Embryo cell detection using regions with convolutional neural networks

Arnoldas Matusevičius[1], Darius Dirvanauskas[1], Rytis Maskeliūnas[1], Vidas Raudonis[2]

[1]Multimedia Engineering Department, Kaunas University of Technology, Kaunas, Lithuania
email: arnoldas.matusevicius@ktu.edu, darius.dirvanauskas@ktu.edu

[2]Department of Automation, Kaunas University of Technology, Kaunas, Lithuania
email: arnoldas.matusevicius@ktu.edu, darius.dirvanauskas@ktu.edu

***Abstract*—this research provide approach for embryo cell detection from images based on convolutional neural networks. Deep neural network used for experiment consist of 15 layers and is trained using GPU for calculations. In research training data set size impact to model training duration is identified. R-CNN model embryo cell detection results are compared to human expert labeling data to evaluate its precision.**

Keywords—Machine vision; Object recognition; Supervised learning

I. INTRODUCTION

Identify and count objects in an image or sequence of images is challenging computer vision problem, which can be found in many applications and systems, ranging from traffic monitoring to biological research. This paper is focused on biological research embryo cells detection. However, developed methodology can be used in numerous medicine procedures that requires counting and detection, such as red or white blood cells count for patient's health, clinical pathology or cell concentration investigation.

Manual embryo cell detection is very monotonous and time-consuming work that is prone to errors. According to this automating, the detection process has many benefits, such as reducing time consumption, minimizing errors possibility and cost. In addition, it is improving consistency of results between individuals and clinics. Our goal is to simplify the task and improve its robustness.

One of the difficulties is to count non-stained cells in dark images, because of constraints, such as the light intensity, transparency or exposure time. All these factors cause image quality and result in faint cell boundaries. One more challenge is that embryos cells has wide variability in appearance and shape. Furthermore, every embryo grows in different individual manner, there cells overlap each over. Also between cells could be found extracellular material hand crafted algorithms.

In this paper, we develop a Convolutional Regression Networks (CNN) approach for regression of density map. Our main goal is to automatically detect and count the number of human cells in developing embryos. In addition, experimental results shows that CNN can be used to provide state-of-the-art cell counting, also detect overlapping cells.

II. GENERAL DESCRIPTION OF R-CNN METHOD

Object detection system R-CNN can be divided into three main modules. The first module is responsible for generation category-independent region proposals [1]. Proposals are used to define the set of possible candidates available to the detector. The second one - convolutional network is used to extract a fixed-length feature vector from each region. The last one module is a set of class-specific linear SVMs.

According to the Pan and Yang taxonomy [2] Regional – Convolutional Neural Networks training is substantiated on learning inductive transfer. For correct R-CNN train, first step is to classify ImageNet as dataset and source task. Second step is network training using supervision, after that network is transferred to the target task and dataset using supervised fine-tuning. At first look, this methodology is related to traditional multi-task learning [3], [4]. However, this training is except for the task sequentially and furthermore, are only based on performing well on the specific target task.

Donahue et al. [5] also mentioned CNNs learning using supervised transfer in work. They state that once trained on ImageNet, further it can be treated as a black box feature extractor. This method is suitable for recognition with scene classification and domain adaptation. One more author Hoffman et al. [6] states transfer learning for R-CNN training is right choice and can be used for image – level label classes, but not for bounding – box training data.

Regions with Convolutional Neural Networks consists of two sibling output layers. The first one is used for discrete probability distribution, $p = (p_0,...,p_K)$, over $K + 1$ categories. Always parameter p is computed by a softmax over outputs of layer. The second one layer outputs bounding – box regression offsets, $t^k = (t^k_x, t^k_y, t^k_w, t^k_h)$ given in [7], t^k is used to specify a scale – invariant translation and log – space height and width shift relative to an object proposal [8].

