Learning to Segment Mouse Embryo Cells

Juan León, Alejandro Pardo, and Pablo Arbeláez

Universidad de los Andes, Bogota, Colombia

ABSTRACT

Recent advances in microscopy enable the capture of temporal sequences during cell development stages. However, the study of such sequences is a complex and time consuming task. In this paper we propose an automatic strategy to address the problem of semantic and instance segmentation of mouse embryos using NYU's Mouse Embryo Tracking Database. We obtain our instance proposals as refined predictions from the generalized Hough transform, using prior knowledge of the embryo's locations and their current cell stage. We use two main approaches to learn the priors: Hand-crafted features and learned features. Our strategy increases the baseline jaccard index from 0.69 to 0.71 using learned features.

Keywords: Cell Segmentation, Video Analysis, Convolutional Encoders

1. INTRODUCTION

Tissue development and function, along with some pathological processes such as tumorige nesis, are driven by complex developmental mechanisms in which cells undergo several processes, among them, cell division, migration, changes in morphology, and death; all of them are critical for tissue formation.[1,2] A better understanding of these mechanisms would require to quantitatively analyze individual cells during their development process, in their natural tissue environment. To examine cell behavior over time, researchers have devised strategies that enable the capture of temporal sequences during several cell developmental stages.[3] However, the analysis of such image sequences is complex and time consuming, as even simple measurements require manual, frame by frame inspection of long video sequences that contain a large amount of targets.[1] To tackle this problem, several strategies have been proposed in the literature[1–5] to segment, detect or track cells during the division process.

In this paper, we explore two fundamental problems for the analysis of video sequences from the developmental process of mouse embryos, namely semantic segmentation and instance detection. We approach these tasks by first establishing a baseline segmentation for the mouse embryos based on local features. Then we address the detection task by refining proposals from the generalized Hough transform, using prior knowledge of the embryos locations and their current cell stage.

The rest of this paper is organized as follows. In Section 2, we present a brief overview of current approaches in mouse embryo cell segmentation. Then we outline the selected representations and overall experimental setup in Section 3. Details and preliminary analysis of the experimental procedures are included in Section 4 along with the design details of the proposed convolutional encoders. Finally we present our conclusions and most relevant findings in Section 5.

2. RELATED WORK

Traditionally works on cell segmentation focus on the identification of nuclei locations or boundaries,[6–8] some recent works also address the task of cell segmentation including both the nuclei and the cytoplasm. Dow et. al.[7] initialize the segmentation process by first identifying, the cell nuclei, and then creating boundaries by region growth. A similar method is presented by Ortiz et. al.[9] where an initial seed inside the cell is used to approximate an snake over its boundary. The strategy proposed by Baggett et. al[3] approaches the boundary segmentation by estimating an optimal border whose local average intensity per unit length is greater than any other possible boundary. Approaching both the segmentation and tracking problems, Meseguer et. al.[10] use static cell models and a particle filter to locate and identify cells in the temporal sequence, finally Cicconet et. al[11] use an extension of the hough transform where the votes are calculated over weighted wavelet kernels.

Further author information: (Send correspondence to)
Juan Leon: E-mail: jc.leon@uniandes.edu.co

