

## MEDICAL IMAGE ANALYSIS WITH SEMANTIC SEGMENTATION AND ACTIVE LEARNING

CHARLES ISAH SAIDU<sup>1,\*</sup> AND LEHEL CSATÓ<sup>2</sup>

**ABSTRACT.** We address object detection using semantic segmentation and apply it for prostate detection in an MRI data-set. Our detection pipeline uses first a segmentation step followed by a classifier with a convolutional neural network (CNN). Since the segmentation provides a set of unbalanced data-sets – where a high accuracy is difficult to obtain – we leverage the prospect of improving detection accuracy using a Bayesian treatment of deep networks and the possibility of better exploiting the data using active learning. The resulting algorithm is both adaptive and data-efficient: by assuming that from a large pool of data only a few are segmented, the active learning module of the algorithm finds the image that improves most detection accuracy. We test our algorithm on a prostate medical image data-set and show that the active learning-based algorithm performs well in the prostate detection class. The resulting system is invariant to translations within the image and the results show improvements when using the pipeline that includes active learning and CNNs.

### 1. INTRODUCTION

Semantic segmentation is the task of *annotating* an image: which *region in an image* belongs to a defined class [14, 31]. Since images are pixels, in practice we have to *classify* every pixel – a difficult task since class information is not local to single pixels; consequently *semantic* refers to the global character of the segmentation task. In this work we focus on a mixed segmentation approach that involves both “traditional” image processing [28] as well as adaptive neural-network-based classification modules [3]. The image processing modules are used to *segment* the images into *superpixels* [2] that are subsequently fed into a neural network-based classifier that classifies the superpixel as belonging to the *region of interest* or not. Superpixels, the result

---

Received by the editors: September 25, 2018.

2010 *Mathematics Subject Classification.* 68T10, 68U10.

1998 *CR Categories and Descriptors.* H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval – *Clustering*; I.4.6 [Image Processing and Computer Vision]: Segmentation – *Edge and feature detection*;

*Key words and phrases.* semantic segmentation, active learning, deep networks.

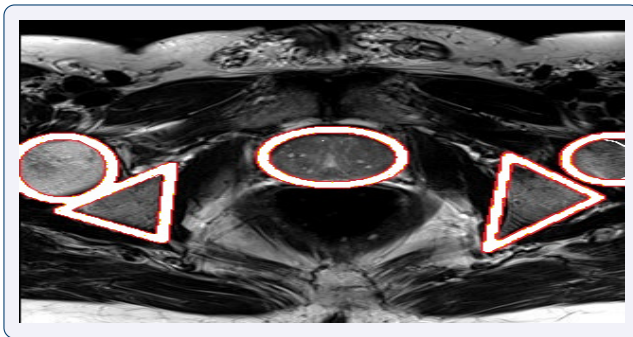


FIGURE 1. Axial scan of prostate gland region. The highlighted areas are similar to the prostate gland – in the centre – and difficult to discriminate, leading to classification errors.

of the semantic segmentation, are the *classification atoms* in this work, this simplifies the classification task and helps to better segment the image.

**1.1. Problem statement.** We aim for a good classifier that is able to “*select*” – or label – the regions of interest (ROI) in an image – an example is provided in Fig. 1. The learning is classification-based and we use a set of labelled images for training. We do not aim for a pixel-level classification and consider the existence of a pre-processing step that identifies the superpixels.

We aim to build a superpixel classification procedure that leads to an efficient labelling of the entire image. Our goal is to have a classifier that is:

- *adaptive* – it is able to learn from labelled data;
- *efficient* – it is able to learn from a small set;
- *generic* – the recognition is location-invariant – exploiting thus the higher-order invariants within the ROI, as illustrated in Fig. 1.

We focus on automated medical image segmentation: identification of the region of interest (ROI) within an image [see for example 19]. We use the prostate data-set from the I2CVB initiative,<sup>1</sup> one image is in Fig 1. This is a natural data-set and is considered as difficult to handle due to low contrast, speckle, micro-calcification in MRI images, as well as the presence of imaging artefacts [10].

