Deep Learning Identifies Cardiomyocyte Nuclei in Murine Tissue with High Precision

Shah R. Ali[1]*, Dan Nguyen[2]*, Brandon Wang[2]*, Steven Jiang[2], Hesham A. Sadek[1,3]

Department of Internal Medicine, Division of Cardiology[1], Department of Radiation Oncology, Medical Artificial Intelligence and Automation Laboratory[2], and Center for Regenerative Science and Medicine[3], The University of Texas Southwestern Medical Center, Dallas, Texas 75390, USA.

*These authors contributed equally to the manuscript

Correspondence to:

Hesham A. Sadek, MD, PhD
Department of Internal Medicine
Division of Cardiology
University of Texas Southwestern Medical Center
6000 Harry Hines Boulevard
Dallas, Texas, 75390, USA
Fax number: 214-648-1450
Telephone number: 214-648-1400
Email: hesham.sadek@utsouthwestern.edu

Steve Jiang, PhD
Medical Artificial Intelligence and Automation Laboratory
Department of Radiation Oncology
University of Texas Southwestern Medical Center
2280 Inwood Road
Dallas, TX 75390, USA
Fax number: 214-645-7622
Telephone number: 214-645-8510
Email: Steve.Jiang@UTSouthwestern.edu

ABSTRACT

Proper identification and annotation of cells in mammalian tissues is of paramount importance to biological research. Various approaches are currently used to identify and label cell types of interest in complex tissues. In this report, we generated an artificial intelligence (AI) deep learning model that uses image segmentation to predict cardiomyocyte nuclei in mouse heart sections without a specific cardiomyocyte nuclear label. This tool can annotate cardiomyocytes highly sensitively and specifically (AUC 0.94) using only cardiomyocyte structural protein immunostaining and a global nuclear stain. We speculate that our method is generalizable to other tissues to annotate specific cell types and organelles in a label-free way.

MAIN

Proper identification and annotation of cells in mammalian tissues is of paramount importance to biological research. Various approaches are currently used to identify and label cell types of interest in complex tissues. In the heart, for example, immunostaining using cell-specific antibodies can be used to identify cardiomyocytes amidst vascular cells, immune cells, and fibroblasts. However, immunostaining can have limitations due to the high background auto-fluorescence of certain tissues, nonspecific antibody binding, and/or epitope non-specificity.[1-4] Another technical issue pertaining specifically to cardiomyocytes is their large size, complex morphology, and multinucleation, which makes identification of their nuclei a complex process that requires training and the application of rigorous techniques. Genetic fluorescent reporter models are ideal, but they require specialized transgenic mice.[5] In the field of mammalian cardiac regeneration, lack of precise cardiomyocyte nuclear markers is likely one source of controversy over the wide range of reported cardiomyocyte renewal rates: misidentification of a dividing nonmyocyte nucleus can lead to calculation of a falsely high myocyte mitotic index.[4,6,7]

Label-free techniques have been increasingly utilized in a wide array of microscopy and biological applications.[8-10] Therefore, we sought to develop an artificial intelligence (AI) tool that can identify cardiomyocyte nuclei in mouse heart tissues. To train an AI to discern cardiomyocyte nuclei from nonmyocyte nuclei using a global nuclear stain (DAPI) and cardiomyocyte structural protein immunostaining (troponin T), those two images were provided from a tissue section – along with a “ground truth” (GT) cardiomyocyte nucleus label by which the AI could iteratively learn (Fig. 1A). GT cardiomyocyte nuclei were identified using Cre staining of Myh6-Cre transgenic mice. Therefore, heart sections from adult Myh6-Cre mice were concomitantly immunostained for troponin T and Cre (Fig. 1A-E). The cardiomyocyte nuclei (red) were easily distinguished from nonmyocyte nuclei. However, we also noted that the Cre

staining has some extranuclear tissue signal, so we subtracted the tissue image (Troponin stain) from the Cre-stained image (Fig. 1F,G): the resulting image was treated as the GT label for training the neural network to learn cardiomyocyte nuclei identification.