R-CNN regions of interest training is labeled with u (ground – truth class) and v (ground – truth bounding – box regression

target). Then multitask loss L for each RoI classification (1).

$$L(p,u,t^u,v) = L_{cls}(p,u) + \lambda[u \geq 1]L_{loc}(t^u,v) \quad (1)$$

Here L_{cls} is equal to log loss for true class $L_{cls}(p,u) = -\log p_u$ the hyper – parameter is used to control the balance between the two task losses. For normalization the ground – truth regression targets v_i is equal to zero mean and unit variance.

In the second task is defined loss over a tuple of true bounding – box regression targets L_{loc} for class u and v and is equal *to* $v = (v_x,v_y,v_w,v_h)$, predicted tuple $t^u=(t^u_x,t^u_v,t^u_w,t^u_h)$ for class *u*. For background regions of interest there is not used notion of ground – truth bounding box also L_{loc} is not involved. In this case for bounding – box regression is used (2) expression.

$$L_{loc}(t^u,v) = \sum_{i\in\{x,y,w,h\}} smooth_L\,(t^u_i - v_i) \quad (2)$$

There $smooth_L$ equal (3).

$$smooth_L\,(x) = \begin{cases} 0{,}5x^2 & if\ |x| < 1 \\ |x| - 0{,}5 & otherwise \end{cases} \quad (3)$$

If the regression targets are unbounded, there is probability, that training with L_2 loss can require tuning of learning rates to prevent gradients explosion.

III. The Algorithm Used in the Study

Deep learning is package of different methods used in machine learning which attempts to present detail features in multiple-layer structure data. R-CNN is one of the most effective learning techniques and is able to minimize learnable parameters significantly by using the same basis function across different image locations.

In this research, we suggest an automatic learning based cell detection framework, which is suitable for 3D and 2D microscopy images. This framework can be used, for the efficiency and accuracy improvement of training a CNN from larger size images, an SVM classifier is applied to detect cell regions for collecting the CNN training set [9].

The exposure time range is dynamical and may not be equal for each session of recording through the light microscope, according to this the color of each stack may be different. Also in this research we apply Image Intensity Standardization (IIS), which was considered in [10] the main advantage is intensity normalization of 2D grayscale images. According to Bogunovic [11] after some modifications IIS algorithm is suitable for normalizing the intensity of the three-dimensional grey scale Rotational Angiography. Furthermore, we use the original Intensity Standardization as a color normalization method for 3D microscopy images. After that, calculation is performed of three histograms of the three channels of the whole RGB stack first. Also, the stack histogram of every used channel is aligned to the corresponding reference based on the non-linear registration method described in [10]. Algorithm of this operation is shown in Fig. 1.