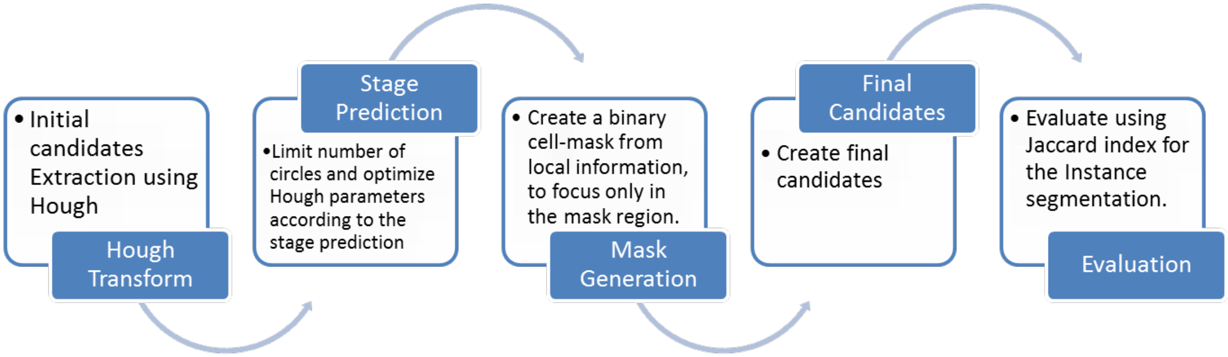
• Initial candidates Extraction using Hough
Hough Transform
Stage Prediction
•Limit number of circles and optimize Hough parameters according to the stage prediction
• Create a binary cell-mask from local information, to focus only in the mask region.
Mask Generation
Final Candidates
• Create final candidates
• Evaluate using Jaccard index for the Instance segmentation.
Evaluation

Figure 1. Our method refines the instance cell segmentation by first establishing a segmentation mask and then establishing the number of cells to be segmented. The final evaluation is performed using the jaccard index.

3. MATERIALS AND METHODS

Given the circular patterns of our detection targets, the generalized Hough Transform[12] is a natural baseline method for the semantic and instance segmentation tasks. However, this approach can be limited given the nature of the time lapse images, where cells overlap, divide, change sizes and position during the division process. To overcome these limitations, we propose to first use local features to approximate the location of the mouse embryo, thereby approaching the semantic segmentation task. Then we estimate the current number of cells and use this information, along with the location prior, to improve the performance of the Hough Transform on the instance segmentation task.

Overall, our method is outlined in Figure 1. We approach embryo segmentation as a classification problem, we learn a mapping $f^s_\phi(L) = C$, $f\colon \mathbb{R}^n \to \mathbb{R}$ that maps a local descriptor L into a binary label C given the parameter set ϕ. Here, the label set C provides an initial classification for every pixel into the labels *cell* and *background*. For a single frame we will refer to the set of cell pixels as the *segmentation mask*. We explore two different strategies to approximate this mapping, their main difference being the feature extraction process. First, we explore traditional hand-crafted features using the STIP descriptor,[13] then we compare its results to an strategy in which local features are learned using a convolutional encoder.[14]

With the approximate location of the mouse embryo, we proceed to estimate the current cell stage as it provides information regarding the amount of cells on the image. Hence, we learn a second mapping $f^i_{\bar{\phi}}(G) = D$, where f^i approximates the image's cell stage given a global descriptor G. For the stage identification we also explore two strategies, again the fundamental difference is the nature of the extracted features. Our first approach uses a classic Bag of Visual Words approach, and the second uses global learned features from a convolutional encoder. As final step we use both the segmentation masks and the amount of cells as prior information for the Hough transform applied to the instance segmentation task.

Spatio-Temporal Interest Points (STIP) were first presented by Ivan Laptev[13] for the task of human action recognition. STIP is an extension of the Harris corner detector[15] for spatio-temporal data. Given an interest point, its descriptor is extracted from 3D normalized Gaussian derivatives as:

$$L_{x^m y^n t^k} = \sigma^{m+n} \tau^k (\partial_{x^m y^n t^k} g) * x. \tag{1}$$

Following this formulation, STIPs, at a given scale, can be characterized by the n-th order local jet:[16]