Since all of the data inputs and labels are on the pixel level, we used an image-to-image deep learning model, a U-net style architecture[11], which is one of the most widely used styles of architectures for imaging tasks that can calculate and maintain both global and local information throughout the network (Fig. 1H).

To evaluate the model's stability and robustness we trained 5 different models, but withheld a portion of the training data to use it as validation data. We divided each complete heart section into 256 x 256 patches. For the 25 tissue sections used as training and validation data (n=3 mice), this resulted in 33,769 total image patches for model development. For each model, 27,016 image patches were used for training and 6,753 were left out and used for validation.

Each instance of the model was trained for a total of 200,000 iterations with a batch size of 4, which took, on average, 20 hours 36 minutes, with a standard deviation of 47 minutes, on an NVIDIA 1080 Ti GPU. For each training instance, the best performing weights with the lowest validation score was used. In total, all 5 model instances took 4.29 days to train for 200,000 iterations each. All 5 models were evaluated on the testing set, to obtain the model's overall performance and variation.

All models performed well: the models' prediction closely matched the ground truth label and had a bright signal where they matched (Fig. 1J-L). The average area under the curve (AUC) across all 5 models was 0.940 ± 0.004 (Fig. 1L), which demonstrates that the models have a high probability of accurately distinguishing a nonmyocyte from a myocyte nucleus (an AUC of 1 would indicate a perfect model). When the model predicted some signal that was

outside of the GT label, the associated signal was weaker and dimmer, showing that the model is confident in its labeling of the cardiomyocyte. For example, a mislabeled nonmyocyte nucleus had a weak red label (arrowhead, Fig. 1K).

A fundamental concern in the study of animal models is the proper identification and annotation of cell types in a given tissue. False attribution can lead to misinterpretation of underlying physiological and pathological processes. Here, we demonstrate that machine learning-based methods may obviate these problems. In the current report, we generated an AI model that uses image segmentation to predict cardiomyocyte nuclei in mouse heart sections without a specific cardiomyocyte nuclear label. Indeed, this tool can annotate cardiomyocytes highly sensitively and specifically (AUC 0.94) using only cardiomyocyte structural protein immunostaining and a global nuclear stain. This method enables calculations that were previously inaccessible to a manual approach, e.g., counting all the cardiomyocyte nuclei in a tissue section. We speculate that our method is generalizable to other tissues to annotate specific cell types and organelles in a label-free way.

APPENDIX

Training Data Preparation

We used the tissue and nuclei stain as the input to teach the deep learning model to predict the cardiomyocyte nuclei. To achieve this, it is required to convert the cardiomyocyte nuclei stain into labels to teach the model. However, since the cardiomyocyte stain still contains a lot of tissue information, we executed multiple steps in order to create the proper image of cardiomyocyte labels, which is outlined below.

The first step is to remove as much of the tissue information as possible, to achieve this we subtracted the tissue image from the cardiomyocyte image,

$$I_D = (I_{MC} - kI_T)_+ \tag{1}$$

where I_{MC} is the cardiomyocyte image, I_T is the tissue image, I_D is the resulting difference image, and k is a scalar constant. The operator, $(\cdot)_+$, projects its argument onto the non-negative orthant. With the goal of removing as much tissue information, we optimized k for each slide image, based on the minimization problem:

$$\underset{k}{minimize}\ \|I_{MC} - kI_T\|_1 \tag{2}$$

where $\|\cdot\|_1$ is the l_1-norm. In mathematics, the l_p-norm of an argument, $x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$, is defined as $\|x\|_p = \sqrt[p]{\sum_{i=1}^{n} |x|^p}$. We chose the l_1-norm as our fidelity term, since the l_1-norm promotes its argument to be sparse in a minimization problem. Since the l_1-norm is non-differentiable at 0, we utilized a first-order primal-dual proximal-class algorithm known as Chambolle-Pock[12,13],

