Appendices

The supplementary material provides details about our implementation (Section A), as well as additional experiments used to evaluate our framework (Section B). In addition to this PDF file, the supplementary material contains video clips that visually demonstrate the constructed cell trajectories for four different C2C12 sequences.

A Implementation Details

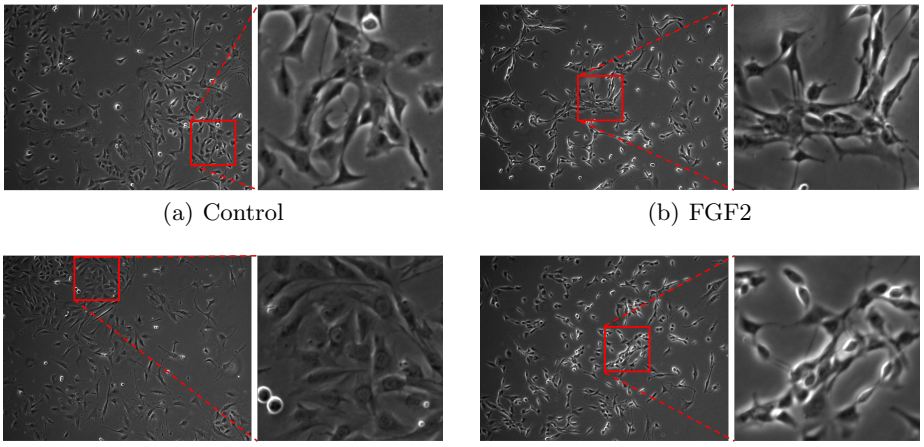
A.1 Graph Neural Network

We implemented the proposed graph neural network (GNN) model using the Pytorch Geometric library [4]. We train our framework with graphs based on microscopy sub-sequences of 10 frames while for the inference we use the entire sequence to construct the input graph. The prediction of all edges (a classification into 'active' and 'non-active' edges) is performed simultaneously.

The spatio-temporal features are normalized by *min-max scaling* for each graph, while the deep metric learning features are not pre-processed. To accommodate the high number of cell instances within a frame and to reduce the computational complexity, cell instances in consecutive frames are connected by edges only if their spatial Euclidean distance is smaller than a predefined threshold that is determined by the cells' *neighborhood region*. The neighborhood region $\mathcal{N}_R$ is defined based on the size of the cells' bounding box $size_{BB}$ and the rate of the cells' movement $size_{move}$. Formally, $\mathcal{N}_R = \alpha \cdot \max(\max_i(size_{BB_i}), \max_j(size_{move_j}))$. The maximization is applied to each axis separately. The hyper-parameter α is set to 2 or 4, depending on the sequence's density. For the graph neural network, we set the number of layers $L = 6$ to perform six message-passing steps, enabling information propagation between cell instances that are 6 frames apart. We set the dimension $d_{\mathcal{V}}$ of the node feature matrix to 32, where $d_{\mathcal{E}} = 64$ for the edge feature matrix. The *Adam* optimizer [7] is used with a learning rate of $1e-3$ and a weight decay of $1e-5$.

A.2 Deep Metric Learning

We use Pytorch metric learning library [10] to train ResNet18 [5] followed by multi-layer perceptron (MLP). The final embedding is $L2$ normalized and $d_{\text{DML}} = 128$. The training is done using batches with a size of 32. Batches are constructed by *m-per-class* sampler, which first randomly samples κ classes, and then randomly samples m images for each of the κ classes. Since the cell's appearance gradually changes during the sequence we perform the *m-per-class* sampling [9] using temporally adjacent frames. We set $\kappa = 8$ and $m = 4$. The ResNet18 and MLP models are optimized using two separated *Adam* optimizers [7] for each model with learning rates of $1e-5$ and $1e-4$, respectively. We also use weight decay of $1e-4$. We use the cell segmentation maps or marker



(a) Control (b) FGF2

(c) BMP2 (d) FGF2 + BMP2

Fig. 1. C2C12 Dataset Visualization. Example frames of four C2C12 sequences with different growth factor conditions [3]. Note the different appearance of the cells between the frames.

annotations to crop each frame into sub-images of all cell instances. We constructed the datasets used for DML training by assigning to each cell instance the index of its biological cell. In case the cell segmentation maps (rather than marker annotations) are available we exploit them to filter out the background via pixel-wise multiplication and extract features such as cell size and intensities.