**1.2. The suggested solution.** We focus on the classifier in the processing pipeline in Fig. 2 and use a *deep convolutional neural network* [3, 13] for classification and we optimise the learning procedure by using active learning. Active learning – also called “query learning” – is important since it allows us

<sup>1</sup>The database is available at <http://i2cvb.github.io/>

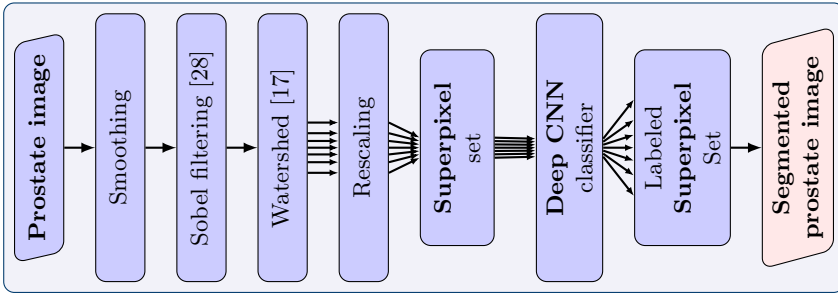


FIGURE 2. The processing pipeline starts with classical image processing, is followed by an adaptive classifier, a convolutional network. The annotated image is built from the labelled units.

to train the system using less data [7, 25]. Active learning is also important since the resulting supersixel classification problem is unbalanced [3]: out of  $\approx 60$  supersixels only a few are positive. Class imbalance leads to problems for the classifier: classifying all supersixels negative is a strong local optimum, and the learning might stop at this state. We use active learning and re-sampling techniques [4] to counter the class imbalance.

**1.3. Structure of the paper.** First, in Sec. 2 we list the related literature, then we discuss our proposed solution in Sec. 3. We test the proposed algorithm in Sec. 4, discuss our findings and further research directions in Sec. 5.

## 2. RELATED WORK

In this section we present two categories of research for medical image segmentation: graph-based segmentation and pixel-level classification based on deep networks.

**2.1. Graph-based semantic segmentation.** These methods build graphs where the nodes are supersixels and the weighted connections are computed using similarity measures between supersixels [7, 30, 31]. It is worth noting that the main motivation is to minimize annotation efforts by labelling groups of pixels – known as supersixels – instead of single pixels.

Vezhnevets et al. [31] used pairwise – label- and “connectedness” – *conditional random fields* (CRFs) over supersixels. The goal was to learn the parameters of this joint model using an energy function that captures both the ability to classify supersixels (unary potential) and the connectedness of supersixels to its neighbouring supersixels. Vezhnevets et al. [31] also applied active learning by designing a query scoring function that attempts to maximize the expected model change on the appearance model parameters.

Fathi et al. [7] focused on semantic video segmentation by building a graph of superpixels connected via a similarity metric. Here an incremental self-training approach was proposed that iteratively first labels the least uncertain frame, followed by the update of similarity metrics based on the extended set of labels.

**2.2. Semantic segmentation with deep networks.** Several techniques that are based on neural networks have been proposed for semantic segmentation and object detection. By far not exhaustive, we mention the work of Ciresan et al. [5] using deep networks, the fully convolutional network by Shelhamer et al. [27], U-NET semantic segmentation by Ronneberger et al. [22], R-CNN and “Fast R-CNN” for object detection [11, 12], “Faster R-CNN” by Ren et al. [21] and Masked R-CNN by He et al. [15].

Ciresan et al. [5]<sup>2</sup> used sliding windows approach. Each sliding window/image patch was labelled and subsequently used as input to a deep neural network. The method was difficult to train: there were a lot of overlapping patches for each image; making it computationally expensive. The patch size induced a trade-off between localization accuracy and patch size: larger patches require more layers and their size reduces localization [see e.g. 22].

Shelhamer et al. [27] extracted individual patterns from the image and classified patches independently. They use multiply connected convolution layers and no fully connected layers. Each convolution layer preserves the size of the input image – the output size is the same size as the input size. Each pixel in the output is annotated and the training was done using uses the cross-entropy likelihood.