$$X = \{L_x, L_y, L_t, L_{x,x}, ...L_{ttt}, ..., .L_{tt...t}\}. \tag{2}$$

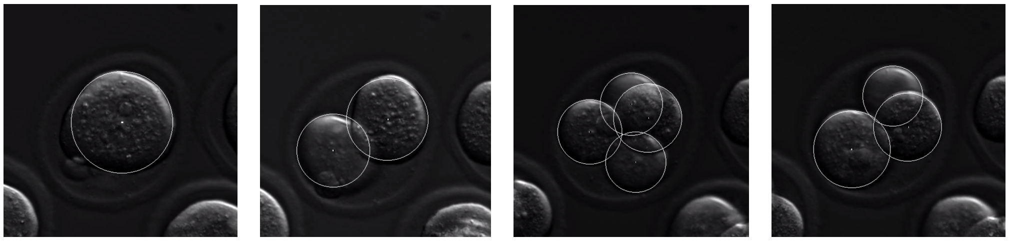

Figure 2. Sample images from the NYU's Mouse Embryo Tracking Databases, from left to right embryo cells in stages 1,2,3. The final image corresponds to the intermediate step between stages 2 and 3.

For the mouse embryo segmentation we use the N-th Order Local Jet descriptor, however, unlike Laptev's work,[13] where the STIP descriptor is used to calculate an inter-class a Mahalanobis distance, we use a Support Vector Machine (SVM) classifier to discriminate between a cell-pixel and a background-pixel.

Learned Convolutional Features Encoders are functions that allow the computation of a feature vector h from an arbitrary input vector x as: $h = f_\phi(x)$. An encoder can be coupled with another function g_ϕ, known as the decoder, whose task is to reconstruct x. Hence the auto-encoder goal is to minimize:

$$\sum_t L(x_t, g_\phi(f_\phi(x_t))) \tag{3}$$

Where L is called the loss function, and can be any suitable reconstruction error.[14] Still in a supervised learning scenario, h can be replaced with a label set Y, under this constraint the encoder will try to minimize L by adjusting its outputs towards the given label set.

In practice, convolutional auto-encoders are multi-layer non-linear convolutional encoders described by:[17]

$$X = \gamma_k(\sigma_k(W_k....\gamma_2(\sigma_2(W_2(\gamma_1(\sigma_1(W_1x)))). \tag{4}$$

Where the function composition $\gamma_k \circ \sigma_k \circ W_k$ is called a layer, x is the N-dimensional input vector, W_k are matrices corresponding to convolutional operations, σ_k are point-wise non-linear functions, and matrices γ_k perform a down-sampling operation also known as feature pooling.

As consequence, minimizing the loss function equals to optimizing over filters W_k, which means the auto-encoder approximates Y by learning a representation X from an optimal set of convolutional filters W_k. In practice, L can be minimized in an iterative process using stochastic gradient optimization and back-propagation.[18]

Bag of Visual Words In a Bag-of-Words representation, local interest patches are first identified on an image, either by dense sampling or using a generic interest point detector.[19] To efficiently handle these interest points, a local descriptor is calculated and then quantized according to a global dictionary in order to extract a key-point descriptor known as *visual word*. An image is then represented by a histogram of these visual words.[19] Classification can then be performed using the histogram as feature vector.

4. EXPERIMENTAL SETUP

For our experiments, we use the NYU Mouse Embryo Tracking Database,[11] which contains 101 hand labeled video sequences of 4 to 6 week old female mice embryos, recorded using an Eclipse TI inverted microscope for up to 120 hours, frames were captured every 420 seconds. For most embryos the division process leads up to the 8-cell stage. Labels are provided for embryos up to the 3rd cell stage in each frame. A sample from the database images and annotations is shown in figure 4.

4.1 Semantic Cell Segmentation using Hand-crafted Features

For the semantic cell segmentation task using hand-crafted features, f_s will be implemented with an SVM. To obtain the best possible mapping, we first have to explore the best parameters for the spatio-temporal descriptor, i.e. find the best combination of voxel size (including its time dimension), and the maximal order of the local derivative. However, changing the descriptor configuration will also change the nature of the SVM inputs, therefore the SVM space parameter must be explored jointly with the descriptor's parameter. This joint search space grows quickly and therefore must be limited. We limit the search space as shown in Table 1:

Parameter	**Explored Value Set**
Voxel x,y size	{5,10,15,20,25,30,35}
Voxel time dimension	{1, 2, 3, 5, 10, 15}
Max order local jet descriptor	{1,2,3,4}
SVM C parameter	{0.1, 1, 10, 100, 1000}
SVM Kernel function	{Lineal, Polynomial, RBF}

Table 1. Summary of the explored parameter set for the first and second step of the SVM training using STIP descriptors.