which can handle non-differentiable functions through the use of the prox operator, which is defined, for some function, $h(\cdot)$, and a step size, t, as $prox_{th}(x) = \underset{u}{argmin}\left(f(u) + \frac{1}{2t}\|u - x\|_2^2\right)$. After solving Equation 2 to find k, we can apply Equation 1 to obtain the difference image, I_D. The reason why we project the difference onto the non-negative orthant is due to the fact that the cardiomyocytes nuclei have higher fluorescence intensity than the surrounding tissue in the Cre immunostain. Therefore, we expect that all cardiomyocyte nuclei will be positive after the operation $I_{MC} - kI_T$. Finally, to create the labels, we must first separate the cardiomyocytes from any residual background in I_D. To do this, we used a method known as Otsu thresholding[14], which is an advanced thresholding technique to separate an image into a foreground and background. It automatically selects the threshold based off an intensity histogram, minimizing the variance within a class and maximizing it between the two classes. We applied this method onto I_D to arrive at our final cardiomyocyte label image, I_{MCL}, shown in Equation 3:

$$I_{MCL} = Threshold_{Otsu}(I_D) \tag{3}$$

This I_{MCL} is then treated as the ground truth label for training the neural network to learn cardiomyocyte nuclei identification.

Deep learning architecture

Since all of the data inputs and labels are on the on the pixel-level, using an image-to-image deep learning model is the natural choice for this study. We utilized a U-net style architecture[11]. The U-net architecture was proposed in 2015 and was originally applied for biomedical image segmentation tasks. The strength of the U-net comes from its capability to calculate and

maintain both global and local information throughout the network, and has become one of the most widely used styles of architectures for imaging tasks. Our exact variant of the U-net is shown in Figure 1H. The basic block of operations used in this study involves a convolutional layer, a rectified linear unit (ReLU) as the non-linear activation, group normalization[15] (GN), and DropBlock[16] (DB). Group normalization was developed in 2018 to overcome the limitations of batch normalization[17], especially with small batch sizes. DropBlock is a special variant of dropout[18], and was shown to work better with convolutional neural networks than the standard dropout. We will refer to this block of operations as Conv-ReLU-GN-DB.

To be specific, our network was developed to take in 2 channels of 256 x 256 image patches and output 1 channel of the same resolution. A total of 5 resolution levels were designed, with downsampling and upsampling operations in between to translate the data between different resolutions. The downsampling consists of 2 parallel operations, whose outputs are then concatenated together. The first operation is a 2 x 2 max pooling[19,20], and second set of operations is a 2 x 2 strided Conv-ReLU-GN-DB with a 2 x 2 kernel. For the upsampling, we performed a nearest neighbor 2 x 2 upsample, followed by a Conv-ReLU-GN-DB operation with a 3 x 3 kernel. Between these downsampling and upsampling operations, 2 sets of Conv-ReLU-GN-DB with a 3 x 3 kernel are performed. Regarding the number of feature maps calculated at each convolution, the first level starts at calculating 32 feature maps at each layer, the lower levels of decreasing resolution double the number of feature maps to 64, 128, and 256, respectively. The model development and training framework was developed using TensorFlow[21]. The Adam optimizer[22] was used during training, with a learning rate of 1×10^{-3}.

Training and Evaluation

To evaluate the model's stability and robustness, we trained 5 different models, omitting a different portion of the training data from training, which was then used as validation data. We

divided each slide image into 256 x 256 patches. For the 25 slides used as training and validation data, this resulted in 33,769 total image patches for model development. For each model, 27016 image patches were used for training and 6753 were left out and used for validation. For each training instance, the weights where the model had performed best on the validation set was then used to evaluate on the testing data, in order to assess the average prediction performance and variation.