Ronneberger et al. [22] proposed U-Net, an extension in architecture of Shelhamer et al. [27]. In their work, only a few training samples are required and the labelled data were in turn augmented. They used up-sampling operators in the decoding part of the network and applied the algorithm successfully to biomedical image segmentation.<sup>3</sup>

Girshick et al. [12] developed the “R-CNN” model for object detection. In contrast to the sliding window approach of [5], they used a selective search for sub-regions, and the ROI was determined using these regions only. The R-CNN model was later extended by the same team under the name “Faster R-CNN” [11]. We also mention “Mask R-CNN” [15] that is an extension of the original R-CNN [12] for object instance segmentation. This latter model uses a “simple model” for image class detection and one for segmentation.

---

<sup>2</sup>The work has won the electron-microscopy image segmentation challenge in 2012.

<sup>3</sup>The work has won the ISBI cell tracking challenge in 2015, [emb.citengine.com/event/isbi-2015/details](http://emb.citengine.com/event/isbi-2015/details), result presented at the ISBI conference (accessed 02.10.2018).

### 3. ACTIVE LEARNING USING DEEP NETWORKS

In what follows, we concentrate on the *classification task* – the *Deep Net classifier* – DNN – from Fig. 2. The inputs to the DNN are the superpixels and we know – see Sec. 1.2 – that the set of labels is extremely unbalanced: often we have 60 negative examples and a single positive one. To counter the negative effect of unbalanced data, a guided data selection – the *active learning* framework [25] – is used, in combination with gradient descent for neural networks and the probabilistic output – necessary for the scoring functions – is provided by the dropout mechanism [8]. We present first the neural network architecture, then the active learning framework, followed by a general overview of the algorithm.

In what follows  $X_u = \{x_1, x_2, \dots, x_n\}$  denotes the set of *re-scaled superpixels*. Since labelling is assumed to be difficult, we only have a small *labelled data-set*  $D_\ell = \{(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\}$ , with  $l \ll n$  and  $X_u = X \setminus X_\ell$  is the pool of unlabelled superpixels.

**3.1. Deep networks with dropout and probabilistic outputs.** For each image, we pre-segment the image using watershed algorithm, hence generating patches of images which we call superpixels - this is opposed to using the sliding-windows approach. These superpixels formed input to the deep neural network. Since each image produces many superpixels set and the task is to identify a region as prostate and not prostate, labelling all superpixels within an image would be laborious. So we endowed the deep neural network with a Bayesian treatment and the started by training only a small sample. The resulting sub-optimal classifier is then used together with active learning technique as a probe for searching the remaining unlabelled superpixels for ONLY informative superpixels that will improve accuracy and generalization. Hence, we use a “deep neural network” with both a *convolution* part and a *fully connected* part. We use  $4 \times 4$  convolution matrices, yielding 32 filters – since we aimed to capture pattern diversity. Each of the two convolution layer is followed by a *relu* non-linearity layer. After the last convolution level we have a *dropout layer* with dropout mean probability of 0.5. The fully connected part contains 3 dense layers each followed by a *relu* and a *dropout* 0.5 layers. The last layer is a *softmax* layer; the network error function is cross-entropy, and the training was done using the gradient descent with ADAM learning [18]. For network – i.e. classifier – optimisation we use gradient descent with the *tensorflow* package [1] and the optimisation uses only the dataset  $D_\ell$ .

Dropout – originally introduced to prevent model over-fitting [29] – uses a “blocking mechanism”: if a binary gate is open, the neuron output is calculated normally, if the gate is closed, the output of the respective neuron is zero.

Aside from its stabilising role, dropout can also be used to calculate predictive uncertainties [8, 9]: we sample the dropout gates and this leads to different weight vectors  $\omega_t$  – with some of the weights set to zero, providing in turn a predictive distribution given as:

$$(1) \quad P(y|x) \approx \frac{1}{T} \sum_{t=1}^T P(y|x, \omega_t)$$

with  $\omega_t = \omega \otimes g_t$  where  $g_t$  is a configuration of the dropout gates. We mention that the above scheme is valid only *if training was made using dropout*. We believe that dropout is important: Gal et al. [9] has shown that dropout is equivalent to performing an approximate Bayesian inference, therefore the samples are approximations of the “true posterior” distribution. The a-posteriori distribution is used to decide which superpixel will be included into the training set: it is based on scoring the super-pixels, the different *query scores* are defined in Sec. 3.2.

