Automatic Detection of Tumor Cells in Microscopic Images of Unstained Blood using Convolutional Neural Networks

Ioana Mocan, Razvan Itu, Anca Ciurte, Radu Danescu
Computer Science Department
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
radu.danescu@cs.utcluj.ro

Rares Buiga
Anatomic Pathology Department
The Oncology Institute "Prof. Dr. Ion Chiricuta"
Cluj-Napoca, Romania
raresbuiga@yahoo.fr

***Abstract*—Accessible high-performance computing power has recently spiked interest in medical image analysis and processing. Biomedical image segmentation has been used to aid in the process of medical analysis and diagnosis. In this paper we present a novel approach to identifying circulating tumor cells (CTCs) using convolutional neural networks on Dark Field microscopic images of unstained blood. We use a modified U-Net that is able to automatically perform image segmentation in order to detect CTCs. We perform detection on our own dataset containing input images and ground truth label images. Detection is done on small image patches using a sliding window mechanism in order to reduce computation time. The final result is reconstructed from the patches and further refined using post-processing. The total number of CTCs is computed from the segmented image using the Hough circle algorithm. We were able to obtain over 99.8% accuracy using our data set.**

Keywords—CTC; cancerous tumor cell; automatic detection; convolutional neural network; segmentation; image processing

I. INTRODUCTION

Medical imaging has been an active interest for many researchers throughout the years. Recently, the field has seen a boost due to the popularity and the continuous development of the convolutional neural networks. These artificial neural networks are based on the convolution operation applied on images and are widely used in different fields, ranging from medical imaging, autonomous vehicles, robotics, and so on. Medical image segmentation represents a complex task. The scope of the presented work is to identify cancerous cells in Dark Field microscopic images of unstained blood. State of the art microscopic image acquisition techniques for living cells are: Phase Contrast, Digital Holographic Microscopy, Dark Field and Differential Interference. The images obtained using Dark Field technique have the appearance of bright objects on a dark background. The acquisition process consists of illumination of the specimens from a lateral point of view, thus being a non-invasive process, meaning that the living cells are not altered or stained.

978-1-5386-8445-0/18/$31.00 ©2018 IEEE

II. RELATED WORK

Interest in computerized diagnostics increased lately due to the higher computing power and the high-performing mining algorithms developed in the last decade. Biomedical image segmentation has been used in recent years to aid in the process of medical analysis and also diagnosis. Such mining algorithms are able to identify new distinctive markers that bring added value to the diagnosis made by the human expert. Convolution neural networks in particular are designed to learn and identify distinctive patterns based on visual representation. Such approaches using CNN are widely used in medicine from micro to macro level. For example, in [1] the authors identified new histopathology image features that directly correlate with specific gene expressions of breast cancer. In [2] authors use deep convolutional neural nets for steatosis detection in ultrasound images of liver. More outstanding examples can be found in [3] where the authors provide an interesting survey on Deep Learning applied on medical image analysis. For cell detection, important results were obtained for problems such as nuclei detection in histopathological images [4] or cell counting [5].

Circulating tumor cells (CTCs) are malignant cells that are spread into the bloodstream. CTCs represent the mechanism by which cancer is spread to other organs. State of the art methods for detecting CTCs and to separate them from blood cells benefit from using their molecular characteristics or physical attributes [6]. The common approach to detect CTCs is as follows: the first step consists of preparing the samples and isolating the tumor cells. The second step is tumor cell staining or oncogene probing and the final step consists of actual detection of tumor cells. This procedure is not only laborious, but also destructive, killing the blood cells, which means that it cannot be applied to re-circulated blood.

In [7], an analysis of image features for CTCs in Dark Field Microscopy was performed, and it was proven that they have a discriminant appearance in comparison with healthy blood cells. This work uses classifiers to discriminate between CTC and non-CTC cells. The work presented in [8] uses SVM and CNN

classifiers to classify image patches into CTC and non-CTC regions, after scaling them down to a fixed size.

The problem of using a binary classifier on image patches is that it has either to rely on cell segmentation, which can be inaccurate, or on a sliding window approach, which may be time consuming and may generate multiple classifications for the same cell, thus a post-processing is required. In our work we propose to use a convolutional neural network for pixel-level segmentation, to directly highlight the CTC regions in the image. There will be no need for sliding windows, but we will use 50% overlapping windows to ensure that each pixel can be seen in a proper context. The neural network is trained using ground truth from the stained version of the samples, therefore no manual labeling of the pixels is necessary.