Even after restricting the search space there are 2520 combinations to explore. To further speed up the process, instead of training a single SVM with N training samples, we train an ensemble of M (N > M) SVMs with N/M training samples each; The final decision follows a voting scheme. This SVM ensemble greatly reduces the training time given the $\mathcal{O}(n^2)$ complexity of the hyper plane optimization algorithm. We also reduce the amount of videos used to explore these hyperparameters and randomly select 50 videos for this process, out of them 35 are randomly selected for the Training set, and the remaining 15 for validation.

Under these training scheme we choose M=10, and exploit its highly parallelizable nature to explore the complete set of parameter in about 24 hours using 40 CPUs. We empirically conclude the best set of parameters is: Voxel-Size:{x=10,y=10,z=5} , Max Order Derivative=4, and C=1000 using an RBF Kernel.

Using the best parameters, we explore four training schemes, first the SVM ensemble used for parameter exploration, second a single SVM trained using the complete set of features, third and fourth, are simple extensions of the first two strategies, where the voxel descriptor is enhanced with the normalized pixel coordinates of its centroid. Table 2 summarizes the results for the cell vs background classification task.

Hand Crafted Features Semantic Segmentation				
Classification Scheme	Avg. Precision	Avg. Recall	Avg. F-Measure	Train Time (Seconds)
STIP + Ensemble	0.79	0.79	0.78	**33.4**
STIP + Single SVM	0.81	0.81	0.81	9650
STIP + Ensemble + Pixel Coordinate	**0.94**	0.94	**0.94**	36.10
STIP + Single SVM + Pixel Coordinate	**0.94**	**0.95**	**0.94**	7013

Table 2. Average results obtained for the semantic segmentation task using an SVM classifier and the STIP representation. For this stage we also evaluate the effectives of including the position prior in the descriptor

As can be seen in Table 2, the average F-Measure significantly improves once the normalized pixel coordinates are included, hence, we will only use descriptors enhanced with the coordinate prior for the segmentation masks.

4.2 Semantic Cell Segmentation Using Learned Features

We explore two main strategies for this task, one that ignores the temporal information, which we call *2D Encoder*, and one that stacks video frames thereby exploring the optimal size for the temporal dimension, we call it the *3D Encoder*. Furthermore, we also explore the patch size that enables the best performance in the segmentation task. Again the parameter space is too large to be explored exhaustively , hence, we try to achieve a sub-optimal configuration by first exploring the optimal parameters of the 2D Encoder, and fix them during the exploration of the stack depth of the 3D Encoder. Finally, we also explore the effect of including normalized pixel coordinates given the large improvements found in the hand crafted feature strategy. This search space is summarized in table 3. We train both encoders using the TensorFlow framework[20] and Keras as frontend.[21]

We star the optimization process from a random weight initialization, and use the Adagrad[22] optimizer with an initial learning rate of 0.001 and a decay 0.0001. We train until convergence on the validation loss.

Network	Parameter	Explored Values
2D Encoder	Patch size	{9, 11, 13, 15, 17, 19}
3D Encoder	Time size	{2, 3, 5, 10, 15}

Table 3. Summary of the explored parameters for the convolutional encoder used in the semantic segmentation task

Table 4 summarizes the final results for the 2D and 3D encoders. Again there is a significant contribution on the average F-Measure when the normalized pixel coordinates are explicitly included. We obtain that patches of size 11 and 13 were the best for the instance segmentation task with an average F-Measure of 0.97.