In total, on the 2 slides used as hold-out testing data, there are 81,330 nuclei in the testing data that was evaluated. To determine whether the model's prediction should be labeled as cardiomyocyte or not, we define the fraction of voxels labeled as positive in the ith nuclei, f_i, as

$$f_i = \frac{\sum_{p \in n_i} y_{pred,p}}{\sum_{p \in n_i} I_{NL,p}} \tag{4}$$

where p is the pixel index, n_i is the ith nuclei, y_{pred} is the model's raw prediction of cardiomyocytes, and I_{NL} is the image of nuclei labels, obtained by performing Otsu thresholding on the nuclei stain image. The summation over $p \in n_i$ means that we are only summing over the voxels defined in the ith nuclei. We can then define the nuclei as cardiomyocyte if v_{f_i} exceeds a defined threshold value

$$n_i \; is \begin{cases} myocardiocyte & if \; f_i > t_{MC} \\ not \; myocardiocyte & else \end{cases} \tag{5}$$

where $t_{MC} \in [0,1]$ is a user defined threshold, and controls the tradeoff between the sensitivity and specificity. We calculated the true/false positives/negatives for each t_{MC} between 0 and 1, with a step size of 0.001. In total, we evaluated the prediction using 1000 t_{MC} values.

REFERENCES

1 Jackson, K. A., Snyder, D. S. & Goodell, M. A. Skeletal Muscle Fiber-Specific Green Autofluorescence: Potential for Stem Cell Engraftment Artifacts. *STEM CELLS* **22**, 180-187, doi:10.1634/stemcells.22-2-180 (2004).

2 del Rosal, B., Villa, I., Jaque, D. & Sanz-Rodríguez, F. In vivo autofluorescence in the biological windows: the role of pigmentation. *Journal of Biophotonics* **9**, 1059-1067, doi:10.1002/jbio.201500271 (2016).

3 Zhuo, L. *et al.* Live astrocytes visualized by green fluorescent protein in transgenic mice. *Dev Biol.* **187**, 36-42 (1997).

4 Ang, K.-L. *et al.* Limitations of conventional approaches to identify myocyte nuclei in histologic sections of the heart. *American Journal of Physiology-Cell Physiology* **298**, C1603-C1609, doi:10.1152/ajpcell.00435.2009 (2010).

5 Muzumdar, M. D., Tasic, B., Miyamichi, K., Li, L. & Luo, L. A global double-fluorescent Cre reporter mouse. *Genesis* **45**, 593-605 (2007).

6 Raulf, A. *et al.* Transgenic systems for unequivocal identification of cardiac myocyte nuclei and analysis of cardiomyocyte cell cycle status. *Basic Research in Cardiology* **110**, 33, doi:10.1007/s00395-015-0489-2 (2015).

7 Walsh, S., Pontén, A., Fleischmann, B. K. & Jovinge, S. Cardiomyocyte cell cycle control and growth estimation in vivo—an analysis based on cardiomyocyte nuclei. *Cardiovascular Research* **86**, 365-373, doi:10.1093/cvr/cvq005 (2010).

8 Moen, E. *et al.* Deep learning for cellular image analysis. *Nature Methods* **16**, 1233-1246, doi:10.1038/s41592-019-0403-1 (2019).

9 Belthangady, C. & Royer, L. A. Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. *Nature Methods* **16**, 1215-1225, doi:10.1038/s41592-019-0458-z (2019).

10 Buggenthin, F. *et al.* Prospective identification of hematopoietic lineage choice by deep learning. *Nature Methods* **14**, 403-406, doi:10.1038/nmeth.4182 (2017).

11 Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 234-241 (2015).

12 Chambolle, A. & Pock, T. A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. *Journal of Mathematical Imaging and Vision* **40**, 120-145, doi:10.1007/s10851-010-0251-1 (2011).

13 Pock, T., Cremers, D., Bischof, H. & Chambolle, A. in *Computer Vision, 2009 IEEE 12th International Conference on.* 1133-1140 (IEEE).