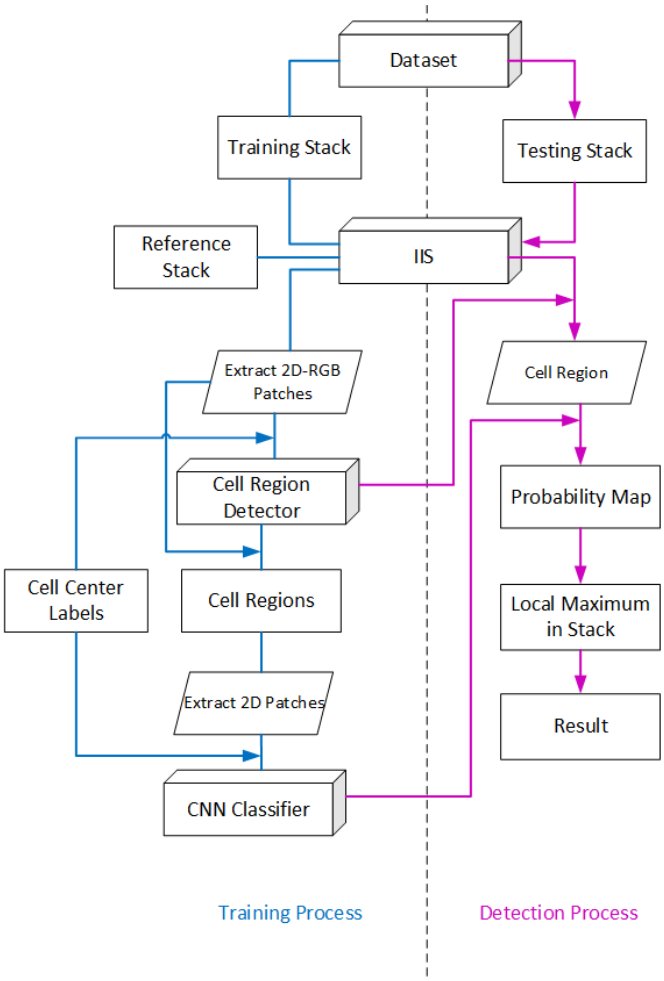

Fig. 1 The workflow of the embryo cell detection framework.

As well, there are more algorithms used for embryo cell detection in machine vision and medical image analysis areas [12, 13]. Nonetheless, most of these automated three-dimensional cell detection methods are not a suitable for manual cell detection [13]. There are two main types of cell detection algorithms. The first one is based on segmentation or thresholding [14] and different software implementations appeared including various plugins as "ImageJ" [12] and the "FARSIGHT" toolkit [15]. The second type is feature or modeling based methods [16, 17]. Due to machine learning techniques development, capabilities of cell detection based on learning are increased. Also, for two-dimensional immunohistochemistry images there are learning based on cell detection methods [18, 19]. However, there is not universal automatic cell detection method for microscopy images.

In this research, cell regions R are determined to discard the irrelevant background regions. Selecting background patches is important for training a CNN. Wherefore, cell regions detection is more efficient and rough using an SVM classifier, after that cell and background training patches are gathered from R instead of the whole stack.

The Support Vector Machine (SVM) detector is used for cell region detection and for collecting CNN training patches,

which are used to remove large part of background pixels. This part of process is like feature selection pre-process. In our case, accuracy of CNN could be increased using cell detection samples in the cell region. Similarly, in the test case, to increase accuracy, in first step we apply the SVM detector to identify those regions. In the training mode of conventional CNN, the cell samples are the same, however the non-cell samples are different.

Then the cell region R is detected using SVM-RGB Histogram detector, second step is to extract cell and patches in region R from all test stacks for training CNN which is also the same size of patches and neighborhood. Pixels in the cell region R have almost same colors. According to this color feature in the cell region is not reliable for distinguishing cell and background patches. On purpose to decrease time range for training all RGB patches are transferred into the YUV color space and only the Y channel patches are needed. Every Y-channel cell patch, is rotated 0, 90, 180, 270 degrees to ensure the detector rotation invariant and increase the amount of cell samples. Also there is probability that cell and background patches can have overlapping pixels. This is useful for increasing the probability of correct cell detection. Approximately half million cell patches are extracted from all training stacks, and the same amount of background patches from the cell region R.

After the last step, max-pooling CNN is ready for testing on the test stacks. The cell region is detected by the SVM RGB Histogram detector for each frame of every stack in the dataset used for testing. After that, the pre-trained CNN is used for identifying embryo cells by scanning each pixel in region and every pixel is given a probability value P.

IV. R-CNN TRAINING

Experiment was done using MATLAB 2016b software in a personal computer with i5-4570 CPU clocked at 3.2 GHz, 8 GB memory 64-bit operating system and video card GeForce GTX 650 Ti. Training process was done with GPU processor instead of CPU to accelerate training procedure.

We train the R-CNN network demonstrated in Fig. 2. It consist of 1 input layer, 13 hidden layers (*convolutional, Relu, Max Pooling, Fully Connected, Softmax*) and classification output layer. Training run for 100 epoch, with base learning rate of 0.001 and Stochastic Gradient Descent training method.

