Few-shot hypercolumn-based mitochondria segmentation in cardiac and outer hair cells in focused ion beam-scanning electron microscopy (FIB-SEM) data

Julia Dietlmeier (1)[1], Kevin McGuinness (1), Sandra Rugonyi (2), Teresa Wilson (2), Alfred Nuttall (2) and Noel E. O'Connor (1)

(1) Insight Centre for Data Analytics, Dublin City University, Dublin, Ireland
(2) Oregon Health and Science University, Portland, Oregon, USA

Abstract

We present a novel AI-based approach to the few-shot automated segmentation of mitochondria in large-scale electron microscopy images. Our framework leverages convolutional features from a pre-trained deep multilayer convolutional neural network, such as VGG-16. We then train a binary gradient boosting classifier on the resulting high-dimensional feature hypercolumns. We extract VGG-16 features from the first four convolutional blocks and apply bilinear upsampling to resize the obtained maps to the input image size. This procedure yields a 2688-dimensional feature hypercolumn for each pixel in a 224×224 input image. We then apply L_1-regularized logistic regression for supervised active feature selection to reduce dependencies among the features, to reduce overfitting, as well as to speed-up gradient boosting-based training. During inference we block process 1728×2022 large microscopy images. Our experiments show that in such a formulation of transfer learning our processing pipeline is able to achieve high-accuracy results on very challenging datasets containing a large number of irregularly shaped mitochondria in cardiac and outer hair cells. Our proposed few-shot training approach gives competitive performance with the state-of-the-art using far less training data.

1 Introduction

Deep learning with convolutional neural networks (CNNs) is currently revolutionizing computer vision and has achieved great success in many applications. Examples include: image classification, hand-written digit recognition, object detection, face recognition, scene understanding, image segmentation, and semantic segmentation. Multilayer CNNs are especially well-suited for computer vision applications because of their ability to hierarchically abstract representations with local operations. Thus, CNNs can hierarchically learn image features starting with low-level features such as edges, corners, and color (shallow layers),

[1] julia.dietlmeier@insight-centre.org

progressing to the higher and more abstract representations such as shape and texture (deeper layers). Usually, deep CNNs for the above application areas are trained on large datasets for a long period of time.

Within deep learning, transfer learning is a branch of techniques that uses pre-trained CNNs (trained on a particular task where a large amount of training data is available) for different new tasks and datasets where a large amount of training data is not available. It is common in practice to use a CNN as a fixed feature extractor [Razavian et al.(2014)]. Instead of training a CNN from scratch, transfer learning-based approaches reuse a pre-trained CNN on a very large dataset (e.g. ImageNet ILSVRC13, which contains 1.2 million images with 1000 classes) for different tasks and different datasets. In particular, the last fully-connected layers are removed from a CNN and the rest of the architecture can be used as a fixed feature extractor for a new dataset. In this paper, we use VGG-16 network trained on ImageNet for a classification task and apply it not only to the new task of segmentation but also to the new biomedical data domain.

Mitochondria segmentation is still a very challenging application area of computer vision. Automated segmentation is usually formulated as supervised or semi-supervised machine learning with handcrafted features. Mitochondria are membrane enclosed organelles that are found inside every living cell. Depending on the tissue type and field of view, complex subcellular environments can contain a very large number of organelles, as seen in Fig.1. Mitochondria have an average diameter of 200nm with large variation in size and shape even within one imaged section. These organelles move within a living cell and also undergo fission and fusion. Mitochondrial morphology depends on the type of biological tissue and further undergoes structural changes during different biochemical processes. These facts lead to the broad range of mitochondrial shapes and textures and present significant challenges to a unified approach to segmentation. Manual segmentation of thousands of images for biomedical image analysis is also very time-consuming, hence the need for automated methods.

The imaging modalities used play an important role in image characteristics and image quality and can add speckle noise, non-uniform illumination and *waterfall* noise [Fitschen et al.(2017)] to the images acquired. Data from a FIB-SEM (Focused Ion Beam Scanning Electron Microscope) are used in the experimental part of this paper. FIB-SEM, or "slice and view," allows 3D imaging of biological samples with nanometer resolution. During image acquisition, the FIB mills thin sections of about 4nm thick from the block, and then the SEM images the block face. This is done repeatedly to obtain a 3D image of the sample. The dataset used in this study contains one hundred 1728×2022 pixel large images of cardiac tissues from chicken embryos during early heart development. Fig.1 shows an example FIB-SEM image. Selected examples of 224×224 training patches containing mitochondria and manually annotated ground truth are shown in Fig.2.

Our mitochondria segmentation approach is partly based on the idea of hypercolumn-based image segmentation first presented in [Hariharan et al.(2015)]. The authors define the "hypercolumn" at a given input location as the outputs

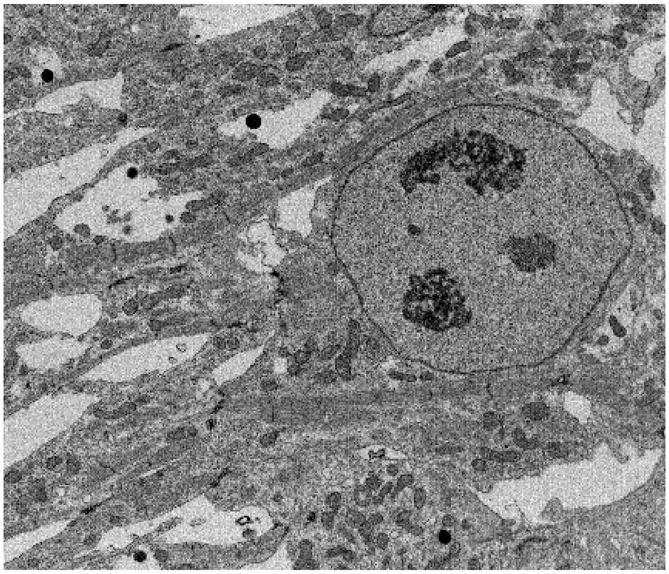

Figure 1: Example of a 1728 × 2022 pixel large FIB-SEM image of a cardiac cell (chicken embryo) containing a nucleus and a large number of mitochondria. Image Source: Oregon Health and Science University (OHSU).