Post-processing of the highlighted pixels will identify the individual cells and count the cells (which is the most important aspect from the clinical point of view).

III. Convolutional Neural Network

A. *Neural network architecture*

Segmentation is performed with the use of a convolutional neural network (CNN). The basic operation of the CNN is to apply a convolution operation on the input image, using a kernel with a smaller size than the input image. By using kernels (filters) on the input image, we obtain activation maps. In order to preserve important local features, the activation maps are further processed using nonlinear functions, usually rectified linear function (ReLU) due to its improved performance. The convolutional neural network used in this work is largely based on the U-NET [9] architecture that is widely used for segmentation purposes and image classification tasks. U-NET is based on an encoder-decoder style, meaning that it can learn segmentation having the raw original image as input and a segmentation image as output. The main advantage of the U-NET is that it performs well when dealing with touching objects that are part of the same class and when using a lower number of annotated images for training.

Most operations in the network represent 3 x 3 convolutions that are followed by non-linear activation functions (ReLU). Max pooling is then used to reduce the size of the feature map and actually propagates the maximum activation from each 2 x 2 window to the next feature map. The feature channels are increased (by a factor of 2) after each max pooling operation, meaning that all of these convolutions and max pool operations will have the effect of spatially contracting the input. In order to generate the segmentation map output, a series of 2 x 2 up-convolutions and concatenations with original (high resolution) features of the input is performed. Each up-convolution is followed by a non-linear activation function.

In this work, we have used a modified U-NET which features 7 steps in the encoding part and 7 decoding steps. Each step down (encoder) is characterized by the following layers and operations:

- 3 x 3 convolution
- batch normalization and non-linear activation (ReLU)
- 3 x 3 convolution
- batch normalization and ReLU
- 2 x 2 max pooling

The decoder layers are characterized by the following:

- 2 x 2 up sampling operation
- 3 x 3 convolution
- batch normalization and ReLU
- 3 x 3 convolution
- batch normalization and ReLU
- 3 x 3 convolution
- batch normalization and ReLU

There are also 2 convolution layers in the center of the network, after the encoder and before the decoders.

The neural network layers are illustrated in figure 1.

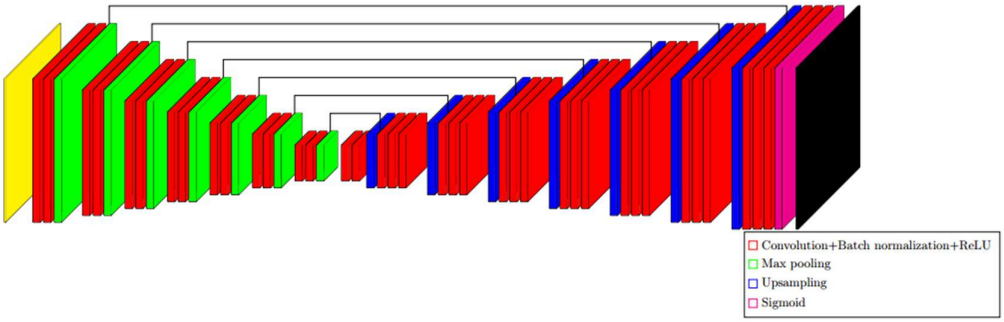

Fig. 1. The neural network architecture.

The batch normalization is used to normalize the output of the previous activation layer. Batch normalization operation will increase the stability and accuracy of the network and also reduce the training time.

B. Data acquisition

The input is a 3 channel RGB image of Dark Field images. Dark Field imaging has been used due to better image quality, where the tumor cell structure is emphasized. The images were acquired using Ximea XiQ camera mounted on an Olympus BX43 fluorescence microscope at a magnification of 20X. The samples were obtained from the Oncology Institute of Cluj-Napoca. DLD-1 cell line of colorectal cancer were used for our tests, which we refer as CTCs. The samples were prepared using blood from two healthy donors that was spiked with CTC's. In order to have an accurate validation of our results, CTCs were marked with the PKH26 fluorescent agent, and during the acquisition we stored pairs of images: one in darkfield – the input for our method, and one in fluorescence – playing the role of ground truth. No manual assistance for pixel labeling was necessary. An example of such pair of images is shown in figure 2.

