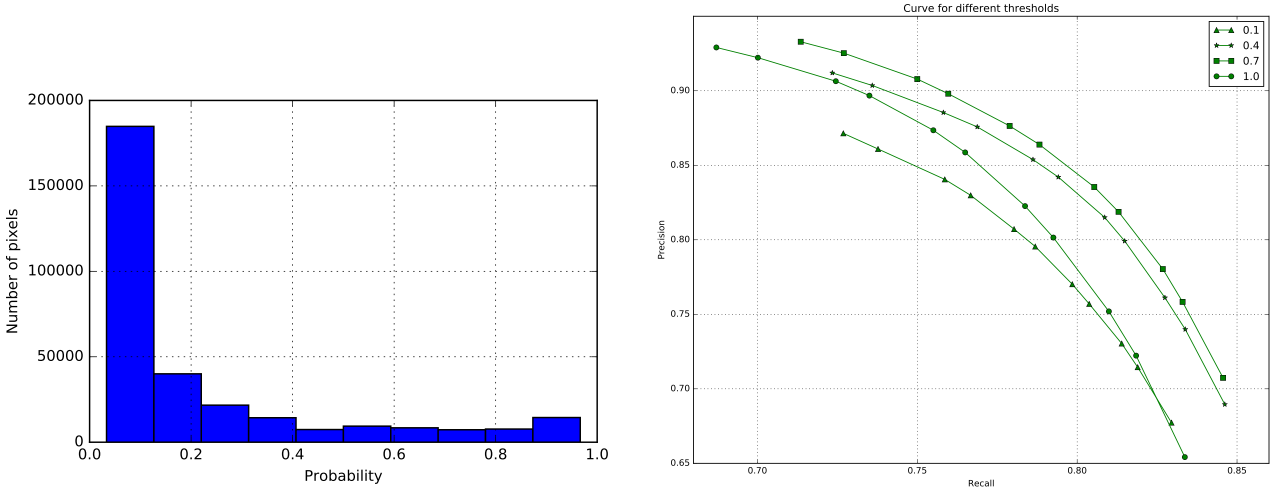
Figure 3.4: Semi-interactive segmentaion with one iteration

The total energy, E, that we want to minimize can be considered as linear combination of data or likelihood term, E_d and prior term (or regularization), E_r. This modifies calculating MAP estimate to:

$$\boldsymbol{M}^* = \arg\min_{\boldsymbol{M}} (E_d(\boldsymbol{I}, \boldsymbol{M}) + E_r(\boldsymbol{M})) \quad .$$

To obtain MAP estimate, we need to formulate likelihood term and prior term. We formulate the prior using Total variation(TV). We can find use of different TV priors such as Wulff shapes etc. In our thesis, as the objects we need to segment are smooth and shaped like a circle, we make use of isotropic total variation, TV. Also, we can try to use isotropic total variation in 2D and 3D as the data we are trying to segment is a 3D stack. For likelihood term, G. Paul et al.[cite] proposed an energy formulation which is not derived from a statistical model but learnt from training set. This gives the advantage of combining example-based and model-based approaches. Similar to Dominic[cite], we formulate the likelihood term as product term of a cost function, C, of soft mask, $\boldsymbol{P}$(probability of each pixel being foreground) learnt from RF and optimal mask to be estimated, $\boldsymbol{M}$. The energy minimization problems becomes:

$$\begin{aligned} E(\boldsymbol{M}) &= E_d(\boldsymbol{I}, \boldsymbol{M}) + E_r(\boldsymbol{M}) \\ &=< C(\boldsymbol{P}), \boldsymbol{M} > + \lambda\ TV(\boldsymbol{M}) + \mathrm{i}_{[0,1]}(\boldsymbol{M}) \quad , \end{aligned}$$



(a) (b)