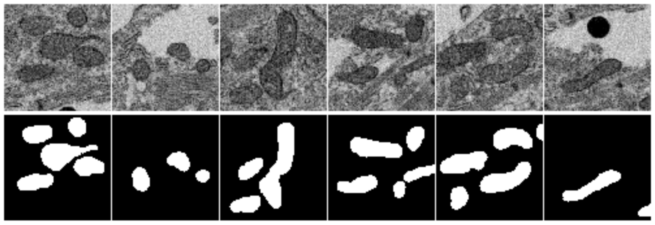

Figure 2: Selected 224 × 224 training samples and labeled mitochondria.

of all units above that location at all layers of the CNN, concatenated into one vector. The information that is generalized in the top layer is present in intermediate layers. Top layers are also more sensitive to semantics. Therefore, as the authors suggest, using multiple levels of abstraction and scale is needed to recover the information of interest which is distributed over all layers of the CNN. Adjacent layers may be correlated, so the authors suggest to sample a few. This proposed framework can solve fine-grained localization tasks by framing them as pixel classification and using hypercolumns as pixel descriptors.

The novelty of our paper is as follows: we combine the idea of hypercolumns with a pre-trained CNN (VGG-16 model) feature extraction step, perform supervised active feature selection using the L_1-regularized logistic regression, and train a gradient boosting classifier to obtain pixel-wise binary classification maps. To our best knowledge, such a few-shot segmentation pipeline has not been previously reported.

2 Related Work

Most approaches to the segmentation of mitochondria apply supervised machine learning with classifiers such as Adaboost, SVM, and Random Forest trained on handcrafted features. To compare the performance of the methods is difficult because they are based on different architectural ideas and are validated on different datasets. The handcrafted features used in the literature include Continuation Energy, Gradient flux, Haar, Radon, intensity and superpixels. These features can be applied to mitochondria in cases where the outer boundary is clearly defined by the membrane [Dietlmeier(2017)]. For irregularly shaped mitochondria, Ray features have been shown to outperform the Haar features [Smith et al.(2009)]. Superpixels are frequently used in combination with spectral graph-based methods to reduce the initial complexity of the input data [Ghita et al.(2014)]. Geometric descriptors such as, for example, Histograms of Oriented Gradients (HOG) [Dalal and Triggs(2005)] have been shown to be highly efficient in the task of human detection. HOG features are obtained by dividing the image into small connected regions and for each region compiling a histogram of gradient directions for the pixels within the region. Haar-like features are based on the gray intensity and are insufficient to describe an object's texture. As noted in [Smith et al.(2009)], Haar and HOG features are inefficient at detecting highly deformable objects such as biological cells and mitochondria.

Most of the reported CNN-based methods are purely supervised. The authors in [Leena and Govindan(2012)] combined a CNN to learn the affinity graph based on perceptual grouping constraints, with the Graph Cut algorithm. A modified CNN which has max-pooling layers instead of sub-sampling layers, was proposed in [Ciresan et al.(2012)] to segment neuronal structures in EM images. Ning et al. [Ning et al.(2005)] have combined the CNN with EBM (Energy Based Model) and [Marquez Neila et al.(2016)] have used CRF (Conditional Random Field) to model probabilistic dependencies. Ronneberger et al. [Ronneberger et al.(2015)] presented a new U-Net CNN architecture for biomedical image segmentation targeting the tasks of segmenting neuronal structures, cells and potentially mitochondria. The U-Net model is essentially an autoencoder, but with convolutions instead of a fully connected layer. The Bayesian SegNet architecture is proposed in [Khobragade and Agarwal(2018)] for multi-class segmentation of neuronal structures and mitochondria in electron microscopy images. It is an encoder-decoder architecture which maps the input to pixel-wise labeled output of the same resolution.

Some AI-based tissue segmentation approaches consider joint task of segmentation of nuclei and classification of cancerous tissue images. Development of accurate and efficient algorithms for these tasks is a challenging problem because of the complexity of tissue morphology and tumor heterogeneity. To address this challenge, [Vu et al.(2019)] presented two algorithms: one designed for segmentation of nuclei and the other for classification of whole slide tissue images. The segmentation algorithm implements a multiscale deep residual aggregation network to accurately segment nuclear material and then separate clumped nuclei into individual nuclei. The classification algorithm initially carries out patch-level classification via a deep learning method, then patch-level statistical and morphological features are used as input to a random forest regression model for whole slide image classification.

In recent years, many machine learning algorithms have been developed to extract features from histopathological images. In [Zheng et al.(2017)], a novel nucleus-guided feature extraction framework based on convolutional neural network is proposed for histopathological images. The nuclei are first detected from images, and then used to train a designed convolutional neural network with three hierarchy structures. Through the trained network, image-level features including the pattern and spatial distribution of the nuclei are extracted. With the nucleus-guided strategy, the network paid more attention to the difference in nucleus appearance and effectively reduced the noise and redundancy caused by stroma. [Manivannan et al.(2016)] proposed another approach based on ensembles of support vector machines (SVMs) for detection and classification of cellular patterns in tissue images. Ensembles of SVMs were trained to classify cells into six classes based on sparse encoding of texture features with cell pyramids, capturing spatial, multi-scale structure. A similar approach was used to classify specimens into seven classes. A comprehensive review of segmentation algorithms for digital pathology and microscopy images is provided in [Xing and Yang(2016)].

