UFRGS-BR

Authors: Lucas Nedel Kirsten, Claudio R. Jung

Email: lnkirsten@inf.ufrgs.br, crjung@inf.ufrgs.br

Platform: Linux

Prerequisites: Python 3

UFRGS-BR: SUMMARY

Our approach follows the typical pipeline of a tracking-by-detection method. First, an object detector is used to detect the cells in each frame using an Oriented Bounding Box (OBB) representation, which are then converted to elliptical representations. Detection results are filtered using a score threshold and non-maximum suppression. Then, we generate tracklets by joining cells in subsequent frames using the Hungarian algorithm with an objective function based on the ProbIoU [5] metric. Finally, a global data association method, based on the work of Bise *et al.* [1], associates the tracklets to obtain cell trajectories and lineage trees.

UFRGS-BR: PREPROCESSING

For training the object detector, we cropped all dataset images using a 512 × 512 pixel-size window with 100 stride. This resulted in, for each sequence, 4,508 training images for **Fluo-N2DH-GOWT1**, 690 for **PhC-C2DL-U373**, and 16,314 for **Fluo-N2DH-HeLa**. During inference, we used the full sized images, except for **Fluo-N2DH-HeLa**, which we used the crops with a 256 stride (to provide some overlap in the borders) and then divided the detection parameters by a factor of 2 to retrieve them with the expected original image sizes. Specifically for **Fluo-N2DH-HeLa**, we augmented all images by a factor of 2 since it contains cells with very different sizes. This allows us to use the same network architecture for the object detector without the necessity of any adjustment in its anchors to better fit this dataset cell sizes. In order to generate the ground-truth OBB from the cell masks to train the detector, we fitted the minimum area rectangle on the segmentation mask of each cell using the OpenCV implementation of such algorithm.

UFRGS-BR: SEGMENTATION

In this work, we advocate the use of Elliptical Bounding Box (EBB) representations for cell detection and segmentation, which is a natural extension of the Oriented Bounding Box (OBB) representation, but usually provides a better fit to the cell shape. In particular, roughly circular shapes induce a naturally

ambiguous angular representation when OBBs are used, since any rotated square fits equally well a circle.

For such, we propose to use off-the-shelf OBB object detectors and then convert the output to elliptical bounding boxes (EBBs). For an OBB with center (*x, y*), width *W*, height *H*, and orientation *Θ*, we generate an ellipse with the same center and orientation, with semi-axes $a = W/2$ and $b = H/2$. If the OBB is clearly oriented (i.e., $W >> H$ or $W << H$), the EBB will preserve the orientation of the OBB. On the other hand, if the OBB is roughly square (i.e., $H \cong W$), the produced EBB will be roughly circular. In the case of a perfect square, the EBB simplifies to a single circle regardless of the orientation of the OBB, which mitigates the orientation ambiguity.

As the object detector, we chose to use the R2CNN [2] provided by the AlphaRotate benchmark. The R2CNN is a two-stage object detector, i.e. it has a region proposed module (RPN) before the detection module, that was first proposed in text detection problems. The model parameters used for all experiments are the same as in the AlphaRotate default architecture. The trained model uses the ResNet50 [3] backbone with pre-trained weights on the ImageNet dataset [4]. Both classification modules (RPN and head) use the categorical cross entropy loss, and both box regression modules use the smooth-L1 loss. Weight decay and momentum are set to 1e-4 and 0.9, respectively. We employ Momentum Optimizer over 1 GPU and 8 images per mini-batch. All models are trained by 100 epochs with learning rate started at 1e-3, except for **Fluo-N2DL-HeLa** which was trained for 24 epoch with learning rate started in 1e-4 to prevent overfitting, since there were over 4 times more training images in this dataset than the other ones. The learning rate was reduced tenfold at epochs 12, 16, and 20. Finally, in all experiments we applied random rotation and flips to augment the training data.

UFRGS-BR: TRACKING

First, we filter the detections, by using only candidates with scores returned by the object detector larger than a predefined threshold τ_s. Still, we usually have several overlapping candidates related to the same object, and use non-maximum suppression (NMS) to retrieve only the candidate with the highest score. In this step, we use the *ProbIoU* [5] geometrical similarity in order to measure the "overlap" degree between detections. Then, we generate a set of "reliable tracklets", obtained by computing the overlap between all detections of subsequent frames with the ProbIoU, and the optimal association between detections is obtained by solving a linear sum assignment problem with the Hungarian algorithm. In order to avoid bad associations caused by false positive detections, we only associate pairs of cells *p* and *q* for which $ProbIoU(p, q) < 0.1 * \sqrt{W^2 + H^2}$. Finally, we use a global data association algorithm based on

[1], in which they formulate a maximum-a-posteriori problem (MAP) solved by linear programming that addresses the tree structure association problem. The MAP problem is solved by defining a set of hypotheses associated with a probability score for combinations of the tracklets (i.e., each tracklet will respond to a set of possible hypotheses with their likelihood). More precisely, they assume the following five possible hypotheses: initiation, termination, translation, mitoses, and false positive. In our work, we propose the following modifications:

- We removed the mitosis detection algorithm. Hence, we decided not to differentiate between mitotic and non-mitotic cells, and consider all cells as potentially mitotic. Furthermore, to address this choice, we employ two different free parameters in order to adjust the probability distribution of the translation and mitoses hypothesis.
- We removed the initiation and termination hypothesis, and replaced them with a *completeness* one.

We re-defined all the probability computations in order to use only information regarding the detected cells (e.g., position, time frame distance, and confidence score returned by the object detector). In particular, we explore the confidence score to discriminate between the true and false positive hypotheses. For the translation and mitoses hypotheses, we used only the center and time distance between detections.

UFRGS-BR: POST-PROCESSING

No post-processing is performed after tracking.

REFERENCES

1. Bise R, Yin Z, Kanade T. Reliable cell tracking by global data association. In *Proceedings of the IEEE International Symposium on Biomedical Imaging*, 1004-1010 (2011).
2. Jiang Y, Zhu X, Wang X, Yang S, Li W, Wang H, Fu P, Luo Z. R2CNN: Rotational region CNN for orientation robust scene text detection. arXiv:1706.09579 (2017).
3. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770-778 (2016).
4. Deng J, Dong W, Socher R, Li L, Li K, Fei-Fei L. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 248-255 (2009).

5. Llerena JM, Zeni LF, Kirsten LN, Jung CR. Gaussian bounding boxes and probabilistic intersection-over-union for object detection. arXiv:2106.06072 (2021).