**3.2. Querying Technique.** The querying technique or acquisition function [26] defines a score for superpixels. Based on this score the unlabelled superpixels will be selected and labelled. The scoring is done such that it brings the most “information” conditioned on the already labelled data. We explored the following scoring functions:

- (1) *Maximum entropy* chooses the superpixels that maximize the predictive entropy [20]:

$$x^* = \arg \max_{x \in D_u} \mathbb{H}[y|x, D_u]$$

- (2) *BALD (Bayesian Active Learning By Disagreement)* looks for the superpixel  $x$  that maximizes the decrease in conditional entropy caused by the posterior [9, 16]:

$$x^* = \arg \max_{x \in D_u} \mathbb{H}[y|x, D_u] - \mathbb{E}_{p(\omega|D_u)} \mathbb{H}[y|x, D_u]$$

- (3) *Random acquisition*: randomly chooses a subset of unlabelled superpixels from the pool.

We compare the acquisition functions in the Results Section 4.3.

**3.3. Oversampling the data.** We found that – in spite of the results of Ertekin et al. [6] showing that active learning provides a natural way to handle imbalanced data – using active learning alone does not eliminate label imbalance. Instead, when running the full pipeline, we found that over-sampling of positive data and under-sampling negative data is useful.

Oversampling is a technique used to adjust class distribution of a dataset: the *synthetic minority oversampling technique*, or SMOTE, replicates samples based on the k-nearest neighbours of the under-sampled items [4]. Aside from

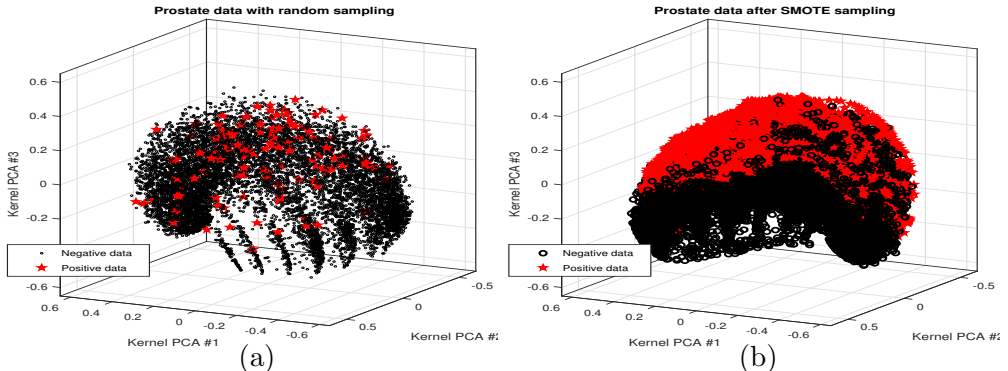


FIGURE 3. Superpixel visualisation – using kernel PCA – with (a) uninformed sampling and (b) informed SMOTE oversampling – image best viewed in colour.

the improved performance, the oversampling method was motivated from a visualization of the superpixel dataset from Fig. 3. In the visualisation we looked at the “topology” in the superpixel space: we wanted to assess the separability of the superpixels as defined by the classification problem. We used the kernel PCA method [24] – a non-linear projection methods that takes into account the distribution of the superpixels – and coloured the ROI superpixels as red, the negative data as black. We see that the oversampling makes the two classes more separable in the latent space.<sup>4</sup>

**3.4. The full algorithm.** We detail the adaptive “Deep CNN classifier” – shown in Algorithm 1 – the part of the image segmentation pipeline from Fig. 2. The input to the algorithm is the collection of resized and cropped superpixels.

To mimic real situation, we consider that superpixel labelling is done on-demand: when required, one can ask an expert to label an image, leading to a labelling of a superpixel set.

The training starts by labelling a small randomly selected set of superpixels – this was set to 50 to mimic the acquisition of superpixels from a single image.

After the initial labelling the algorithm proceeds by using the active learning technique described in sec. 3.2: we select from a pool of unlabelled superpixels  $D_u$  to add to the training set  $D_\ell$ , then we re-train the model.