In a few-shot learning setting, the traditional machine learning algorithms attempt to learn from very few training samples. More specifically, few-shot learning aims to learn the pattern of new concepts unseen in the training data, given only a few annotated examples. Sometimes there is only one example available for each class [Dong and Xing(2018)]. Few-shot learning is an active research area, motivated by the fact that traditional deep learning methods require large amounts of training data. The availability of manually annotated data becomes even more challenging in segmentation since pixel-level annotation in segmentation task is more labor-intensive to acquire [Hu et al.(2019)]. In the literature, few-shot learning mainly focuses on the classification task and rarely on the segmentation and object detection [Fan et al.(2019)]. Few-shot segmentation task can be split into two components: detect the object in the scene and then segment it. [Michaelis et al.(2018)], for example, proposed a system that performs the detection part with a Siamese net applied in sliding windows over the scene to produce a heat map of candidate locations. The segmentation mask is then generated by a deconvolutional net with skip connections from the encoder. [Hu et al.(2019)] designed an Attention-based Multi-Context Guiding network (A-MCG) that incorporates multi-level concentrated context. The benefits of this processing pipeline are that the shallow part of the network generates low-level semantic features while the deep part captures high-level semantics. Inspired by few-shot classification, [Dong and Xing(2018)] proposed a generalized framework for few-shot semantic segmentation with an alternative training scheme. The framework is based on prototype learning and metric learning. Generally, few-shot learning has been investigated in many computer vision tasks such as image recognition and domain adaptation. However, the few-shot segmentation task is still considered underexplored [Dong and Xing(2018)].

3 Processing Pipeline

In this paper, we apply transfer learning in a few-shot supervised setting to extract the convolutional features and to train a gradient boosting classifier on the formed hypercolumns to classify pixels belonging to mitochondria. We adopt the CNN-based feature extraction with a gradient tree boosting method to capture complex non-linear mitochondrial morphologies in subcellular environments. Our processing pipeline is illustrated in Fig.3. In particular, we develop an automated image processing algorithm to extract a high dimensional mitochondrial feature set (2688 features) from each 224 × 224 input image and then use machine learning-based methods to build models for dense pixel-wise predictions. In contrast to most prior studies on segmentation of mitochondria where authors use handcrafted features, we automatically extract features from a pre-trained VGG-16 network. We reason that these features have good discriminative and generalization properties to be combined with gradient boosted decision trees to produce accurate pixel-wise binary predictions for mitochondria.

As can be seen from Fig.3, our processing pipeline has a hybrid model structure: the concatenation of a feature extractor (pre-trained VGG-16), sparse linear classifier (L1-LR) and boosted decision trees (XGB). We combine L1-regularized logistic regression with the gradient boosting implemented in the XGBoost package [Chen and Guestrin(2016)]. This particular implementation of gradient boosting is consistently used to win machine learning competitions on Kaggle. XGBoost also incorporates regularization to prevent overfitting.

The key advantage of L1-LR is the scalability to very large datasets as noted by [Zakharov and Dupont(2011)]. L1-LR is a linear model and its predictive performances might be limited in the presence of non-linear relationships in the data. On the other hand, gradient boosting decision trees have shown to learn higher-order interactions between the features. In particular, XGBoost is implemented with the gradient boosted decision trees, which in contrast to lasso and ridge regression methods, incorporates complex non-linear feature interactions into prediction models in a non-additive form [Chen and Guestrin(2016)]. Our combined pipeline is simple yet very effective to identify complex mitochondrial morphologies in a challenging FIB-SEM dataset.

We combine the hypercolumns [Hariharan et al.(2015)] with the feature extraction using convolutional layers of a pre-trained VGG-16 network presented in [Simonyan and Zisserman(2015)]. Convolutional feature maps in each block are resized to 224 × 224 using bilinear upsampling to form a hypercolumn for each pixel.

Hypercolumns have been introduced in [Hariharan et al.(2015)] with the motivation that most CNN-based recognition algorithms use the output of the

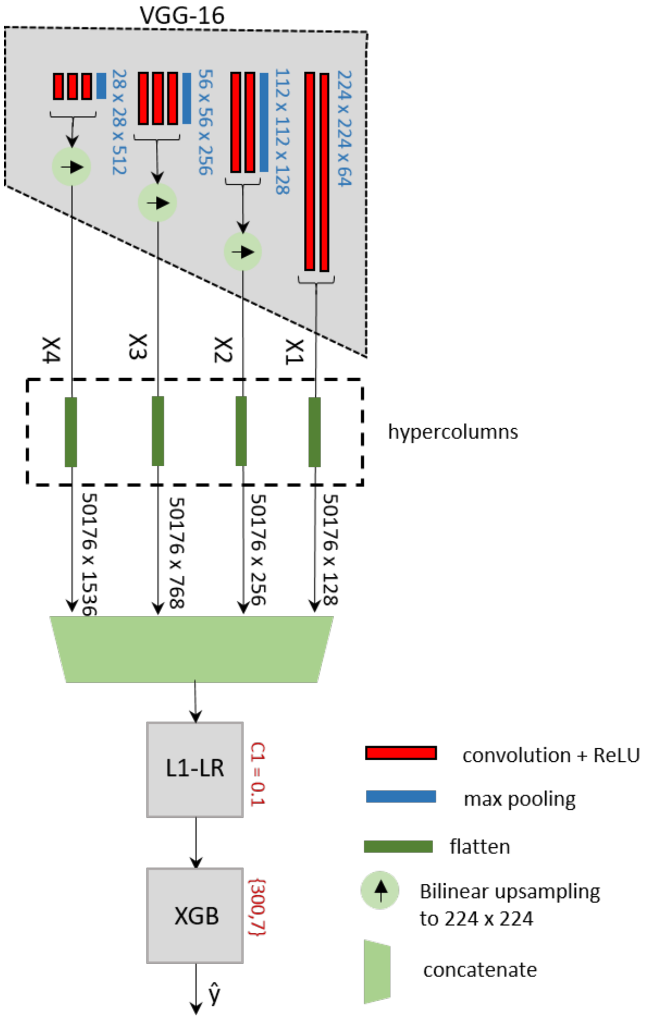

Figure 3: Our Processing pipeline. The input to VGG-16 during feature extraction is a 224 × 224 image. We extract features from the first four (X1 to X4) convolutional blocks of VGG-16 and further apply bilinear upsampling to the feature maps. For each image pixel we concatenate upsampled features and form hypercolumns. All concatenated features are passed through the L1-regularized logistic regression (L1-LR) followed by gradient boosted decision trees (XGB) which return binary predictions.