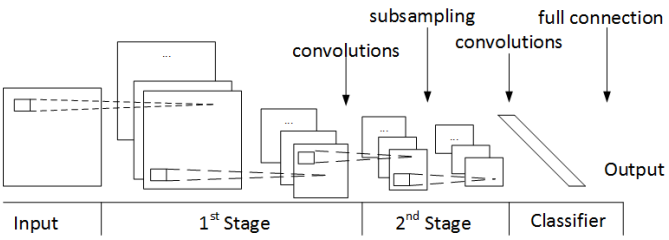

Fig. 2 The outline of the convolutional neural network architecture.

For experiment there was randomly selected thousand embryo photos Fig. 3 which was labeled by human expert. To evaluate training data set size impact to detection precision there was trained 14 R-CNN networks with different size training data set. Data set size for training was increased from 5% to 70% with 5% steps. 30% of data set was used to evaluate network cell detection precision. To decrease training time there was used pre trained CIFAR-10 network. Pre trained network biases and neuron weights there adjust to detect embryos cells in photos.

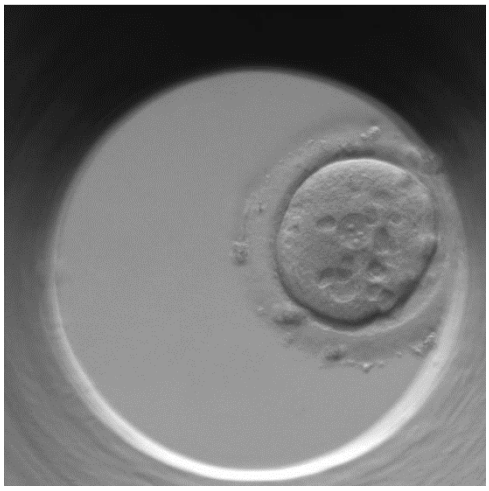

Fig. 3 Embryo images

Training data set size impact to training time is linear and it can be seen in Fig. 4. Training duration using biggest training dataset with 700 images was 38 minutes 40 seconds.

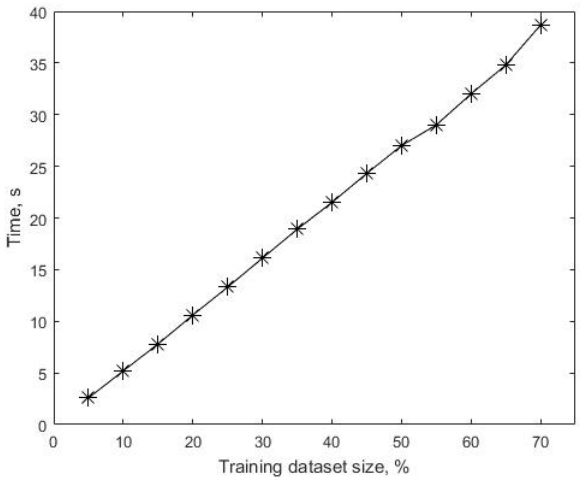

Fig. 4 Neural network training time

V. SIMULATION RESULTS

Trained R-CNN network was tested with new, do not used at training process, 300 embryo images Fig. 5. Predicted embryo position and size was compared with human expert labeled embryo size and position results.

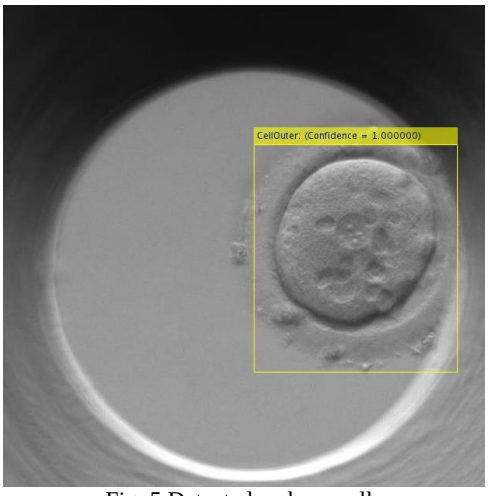

Fig. 5 Detected embryo cells

After comparing specialist data labeling results with deep neural network result, received size mean squared error and standard deviation presented at Table. 1. Some trained neural networks do not detected one or two embryos cell at images. Models with 30% or bigger size training dataset detected all embryos in images.