The iteration follows as long as a stop-condition is not met or all data has been labelled, i.e.  $X_u = \emptyset$ . In our experiments we use a fixed number of iterations as stop-condition, this number resulted from testing the algorithm and finding that going over 30 iterations makes no difference in performance.

<sup>4</sup>Since the classification is not performed by a kernel machine [23], the plots provide only hints about the difficulty of the classification task.

**Algorithm 1** The Active Segmentation Algorithm

---

```

1: procedure TRAINING( $X_u$ )
2:   Select  $X_{init}$ ;  $X_u = X_u \setminus X_{init}$            ▷ superpixels from a single image
3:    $D_\ell \leftarrow \text{oracle}(X_{init})$                  ▷ labelling
4:   trainedModel  $\leftarrow \text{deepConvNet}(D_\ell)$        ▷ initial training
5:   repeat
6:      $X_{sub} \subset D_u$ 
7:      $S_{sub} \leftarrow \text{scoreSuperpixel}(\text{trainedModel}, X_{sub})$ 
8:      $k \leftarrow \arg \max\{s_i \mid s_i \in S_{sub}\}$      ▷ index of the “best” s.pixel
9:      $y_k \leftarrow \text{oracle}(x_k)$                  ▷ labelling the selected superpixel
10:     $X_k \leftarrow \{(x_k, y_k)\}$                  ▷ if oversampling, then
11:     $X_k \leftarrow \text{oversample}((x_k, y_k))$ 
12:     $D_\ell = D_\ell \cup X_k$ 
13:     $X_u = X_u \setminus \{x_k\}$ 
14:    trainedModel  $\leftarrow \text{deepConvNet}(D_\ell)$        ▷ re-training
15:  until stopCondition  $\vee X_u = \emptyset$ 
16:  return trainedModel

```

---

## 4. EXPERIMENTS AND RESULTS

The experiments used the pipeline from Fig. 2: we first perform a Gaussian smoothing, followed by a Sobel filtering [28], before using watershed to generate the superpixels. These patches – after labelling – will be combined into a *segmented* image as a result of the processing pipeline.

We use MRI axial scans from a total of 30 patients. Each patient’s database consists on average of 20 axial slices of the prostate region at different levels. For each image slice, we perform over-segmentation of the regions within the image and generate an average of 60 superpixels per slice, making an approximate total of 36,000 superpixels that can potentially be used for training.

**4.1. No-oversampling Experiment.** To handle location-invariance, we crop out each superpixel/patch and resize to a standard size of  $40 \times 40$  pixels before feeding into the convolutional network. This process further constrains and makes ROI detection a bit more difficult for this data-set. In particular – as in Fig. 1 – it is obvious that there are huge similarities in feature space between highlighted regions thus making it a lot more difficult for the classifier to learn which of these regions is prostate and which is not. In addition, our approach creates a hugely imbalanced data-set shown in Fig. 3.b, where we show a kernel PCA projection of the data-set. Consequently, we observed that active



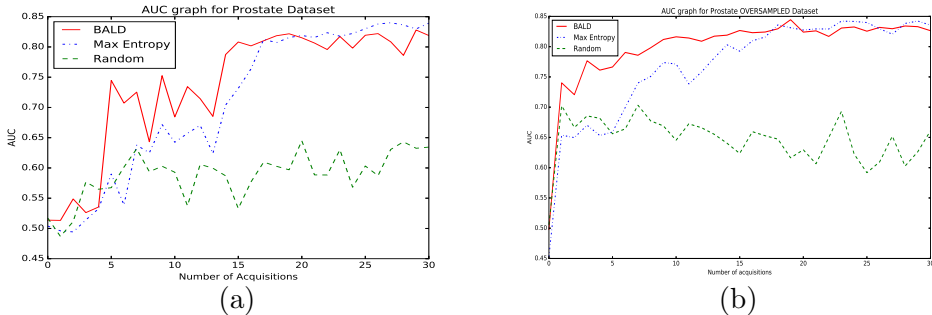


FIGURE 4. Area Under the ROC Curve (AUC) for (a) non-oversampled and (b) oversampled data-set.

learning alone did not deal implicitly with the issue of class imbalance as reported by [6] - A case that needs further investigation. However, after actively oversampling during each training step, considerable performance gains were observed as depicted in 4.3.