last layer as a feature representation. The information in this layer may be too coarse spatially to allow precise segmentation. Earlier layers are precise in localization but will not capture semantics. To combine these two aspects, the

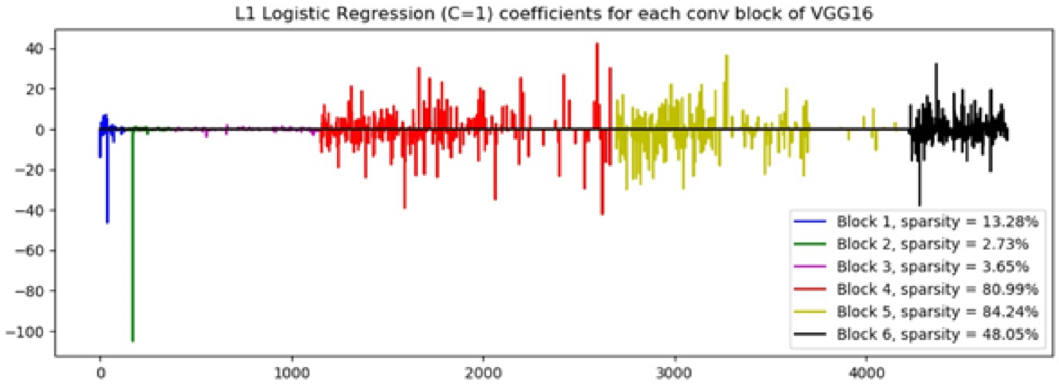

Figure 4: Importance coefficients of the L1-regularized logistic regression computed for each convolutional block of the VGG-16 network.

hypercolumn is defined at a pixel to be the vector of activations of all CNN units above that pixel.

We use the first four convolutional blocks X1 to X4 (see Fig.3). After bilinear upsampling, we flatten the features to 224 × 224 = 50176 long vectors and build the 50176 × 2688 large feature matrix where each pixel has a corresponding 2688-dimensional hypercolumn. We then perform feature selection with the L1-regularized logistic regression (L1-LR) and train the gradient boosting classifier (XGB) with very few training samples on selected features to obtain binary pixel predictions.

3.1 Feature extraction with VGG-16 network

VGG is a particular type of a CNN proposed by K. Simonyan and A. Zisserman from the University of Oxford in [Simonyan and Zisserman(2015)]. The VGG model achieves 92.7 % top-5 classification test accuracy on the ImageNet dataset which contains over 14 million images with 1000 classes. VGG is one of the best performing convolutional neural networks on the ImageNet challenge since 2015. Gradual increase in semantic complexity with the depth of the network is the key feature of the VGG-like CNN architectures. This hierarchical property facilitates the adaptability of extracted features across different datasets and tasks. Our motivation for using VGG is that these networks are especially suited for transfer learning because the learned representations progress from being simple and local to abstract and global. In addition, deep VGG-like networks are shown to generalize well to images other than the ImageNet dataset [Razavian et al.(2014)]. Thus, features extracted at lower levels of the hierarchy tend to be common across different tasks [Hadji and Wildes(2018)]. For example, CNNs trained with ImageNet for the classification task have been applied to other datasets [Lin et al.(2014)], [Zeiler and Fergus(2014)]. Texture recognition had been studied in [Cimpoi et al.(2014)], and other applications involved object detection and semantic segmentation [Girshick et al.(2014)]. To our best knowledge, we are the first to use a pre-trained VGG network for the task of mitochondria segmentation in the FIB-SEM data.

For our experiments we use the 16-layer deep VGG network implemented in the Keras library [Chollet(2018)]. The input to the VGG-16 is a fixed-size 224×224 RGB image. The image is passed through a stack of convolutional layers of decreasing size with filters having a very small receptive field of 3×3. Spatial pooling is carried out by five max-pooling layers, which follow some of the convolutional layers, to reduce volume size. A stack of convolutional layers is followed by three fully-connected layers: the first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels. The final layer is the soft-max classification layer [Simonyan and Zisserman(2015)].

In our experiments we extract features from the first four (X1 to X4) convolutional blocks (10 first convolutional layers) of VGG-16 as illustrated in Fig.3. We do not include the last two convolutional blocks in our experiments due to the small size of the feature maps ($14 \times 14 \times 512$ and $7 \times 7 \times 512$), which results in a very coarse resolution after bilinear upsampling.

3.2 Feature Selection

Our VGG-16 based feature extraction procedure results in a 2688-dimensional feature hypercolumn for each pixel in the input image. Therefore, for one 224×224 flattened input image sample we obtain a 50176×2688 feature matrix. To reduce dependencies and collinearities among the features, and to reduce overfitting due to a small sample size, we apply L1-regularized logistic regression (L1-LR) prior to training the XGBoost model. This feature selection process helps to identify active features and to reduce computational load of using the XGBoost algorithm. Feature selection, in general, has been shown to improve the interpretability and predictive performances of various classifiers [Zakharov and Dupont(2011)]. As can be seen from Fig.4, deeper VGG-16 layers have higher L1-LR coefficient sparsity. In particular, the convolutional block X4 (Block 4) has about 81% zero L1-LR coefficients.

