TOWARDS AUTOMATIC EMBRYO STAGING IN 3D+T MICROSCOPY IMAGES USING CONVOLUTIONAL NEURAL NETWORKS AND POINTNETS

Manuel Traub\textsuperscript{1,2}       Johannes Stegmaier\textsuperscript{1,*}

\textsuperscript{1}Institute of Imaging and Computer Vision, RWTH Aachen University, Aachen, Germany
\textsuperscript{2}Institute for Automation and Applied Informatics, Karlsruhe Institute of Technology, Karlsruhe, Germany

ABSTRACT

Automatic analyses and comparisons of different stages of embryonic development largely depend on a highly accurate spatio-temporal alignment of the investigated data sets. In this contribution, we compare multiple approaches to perform automatic staging of developing embryos that were imaged with time-resolved 3D light-sheet microscopy. The methods comprise image-based convolutional neural networks as well as an approach based on the PointNet architecture that directly operates on 3D point clouds of detected cell nuclei centroids. The proof-of-concept experiments with four wild-type zebrafish embryos render both approaches suitable for automatic staging with average deviations of 0.45 – 0.57 hours.

Index Terms— Convolutional Neural Networks, PointNet, Regression, Developmental Biology, Embryo Staging

1. INTRODUCTION

Embryonic development is characterized by a multitude of synchronized events, cell shape changes and large-scale tissue rearrangements that are crucial steps in the successful formation of a new organism \textsuperscript{2}. To be able to compare these developmental events among different wild-type individuals, different mutants or upon exposure to certain chemicals, it is highly important to temporally synchronize acquired data sets, such that corresponding developmental stages are compared to one another \textsuperscript{3,4}. While the temporal synchronization is typically easy in small 2D screens, it becomes already a tedious undertaking when analyzing high-throughput screens consisting of thousands of repeats. Shifting dimensions to 3D the challenge of reproducible temporal synchronization becomes even more difficult and finally almost impossible for a human observer if the time domain is additionally considered. Current approaches to embryo staging largely rely on human intervention with the risk of subjective bias and might require specific labeling strategies or sophisticated visualization tools to cope with large-scale time-resolved 3D data \textsuperscript{5,6}.

Under the assumption that development progresses equally in all embryos, a rough temporal synchronization of the data sets can be obtained by measuring the time between fertilization and image acquisition \textsuperscript{7}. Moreover, a visual staging can be performed after image acquisition by identifying certain developmental characteristics of a single snapshot of a time series or after chemical fixation of the embryo and by comparing it with a series of standardized views that show the characteristic development of a wild-type specimen at a standardized temperature \textsuperscript{8}. In early phases of development, synchronized cell divisions regularly double the cell count and can be used to reliably align early time points \textsuperscript{9,10}. Villoutreix et al. use measured cell counts over time to specify an affine transformation on the temporal domain, \textit{i.e.}, the time axis is scaled such that the cell count curves of multiple embryos best overlap \textsuperscript{11}. While cell counts might be reliable in specimens like the nemathode \textit{C. elegans}, where adult individuals exhibit a largely identical number of cells, using the cell counts for synchronization in more complex animal models becomes more and more ambiguous. As development progresses, strong variation of total cell counts, cell sizes, cell shape and tissue formation are observed even among wild-type embryos \textsuperscript{12,13}. If the number of individuals is limited and if later time points are considered, a manual identification of characteristic anatomical landmarks can be used to assign a specific developmental stage to selected frames \textsuperscript{14,15,16}. With steadily increasing degrees of automation of experimental setups, it will be impractical to perform the staging with the human in the loop and automated approaches could thus help to further automate experimental protocols.

In this contribution, we analyzed two learning-based approaches for their suitability to automate embryo staging in large-scale 3D+\textit{t} microscopy experiments. The methods comprise image-based convolutional neural networks (CNNs) as well as a point cloud-based approach using the PointNet architecture \textsuperscript{16}. Both approaches were adapted for regression tasks and assessed under different hyperparameter and training conditions. We applied the methods to 2D maximum intensity projections and 3D point clouds of cell nuclei centroids (Fig.\textsuperscript{1}) that were obtained from terabyte-scale...
3D+t light-sheet microscopy experiments of four wild-type zebrafish embryos (*D. rerio*) ubiquitously expressing a green fluorescent protein (GFP) in their cell nuclei [10].

2. AUTOMATIC EMBRYO STAGING AS A REGRESSION PROBLEM

We analyzed two conceptually different approaches to automatically predict the hours post fertilization (hpf), a common measure for staging zebrafish embryos, either from 2D maximum intensity projection images or from 3D point clouds of centroids of fluorescently labeled cell nuclei. A schematic overview of both investigated approaches is depicted in Fig. 2 and the details of the employed methods are summarized in the remainder of this section.

