

## HD-Hau-GE

Authors: Carsten Haubold, Martin Schiegg, David Stöckel, Steffen Wolf, Fred A. Hamprecht

Email: [fred.hamprecht@iwr.uni-heidelberg.de](mailto:fred.hamprecht@iwr.uni-heidelberg.de)

Platform: Linux

Prerequisites: None

### *HD-Hau-GE: SUMMARY*

Our tracking-by-assignment approach, *Conservation Tracking* [1], incorporates a pixel-wise classification into foreground and background, followed by connected component extraction to obtain the segmentation. To solve the assignment problem, for each detection we build a graphical model containing random variables, which are connected to possible successors via transition nodes. The most probable configuration is found by running a global optimization. To cope with under-segmentation, a detection is allowed to contain more than one object. Two classifiers are trained to support the optimization in deciding how many cells each detection contains, and which cells possibly divide. The training of all these classifiers was facilitated by the open source software framework *ilastik* [2]. All three classifiers are Random Forests, trained by manual annotation of the Training datasets with interactive feedback in *ilastik*. Only 5 to 10 annotated examples for every case were required to reach the submitted quality.

### *HD-Hau-GE: PREPROCESSING*

No preprocessing is carried out before segmentation.

### *HD-Hau-GE: SEGMENTATION*

To extract detections from the raw data, we interactively train a random forest by sparsely annotating pixels (or voxels), and let it predict probabilities for all pixels indicating whether they belong to the foreground or background class. The annotations are drawn by the user with a brush tool, while the predictions are presented as interactive feedback. The annotations are refined by drawing additional strokes until the predictions yield properly segmented cells or nuclei. The random forest works with a set of features computed from the raw images (e.g., Gaussian gradient magnitude or the structure tensor eigenvalues). We threshold the pixel-wise probability for the foreground class at  $\theta_{thresh}$ , and extract the connected components to get the segmentation. We also filter detections that are too small and comprise less pixels than  $s_{min}$ , or are larger than  $s_{max}$ .

## HD-Hau-GE: TRACKING

The tracking routine [1] builds a graphical model of the detections and the possible transitions. Conservation Tracking is not restricted to tracking cells or nuclei, but can track arbitrary dividing or non-dividing objects. It allows objects to merge, to split after a merge, and to divide. To be able to distinguish the very similar events of splitting and dividing, it uses two classifiers: one for object count per detection, and one for divisions. The first classifier is trained to detect how many cells or nuclei are combined into a segment. The division classifier learns, based on a mother cell and two children candidates in the next time frame, which features indicate a division. The classifiers are trained in a similarly interactive fashion as the pixel classification mentioned above: a user assigns the correct class to some detections in the GUI, and the current predictions for all other detections are presented as colored overlay over the data. By annotating more detections, the user can improve the classifiers to his/her satisfaction. The optimization procedure uses those classifiers as a guide, but can contradict them to conserve objects in the global temporal context. The classifier influence can be controlled using parameters as described below.

*Graphical Model and Global Optimization.* As mentioned before, we build a probabilistic graphical model and add random variables for all detections and possible transitions. Every detection can contain between 0 and  $m$  objects. Each of these configurations denotes a state in the optimization, and has an energy value attached. This energy is the negative logarithm of the probability for this state as it was predicted by the object count classifier. It can be adjusted with the factor  $w_{\text{det}}$  (we use the same notation for parameters as in the original paper [1]). The energy of an appearing cell can be controlled via  $w_{\text{app}}$ , and disappearances by  $w_{\text{van}}$ . The energy decreases linearly from a distance of  $d_{\text{border}}$  to the border. Cells that have their center closer than  $d_{\text{crop}}$  to their border, are discarded. A transition node is created for every detection-pair in consecutive temporal frames, if the Euclidean distance  $d$  between their centers is less than  $d_{\text{max}}$  pixels. Transition probabilities are defined relative to the Euclidean distance  $d$  as

$$p_{\text{tr}}(d) = \begin{cases} \exp\left(-\frac{d}{\alpha}\right) & \text{if this transition is active,} \\ 1 - \exp\left(-\frac{d}{\alpha}\right) & \text{otherwise.} \end{cases} \quad (1)$$

Transition utility energies are thus  $-w_{\text{tr}} \cdot \log(p_{\text{tr}}(d))$ . Each detection node in the graph that is connected to two outgoing transition nodes, meaning there are two possible fates in  $t + 1$ , could possibly divide. Then, we add an extra random variable that captures whether this division is active. The negative logarithm of the division probability predicted by the classifier is used as energy there, scaled by  $w_{\text{div}}$ . The optimal parameters for tracking ( $w_{\text{app}}$ ,  $w_{\text{van}}$ ,  $w_{\text{div}}$ ,  $w_{\text{det}}$ , and  $w_{\text{tr}}$ ) were obtained through structured learning on the

training datasets, the other parameters were found by a grid search. We find the tracking solution with minimal energy by transforming the graph described above to an integer linear program with indicator variables for each detection-, transition- and division hypothesis. We integrate conservation constraints, and optimize using the commercial CPLEX solver [3].

#### *HD-Hau-GE: POST-PROCESSING*

For the **PhC-C2DL-PSC** dataset, our segmentation was trained such that it separates most objects, and thus the detections did not cover the complete cells in the raw data. We applied a dilation operation to the segmentation with a structured disc element of radius  $r_{\text{dilate}} = 2$ . We did not dilate the results of any other dataset.

#### **REFERENCES**

1. Schiegg M, Hanslovsky P, Kausler BX, Hufnagel L, Hamprecht FA. Conservation tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, 2928-2935 (2013).
2. Sommer C, Strähle C, Köthe U, Hamprecht FA. Ilastik: Interactive learning and segmentation toolkit. In *Proceedings of the 8th IEEE International Symposium on Biomedical Imaging*, 230-233 (2011).
3. <http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/>