For the 3D Encoder, we explore the size of the temporal stack for patches with size 11 and 13, as both configurations show the same performance without stacking frames. Table 5 shows that adding temporal information slightly improves the results for both patch sizes. A temporal stack of only two frames produces the best improvement, while larger stacks show smaller improvements. For the final results in the semantic segmentation task, we choose a patch size of 11x11 as its train time is shorter.

Finally, we use the encoder with candidates of patch size 11x11 and time size of 2 to predict the segmentation mask on the test set. Some qualitative results can be seen in Figure 3

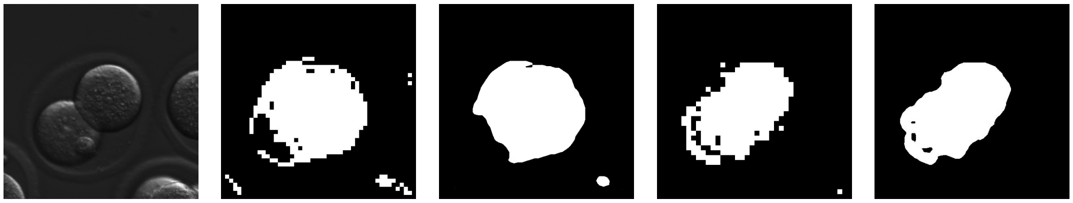

Figure 3. Sample results of the Cell Mask estimation task, from left to right, original image, raw mask using SVM + Local STIP descriptor, post-processed mask (SVM STIP) using a median filter of 15x15, raw mask using 2d encoder, post-processed mask (2d encoder) using a median filter of 15x15.

4.3 Cell Instance Segmentation Baseline

Our baseline for the instance segmentation task is also the Hough Transform. Unlike the semantic segmentation task, we do not use the union of the predicted circles, but rather consider each prediction as a cell instance. We optimize the parameters of the Hough Transform, empirically we find the best parameter set for the detection task to be:

- Minimum centroid distance: 10 pixels
- Accumulator threshold: 50
- Maximum Radius: 140 pixels (largest cell in stage1)
- Minimum Radius: 70 pixels

As outlined in section 3, we enhance the effectiveness of the baseline by adding prior information of the embryo's location and the the amount of cells. The former is already available from the segmentation masks obtained, the later will be obtained by estimating the embryos current stage.

2D Convolutional Auto-encoder Semantic Segmentation		
Voxel Size	11	13
Local Information	0.95	0.95
Local Information+ Pixel Coordinate	**0.97**	**0.97**

Table 4. Average F1 measure for the 2D convolutional encoder using only local image information and adding normalized pixel coordinate

Convolutional Features Semantic Segmentation

Patch Size/ Stack Size (frames)	2	3	4	5
11x11	**0.97**	0.96	0.96	0.96
13x13	**0.97**	0.96	0.96	0.95

Table 5. Results of temporal stack explorations,

4.4 Embryo Stage Classification Using Handcrafted Features

For the embryo stage prediction we establish four different categories according to the cell stage, *Stage 1*: 1 cell , *Stage 2*: 2 cells, *Stage 3*: 4-cells, we also add an artificial 'stage 0' which accounts for the transition between stage 2 and 3 as in this scenario we have 3 cells. The distribution of classes for this division is shown in Table 6.

Dataset Instance Distribution for Stage Classification

Set	Stage 1	Stage 2	Stage 3	Stage 0	Total
Train	3260 (14%)	11573 (50%)	7621 (33%)	500 (0.02%)	22954
Test	1482 (15%)	4959 (50%)	3112 (32%)	266 (0.03%)	9819

Table 6. Distribution of the train and test sets for the stage classification, instance number and percentage per class

For the handcrafted feature approach, we use a classic Bag of Visual Words approach, in which we replace the interest point detection task, with a dense sampling on a grid with cell size of 32x32 pixels. Each cell on the grid is then represented by its SIFT descriptor[23] which is calculated using 8 directions and 16 subregions. On the clustering step, we use the K-means algorithm and explore its K parameter to optimize the classification F-Measure. Empirically we find the best value for K to be 130. Just like in Section 4, we explore the best parameters for the SVM classifier. Results for the best SVM are shown in Table 7.