Figure 3.5: (a) Histogram of probabilities. (b) Precision-recall curve for varying thresholded RF mask. Different curves for different annotation budget

where $\mathrm{i}_{[0,1]}(\boldsymbol{M})$ is an indicator function to ensure values of $\boldsymbol{M}$ remain in [0,1]. In addition to use of cost function, we enforce constraint of preserving the pixels annotated by experts as foreground and background in our energy minimization problem. We used an indicator function, $\mathrm{i}_{fg}(\boldsymbol{M})$, to ensure pixels in foreground scribbles have value of 1 in mask, and indicator function, $\mathrm{i}_{bg}(\boldsymbol{M})$, to ensure pixels in background scribbles have value of 0 in mask. Using discrete implementaion of TV, the final energy minimization problem is formulated as given below:

$$E(\boldsymbol{M}) =< C(\boldsymbol{P}), \boldsymbol{M} > +\lambda \,\||\boldsymbol{D}\boldsymbol{M}|_2\|_1 + \mathrm{i}_{[0,1]}(\boldsymbol{M}) + \mathrm{i}_{fg}(\boldsymbol{M}) + \mathrm{i}_{bg}(\boldsymbol{M})$$

The optimization problem is solved using Alternating Split Bregman method (ASB), as described in Dominic[cite]. The advantage of using ASB is that it splits the above problem into subproblems. Each subproblem is easy to solve and can be solved independently. The final solution to the problem is obtained by iterating updates. The details of implementation and solution can be obtained from Dominic[cite]. The results for RF with variational segmentation (with ***anti-nll*** cost function described in following section) can be seen in figure 3.6. We can observe different improvement due to use of variational method for different regularisation parameter (λ). The boost obtained from variational image processing (VIP) with best among different λ can be seen in figure 3.7. The advantage can be seen that the boost is always positive implying that use VIP never degrades the performance of RF. In the following section, we describe and compare use of different cost functions in Variational segmentation methods.

3.2.1 Different likelihhood formulations

In literature, people have formulated the cost function either directly using the soft mask ($\boldsymbol{P}$) obtained from RF or, some linear or non-linear function of the mask. Santner[cite] formulates likelihood term as linear function of mask, C_l, as given below:

$$C_l(\boldsymbol{P}) = -4\,(\boldsymbol{P} - 0.5) \quad .$$

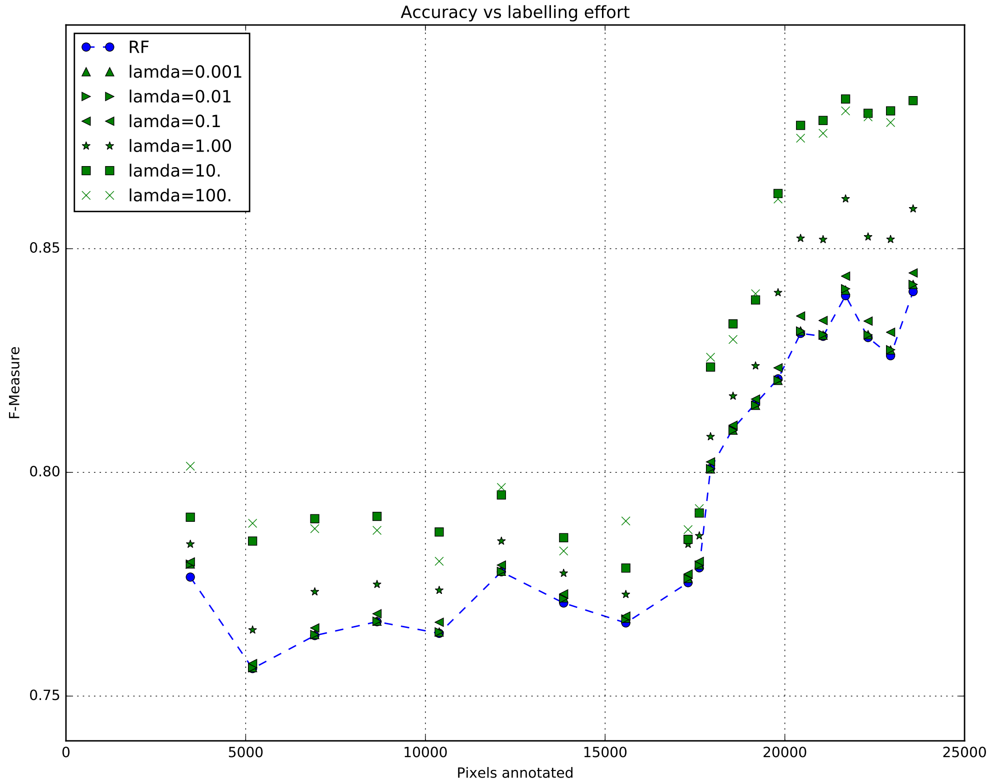


Figure 3.6: Segmentation score with RF and prior(TV) for different annotation budget

We used, ***anit-nll*** cost function, C_{nll}, as given below:

$$C_{nll}(\boldsymbol{P}) = \begin{cases} 0, & \text{if } pixel \in Scribbles \\ -\log \frac{\boldsymbol{P}}{1-\boldsymbol{P}}, & \text{else} \end{cases}$$

For *anti-nll* cost function, we need to use foreground and background constraint for pixel with probability 0 or 1 as we do for scribbles. In addition to the cost function, Santner[cite] mentioned use of hard constraint for pixels in foreground or background scribbles i.e. using cost of $-\infty$ for foreground scribbles and ∞ for background scribbles. They didn't show a way to enforce this constraint with *linear* cost function. We enforced this constraint with use of indicator functions.

We used different amount of annotated pixels, randomly selected from ground truth and generated segmentation mask using RF and TV with different cost functions. We generated results to compare 3 cost functions: *linear*, *linear with constraints* and *anti-nll (with constraints)*. The results can be observed in figure 3.8. The *linear* cost function does not work well with large values of λ and also deteriorates the mask obtained from RF. This does not happen with *linear with constraints* and *anti-nll*, where VIP performs always better than RF. This shows robustness of using *linear with constraints* and *anti-nll* cost functions.

3.2.2 Effect of regularisation parameter

The use of prior information boosts up the performance but we get different boost for different value of λ. The regularisation parameter decides the weight of TV cost. One expects smooth boundaries in segmentation