B Experiments

We present additional experiments used to asses the proposed framework. In Section B.1, we provide more details regarding the evaluated datasets. Then, in Section B.2, we discuss the qualitative tracking results obtained for the C2C12 dataset [3]. Comparison to graph-based solutions is summarized at Section B.3. Further ablation study experiments are presented in Section B.4. In Sections B.5 and B.6, we present an edge case and the run time of our framework, respectively. Last, in Section B.7, we elaborate about the experiments used to evaluate our *deep-metric-learning* based feature extractor.

B.1 Datasets

C2C12 dataset [3] is acquired with four different growth factor conditions:fibroblast growth factor 2 (FGF2), bone morphogenetic protein 2 (BMP2), FGF2 + BMP2, and control (no growth factor). In Fig. 1, we present example frames for each condition. Note the different appearance of cells depending on the frames' growth factor. The FGF2 cells are partially overlapped and become thinner along the sequence while the BMP2 cells are spread. In BMP2+FGF2, we can observe both

Table 1. Cell tracking challenge [8,14] datasets properties. The table provides details regarding the datasets dimension, cell type, acquisition method (phase-contrast or fluorescence microscopy), number of frames, and spatial resolution.

Dataset	Dim.	Cell Type	Acq.	# Frames	Resolution
PhC-C2DH-U373	2D	U373	Ph.-C.	115	696×520
Fluo-N2DH-SIM+	2D	HL60	Fluo.	Varies	Varies
Fluo-N3DH-SIM+	3D	HL60	Fluo.	Varies	Varies
Fluo-C2DL-Huh7	2D	HCC	Fluo.	92	1024×1024
Fluo-N2DL-HeLa	2D	HeLa	Fluo.	30	1100×700

phenomena. In Table 1, we summarize the properties of the CTC datasets [8,14] used to evaluate our method.

B.2 Qualitative Tracking Results

Fig. 2 visually presents the trajectories of the C2C12 sequences. The figure illustrates the dense and cluttered cell environment and the prevalence of mitotic events and trajectory intersections. The video clips enclosed in this supplementary material further demonstrate the complexity of the extracted lineage trees and the strengths of the proposed method.

B.3 Comparison to Graph-Based Methods

As mentioned in the main paper, we compare our method in Table 1 (main paper) with two classical graph-based methods, namely Asymmetric Graph Cut (AGC) [1], Spatio-Temporal Global Data Association (ST-GDA) [2]. Our method far surpasses AGC and ST-GDA, where the average results are improved by more than 15% for the AA and the TE. Moreover, a few algorithms which competed at the Cell Tracking Challenge (CTC) are based on a graph strategy. For example, Scherr et al. [12] adopt the coupled minimum-cost flow algorithm suggested in [11]. Table 2 (main paper) presents our ranks for different CTC datasets with respect to all competing methods (three times top-1, one top-2, once top-4).

B.4 Ablation Experiments - Cont.

We present more ablation experiments in Table 2 to demonstrate the contribution of the distance and the similarity features in the edge encoder to the overall cell tracking performance. As expected, the distance features are slightly better than the similarity, as more entries in the feature vector are represented by them. When using both features the results are more robust over all the datasets and improved. Last, by comparing first and last rows, the significant contribution of