Results of Stage Classification using Bag of Words

class	Precision	Recall	F-Measure
Stage 0	0.10	0.007	0.08
Stage 1	0.81	0.90	0.85
Stage 2	0.83	0.17	0.28
Stage 3	0.74	0.87	0.80
Weighted Avg.	0.79	0.75	0.72

Table 7. Results for the BOW approach for embryo stage classification

4.5 Embryo Stage Classification Stage Classification Using Learned Features

Global features for the Cell Segmentation task are learned using a convolutional encoder based on the squeeze net,[24] a deep neural network that has a similar performance to the well known AlexNet[25] in the ImageNet challenge, however it contains significantly fewer parameters. The squeeze architecture is composed by 'fire modules' that are small ensembles of layers with filters of dimension 1x1 concatenated to two parallel layers, one with filters of 1x1 and the other with kernels of 3x3. We use a smaller version of the original architecture with only 6 fire-modules and enhance it with two skip connections as proposed in,[26] these connections create two residual paths which bring a small improvement on the encoder's performance. We train again from random weight initialization using the Keras Framework[21] using the Adam optimizer with learning rate of 0.001 and decay of 0.0001. Table 8 summarizes the results for the embryo stage classification task using the proposed encoder. Results show that the convolutional encoder significantly improves the classification accuracy over the BOW strategy, nevertheless the stage 0 class remains challenging for any of the proposed strategies.

Finally, we use the stage classification information to improve the detection results of the Hough transform, we limit the number of cells according to the predicted cell stage and further optimize its remaining parameters for each stage. Additionally, given the good results for the semantic segmentation, we eliminate predictions that are located out of the segmentation mask.

The final results show that the proposed strategy improves the average jaccard index for stages 1 and 3 while remaining comparable with the baseline results at stages 0 and 2. Overall the proposed optimization improves 2% in the global evaluation. Table 9 summarize the results.

Results of Stage Classification using Convolutional Encoder

class	Precision	Recall	F-Measure
Stage 0	0.23	0.09	0.13
Stage 1	0.99	0.88	0.93
Stage 2	0.96	0.90	0.93
Stage 3	0.83	1.00	0.96
Weighted Avg.	0.95	0.95	0.95

Table 8. Evaluation of the global learned features on the stage classification subtask

Results of Hough Transform for cell detection

Approach	Stage 0	Stage 1	Stage 2	Stage 3	Weighted Avg.
Hough	**0.68**	0.77	**0.71**	0.63	0.69
Hough + BOW Stages	0.63	0.77	0.66	0.60	0.65
Hough + CNN Stages	0.64	**0.90**	0.69	0.65	0.70
Hough + BOW Stages + STIP Mask	0.63	0.77	0.67	0.60	0.66
Hough + CNN Stages + CNN Mask	0.65	**0.90**	0.69	**0.67**	**0.71**

Table 9. Ablation experiments for the proposed approach using both hand-crafted features and automatically learned features

5. CONCLUSIONS

We presented a novel strategy to refine the instance segmentation of mouse embryo cells using reliable prior information from the embryo's locations and its current cell stage, which we use to approximate the number of cell instances. Both priors seem to have a significant impact in the cell instance segmentation task as it raises the baseline Jaccard coefficient from 0.69 up to 0.71.

The two main task were explored using approaches based on handcrafted features and automatically learned ones. Experimental evaluation shows an improvement of the base performance for one source of features, automatically learned features consistently outperform hand crafted ones in the evaluation of every subtask, and in the final instance segmentation. This suggest that, while the handcrafted features have are capable of capturing the relevant information at both local and global scale, the automatically learned features are better choices for the tasks explored in this paper.