To select active features from all (X1 to X4) convolutional blocks, we standardize the extracted features first and then use the L1-regularized logistic regression algorithm implemented in the scikit-learn Python package. This algorithm fits the sample to a logistic curve by minimizing a loss function based on the feature values. To reduce overfitting, we use the L1-regularization which minimizes the absolute difference of each feature from its predicted value. We run the feature selection process for all 2688 (X1 to X4) concatenated convolutional features and the obtained features with non-zero coefficients are identified as active features. The parameter C controls the sparsity of the L1-LR. For the value of $C = 1$ we, for example, obtained 46.2% reduction in feature matrix size. Smaller C values lead to stronger regularization and higher sparsity among L1-LR coefficients. Decreasing the C parameter to $C = 0.1$ results in 66.7% reduction in the size of the feature matrix.

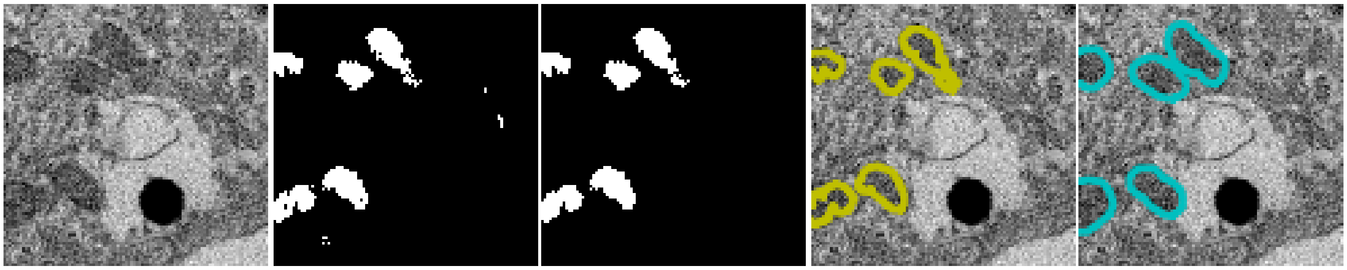

Figure 5: Example of mitochondria segmentation. From left to right: original 224 × 224 image, XGB prediction, postprocessed XGB prediction, segmentation contours (in yellow) overlaid on the original image, ground truth contours (in cyan). Diagram is best viewed in color.

3.3 Gradient Boosting Classifier

The last module in our processing pipeline is the gradient boosting classifier which produces binary prediction maps. Boosting, in general, is an ensemble approach for combining various learning models to create a more powerful predictor. By doing so, one can improve model predictions of any learning algorithm. The idea of boosting is to combine the outputs of several "weak" learners to build a more powerful ensemble with improved generalization [Friedman(2000)]. AdaBoost and Random Forest are two popular ensemble algorithms which together with handcrafted features are being used in a number of works on biomedical imaging. For example, AdaBoost has been used in [Smith et al.(2009)], [Narasimha et al.(2009)] for cells, in [Smith et al.(2009)], [Narasimha et al.(2009)], [Li et al.(2016)] for mitochondria segmentation and in [Becker et al.(2012)], [Navlakha et al.(2013)] for synapses segmentation.

Gradient boosting is similar to AdaBoost and it works by sequentially adding predictors to an ensemble, each one correcting its prior version. AdaBoost changes the weights for every incorrect classified observation at every iteration. On the other hand, gradient boosting tries to fit the new predictor to the generalized residual errors made by the previous predictor. Friedman showed that AdaBoost can be generalized to gradient boosting to handle a variety of loss functions [Friedman(2000)]. Gradient boosting has shown significant performance improvements in many classification problems as compared to classic AdaBoost [Caruana and Niculescu-Mizil(2006)].

We use the XGBoost algorithm, which is a fast and an efficient implementation of the gradient tree boosting method with fully tunable parameters. XGBoost stands for Extreme Gradient tree Boosting – an approach which has proven successful in several applications [Chen and Guestrin(2016)].We tune the following XGBoost hyperparameters (i) number of decision trees and (ii) size of decision trees which is used to control overfitting. Parameters for XGBoost have been identified using 2-fold cross-validation grid search procedure from *sklearn* Python package. The parameters returned are n_estimators=500 and

Table 1: Performance metrics used in validation

Metrics	Mathematical expression
Accuracy	(TP+TN)/(TP+TN+FP+FN)
Precision	TP/(TP+FP)
Recall	TP/(TP+FN)
F1-Score	2×Precision×Recall/(Precision+Recall)

Table 2: Comparison of mitochondria segmentation methods reviewed

Source	**Accuracy**	**Precision**	**F1-score**	**Microscope**	**Image size**	**Features**
[Kumar et al.(2010)] mouse neuropil	-	78%	0.8	EM	1024 × 1024 512 × 512	Radon-like
[Seyedhosseini et al.(2013)] mouse neuropil	-	82.51%	0.82	EM	700 × 700	algebraic curves
[Ghita et al.(2014)] ductulus efferens	98%	97%	0.96	EM/ssTEM	-	superpixels clustering
[Marquez Neila et al.(2016)] rat somatosensory cortex	96%	-	-	FIB-SEM	700 × 700	F2D
[Khobragade and Agarwal(2018)] Drosophila VNC	98%	-	-	ssTEM	1024 × 1024 512 × 512	Bayesian SegNet
Ours (20 training samples) Cardiac cells (chicken embryo)	96.74%	81.57%	0.76	FIB-SEM	1728 × 2022	CNN (VGG-16)

max_depth=5. Other parameters are as follows: gamma=0, subsample=0.75, colsample_bytree=1, min_child_weight = 1). We chose the learning rate parameter to be learning_rate=0.1.

4 Experimental results

A stack of 100 (1728 × 2022 pixels) sections from FIB-SEM of cardiac tissue from chicken embryos provided by the Oregon Health and Science University (Portland, OR, USA) is used in this study for training and testing our processing pipeline. We do not apply any preprocessing to the dataset.

We test the accuracy of our proposed segmentation method against manual annotations. The ground truth labeling is done with the Amira (Thermo Fisher Scientific) software. For the quantitative assessment we calculate Accuracy, Precision and F1-Score performance metrics which are the combinations of True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) as shown in Table 1.