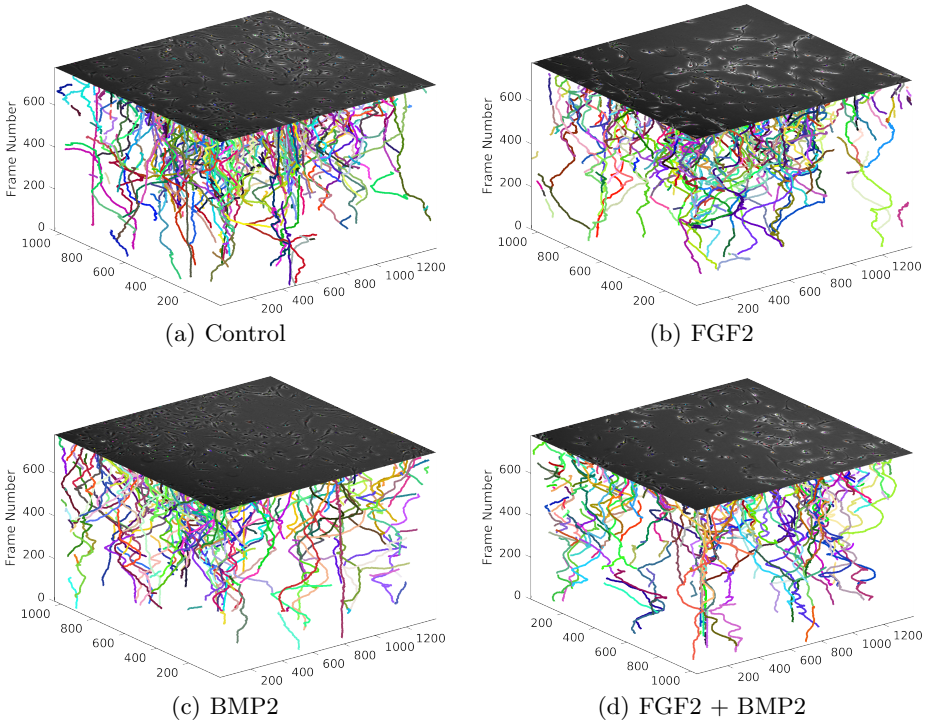
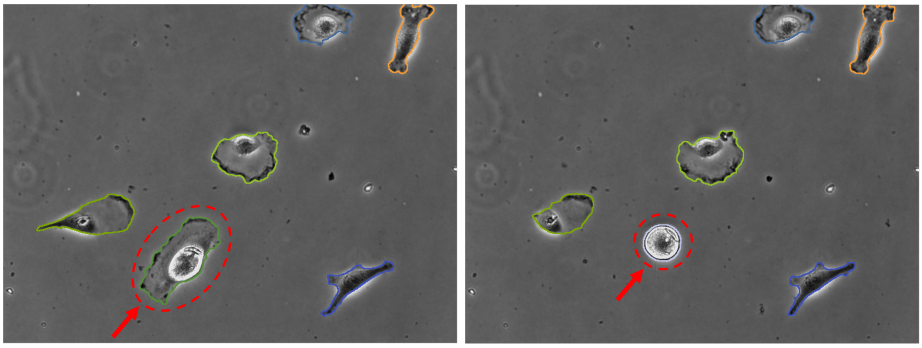


Fig. 2. 3D trajectories visualization. Example trajectories of four different C2C12 sequences dataset [3]. The X-Y axes present the original frame resolution (1392×1040 pixels), while Z axis presents the frame number. Note the clutter, density, overlap, and the random cell movements in each of the sequences.

the distance similarity block is highlighted. The results demonstrate the exclusive and the joint contributions of the distance and the similarity features in the edge encoder to the overall cell tracking performance.

Table 2. Edge encoder quantitative ablation study. The contribution of the distance and the similarity features in the edge encoder to the overall cell tracking performance. The symbols ✓ and ✗ indicate included or excluded, respectively. AA and TE stand for association accuracy and target effectiveness scores.

Edge Encoder Components		Results	
Distance	Similarity	AA	TE
✗	✗	0.840	0.598
✗	✓	0.991	0.987
✓	✗	0.993	0.988
✓	✓	**0.994**	**0.989**



(a) Frame t (b) Frame $t+1$

Fig. 3. Edge case visualization. Example of two consecutive frames from PhC-C2DH-U373 dataset. The proposed framework may fail associating cell instances in consecutive frames when both cell dynamics and appearance change where the difference in either of them is significant and abrupt.