2.1. CNN-based Embryo Staging on Downsampled Maximum Intensity Projections

For the image-based approach we selected three popular CNN architectures, namely VGG-16, ResNet-18 and GoogLeNet [17, 18, 19]. In all cases, the classification output layer was replaced with a single regression node with a linear activation to predict the stage of the current input in hours post fertilization. In addition to the relatively large pretrained networks, we added a more shallow VGG-like network consisting of four blocks of convolutions followed by ReLU activation (3 × 3 kernels, stride 1, 32 layers in the first convolutional layer and doubling the depth after each pooling operation), three max-pooling layers (2 × 2 kernels, stride 2) and two fully-connected layers (128 and 64 nodes, respectively, each followed by a dropout layer with dropout probability of 0.2) before mapping to the final regression node (Fig. 2 top).

2.2. Regression PointNet for Automatic Embryo Staging

To perform the automatic staging on 3D point clouds, we equipped the original PointNet architecture by Qi et al. [16] with a regression layer as the target instead of classification scores (Fig. 2 bottom). In brief, the PointNet architecture processes input points separately by first putting them through learnable input and feature transformation modules (3 × 3 and 64 × 64) that are connected via a point-wise multilayer perceptrons with shared weights. This step is intended to orient the point cloud consistently and thus to obtain invariance with respect to input transformations. The transformed points are then fed through another set of multilayer perceptrons with shared weights (64, 128 and 1024 nodes in the hidden layers), a max-pooling operation that serves as a symmetric function to compensate for permutations of the input points and a final fully-connected multilayer perceptron with 512 and 256 hidden layers that map to k output scores in the original classification PointNet. In the final fully-connected layer, however, we map to a single output neuron with linear activation that...
The CNN-based approaches and pretrained networks were implemented in MATLAB using the Deep Learning Toolbox. We used the ADAM optimizer, tried different mini-batch sizes of 8, 16, 32 and 64 and trained for 100 epochs. Optionally, we used data augmentation including reflection, scaling (0.9 – 1.0), random rotations (0 – 180°) and random translations (± 5 px). In addition to training the networks from scratch, we also investigated the performance of fixing the pretrained weights of the networks at different layers and only fine-tuned the deeper layers to the new task. For VGG-16, we tested all positions prior to the max-pooling layers (5 possibilities), for ResNet-18, all positions before and after pairs of ResNet modules (5 possibilities) and for GoogLeNet, all locations between Inception modules (9 possibilities).

The PointNet was implemented using TensorFlow by modifying the original repository [16]. Training was performed using the ADAM optimizer and we randomly sampled 4096 centroids of each time point at each training iteration as the fixed-size input to the PointNet. As opposed to the original implementation, we consistently obtained better results when disabling dropout and batch normalization during training and thus only report these results. For augmentation, random rotations around the origin were applied to all points and point jittering was used to apply small random displacements to each of the 3D input points.

In both scenarios, we performed the training with a 4-fold cross-validation using three embryos for training and one embryo for testing in each fold, respectively. Reported average values and standard deviations were computed on all folds. Both the CNNs and the PointNet were not aware of the total number of cells present in a particular frame, i.e., the prediction was solely based on the appearance, shape and orientation of the images and point clouds.

4. RESULTS AND DISCUSSION

The results of the best combination of architecture, augmentation and hyperparameters are summarized in Tab. [1] The

3. EXPERIMENTS

3.2. Implementation and Training Details

Fig. 2. Illustration of the two different approaches used for automatic embryo staging. Different flavors of CNNs are applied to 2D maximum intensity projection images (top) and a PointNet [16] is applied to 3D point clouds of detected cell nucleus centroids (bottom). Both methods map to a final output node that predicts the hours post fertilization of the input. represents the regression target. Instead of the sparse softmax cross-entropy loss as performed for the classification task in [16], we use a mean squared error loss to regress to the targeted measure in hpf.

3.1. Data Acquisition and Training Data Generation

Raw image data was acquired using a custom-made lightsheet microscope as described in [10]. Experiments considered in this contribution comprise four wild-type zebrafish embryos expressing a transgenic H2A-GFP fusion protein in their cell nuclei that were imaged from about 2.5 to 16 hpf and were performed in accordance with the German animal protection regulations, approved by the Regierungspräsidium Karlsruhe, Germany (Az. 35-9185.64) [10]. A single experiment acquired with these settings accumulates about 10 terabytes of raw image data. Cell nuclei of all four embryos were automatically detected and we aligned the 3D+t point clouds in a consistent orientation with the prospective dorso-ventral axis oriented along the y-axis [1] [10] [6]. For each data set we cropped a temporal window spanning from 4.7 – 10 hpf by manually identifying the 10 hpf stage of all data sets as a synchronization time point, assuming constant development of all embryos prior to that stage. Stage identification was performed on 3D+t point clouds visualized in KitWare’s ParaView [20]. We note that this manual stage identification required several hours of interactive data visualization and analysis, which could largely benefit from automation.