**4.2. Oversampling Experiment.** In order to evaluate the effect of feature space similarities in superpixels and class imbalance in the task of detection and agglomeration of the prostate region, we set up the experiment with a slight modification to the algorithm. This is prompted by the kernel PCA visualization of the data as a result of oversampling as shown in Fig. 3. In the modification, we oversampled *only after each batch acquisition from the data-set*. Algorithm 1 captures the idea if the oversampling flag set to *True*.

**4.3. Results.** Fig. 4 shows the results of successive Area Under the Receiver Operating Characteristic curve (AUC-ROC). From the figure, it would be misleading to think that performance improves as more superpixels are added to the training set. However, to the contrary, we obtained zero precision and zero recall when increasing the data-set size. Recall from Sec. 1.1 and the illustration in Fig. 1 that there is really very marginal distinction between regions of the prostate and non-regions of the prostate especially when the images are cropped, coupled with the fact that the imbalance ratio between negative and positive class is 60 to 1.

Consequently, we decided to over-sample the most informative superpixels after acquisition so as to give the algorithm more representation of what the actual prostate superpixels look like. This led to Fig. 4.b and Fig. 5 in which Fig. 5 shows improvement in precision of the prostate region after the 15-th acquisition.

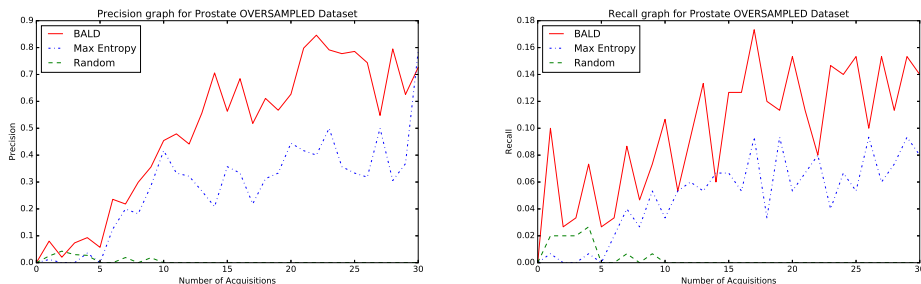


FIGURE 5. Precision (left) and recall (right) for oversampled data.

## 5. CONCLUSIONS AND FUTURE RESEARCH

We presented a semantic segmentation pipeline using active learning, providing a non-pixel-based adaptive segmentation; applied to medical images. Within the active learning, to achieve better segmentation results, we had to over-sample the positive class of superpixels, improving considerably the accuracy of the system, measured through the AUC curve. We also obtained a higher precision and recall compared to randomly selecting superpixels to be used for training the neural network. Overall, we observed that active learning technique could be complemented with oversampling techniques for better results.

In what follows, we plan to explore means of integrating active learning in the U-NET-style of semantic segmentation. Researchers measured the capability of the detection pipelines [22] using a pixel-wise matching of the desired and the true ROI. Applying the intersection over union (IoU) metric [22] means that the whole processing pipeline is evaluated, therefore one might want to optimise all other parameters also. An interesting prospect would be the use of 3D kernels for segmentation: given that there are several data-sets where the successive nature of the images can be exploited and the exploitation of this extra information is a promising research direction.<sup>5</sup>

**5.1. Acknowledgements.** C.I. Saidu wants to thank the African Development Bank for support via its home university. L. Csató acknowledges financial support of the European Regional Development Fund and the Romanian Government through the Competitiveness Operational Programme 2014-2020, project P/37/679, contract no. 157/16.12.2016.

<sup>5</sup>An example is the project “Lung cancer detection using 3D CNN’s” (link).