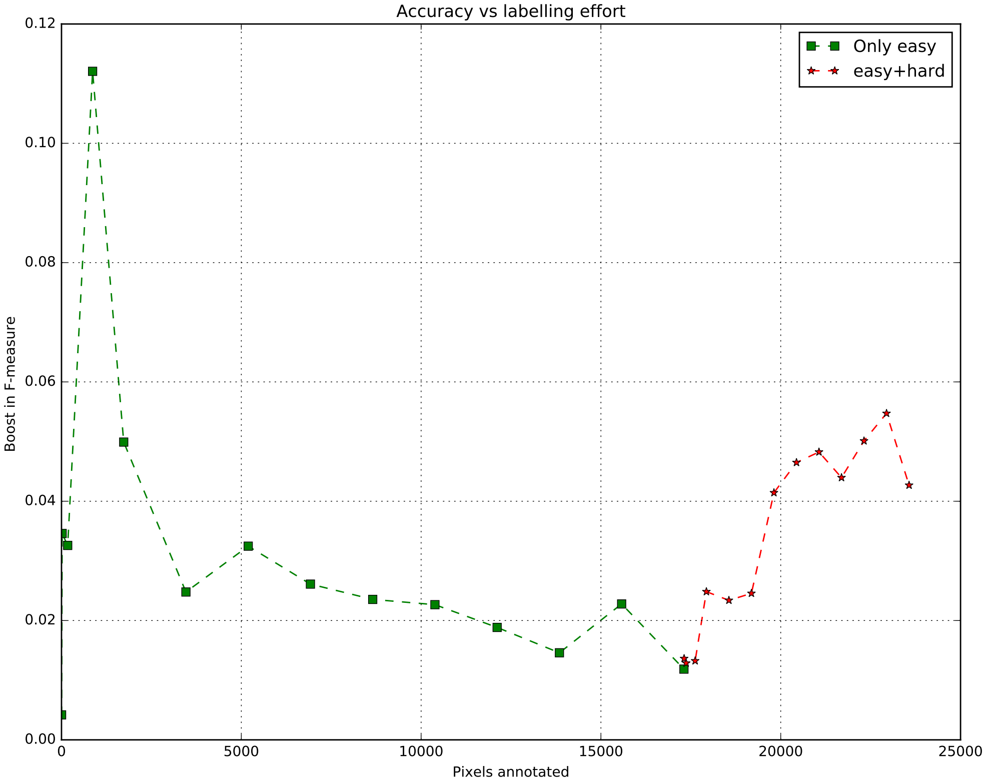


Figure 3.7: Boost obtained with VIP over RF Segmentation score with RF and prior(TV) for different annotation budget

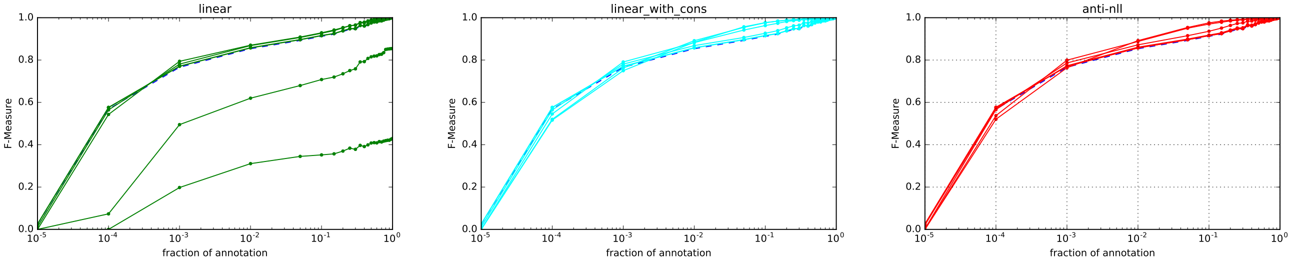


Figure 3.8: VIP with different cost functions

mask for high values, but this also effect mask for small objects. We can observe this in figure 3.9. The upper row shows that for larger vesicles, the boundary gets smooth for higher values of λ, while the bottom row shows that tiny vesicles in image tend to diminish for high values of λ. This shows that the ideal case will be to choose different λ for different part of images or to combine results for different λ.

3.3 Semi-interactive segmentation

We realized the need for iterative semi-interactive segmentation in section 3.1.2. With 1 iteration of interactive segmentation, we were able to improve f-measure results . Now, we combine semi-interactive