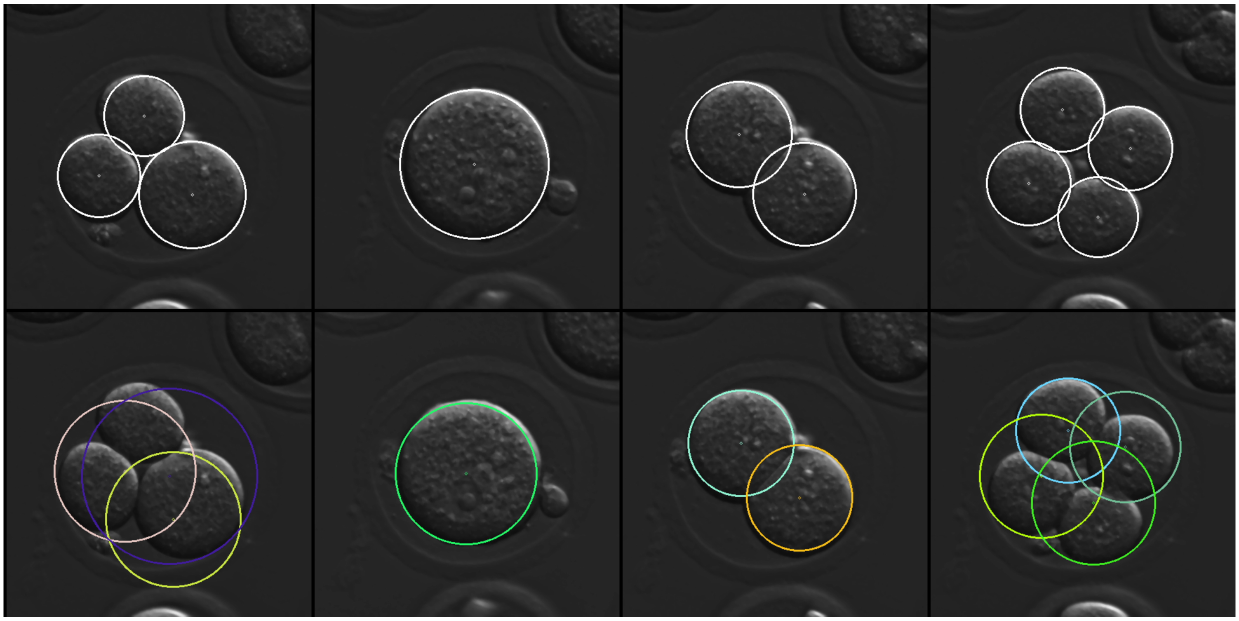

Figure 4. Qualitative results for the instance segmentation task, top row shows some 'Best case scenario' (little cell overlap and circular shapes) segmentations across different cell stages. Bottom row shows fail condition for stages 0 and 4 where the overlap and shape variations yield poor results.

REFERENCES

[1] Al-Kofahi, O., Radke, R. J., Goderie, S. K., Shen, Q., Temple, S., and Roysam, B., “Automated cell lineage construction: a rapid method to analyze clonal development established with murine neural progenitor cells,” *Cell cycle* **5**(3), 327–335 (2006).

[2] Khodadadi, V., Fatemizadeh, E., and Setarehdan, S. K., “Optimized kalman filter based on second momentum and triple rectangular for cell tracking on sequential microscopic images,” in [*Biomedical Engineering (ICBME), 2015 22nd Iranian Conference on*], 251–256, IEEE (2015).

[3] Baggett, D., Nakaya, M.-a., McAuliffe, M., Yamaguchi, T. P., and Lockett, S., “Whole cell segmentation in solid tissue sections,” *Cytometry Part A* **67**(2), 137–143 (2005).

[4] Jonaitis, D., Raudonis, V., and Lipnickas, A., “Application of computer vision methods in automatic analysis of embryo development,” in [*Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2015 IEEE 8th International Conference on*], **1**, 258–260, IEEE (2015).

[5] Lundin, K. and Ahlström, A., “Quality control and standardization of embryo morphology scoring and viability markers,” *Reproductive biomedicine online* **31**(4), 459–471 (2015).

