

# Flash Scanning Electron Microscopy

Raphael Sznitman, Aurelien Lucchi, Marco Cantoni,  
Graham Knott, and Pascal Fua

École Polytechnique Fédérale de Lausanne, Switzerland  
`firstname.lastname@epfl.ch`

**Abstract.** Scanning Electron Microscopy (SEM) is an invaluable tool for biologists and neuroscientists to study brain structure at the intra-cellular level. While able to image tissue samples with up to  $5nm$  isotropic resolution, image acquisition is prohibitively slow and limits the size of processed samples. In this work, we propose a novel approach to speeding up imaging when looking for specific structures. Unlike earlier methods, we explicitly balance the conflicting requirements of spending enough time scanning potential regions of interest to ensure that all targets are found while not wasting time on unpromising regions. This is achieved by using a Markov Random Field to model target locations and optimizing scanning locations by using a Branch-and-Bound strategy. We show that our approach significantly outperforms state-of-the-art methods to locate mitochondria in brain tissue.

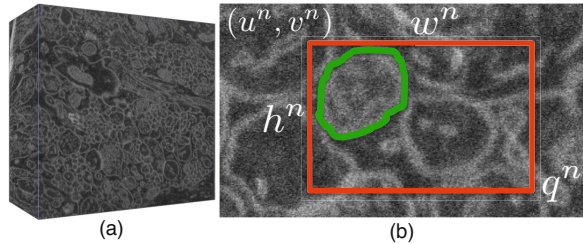
## 1 Introduction

Modern neuroscience has greatly benefited from Scanning Electron Microscope (SEM) technology. With its ability to image with  $nm$  and isotropic resolution, SEMs allow detailed visualization of intra-cellular structures as depicted in Fig. 1. This has already produced important advances in the study of neuro-degenerative diseases [1,2] and is expected to play a critical role in further understanding of brain function [3,4].

However, the very slow scanning speed of SEMs stands in the way of that promise because it severely limits the size of tissue samples that can be imaged. To image a volume, the electron beam has to scan the tissue surface multiple times, accumulate pixel information at each pass, and then average pixel samples to provide a clear and coherent image of the tissue surface. The top layer, or *slice*, is then removed using a diamond cutter or Focused Ion Beam and the process repeated for the newly visible layer. This process may take days for volumes as small as  $10\mu m^3$ . This is particularly burdensome for hypothesis testing procedures where the goal is to perform many different experimental manipulations and verify their effects on specific properties of tissue samples, such as counting the number of mitochondria or synapses per  $\mu m^3$ .

In this paper, we therefore address the problem of speeding up SEM imaging for the purpose of detecting and measuring specific intra-cellular structures. While the majority of related research has focused on providing automatic segmentation and labeling tools [5,6,7], very few have addressed this throughput

**Fig. 1.** (a) Volume generated from slice by slice SEM scanning. (b) Sample slice image with a mitochondrion outline drawn in green. In red, the scanned region  $q^n = (u^n, v^n, w^n, h^n)$  at time step  $n$ .



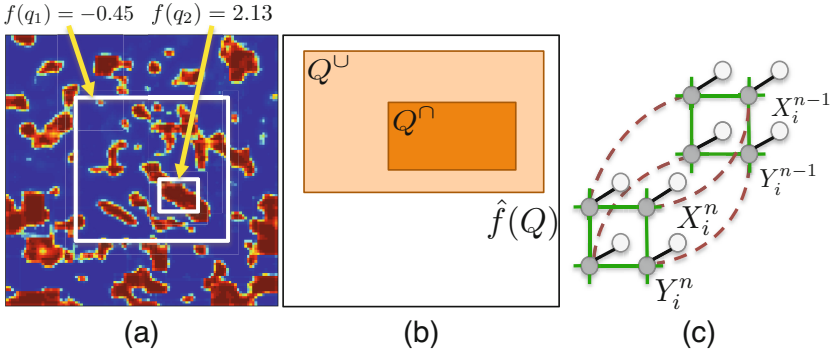
issue. Among these, hypothesize-and-scan strategies [7,8] save time by first scanning the image fewer times than typically required, using the resulting noisy image to hypothesize regions very likely to contain target structures, and re-scanning these regions. The procedure is repeated until only parts of the image that contain target structures are imaged with maximum quality. In [7], this is achieved by using a cascade of detectors to sequentially identify promising image regions. This requires correctly setting numerous parameters in the cascade and, as we will show in our experiments, is challenging to tune. By contrast, the scanning strategy of [8] is optimal in theory, but only when there is a single target in the field of view, an assumption that is systematically violated in practice and leads to inefficiencies. Furthermore, only results on simulated data are reported in these two papers [7,8].