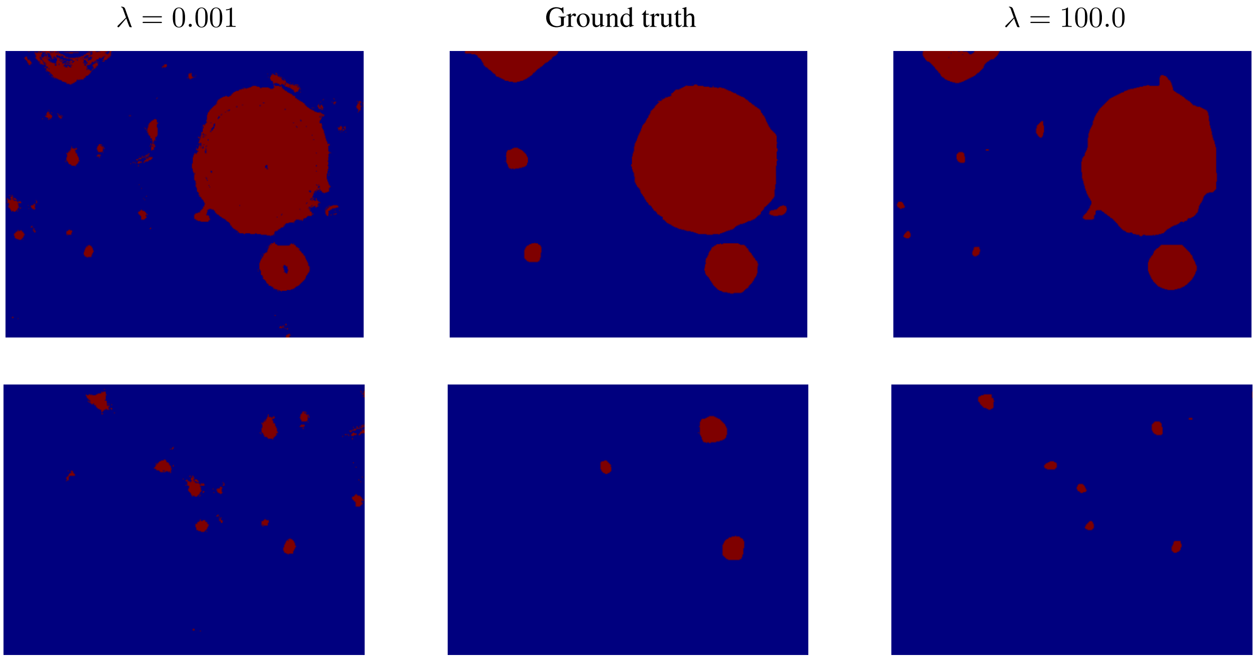


Figure 3.9: Segmentation mask for differnt λ for 2 crops of image

annotation with use of variational segmentation. We take segmentation mask obtained from variational segmentation and add scribbles to image where the RF and VIP together is unable to segment correctly. Then, we retrain Random forest and generate new mask using variational segmentation. The image, mask and scribbles are shown in figure 3.10. We can observe the improvement in mask with addition of only 5000 pixels (less 0.2% of pixels in image).