TABLE I. PREDICTED SIZE RESULTS

Training data set size	Mean square error, %	Standard deviation, %	Undetected embryos
5%	20,73	13,86	1
10%	17,32	11,08	2
15%	13,82	8,74	0
20%	11,55	7,54	1
25%	15,79	10,83	1
30%	13,48	8,53	0
35%	13,43	8,87	0
40%	14,67	8,41	0
45%	12,40	7,88	0
50%	12,57	7,77	0
55%	17,60	9,31	0
60%	18,32	8,99	0
65%	12,15	7,65	0
70%	11,92	7,18	0

From Fig. 6 it is seen that best results got with 20% and 70% size of training data set where models results compared with human expert gives 11.92% mean square error. It shows that not only training data set size impacts model accuracy, but images distribution in training data set influence model prediction accuracy.

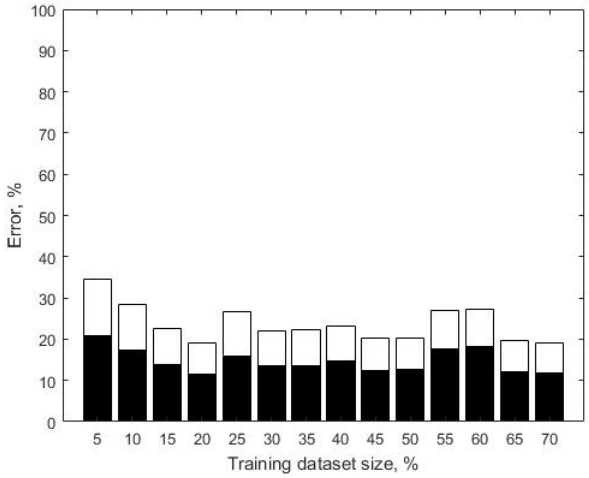

Fig. 6 Detected cell size error

Comparing model position predicting results with human expertise prediction from Table 2 it seen that error rate is smaller than size error rate. Smallest mean square error rate got using model trained with 30%, 40% and 65% training data set size. Close error rate got using 25% training data set, but this model do not detect one embryo cell.

TABLE II. PREDICTED POSITION RESULTS

Training data set size	Mean square error, %	Standard deviation, %	Undetected embryos
5%	6,05	2,55	1
10%	5,59	2,42	2
15%	5,45	2,76	0
20%	5,29	2,58	1
25%	4,64	2,07	1
30%	4,64	2,17	0
35%	6,82	2,92	0
40%	4,63	2,28	0
45%	5,29	2,5	0
50%	5,06	2,25	0
55%	5,42	2,31	0

60%	5,35	2,35	0
65%	4,62	2,18	0
70%	5,68	2,49	0

At Fig. 7 it seen whole error distribution. Inaccuracies appears using model with 35% training data set. That means few images could distort model parameters and decrease its accuracy.

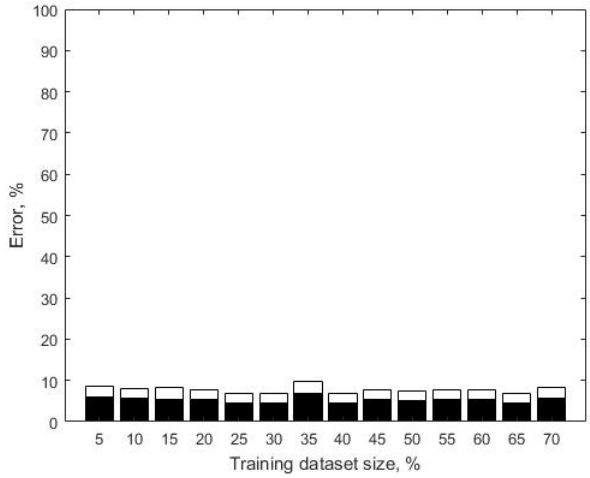

Fig. 7 Detected cell position error

VI. CONCLUSIONS

From experiment results it is possible to confirm that deep neural network training time is linearly dependent to training data set size. After detected region size comparison with human expertise prediction best result with mean square error rate 11.92% without any undetected embryos cell got using biggest 70% training data set size. More precise result got comparing embryos cell position. Smallest error 4.62% got using 65% training data set size. This shows that offered model better works for position prediction.