14 Otsu, N. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics* **9**, 62-66 (1979).

15 Wu, Y. & He, K. in *Proceedings of the European Conference on Computer Vision (ECCV).* 3-19.

16 Ghiasi, G., Lin, T.-Y. & Le, Q. V. in *Advances in Neural Information Processing Systems.* 10727-10737.

17 Ioffe, S. & Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*, 448-456 (2015).

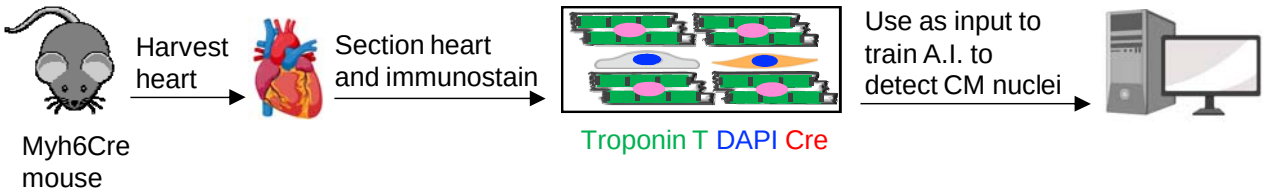
18 Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research* **15**, 1929-1958 (2014).

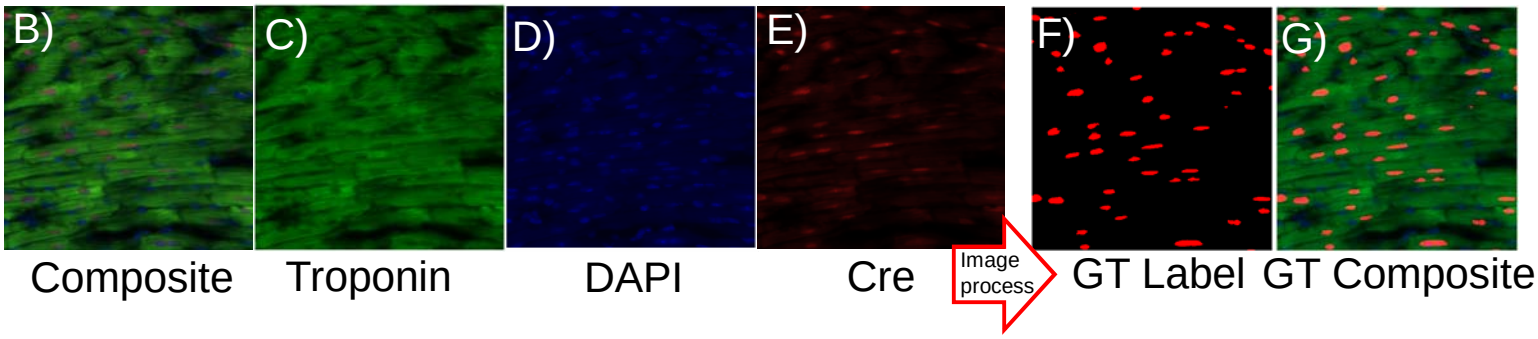
19 Ranzato, M. a., Huang, F. J., Boureau, Y.-L. & LeCun, Y. in *2007 IEEE conference on computer vision and pattern recognition.* 1-8 (IEEE).
20 Scherer, D., Müller, A. & Behnke, S. in *International conference on artificial neural networks.* 92-101 (Springer).
21 Abadi, M. *et al.* Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).
22 Kingma, D. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