Part of the problem with these earlier approaches is that they do not explicitly optimize for scanning time. The speed-ups are only byproducts of performing fewer scans at uninteresting locations. By contrast, in this work, we propose a hypothesize-and-scan approach that avoids the aforementioned problems by explicitly formulating the selection of regions to be scanned and re-scanned as finding the optimal compromise between maximizing the probability of finding target structures and minimizing the cost of scanning. In practice, this is done by using a Markov Random Field (MRF) to model the likelihood of targets being at different locations and performing mean-field approximate inference to estimate Maximum Posterior Marginals (MPM). Finding the optimal region to scan can then be solved exactly and yields regions to be re-scanned that are both compact and likely to contain targets. This strategy is simple to implement and we demonstrate significant accuracy and speed improvements on real brain tissue.

## 2 Flash Scanning for Target Discovery

The goal of our *Flash Scanning* approach is to scan as many times as necessary *all* the target structures and the rest as sparsely as possible. In essence, this means simultaneously satisfying two conflicting goals—minimizing the size of the scanned regions, while maximizing the number of targets within them—that is, finding a Pareto-Optimum.

To this end, for each slice, we first scan each pixel once and use a mean-field approximation to estimate the probability of a pixel belonging to a target



**Fig. 2.** (a) Visual representation of  $P(Y_i^n | X^n, Y^{n-1})$ , with high values in red and low values in blue. Two rectangular regions,  $q_1$  and  $q_2$ , are shown and their corresponding  $f$  scores. (b) Branch-and-Bound: depiction of regions  $Q^u$  and  $Q^n$  when optimizing the bounding function  $\hat{f}(Q)$ . (c) MRF depiction with 4-connected neighbourhood and dependencies on previous slices.

structure, based both on learned texture measures and the strong correlation between slices. We then find a rectangle that maximizes the integral of the probability density within it, minus its area multiplied by a fixed coefficient. In essence, the multiplicative coefficient selects one among many possible Pareto-Optima because the scanning time is proportional to the rectangle area. We then solve our optimization problem exactly using a branch-and-bound approach, re-scan the optimal rectangle, update our probabilities and iterate. This process is visually depicted in the supplementary video associated with this paper.

## 2.1 Notation

For a single slice,  $S$ , let  $n = 0, \dots, N$  be discrete time steps of our iterative process. Let  $I^n$  be the reconstructed image of the slice at step  $n$ , with  $I^0 = 0$  at each pixel location. Our goal is to scan the slice such that we reconstruct clearly all targets by time step  $N$ .

At each step  $n$ , we must hence direct the electron beam to scan a rectangular region of the slice **once** and denote this region  $q^n = (u^n, v^n, h^n, w^n)$ , as depicted by Fig. 1(b). Let  $C_p$  be the number of times pixel  $p$  has been scanned and let  $C_{\max}$  be the maximum number of times a pixel can be scanned. For each scanned region  $q^n$ , we acquire the corresponding new pixel intensities  $\{s_p^n\}_{p=1}^{|q^n|}$ , and incorporate them into  $I^n$  by weighted averaging for all  $p \in q^n$ ,  $I_p^n = I_p^{n-1}(1 - \alpha_p) + s_p^n \alpha_p$ , where  $\alpha_p = 1/C_p$ . Furthermore, let  $I^n$  be decomposed into  $r \times r$  image patches and let  $Y_i^n \in \{-1, 1\}$  be a random variable denoting the presence of a target in image patch  $i$  at iteration  $n$ . Also, let  $X_i^n \in \{-1, 1\}$  be a random image observation at location  $i$  and computed from the image  $I^n$ .