Our training set consists of 20 training 224 × 224 patches from the first and the last image from the FIB-SEM stack (100 sections in total). All other

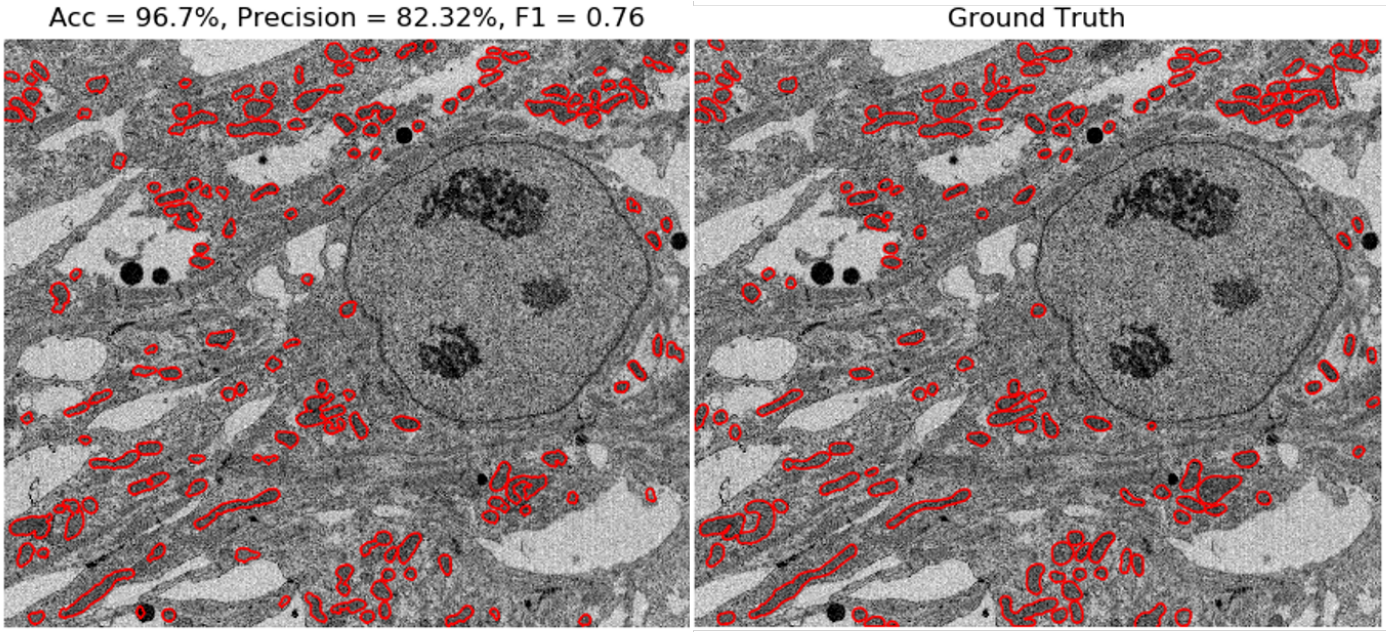

Figure 6: Selected 1728 × 2022 result (left) and the corresponding ground truth (right). Diagram is best viewed in color.

98 large-scale images are used for testing. Therefore, we are using only two 1728 × 2022 images for training purposes equivalent to a 2%-98% training-test split.

We also conduct experiments with random training-test splits. We select randomly two training images out of 100 and validate on the remaining 98 images. For 10 splits, we report the expected accuracy of 96.42% and the variance of 0.6%.

During training we concatenate flattened training patches into one long column. Due to the limited memory resources (32GB RAM on Dell Latitude 5580) we randomly subsample the features. For example, instead of using $224 \times 224 \times 20 = 1003520$ row dimension of the 1003520×2688 feature matrix, we only use a subset of n_sub=3000 rows for each image resulting in 60000×2688 feature matrix. This subsampling procedure results in about 94% compression rate of the feature matrix.

Currently, our block processing approach results in oversegmentation. We apply minor postprocessing with mathematical morphology to remove small objects, fill holes and close small concave regions. An example of postprocessing can be seen in Fig.5. Also, block processing results in structures detected inside the nucleus where mitochondria should not be present. To filter out the structures inside the nucleus we train our processing pipeline on two annotated nuclei images and during inference phase subtract the obtained nucleus XGB predictions from

Table 3: Ablation study results. Inference time is given per one 224 × 224 image.

Component	**Accuracy**	**Precision**	**F1-score**	**training time**	**inference time**
L1-LR only (C=10)	96.17%	74.17%	0.73	58.96 sec	0.65 sec
XGB only	96.76%	83.18%	0.756	7606.5 sec	6.53 sec
L1-LR + XGB	96.74%	81.57%	0.76	3272.69 sec	3.27 sec

the XGB predictions for mitochondria. An example of segmented large-scale 1728 × 2022 image and the corresponding ground truth annotation can be seen for comparison in Fig.6.

As shown in Table 2, we are the first to perform experiments on the large-scale FIB-SEM dataset. We are competitive with other approaches reviewed and we are the first to report results for large-scale 1728 × 2022 biomedical images, which contain large number of irregularly shaped mitochondria.

We perform an ablation study to gain an insight into performance of separate components of our pipeline. Here we systematically remove L1-LR and XGB to see how it affects performance. Table 3 presents the results. Performance figures are marginally the same, with the highest performance given by the XGB trained directly on VGG-16 features. The difference is in training and inference times: XGB is only marginally better than the complete pipeline (L1-LR + XGB) but requires significantly more computational time. We plot the learning curve for the cardiac cells dataset in Fig.7. It can be seen that by using only two training samples (two-shot learning) we are able to achieve the segmentation accuracy of almost 96%.