For the image-based CNNs, we computed 2D maximum intensity projections for all data sets and all time points along the axial direction. Intensity values of all frames were scaled to the 8 bit range and contrast adjusted such that 0.3% of the pixels at the lower and the upper end of the intensity range were saturated. To be able to use networks that were pretrained on the ImageNet database, we converted the single channel 8 bit images to the required input resolution of 224 × 224 pixels with three redundant channels. We ended up with 370 frames for each embryo spanning a duration of 4.7 – 10 hpf, i.e., 1480 images for all embryos in total. The PointNet was trained on 3D centroids of detected nuclei, which were extracted from the same four embryos as for the image-based experiments (between 4160 to 19794 per data set) [6].
Table 1. Staging accuracy of the different methods with top-scoring methods highlighted in bold-face letters.

| Method       | Mean Dev. ± Std. Dev. (h) | RMSD (h) |
|--------------|---------------------------|----------|
| VGG-Simple   | 0.51 ± 0.40               | 0.65     |
| VGG-16 [17]  | 0.57 ± 0.41               | 0.70     |
| ResNet-18 [18]| 0.45 ± 0.30               | 0.54     |
| GoogLeNet [19]| 0.50 ± 0.37               | 0.62     |
| PointNet [16]| 0.47 ± 0.29               | 0.55     |

The smallest mean deviation from the ground truth was obtained with the ResNet-18 approach with an average deviation of 0.45 ± 0.30 hours (mean ± std. dev.) with a mini-batch size of 32 and with freezing the pretrained weights of the first four ResNet modules. These results were closely followed by the PointNet approach (0.47 ± 0.29 hours), the GoogLeNet trained from scratch with enabled data augmentation and a mini-batch size of 8 (0.50 ± 0.37 hours) and the Simple-VGG architecture (0.51 ± 0.40 hours) with augmentation enabled and a mini-batch size of 16. VGG-16 yielded the poorest performance in all investigated settings and the best results reported in Tab. [1](0.57 ± 0.41 hours) were obtained using a fine-tuning approach with keeping the weights up to the third convolutional block fixed, with enabled data augmentation and a mini-batch size of 8. The 138 million parameters of VGG-16 combined with a limited amount of training images showed a tendency to overfit despite using data augmentation and dropout regularization. However, with fixed pretrained weights and enabled data augmentation the qualitative trend did not change and the network even failed to learn the training data properly, which is probably due to the differences of the ImageNet data from fluorescence microscopy images.

The performance of the image-based projection approach directly depends on the initial orientation of the embryo during image acquisition. On the contrary, the point cloud-based method is essentially independent of the initial orientation and randomized orientation of the point clouds are explicitly used as augmentation strategy during the training process to further improve the invariance of the stage prediction with respect to the spatial orientation. The staging can thus be performed even without a tedious manual alignment of different experiments or the use of spatial registration approaches, which might be required for the image-based approach. Robustness with respect to specimen orientation could be achieved for the image-based CNN approach by using 3D CNNs that are directly applied on downsampled raw image stacks [21]. However, this would require extensive preprocessing of the terabyte-scale data sets like multi-view fusion and downsampling, which might be impractical in real-time scenarios. We currently perform these steps solely in feature space on extracted cell centroids for performance reasons.

The general average trends obtained with the ResNet-18, GoogLeNet and the PointNet approaches nicely resemble the ground truth and are arguably the most promising candidates among the presented methods. However, an average deviation of about 27 – 30 minutes from the real developmental time is still too coarse for some practical applications, where rapid tissue changes can happen in a matter of a few minutes. We thus have to further improve the precision of the presented approaches and consider it a first proof-of-concept for automatic embryo staging in 3D+t experiments. To compensate the lack of training data, we used several data augmentation strategies including various image transformations, point jittering and random rotations of the entire data set. However, data augmentation in the image-space did not consistently improve the results. As a straightforward extension, we could simply enlarge the training set by additional experiments and add realistic simulations like our previously presented approach for semi-synthetic data generation [22]. Moreover, the data sets we used for demonstration were manually synchronized and assumed constant development of the embryo prior to the synchronization time point, with a potential subjective bias. To obtain a more objective ground truth, calibration experiments would be required, e.g., by imaging multiple channels of reporter genes that are known to be expressed within a distinct time window. Finally, an automatic spatial registration of the temporally aligned data sets will be the next logical step in the pipeline to eventually be able to automatically register numerous data sets in space and time.
5. REFERENCES

[1] J. Stegmaier, J. C. Otte, A. Kobitski, et al., Fast Segmentation of Stained Nuclei in Terabyte-Scale, Time Resolved 3D Microscopy Image Stacks,” PLoS ONE, vol. 9(2), pp. e90036, 2014.