## 2.2 Objective and Optimization

Our goal at each time step  $n$  is to compute the next region  $q^{n+1}$  to scan in order to image as fast as possible all the targets with the highest accuracy. In particular, we want regions with high target probability to be scanned many times, while avoiding scanning large regions that are irrelevant. For this reason, we take  $q^{n+1}$  to be

$$q^{n+1} = \arg \max_{q \in \mathcal{Q}} f(q) = \arg \max_{q \in \mathcal{Q}} \sum_{i \in q} P(Y_i^n = 1 | X^n, Y^{n-1}) \mathbb{1}_{\{C_i \leq C_{\max}\}} - \lambda |q|, \quad (1)$$

where  $P(Y_i^n = 1 | X^n, Y^{n-1})$  is the conditional probability that a target is at patch  $Y_i^n$ ,  $|q|$  is the area of region  $q$ ,  $\mathcal{Q}$  is the space of all rectangular patch regions and  $\lambda > 0$  is a user parameter that balances accuracy and speed. In addition,  $\mathbb{1}_{\{C_i \leq C_{\max}\}}$  is 1 as long as  $C_i \leq C_{\max}$  and 0 there after, hence penalizing regions that have been over-scanned.

To give some intuition for the meaning of Eq. (1), Fig. 2(a) depicts the conditional probability at each location  $i$ , two candidate regions ( $q_1, q_2$ ) and their respective  $f$  scores. Observe that the objective favors regions very likely to contain targets, while penalizing large ones that are slow to scan, which results in densely packed target regions.

Given Eq. (1) is additive in  $P(Y_i^n = 1 | X^n, Y^{n-1})$ , we may use a Branch-and-Bound optimization strategy [9] to solve the global optimum exactly. The idea is to evaluate bounds on subsets of regions  $Q \subset \mathcal{Q}$  in a divide-and-conquer approach. This is achieved by defining a bounding function  $\hat{f} : \mathcal{Q} \rightarrow \mathbb{R}$  for subsets of regions. We write

$$\hat{f}(Q) = \sum_{i \in Q^\cup} P(Y_i^n = 1 | X^n, Y^{n-1}) \mathbb{1}_{\{C_i \leq C_{\max}\}} - \lambda |Q^\cap|, \quad (2)$$

where  $Q^\cup$  is the union of all rectangles in  $Q$  and  $Q^\cap$  is the intersection of all rectangles in  $Q$ , as depicted in Fig. 2(b). Clearly,  $\hat{f}$  is a proper bounding function as  $f(q) = \hat{f}(Q)$  when  $Q = q$ , and  $\hat{f}(Q) \geq \max_{q \in Q} f(q)$ .

## 3 Implementation

Given the optimization problem of Eq. (1), we now discuss how we compute the image observations  $X_i^n$  and the posterior distributions  $P(Y_i^n = 1 | X^n, Y^{n-1})$ , as well as our Flash Scanning algorithm.

**Target Detectors and Observation Model.** The observation  $X_i^n$ , describing the evidence of a target at location  $i$ , is computed directly from the reconstructed image  $I^n$ . As in [5], we use simple feature extraction with SVM classification to describe this evidence. Specifically, for node  $i$ , we compute 1) a feature dedicated to node  $i$  and 2) a feature from the 8-connected neighbours of node  $i$ . For 1) we compute for image patch  $i$ , the histogram of intensity (10 bins), the co-occurrence matrix (64 bins) and concatenate the two. For 2), we extract the

same features as in 1) from each neighbour of  $i$ , adding intensity histograms together and co-occurrences together. These are then normalized by the number of neighbours and concatenated, yielding a second 10+64 bin feature. Both 74-bin feature vectors are then concatenated and the estimator,  $X_i^n$  is the output of a trained RBF-kernel binary SVM (target=1 or background=-1) evaluated with the extracted 148 bin feature vector. Given that image noise varies as a function of how many scans have been performed at a location, a dedicated c-scan SVM is used when estimating regions observed with  $c$  scans. Note that other features may improve performances and in effect shift the pareto-optimum higher.