Next, mitochondria segmentation is performed to identify mitochondria within individual mouse auditory outer hair cells (OHC). OHC mitochondria are generally punctate in morphology making them an ideal subject for automated segmentation. Preliminary qualitative results using a few-shot training procedure on several high resolution TEM images (4512 × 3552 pixels) are presented in Fig.8. The speed of automatic segmentation will allow for rapid assessment of changes in OHC mitochondrial dynamics in large-scale FIB-SEM data sets. The objective of this experiment is to segment mitochondria in each of three OHC cells. First, we train our model to segment mitochondria in all cells present in the image. Then, we use two annotated middle cell images to train our model to recognize a middle cell in the unseen test data. This information about the cell outline is used to filter out mitochondria outside the middle cell. This experiment is part of our preliminary work on automated OHC quantification, and the ground truth is not available to compute performance metrics at this time.

Finally, we report results for the Drosophila VNC dataset. The state-of-the-art 98% accuracy is reported in [Khobragade and Agarwal(2018)]. We achieve 97.88% segmentation accuracy using single-shot training procedure. That is, we use only one resized 224 × 224 ssTEM image for training. Thus, the proposed single-shot approach on the Drosophila VNC dataset gives competitive perfor-

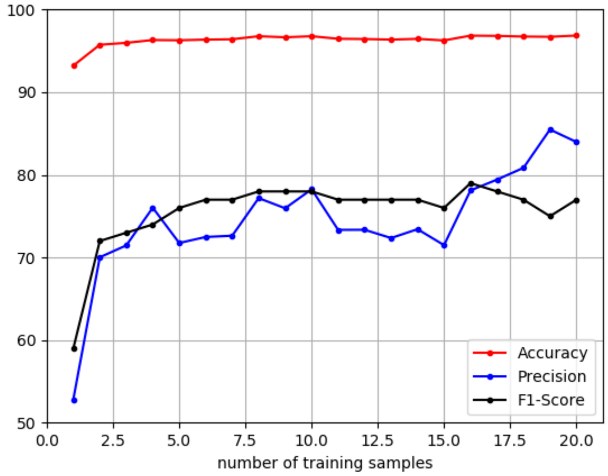

Figure 7: Learning curve of our few-shot segmentation approach. The first six training samples and the corresponding labels are shown in Fig.2.

mance with the state-of-the-art with far less training data and without data augmentation.

5 Conclusion

Despite being pre-trained on the ImageNet dataset, which contains RGB natural images and does not contain any mitochondria, extracted VGG-16 features have been shown to generalize well to the biomedical domain. Our processing pipeline is able to work with little training data in a few-shot segmentation setting. The method presented in this paper is a part of an ongoing work to develop an automated solution to segmentation of mitochondria in large-scale FIB-SEM images. In future we plan, therefore, to improve the accuracy and precision metrics.

Acknowledgments

The authors would like to thank Rachel Dumont as well as the OHSU Multiscale Microscopy Core and its Director, Dr. Claudia López, for technical assistance in acquiring the EM images. This publication was made possible with support from the following grants NIH R01 HL094570 (SR), R01 DC00105 (AN), and P30 NS061800 (Sue Aicher, PI). This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under grant number SFI/12/RC/2289 and SFI/15/SIRG/3283. The content is solely the responsibility of the authors and does not necessarily represent the official

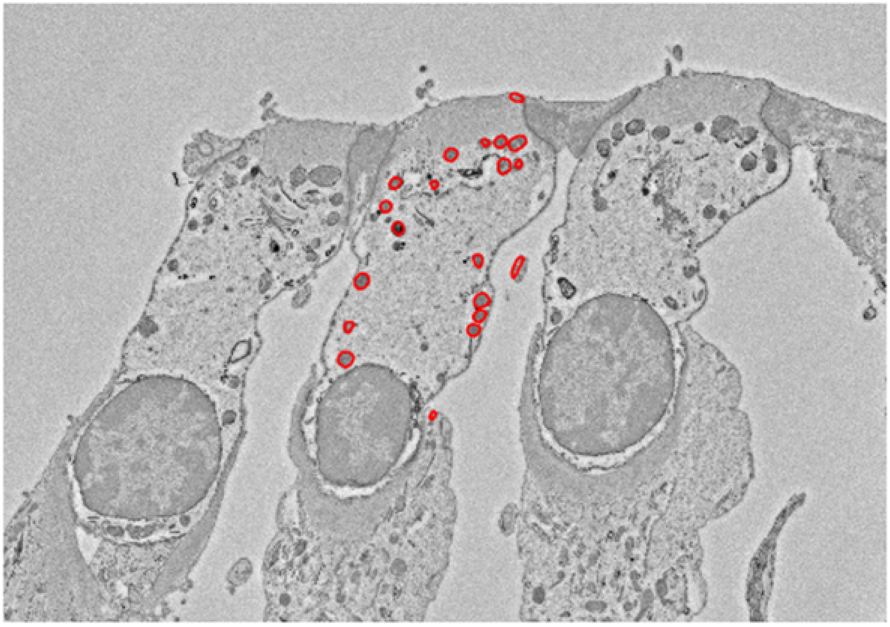

Figure 8: Mitochondria segmentation results for the middle OHC (outer hair cell). Diagram is best viewed in color.

views of grant giving bodies. We would like to thank anonymous reviewers for their valuable comments that helped to improve this manuscript.

References

[Becker et al.(2012)] Becker, C., Ali, K., Knott, G., Fua, P., 2012. Learning context cues for synapse segmentation. IEEE Trans. on Medical Imaging 31, 474—486.

[Caruana and Niculescu-Mizil(2006)] Caruana, R., Niculescu-Mizil, A., 2006. An empirical comparison of supervised learning algorithms. ICML .

[Chen and Guestrin(2016)] Chen, T., Guestrin, C., 2016. XGBoost: A scalable tree boosting system. 22nd Int. Conf. ACM SIGKDD .

[Chollet(2018)] Chollet, F., 2018. Deep Learning with Python. Manning Publications Co.

[Cimpoi et al.(2014)] Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A., 2014. Describing textures in the wild. CVPR .

[Ciresan et al.(2012)] Ciresan, D., Giusti, A., Gambardalla, L.M., 2012. Deep neural networks segment neuronal membranes in electron microscopy images. Advances in NIPS , 2843—-2851.