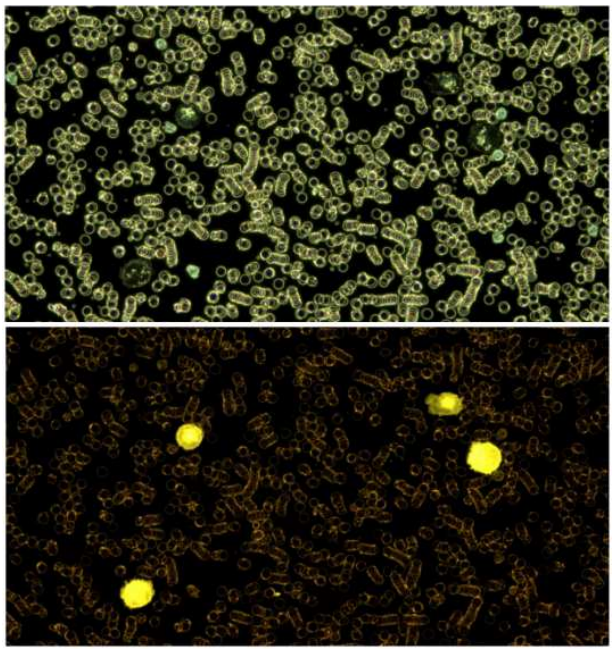

Fig. 2. Top: Dark Field microscopic image – the input image from the data set. Bottom: CTCs marked with PKH26 fluorescent agent – the input ground truth (label).

C. Dataset preparation and augmentation

The initial dataset consists of 120 images that are actually at a resolution of 2040 x 1080 pixels. The dataset was pre-processed and the input images split into 128 x 128 patches that are centered around the cancerous cells, meaning that the patches contain the actual and most relevant features that we want to detect. By generating these small patches, we were able to obtain over 56000 images. The ground truth (label) images for the neural network are generated from the fluorescent images (figure 2 bottom) by converting them to grayscale and then applying a Gaussian filter to reduce the noise. The next step is to perform binarization, followed by erosion to further eliminate the noise. In the resulting image we search for contours and compute the area of each detected contour. Objects with an area smaller than 1200 pixels are eliminated. We compute the center of mass for each remaining object and then generate the small 128 x 128 patches centered around this point. The process is illustrated in figure 3.

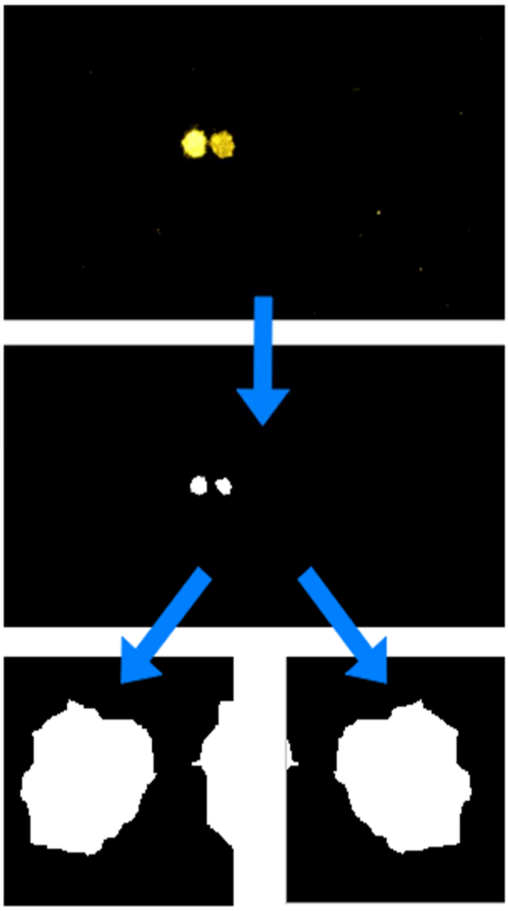

Fig. 3. Generating the patches from the original ground truth labels.

The dataset was divided into training and testing data sets (70% - 30%). We then performed data augmentation on the initial images. For this approach, we have used only 3 augmentation techniques: shifting, scaling and rotating images. Another common augmentation technique is to randomly flip the images (ex: horizontally or vertically), but we didn't use it. An example of extracted patches is illustrated in figure 4.