[6] Ortiz De Solórzano, C., Garcia Rodriguez, E., Jones, A., Pinkel, D., Gray, J. W., Sudar, D., and Lockett, S. J., “Segmentation of confocal microscope images of cell nuclei in thick tissue sections,” *Journal of Microscopy* **193**(3), 212–226 (1999).

[7] Dow, A. I., Shafer, S. A., Kirkwood, J. M., Mascari, R. A., and Waggoner, A. S., “Automatic multiparameter fluorescence imaging for determining lymphocyte phenotype and activation status in melanoma tissue sections,” *Cytometry Part A* **25**(1), 71–81 (1996).

[8] Lin, G., Chawla, M. K., Olson, K., Guzowski, J. F., Barnes, C. A., and Roysam, B., “Hierarchical, model-based merging of multiple fragments for improved three-dimensional segmentation of nuclei,” *Cytometry Part A* **63**(1), 20–33 (2005).

[9] Ortiz de Solorzano, C., Malladi, R., Lelievre, S., and Lockett, S., “Segmentation of nuclei and cells using membrane related protein markers,” *journal of Microscopy* **201**(3), 404–415 (2001).

[10] Meseguer, M., Herrero, J., Tejera, A., Hilligsøe, K. M., Ramsing, N. B., and Remohí, J., “The use of morphokinetics as a predictor of embryo implantation,” *Human reproduction* **26**(10), 2658–2671 (2011).

[11] Cicconet, M., Gutwein, M., Gunsalus, K. C., and Geiger, D., “Label free cell-tracking and division detection based on 2d time-lapse images for lineage analysis of early embryo development,” *Computers in biology and medicine* **51**, 24–34 (2014).

[12] Ballard, D. H., “Generalizing the hough transform to detect arbitrary shapes,” *Pattern recognition* **13**(2), 111–122 (1981).

[13] Laptev, I. and Lindeberg, T., “Space-time interest points,” in [*9th International Conference on Computer Vision, Nice, France*], 432–439, IEEE conference proceedings (2003).

[14] Bengio, Y., Courville, A., and Vincent, P., “Representation learning: A review and new perspectives,” *IEEE transactions on pattern analysis and machine intelligence* **35**(8), 1798–1828 (2013).

[15] Harris, C. and Stephens, M., “A combined corner and edge detector.,” in [*Alvey vision conference*], **15**(50), 10–5244, Citeseer (1988).

[16] Koenderink, J. J. and van Doorn, A. J., “Representation of local geometry in the visual system,” *Biological cybernetics* **55**(6), 367–375 (1987).

[17] Paulin, M., Mairal, J., Douze, M., Harchaoui, Z., Perronnin, F., and Schmid, C., “Convolutional patch representations for image retrieval: an unsupervised approach,” *International Journal of Computer Vision* , 1–20 (2016).

[18] LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R., “Efficient backprop,” in [*Neural networks: Tricks of the trade*], 9–48, Springer (2012).

[19] Zhang, Y., Jin, R., and Zhou, Z.-H., “Understanding bag-of-words model: a statistical framework,” *International Journal of Machine Learning and Cybernetics* **1**(1-4), 43–52 (2010).

[20] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X., "TensorFlow: Large-scale machine learning on heterogeneous systems," (2015). Software available from tensorflow.org.

[21] Chollet, F., "keras." `https://github.com/fchollet/keras` (2015).

[22] Duchi, J., Hazan, E., and Singer, Y., "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research* **12**(Jul), 2121–2159 (2011).

[23] Lowe, D. G., "Object recognition from local scale-invariant features," in [*Computer vision, 1999. The proceedings of the seventh IEEE international conference on*], **2**, 1150–1157, Ieee (1999).

[24] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K., "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360* (2016).

[25] Krizhevsky, A., Sutskever, I., and Hinton, G. E., "Imagenet classification with deep convolutional neural networks," in [*Advances in neural information processing systems*], 1097–1105 (2012).

[26] He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*], 770–778 (2016).