REFERENCES

[1] Region-based Convolutional Networks for Accurate Object Detection and Segmentation.

[2] S. J. Pan, Q. Yang, "A survey on transfer learning," TPAMI, 2010.

[3] R. Caruana, "Multitask learning: A knowledge-based source of inductive bias," in ICML, 1993.

[4] S. Thrun, "Is learning the n-th thing any easier than learning the first?" NIPS, 1996.

[5] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, T. Darrell, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition," in ICML, 2014.

[6] J. Hoffman, S. Guadarrama, E. Tzeng, J. Donahue, R. Girshick, T. Darrell, K. Saenko, "From large-scale object classifiers to large-scale object detectors: An adaptation approach," in NIPS, 2014.

[7] R. Girshick, J. Donahue, T. Darrell, J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," In CVPR, 2014. 1

[8] Ross Girshick Microsoft Research, "Fast R-CNN" 27 Sep. 2015

[9] B. Dong, L. Shao, M. Da Costa, O. Bandmann, A. F. Frangi, "Deep Learning for Automatic Cell Detection in Wide-Field Microscopy Zebrafish Images," IEEE, pp. 772-776, 2015.

[10] L. G. Nyu, J. K. Udupa, "On standardizing the MR image intensity scale," Magnetic Resonance in Medicine, vol. 42, pp. 1072–1081, 1999.

[11] H. Bogunovic, J. M. Pozo, M. C. Villa-Uriol, C. B. Majoie, R. van der Berg, H. A. Gramata van Andel, J. M. Macho, J. Blasco, L. S. Roman, A. F. Frangi, "Automated segmentation of cerebral vasculature with aneurysms in 3DRA and TOFMRA using geodesic active regions: An evaluation study," Medical Physics, vol. 38(1), pp. 210-222, 2011.

[12] M. D. Abramoff, P. J. Magalhaes, S. J. Ram, "Image processing with ImageJ," Biophotonics International, vol. 11(7), pp. 36-43, 2004.

[13] C. Schmitz, B. S. Eastwood, S. J. Tappan, J. R. Glaser, D. A. Peterson, P. R. Hof, "Current automated 3D cell detection methods are not a suitable replacement for manual stereologic cell counting," Frontiers in Neuroanatomy, vol. 8, 2014.

[14] Y. Al-Kofahi, W. Lassoued, W. Lee, B. Roysam, "Improved automatic detection and segmentation of cell nuclei in histopathology images," Biomedical Engineering, IEEE Transactions on, vol. 57(4), pp. 841-852, 2010.

[15] G. Lin, M. K. Chawla, "A multi-model approach to simultaneous segmentation and classification of heterogeneous populations of cell nuclei in 3D confocal microscope images," Cytometry Part A, vol. 71(9), pp. 724-736, 2007.

[16] M. K. K. Niazi, A. A. Satoskar, M. N. Gurcan, "An automated method for counting cytotoxic T-cells from CD8 stained images of renal biopsies," in SPIE Medical Imaging, vol. 8676, 2013.

[17] S. Wienert, D. Heim, K. Saeger, "Detection and segmentation of cell nuclei in virtual microscopy images: a minimum-model approach," Scientific reports, vol. 2, 2012.

[18] T. Chen, C. Chefdhotel, "Deep learning based automatic immune cell detection for immunohistochemistry images," in Machine Learning in Medical Imaging, pp. 17-24, 2014.