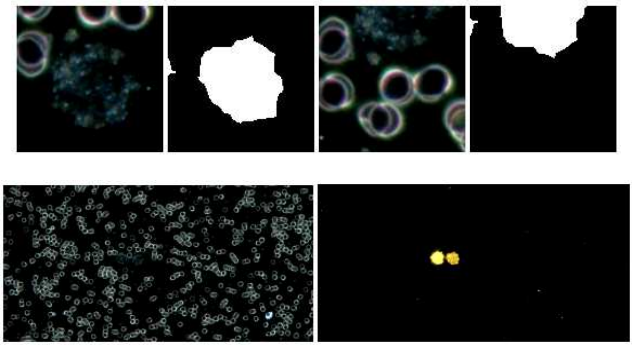

Fig. 4. Examples of patches (top) extracted from the original images (bottom). The original image source (left side) and the ground truth (right side) containing CTCs.

Figure 5 shows an example of the original microscopic image in Dark Field (left) along with fluorescence microscopic image of the same view (right) representing the ground truth. In figure 4 are actually the patches extracted from the original input image illustrated in figure 5. In figure 4, the second and fourth images represent the CTCs ground truths corresponding to the first and third images.

D. Network training

For training we have used a desktop computer equipped with Intel i7 CPU and 2 x Nvidia GeForce 1080 Ti GPUs to speed up the process. We have used the Adam optimizer with a learning rate of 0.0001, an extension of the stochastic gradient descent that automatically adjusts the learning rate and a modified intersect over union (IoU) as a loss function during training. The objective for the network is to minimize the loss, meaning that we need to define a metric that favors predicted masks that are heavily overlapped over the ground truth masks. IoU is a popular metric that computes the intersect over union between the output of the network (the prediction) and the labeled input mask (the ground truth). The IoU is ranged between 0 and 1 (1 being equivalent to fully overlapped images), therefore in order to maximize the IoU, we actually have to minimize the negative of the IoU result during training of the network.

To obtain a lower computational time during each epoch, we have used a modified Sorensen-Dice coefficient [10] for computing the IoU loss:

$$IoU_{loss} = \frac{2(predicted_{output} \times label)}{predicted_{output} + label} \qquad (1)$$

To determine the modified Sorensen-Dice coefficient [9], a measure of similarity between two images, we must compute: 2 times the intersection between the predicted output and the ground truth, divided by the sum of both.

For deploying the network, we have used the Keras framework on top of TensorFlow library and Python as programming language.

IV. *POST-PROCESSING*

We do post-processing in order to further refine the detection results of the CNN. The general aspect of CTCs is circular, rounded, meaning that we can take advantage of this geometric property. Therefore, we use the Hough circle algorithm to detect the circular shapes in the prediction image. We further trim the false positives by using a fixed threshold of 1200 pixels on the Hough circles area. This value was determined experimentally, we found that circular objects smaller than 1200 pixels usually represent white cells. In figure 6 we show the results after post-processing.

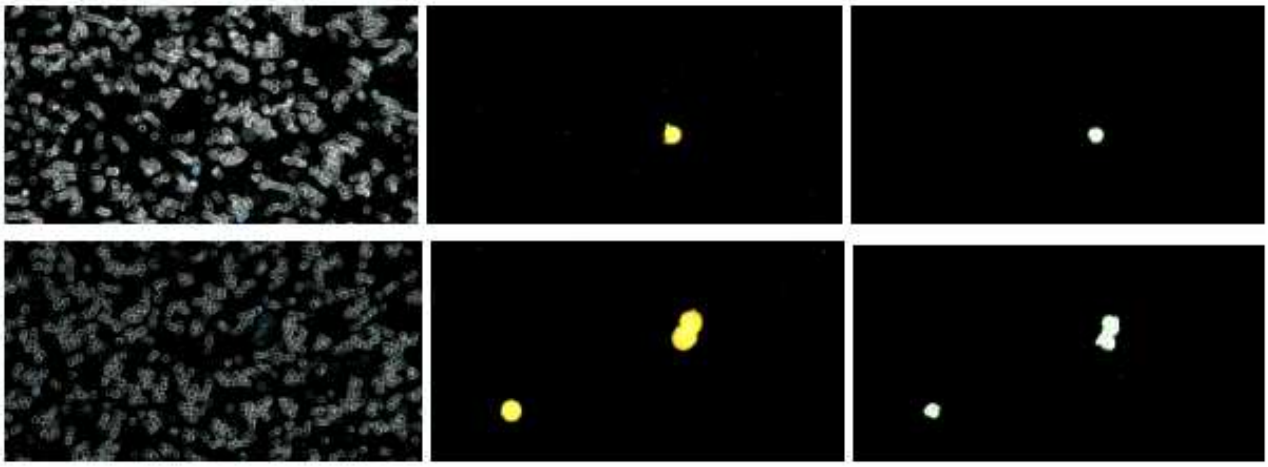

Fig. 5. Results after post-processing: left column contains the input images, the center column contains the ground truth, and the right column contains the post-processed segmentation result.