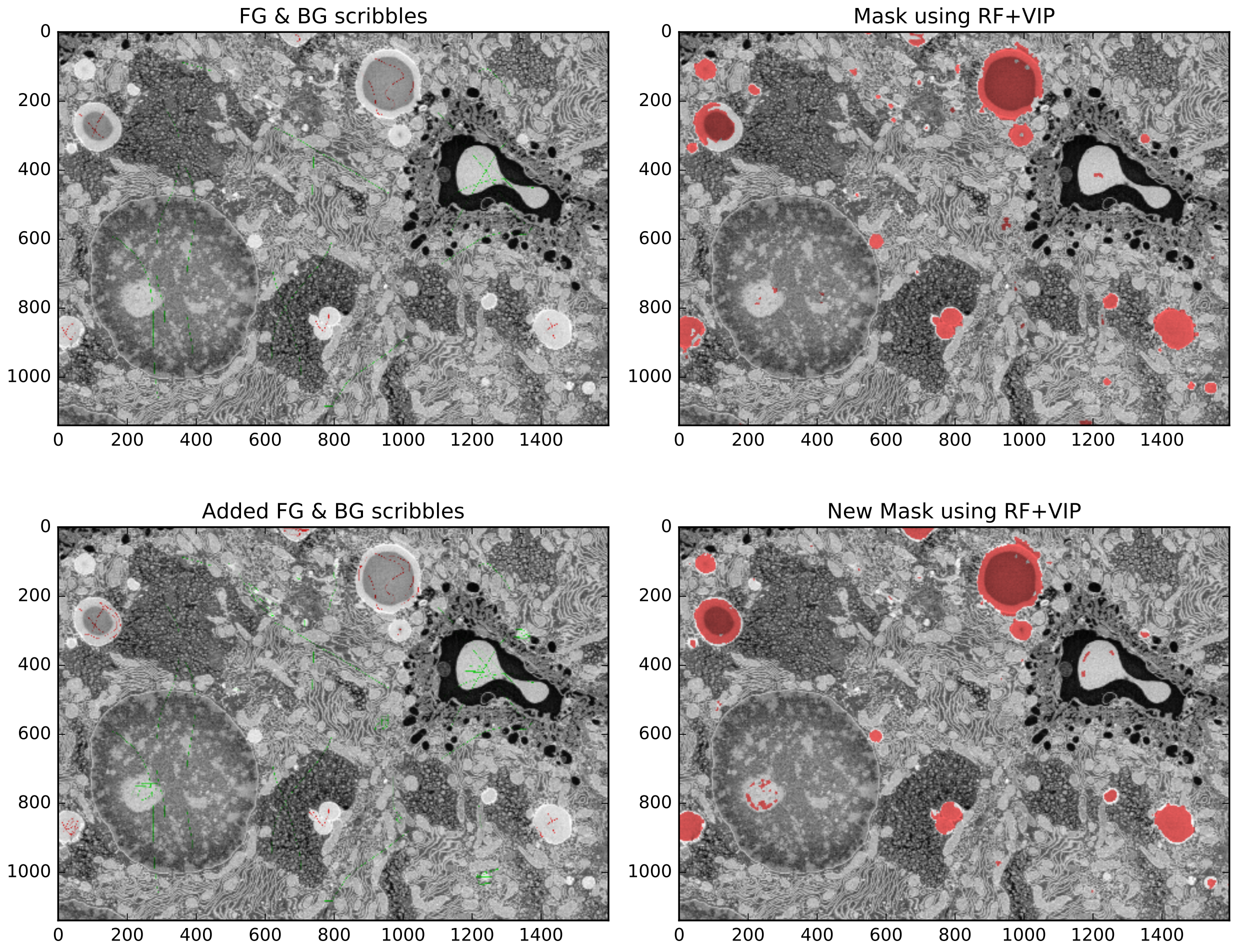


Figure 3.10: Semi-interactive segmentaion with one iteration (RF+VIP)

Chapter 4

Training CNN from scribbles

The use of Variational image processing parametrized with cost learnt from RF provides a segmentation mask with good accuracy. The RF was trained with features described in WEKA toolset. These features may not work well for certain medical images and thus, use of CNN proves beneficial as it learns different feature maps according to task. The initial layers learns basic image features while the final layers get trained for filters to compute problem specific results. In addition to this, the choice of λ is always a problem in using VIP. As we showed in section 3.2.2, different parts of images need different values of λ to produce best segmentation mask. Ranftl[cite] uses CNN with VIP and modifies the loss function accordingly and learns optimal values of λ along with CNN parameters. The paper describes a method of combining CNN(5 layers) with a final variational/inference layer. The inference layer has activation function in form of Total variation. Similarly, Taylor et al.[cite] implemented CNN as a scalable ADMM approach. They splitted objective function into subproblems (as we did using ASB) and trained CNN without gradients. These papers attempts to couple CNN with VIP to gain from both approaches.

This motivated us to replace RF with CNN to parametrize cost function. In literature, we can find multiple approaches to train CNN using scribbles. Gonda et al.[cite] uses an interactive approach to train deep neural networks for segmentation of neuronal structures using scribbles. Lai et al.[cite] uses patch-based 3D image segmentation. They make use of patches around pixels annotated to train neural network. These ideas takes each pixel as an sample and CNN is trained as a classifier to classify each pixel. The disadvantage is that we remove one major property of CNN to adapt its final layers according to full image for segmentation task and also, it needs sufficient data to train CNN. Therefore, we tried to train OSVOS network (explained in Section 2) from scibbles using **cross entropy scribble loss**. The simple trick we used was to replace computation of cross-entropy loss function for complete image by computing loss for only annotated scribbles. The **cross entropy scribble loss** can be computed at each pixel, x with probability p, as defined below:

$$l_{scribble}(x) = \begin{cases} -z(x)\, log(p(x)) - (1 - z(x))\, log(1 - p(x)), & \text{if } x \in \text{Scribbles} \\ 0, & \text{else} \end{cases}$$

We succeded in training our network with scribble loss and results can be seen in figure 3.11.

Chapter 5

Conclusion

Bibliography