**Target Model and Posterior Update.** Given we are interested in inferring each of the variables  $Y^n$ , we use a Markov Random Field (MRF) to describe the joint probability distribution of all the target locations  $Y^n$  and observations  $X^n$ . We write

$$P(X^n, Y^n | Y^{n-1}) = \frac{1}{Z} \prod_i P(Y_i^n | Y_i^{n-1}) \prod_{j \in \mathcal{N}_i} P(Y_i^n Y_j^n) \prod_i P(X_i^n | Y_i^n), \quad (3)$$

where  $\mathcal{N}_i$  is a 4-connected neighbourhood of node  $i$  (depicted in Fig. 2(c)) and  $Z$  is a normalizing constant. This model allows us to describe the likelihood of a given patch of being a target, while taking into account it's local neighbourhood. Here, we have assumed that  $X_i^n$  is conditionally independent of  $Y_i^{n-1}$  given  $Y_i^n$ , and  $P(Y^n | Y^{n-1}) = \prod_i P(Y_i^n | Y_i^{n-1}) \prod_{j \in \mathcal{N}_i} P(Y_i^n Y_j^n)$ . This MRF model is convenient as it allows us to represent where all the targets lie on a current slice, as a function of occupancy in previous slices.

Within this MRF model,  $P(Y_i^n | X^n, Y^{n-1})$  is known as the Maximum Posterior Marginal (MPM). To compute these for all  $Y_i^n$ , we use the Mean-Field [10] approximation method, which involves setting  $P(Y_i^n | X^n, Y^{n-1}) = \mu_i$ , where  $\mu_i$  is initially 0 and then iteratively updating

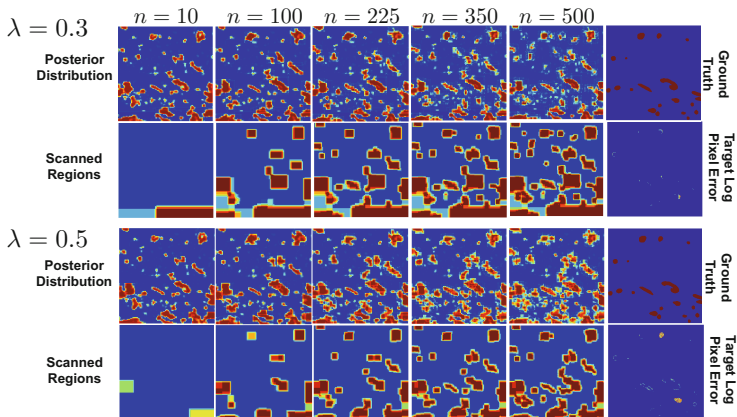
$$\mu_i = \mu_i \beta + (1 - \beta) \tanh\left(\sum_{j \in \mathcal{N}_i} \mu_j + P(Y_i^n | X_i^n) + P(Y_i^n | X_i^n, Y_i^{n-1})\right), \quad (4)$$

where  $\beta$  is a damping rate and  $P(X_i^n | Y_i^n) \propto (1 + \exp(g(X_i^n)))^{-1}$  with  $g(X_i^n)$  being the output of the SVM. In our experiments, we set  $\beta = 0.5$ , iterate 10 times for each node and set  $P(Y_i^n | Y_i^{n-1}) = \exp(Y_i^n Y_i^{n-1} / T)$  and  $P(Y_i^n, Y_j^n) = \exp(Y_i^n Y_j^n / T)$ , where  $T = 5$  is a temperature.

**Algorithm:** The proposed algorithm is summarized in Alg. 1. For each slice, the user provides an initial probability distribution for each node  $P(Y_i^0)$ ,  $\lambda$  and the total number of iterations permitted  $N$ . Both the final MPM and the scanned image  $I^N$  are returned by our procedure. For a given slice, we first scan each pixel location once (line 1), then iterate lines 3-6  $N$  times. For subsequent slices, we first apply temporal smoothing of the form  $P(Y_i) = P(Y_i)\alpha + 1/2(1 - \alpha)$ , where  $\alpha = 0.75$  in our experiments, and then use the smoothed probability as the prior for the next slice. As in [7], this is to account for the 3D nature of the targets and promote regions that have been heavily scanned in previous locations, as they are likely to be in nearby locations. For the first slice, we set  $P(Y_i^0) = 1/2$ .