V. RESULTS

The network takes between 0.02 seconds and 0.03 seconds to predict on a 128 x 128 image, meaning that the proposed solution could achieve a frame rate between 30 - 50 Hz. Figure 7 illustrates some detection results using the trained artificial neural network.

In order to generate the final prediction, we apply the classifier on overlapping windows (50% overlap) for segmenting the whole image. The segmentation result is compared to the ground truth, and the accuracy is computed using equation (2), where TP denotes the number of true positives, TN the number of true negatives, FP the number of false positives, and FN the number of false negatives. Each of these measures refers to pixels, which are correctly or incorrectly labeled.

$$ACC = \frac{TP+TN}{TP+FP+FN+TN} \qquad (2)$$

The accuracy of the segmentation is estimated on the test set, images that were not used in the network training process. The graph in Figure 8 represents the pixel classification accuracy for multiple images of the test set.

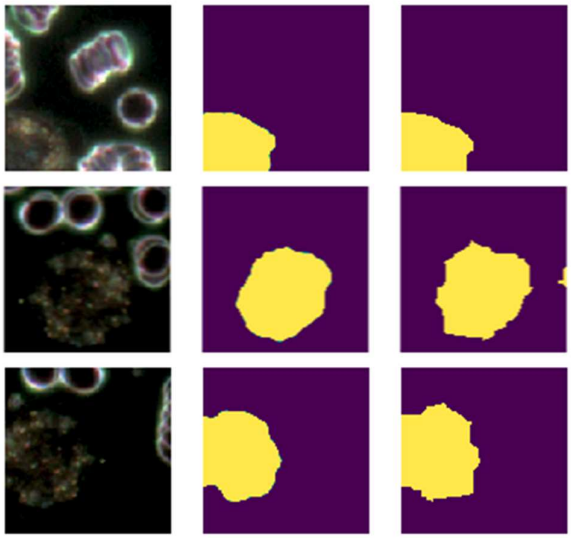

Fig. 6. Segmentation results: left column shows the inputs, the center column shows ground truth images, and the images on the right column represent the prediction (network output).

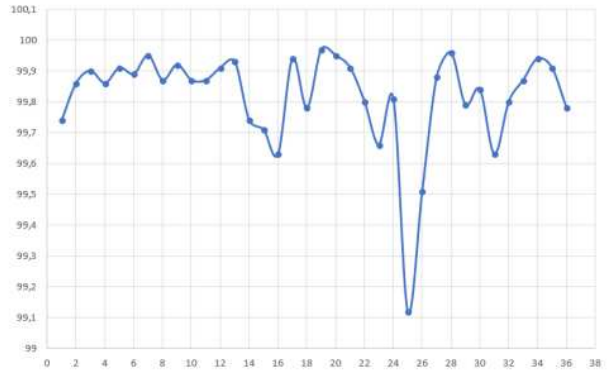

Fig. 7. Accuracy of the segmentation, for each image in the test set.

Using our test data set we were able to obtain an average accuracy of 99.81%.

Furthermore, the test set was evaluated using another metric: we compared the number of detected cells with the ground truth (actual) number of CTCs in the input image. From the medical point of view, this is the most important result, as it allows the estimation of the tumoral cell density in the blood. The result is presented in figure 9.

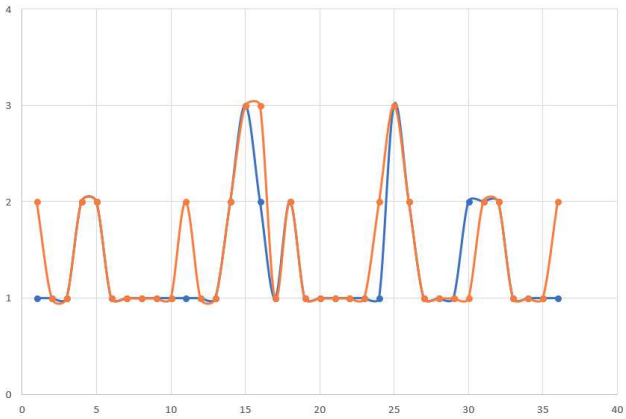

Fig. 8. Detected cells vs. actual number of cells in the image. The ground truth is illustrated with blue, and the prediction with orange.