[Dalal and Triggs(2005)] Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. CVPR .

[Dietlmeier(2017)] Dietlmeier, J., 2017. A machine learning approach to the unsupervised segmentation of mitochondria in subcellular electron microscopy data. PhD Thesis. www.doras.dcu.ie .

[Dong and Xing(2018)] Dong, N., Xing, E.P., 2018. Few-shot semantic segmentation with prototype learning. BMVC .

[Fan et al.(2019)] Fan, Q., Zhuo, W., Tai, Y.W., 2019. Few-shot object detection with attention-rpn and multi-relation detector. arXiv .

[Fitschen et al.(2017)] Fitschen, J., Ma, J., Schuff, S., 2017. Removal of curtaining effects by a variational model with directional forward differences. Computer Vision and Image Understanding 155, 24–32.

[Friedman(2000)] Friedman, J., 2000. Greedy function approximation: A gradient boosting machine. Annals of Statistics .

[Ghita et al.(2014)] Ghita, O., Dietlmeier, J., Whelan, P., 2014. Automatic segmentation of mitochondria in EM data using pairwise affinity factorization and graph-based contour searching. IEEE Trans. on Image Processing 24, 4576–4586.

[Girshick et al.(2014)] Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. CVPR .

[Hadji and Wildes(2018)] Hadji, I., Wildes, R.P., 2018. What do we understand about convolutional networks. arXiv .

[Hariharan et al.(2015)] Hariharan, B., Arbelaez, P., Girshick, R., Malik, J., 2015. Hypercolumns for object segmentation and fine-grained localization. CVPR .

[Hu et al.(2019)] Hu, T., Yang, P., Zhang, C., Yu, G., Mu, Y., Snoek, C., 2019. Attention-based multi-context guiding for few-shot semantic segmentation. AAAI .

[Khobragade and Agarwal(2018)] Khobragade, N., Agarwal, C., 2018. Multiclass segmentation of neuronal electron microscopy images using deep learning. Proceedings of SPIE 10574.

[Kumar et al.(2010)] Kumar, R., Reina, A.V., Pfister, H., 2010. Radon-like features and their application to connectomics. CVPRW .

[Leena and Govindan(2012)] Leena, S.M., Govindan, V.K., 2012. Enhanced CNN based electron microscopy image segmentation. Cybernatics and Information Technologies 12, 84–97.

[Li et al.(2016)] Li, W., Rao, Q., Chen, X., Li, G., Zhang, D., 2016. Segmentation of mitochondria based on sem images. ICMA .

[Lin et al.(2014)] Lin, M., Chen, Q., Yan, S., 2014. Network in network. ICLR .

[Manivannan et al.(2016)] Manivannan, S., Li, W., Akbar, S., Wang, R., Zhang, J., McKenna, S., 2016. An automated pattern recognition system for classifying indirect immunofluorescence images of hep-2 cells and specimens. Pattern Recognit. 51, 12–26.

[Marquez Neila et al.(2016)] Marquez Neila, P., Baumela, L., Gonzalez-Soriano, J., Rodriguez, J.R., DeFelipe, J., Merchán-Pérez, A., 2016. A fast method for the segmentation of synaptic junctions and mitochondria in serial electron microscopic images of the brain. Neuroinformatics 14, 235—-250.

[Michaelis et al.(2018)] Michaelis, C., Bethge, M., Ecker, A., 2018. One-shot segmentation in clutter. International Conference on Machine Learning (ICML) .

[Narasimha et al.(2009)] Narasimha, R., Ouyang, H., Gray, A., McLaughlin, S.W., Subramaniam, S., 2009. Automatic joint classification and segmentation of whole cell 3d images. Pattern Recognition 42, 1067—-1079.

[Navlakha et al.(2013)] Navlakha, S., Suhan, J., Barth, A.L., Bar-Joseph, Z., 2013. A high-throughput framework to detect synapses in electron microscopy images. Bioinformatics 29, i9—i17.

[Ning et al.(2005)] Ning, F., Delhomme, D., LeCun, Y., Piano, F., Bottou, L., Barbano, P.E., 2005. Toward automatic phenotyping of developing embryos from videos. IEEE Transactions on Image Processing 14, 1360—1371.

[Razavian et al.(2014)] Razavian, A., Azizpour, H., Sullivan, J., Carlsson, S., 2014. CNN features off-the-shelf: an astounding baseline for recognition. CVPR Workshop .

[Ronneberger et al.(2015)] Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation. MICCAI .

[Seyedhosseini et al.(2013)] Seyedhosseini, M., Ellisman, M.H., Tasdizen, T., 2013. Segmentation of mitochondria in electron microscopy images using algebraic curves. ISBI .

[Simonyan and Zisserman(2015)] Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. ICLR .

[Smith et al.(2009)] Smith, K., Carleton, A., Lepetit, V., 2009. Fast ray features for learning irregular shapes. ICCV .

[Vu et al.(2019)] Vu, Q., Graham, S., Kurc, T., To, M., 2019. Methods for segmentation and classification of digital microscopy tissue images. Frontiers in Bioengineering and Biotechnology .

[Xing and Yang(2016)] Xing, F., Yang, L., 2016. Robust nucleus/cell detection and segmentation in digital pathology and microscopy images: a comprehensive review. IEEE Reviews in Biomedical Engineering 9, 234–263.

[Zakharov and Dupont(2011)] Zakharov, R., Dupont, P., 2011. Ensemble logistic regression for feature selection. Pattern Recognition in Bioinformatics .

[Zeiler and Fergus(2014)] Zeiler, M.D., Fergus, R., 2014. Visualizing and understanding convolutional networks. ECCV .

[Zheng et al.(2017)] Zheng, Y., Jiang, Z., Xie, F., Zhang, H., Ma, Y., Shi, H., 2017. Feature extraction from histopathological images based on nucleus-guided convolutional neural network for breast lesion classification. Pattern Recognit. 71, 14–25.