## REFERENCES

- [1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattemberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015.
- [2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.
- [3] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York, NY, USA, 1995. ISBN 0198538642.
- [4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.
- [5] D. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2843–2851. Curran Associates, Inc., 2012.
- [6] S. Ertekin, J. Huang, L. Bottou, and L. Giles. Learning on the border: active learning in imbalanced data classification. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, CIKM '07*, pages 127–136, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-803-9. doi: 10.1145/1321440.1321461.
- [7] A. Fathi, M. F. Balcan, X. Ren, and J. M. Rehg. Combining self training and active learning for video segmentation. In *Proceedings of the British Machine Vision Conference*, pages 78.1–78.11. BMVA Press, 2011. ISBN 1-901725-43-X.
- [8] Y. Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.
- [9] Y. Gal, R. Islam, and Z. Ghahramani. Deep Bayesian active learning with image data. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1183–1192, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.
- [10] S. Ghose, A. Oliver, R. Martí, X. Lladó, J. C. Vilanova, J. Freixenet, J. Mitra, D. Sidibé, and F. Meriaudeau. A survey of prostate segmentation methodologies in ultrasound , magnetic resonance and computed tomography images. *Comput. Methods Programs Biomed.*, 108(1):262–287, 2012. ISSN 0169-2607. doi: 10.1016/j.cmpb.2012.04.006.
- [11] R. Girshick. Fast r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.
- [13] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.
- [14] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1–8. IEEE, 2009.
- [15] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017.

- [16] N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel. Bayesian active learning for classification and preference learning. *ArXiv e-prints*, 2011.
- [17] Z. Hu, Q. Zou, and Q. Li. Watershed superpixel. *2015 IEEE International Conference on Image Processing (ICIP)*, pages 349–353, 2015. doi: 10.1109/ICIP.2015.7350818.
- [18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.
- [19] G. Litjens, R. Toth, W. van de Ven, C. Hoeks, S. Kerkstra, B. van Ginneken, G. Vincent, G. Guillard, N. Birbeck, J. Zhang, R. Strand, F. Malmberg, Y. Ou, C. Davatzikos, M. Kirschner, F. Jung, J. Yuan, W. Qiu, Q. Gao, P. Edwards, B. Maan, F. van der Heijden, S. Ghose, J. Mitra, J. Dowling, D. Barratt, H. Huisman, and A. Madabhushi. Evaluation of prostate segmentation algorithms for MRI: The PROMISE12 challenge. *Medical Image Analysis*, 18(2):359–373, 2 2014. ISSN 1361-8415.
- [20] D. J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, New York, NY, USA, 2002. ISBN 0521642981.
- [21] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015.
- [22] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.
- [23] B. Scholkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001. ISBN 0262194759.
- [24] B. Schölkopf, A. Smola, and K.-R. Müller. Kernel principal component analysis. In B. Schölkopf, C. J. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 327–352. MIT Press, 1999.
- [25] B. Settles. Active Learning Literature Survey. *Mach. Learn.*, 15(2):201–221, 2010. ISSN 00483931. doi: 10.1.1.167.4245.
- [26] B. Settles, M. Craven, and S. Ray. Multiple-instance active learning. In *Advances in Neural Information Processing Systems 20 - Proceedings of the 2007 Conference*, volume 20, pages 1289–1296, 2008. ISBN 160560352X.
- [27] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, Apr. 2014. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2572683.
- [28] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering, 2007. ISBN 049508252X.
- [29] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1): 1929–1958, Jan. 2014. ISSN 1532-4435.
- [30] A. Vezhnevets, V. Ferrari, and J. Buhmann. Weakly Supervised Semantic Segmentation with Multi Image Model. *Proc. Int’l Conf. Comput. Vis.*, 2011.
- [31] A. Vezhnevets, J. M. Buhmann, and V. Ferrari. Active learning for semantic segmentation with expected change. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pages 3162–3169, 2012. ISSN 10636919.

<sup>1</sup> COMPUTER SCIENCE DEPARTMENT, AFRICAN UNIVERSITY OF SCIENCE AND TECHNOLOGY, AIRPORT ROAD, 10 KM, ABUJA, NIGERIA

<sup>2</sup> FACULTY OF MATHEMATICS AND COMPUTER SCIENCE, BABEȘ-BOLYAI UNIVERSITY, 1  
KOGĂLNICEANU, RO-400084 CLUJ-NAPOCA, ROMANIA

\* WORK PARTIALLY DONE WHILE AT INTERNSHIP AT BABEȘ-BOLYAI UNIVERSITY  
*Email address: isaidu@aust.edu.ng and lehel.csato@cs.ubbcluj.ro*