We also computed the sensitivity (also known as true positive rate or recall), a measure of the proportion of true positives that are properly detected. The results are illustrated in figure 10. The specificity, or true negative rate, is the proportion of true negatives that are accurately detected. Figure 11 represents the specificity results. Specificity will quantify the avoidance of false positives, whereas sensitivity will quantify avoiding false negatives.

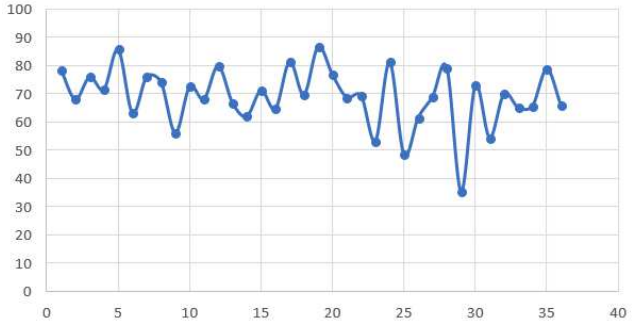

Fig. 9. Sensitivity computed on the test data set.

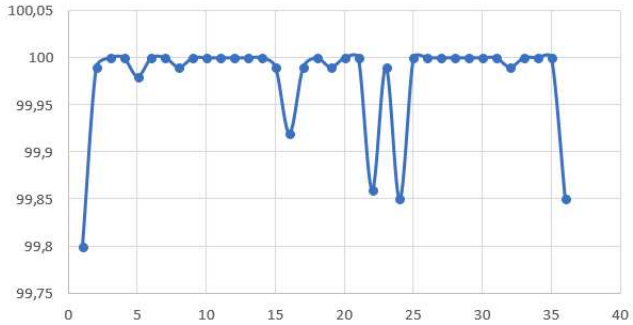

Fig. 10. Specificity computed on the test data set.

From figures 10 and 11 we can see that the system produces very few false positive CTC pixels, while detecting about 70% of the true positive pixels. However, even if some CTC pixels are lost, the cells are detected and counted accurately, as seen from figure 9, and therefore the system is able to accurately provide a statistic of CTC density in a blood sample image.

VI. Conclusion

In this paper we have presented a unique methodology to automatically detect and segment circulating tumor cells in Dark Field microscopic images of unstained blood. The segmentation is done using an encode-decoder type convolutional neural network that has the images acquired from the microscope as input, and provides the segmented result as output. The results are further improved using Hough circle detection that filters out false positives. The method proposes a completely automatic detection of CTCs that can improve the diagnosis and treatment of patients. From our tests, we were able to obtain over 99.8% pixel segmentation accuracy, and we were able to reliably count the CTC cells in the given samples.

References

[1] V Popovici, E Budinská, L Čápková, D Schwarz, L Dušek, J Feit, R Jaggi, "Joint analysis of histopathology image features and gene expression in breast cancer", *BMC bioinformatics*, Vol 17 (1), 2016.

[2] C Vicas, I Rusu, N Al Hajjar, M Lupşor-Platon, "Deep convolutional neural nets for objective steatosis detection from liver samples", *Intelligent Computer Communication and Processing (ICCP),* 2017, pp 385-390.

[3] G. Litjens, et al, "A Survey on Deep Learning in Medical Image Analysis", 2017, available at arXiv:1702.05747.

[4] Y. Xue, N. Ray, "Cell Detection in Microscopy Images with Deep Convolutional Neural Network and Compressed Sensing ", 2018, available at arXiv:1708.03307.

[5] Y. Xue, et al, "Cell Counting by Regression Using Convolutional Neural Network", Computer Vision – ECCV 2016 Workshops.

[6] C. Alix-Panabieres and K. Pantel, "Circulating tumor cells: liquid biopsy of cancer", *Clinical chemistry*, vol. 59, no. 1, pp. 110–118, 2013.

[7] A Ciurte, T Marita, R Buiga, "Circulating tumor cells classification and characterization in dark field microscopic images of unstained blood", *Intelligent Computer Communication and Processing (ICCP)*, 2015, pp. 367-374.

[8] Y. Mao, Z. Yin, J. M. Schober, "Iteratively Training Classifiers for Circulating Tumor Cell Detection", *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, 2015, pp. 190–194.

[9] O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", *Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer, LNCS*, Vol. 9351: 234--241, 2015, available at arXiv:1505.04597.

[10] T. Sorensen, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons", Kongelige Danske Videnskabernes Selskab, Vol 5 (4), pp. 1–34, 1948.