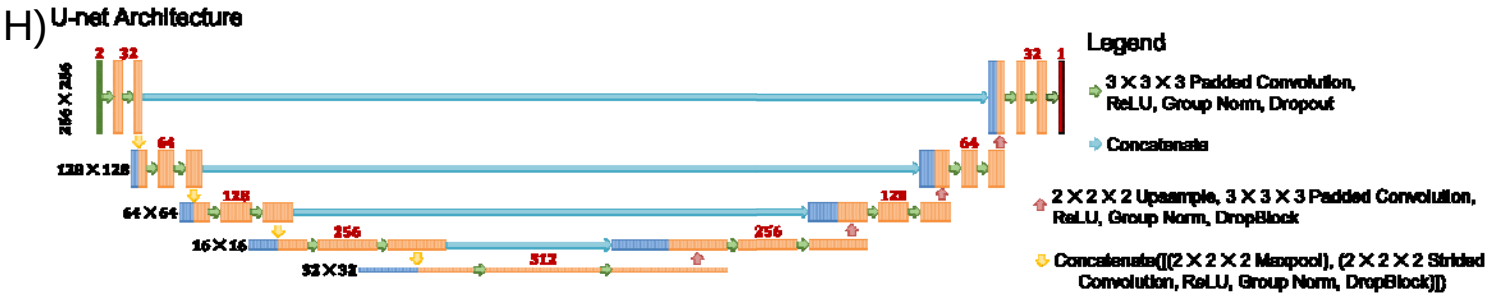
Figure Legend

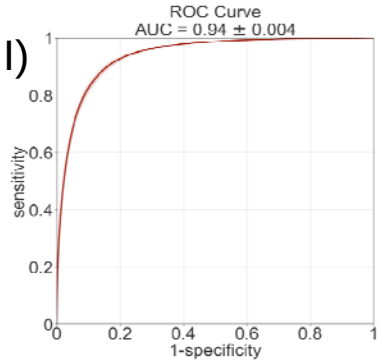
Fig. 1: A) Schematic of overall approach. B-E) Immunostaining of Myh6Cre heart sections (8 um) with Troponin (Millipore 06-570, 1:100) and Cre (Millipore 69050, 1:10000) antibodies. F-G) Image processing of Cre-immunostain image renders a Ground Truth (GT) label that is used to train the A.I. H) Schematic of the U-net architecture used in this study. The black numbers at the left represent the image resolution at each level of the U-net. The red numbers represent the number of feature maps that are calculated when the convolution is applied. I) Receiver-operating curve (ROC) of all 5 models. J-K) Image of AI-predicted cardiomyocyte (CM) nuclei and composite image; arrowhead in (K) points to nonmyocyte nucleus mislabelled as myocyte nucleus by the AI). L) Original composite image is provided to enable comparison to AI's predictions of CM labels.

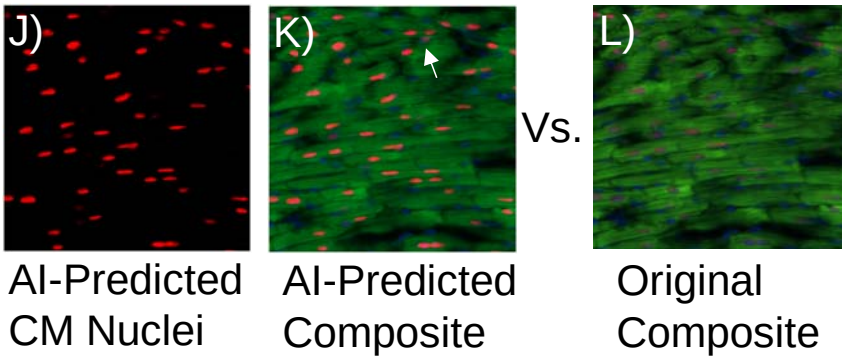
A) Schematic


Harvest heart
Section heart and immunostain
Use as input to train A.I. to detect CM nuclei
Myh6Cre mouse
Troponin T DAPI Cre


B)
C)
D)
E)
F)
G)
Composite
Troponin
DAPI
Cre
Image process
GT Label
GT Composite


H)
U-net Architecture
Legend
3 X 3 X 3 Padded Convolution, ReLU, Group Norm, Dropout
Concatenate
2 X 2 X 2 Upsample, 3 X 3 X 3 Padded Convolution, ReLU, Group Norm, DropBlock
Concatenate([(2 X 2 X 2 Maxpool), (2 X 2 X 2 Strided Convolution, ReLU, Group Norm, DropBlock)])
256 X 256
128 X 128
64 X 64
16 X 16
32 X 32


I)
ROC Curve
AUC = 0.94 ± 0.004
sensitivity
1-specificity


J)
K)
Vs.
L)
AI-Predicted CM Nuclei
AI-Predicted Composite
Original Composite