**Algorithm 1.** Flash Scanning ( $P(Y^0), \lambda, N$ )

- 
- 1: Scan the entire slice one.
  - 2: **for**  $n = 1, \dots, N$  **do**
  - 3:   Select region to scan:  $q^n = \arg \max_{q \in \mathcal{Q}} f(q)$
  - 4:   Scan region  $q^n$  and update  $I^n$
  - 5:   Compute target estimators:  $\{X_i^n\}_{i \in q^n}$  from  $I^n$
  - 6:   Compute MPMs using Eq. (4)
  - 7: **end for**
  - 8: **return**  $P(Y^N)$  and  $I^N$
- 

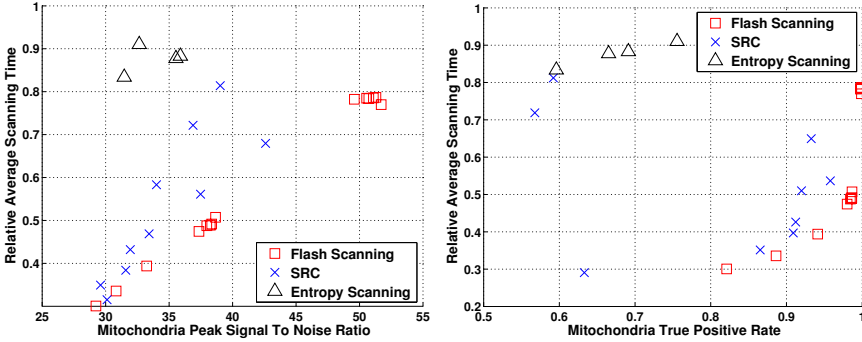


**Fig. 3.** Flash Scanning iterations for a slice when  $\lambda = 0.5$  and  $0.3$ . In each case, we show the evolution of the posterior distribution and the scanned locations for different values of  $n$ . At  $n = 500$ , we show the log-pixel errors for mitochondria locations (yellow are high errors and blue are low errors).

## 4 Experiments and Results

To test our approach we imaged 120,  $5nm$  thick slices using a Zeiss NVision40 FIBSEM microscope, such that each slice consisted of a  $1280 \times 1280$  image with  $5nm$  pixel resolution. Each slice was scanned a total of 10 times to produce 10 noisy  $1280 \times 1280$  images, *i.e.*  $C_{\max} = 10$ , and we collected the individual pixel values from each scan using a Fibics scanning head [11].

As in [7,8], we use mitochondria as our target structures. Mitochondria pixels were hand labeled in the entire 120-image stack by an expert. The first 20 images were used to train the different  $c = \{1, \dots, C_{\max}\}$  scan SVM classifiers as described in Sec. 3 and the rest for testing purposes. Each  $Y_i$  node of the MRF consisted of an image patch of size  $12 \times 12$  pixels. Computations were performed offline using a 2.3GHz PC. In practice, steps 3 and 6 of Alg. 1 take approximately 0.2 seconds, which is negligible considering the days of scanning performed during typical experiments.



**Fig. 4.** Scanning time, PSNR and TPR performances of Flash Scanning and earlier methods. For each method, a data point corresponds to a set of parameters used. In general, Flash scanning consistently outperforms earlier approaches by being faster and more accurate at scanning mitochondria.

**Evaluation:** In Fig. 3, we show a sequence of iterations of the Flash Scanning algorithm for a given slice with parameters  $N = 500$  and  $\lambda = \{0.3, 0.5\}$ . We show the evolution of the MPMs and the number of scans dedicated to each region of the space as  $n$  approaches 500 (large values in red, small values in blue). We also show the ground truth locations of mitochondria for the evaluated slice, as well as the log pixel error between the reconstructed images and the maximally scanned images at mitochondria locations.