[2] T. Lecuit and L. Le Goff, Orchestrating Size and Shape during Morphogenesis,” Nature, vol. 450, no. 7167, pp. 189, 2007.

[3] C. Castro-González, M. A. Luengo-Oroz, L. Duloquin, et al., A Digital Framework To Build, Visualize and Analyze a Gene Expression Atlas With Cellular Resolution in Zebrafish Early Embryogenesis,” PLOS Computational Biology, vol. 10, no. 6, pp. e1003670, 2014.

[4] L. Guignard, C. Godin, U.-M. Fiuza, et al., Spatio-Temporal Registration of Embryo Images,” in Proceedings of the IEEE International Symposium on Biomedical Imaging, 2014, pp. 778–781.

[5] T. Pietzsch, S. Saalfeld, S. Preibisch, and P. Tomancak, BigDataViewer: Visualization and Processing for Large Image Data Sets,” Nature Methods, vol. 12, no. 6, pp. 481–483, 2015.

[6] B. Schott, M. Traub, C. Schlagenhauf, et al., EmbryoMiner: A New Framework for Interactive Knowledge Discovery in Large-Scale Cell Tracking Data of Developing Embryos,” PLOS Computational Biology, vol. 14, no. 4, pp. 1–18, 04 2018.

[7] M. Shahid, M. Takamiya, J. Stegmaier, et al., Zebrafish Biosensor for Toxicant Induced Muscle Hyperactivity,” Scientific Reports, vol. 6, no. 23768, pp. 1–14, 2016.

[8] C. B. Kimmel, W. W. Ballard, S. R. Kimmel, B. Ullmann, and T. F. Schilling, Stages of Embryonic Development of the Zebrafish…” Developmental Dynamics: An Official Publication of the American Association of Anatomists, vol. 203, no. 3, pp. 253–310, 1995.

[9] E. Faure, T. Savy, B. Rizzi, et al., A Workflow to Process 3D+Time Microscopy Images of Developing Organisms and Reconstruct Their Cell Lineage.” Nature Communications, vol. 7, no. 8674, pp. 1–10, 2016.

[10] A. Kobitski, J. C. Otte, M. Takamiya, et al., An Ensemble-Averaged, Cell Density-based Digital Model of Zebrafish Embryo Development Derived from Light-Sheet Microscopy Data with Single-Cell Resolution,” Scientific Reports, vol. 5, no. 8601, pp. 1–10, 2015.

[11] P. Villoutreix, J. Delile, B. Rizzi, et al., An Integrated Modelling Framework from Cells To Organism Based On a Cohort of Digital Embryos,” Scientific Reports, vol. 6, pp. 1–11, 2016.

[12] C. C. Fowlkes, C. L. Luengo Hendriks, S. V. E. Keränen, et al., A Quantitative Spatiotemporal Atlas of Gene Expression in the Drosophila Blastoderm,” Cell, vol. 133, no. 2, pp. 364–374, 2008.

[13] K. McDole, L. Guignard, F. Amat, et al., In Toto Imaging and Reconstruction of Post-Implantation Mouse Development at the Single-Cell Level,” Cell, vol. 175, no. 3, pp. 859–876, 2018.

[14] S. E. Muenzing, M. Strauch, J. W. Truman, et al., larvalign: Aligning Gene Expression Patterns from the Larval Brain of Drosophila melanogaster,” Neuroinformatics, vol. 16, no. 1, pp. 65–80, 2018.

[15] K. Winkley and M. Veeman, A Temperature-Adjusted Developmental Timer for Precise Embryonic Staging,” Biology Open, vol. 7, no. 6, pp. bio032110, 2018.

[16] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.

[17] K. Simonyan and A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv preprint [arXiv:1409.1556], pp. 1–14, 2014.

[18] K. He, X. Zhang, S. Ren, and J. Sun, Deep Residual Learning for Image Recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[19] C. Szegedy, W. Liu, Y. Jia, et al., Going Deeper With Convolutions,” Proceedings of the IEEE Computer Society Conference On Computer Vision and Pattern Recognition, pp. 1–9, 2015.

[20] U. Ayachit, The Paraview Guide: A Parallel Visualization Application, Kitware, Inc., 2015.

[21] C. R. Qi, H. Su, M. Nießner, et al., Volumetric and Multi-View CNNs for Object Classification on 3D Data,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5648–5656.

[22] J. Stegmaier, J. Arz, B. Schott, et al., Generating Semi-Synthetic Validation Benchmarks for Embryomics,” in Proceedings of the IEEE International Symposium on Biomedical Imaging, 2016, pp. 684–688.