From the figure, we see that our objective Eq. (1) concentrates scanning on target locations and generally avoids scanning irrelevant regions. We also see that when  $\lambda = 0.5$ , scanned regions are smaller and more compact than when  $\lambda = 0.3$ . This is consistent with Eq. (1) that imposes heavier costs for scanning large windows for large values of  $\lambda$ . In other words, larger values of  $\lambda$  means faster scanning at the risk of missing some target structures. This effect is clearly visible at  $n = 500$  where some mitochondria have not been scanned when  $\lambda = 0.5$  but have been when  $\lambda = 0.3$ . The full sequence of iterations can be viewed in our supplementary video.

**Comparison:** We evaluated our method against the SRC [7] and the Entropy Scanning (ES) [8] algorithms. For each tested method, including ours, we evaluated the scanning time relative to scanning the entire block with 10 scans, the Peak Signal-To-Noise Ratio (higher values being better) at mitochondria locations and the True Positive Rate (TPR) for regions scanned maximally. To show the effect of using different parameters, we ran our algorithm with different parameter values,  $\{N \in (300, 1000), \lambda \in (0.3, 0.6)\}$ . We did the same for both SRC and ES, while using the same SVM classifiers for all three methods. The results are shown in Fig. 4.

Flash scanning outperforms SRC, which itself does much better than ES. More specifically, for the tested set of parameters, Flash scanning significantly improves the speed and accuracy compared to SRC. For example, at roughly half traditional scanning times, our method has an average PSNR of over 37,

while SRC is of under 34. Similarly, for a PSNR of 37, SRC is 8% slower than our method. Note, that these differences are statistically significant, and similar observations can be made with respect to the reported TPR results. Note that similar results were also attained by performing experiments on synthetic data generated as described in [7].

## 5 Conclusion

We have presented a new method for efficiently scanning tissue samples using SEMs when imaging predefined targets. Our method uses an intuitive objective function that selects scanning regions that are likely to contain targets, but which are small in order to avoid taking too long to scan. In addition, our optimization is solved exactly and use a MRF to allow localization of multiple targets simultaneously. Our method requires far fewer parameters to tune compared to earlier approaches and out-performs them in terms of speed and accuracy. In the future, we plan to investigate how to integrate the cost of beam reposition when selecting regions to scan.

**Acknowledgements.** This work was supported in part by the ERC grant MicroNano.

## References

1. Knott, G., Marchman, H., Wall, D., Lich, B.: Serial Section Scanning Electron Microscopy of Adult Brain Tissue Using Focused Ion Beam Milling. *Journal of Neuroscience* 28(12), 2959–2964 (2008)
2. Van-Rijnsoever, C., Oorschot, V., Klumperman, J.: Correlative light-electron microscopy combining live-cell imaging and immunolabeling of ultrathin cryosections. *Nature Methods* 5, 973–980 (2008)
3. Alivisatos, P., Chun, M., Church, G., Greenspan, R., Roukes, M., Yuste, R.: The brain activity map project and the challenge of functional connectomics. *Neuron* 74, 970–974 (2012)
4. HBP: EU human brain project (2013), <http://www.humanbrainproject.eu/>
5. Lucchi, A., Smith, K., Achanta, R., Knott, G., Fua, P.: Supervoxel-Based Segmentation of Mitochondria in EM Image Stacks with Learned Shape Features 31(2), 474–486 (2011)
6. Laptev, D., Vezhnevets, A., Dwivedi, S., Buhmann, J.M.: Anisotropic ssTEM image segmentation using dense correspondence across sections. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part I. LNCS, vol. 7510, pp. 323–330. Springer, Heidelberg (2012)
7. Sznitman, R., Lucchi, A., Pjescic-Emedji, N., Knott, G., Fua, P.: Efficient Scanning for EM Based Target Localization. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part III. LNCS, vol. 7512, pp. 337–344. Springer, Heidelberg (2012)
8. Sznitman, R., Lucchi, A., Frazier, P., Jedynek, B., Fua, P.: An optimal policy for target localization with application to electron microscopy. In: ICML (2013)
9. Lampert, C., Blaschko, M., Hofmann, T.: Efficient Subwindow Search: A Branch and Bound Framework for Object Localization. *PAMI* 31, 2129–2142 (2009)
10. Murphy, K.: *Machine Learning: A Probabilistic Perspective*. MIT (2012)
11. Fibics (2012), <http://www.fibics.com>