B.5 Limitations and Edge Cases

The proposed tracker performs well even in the presence of abrupt changes in either cell appearance or dynamics (spatio-temporal changes). However, when both changes occur simultaneously the tracker may fail as illustrated in Fig. 3.

B.6 Run time

We trained and tested our framework using NVIDIA Tesla V100 DGXS 32-GB GPU. Training and evaluation run times varied between a few minutes to a maximum of one hour depending on the sequence length (number of frames), cell density, and dimension (2D or 3D). Most of the time is spent on the construction of the graph. It is worth noting that cell tracking methods mainly run offline, after the acquisition of the entire time-lapse microscopy sequence. Therefore, the run time is not critical.

B.7 Deep Metric Learning

In this section we present the experiments conducted to evaluate the DML component in our model.

Evaluated Metrics. For evaluation, we employ Adjusted Mutual Information (AMI) and Normalized Mutual Information (NMI), Precision at 1 (P@1), R-Precision (RP), and Mean Average Precision at R (MAP@R) [9] scores. AMI and NMI quantify the clustering performances and are based on k-mean algorithm, while P@1, RP, and MAP@R measure the neighboring area and are based on k-nearest neighbors (k-nn) algorithm. Our primary evaluation metric

Table 3. Deep metric learning ablation study scores (%)

Method	Sampler	P@1	RP	MAP@R
Triplet+L2 [6]	Proposed	80.6	33.2	27.4
Triplet+CS [6]	Proposed	82.3	35.1	29.7
Circle [13]	Proposed	83.1	36.8	31.7
MS [15]	m-per-class	79.9	35.2	29.8
MS [15]	Proposed	**84.7**	**37.8**	**32.8**

is MAP@R proposed recently in [9]. This metric is stable and suited for the selection of the best performing model checkpoints.

Ablation Study. We conducted an ablation study to justify the use of *multi-similarity loss* and miner [15]. We compare the results with those obtained by the triplet loss [6] with $L2$ distance and Cosine similarity in the embedded space. In both setting we use the *triplet margin* miner. We also examine the recently proposed *circle loss* [13]. Finally, we compare the performances of the proposed sampling mechanism which selects within-class samples from temporally adjacent frames.

Table 3 presents the results obtained for each setup for the CTC Fluo-N2DL-HeLa dataset. We can observe that the *multi-similarity loss* and the miner that was trained with the proposed sampler performs better than the other setups. Furthermore, the proposed modified sampling scheme outperforms the 'traditional' *m-per-class* sampler [9] in a significant. This demonstrates its suitability to our task.

Results. We report our results for all the datasets in Table 4. The results differ due to the significant variability between the datasets. The better scores obtained for the 3D sequences Fluo-N3DH-SIM+ with respect to the 2D Fluo-N2DH-SIM+ sequences demonstrate the importance of the additional dimension for distinguishing between cell instances.

Table 4. Accuracy measures (%) for the DML component. Performances on the evaluated metrics for each dataset under the same setting.

Dataset	AMI	NMI	P@1	RP	MAP@R
Fluo-N3DH-SIM+	73.1	82.4	98.3	67.7	65.1
Fluo-N2DH-SIM+	55.6	73.2	84.8	42.9	38.1
PhC-C2DH-U373	45.9	46.8	81.6	51.6	39.3
Fluo-C2DL-Huh7	86.8	89.1	94.5	65.1	61.2
Fluo-N2DL-HeLa	60.5	74.6	82.8	39.3	34.1
C2C12 [3]	67.5	76.4	97.9	46.8	42.4

References

1. Bensch, R., Ronneberger, O.: Cell segmentation and tracking in phase contrast images using graph cut with asymmetric boundary costs. In: IEEE International Symposium on Biomedical Imaging (ISBI). pp. 1220–1223. IEEE (2015)
2. Bise, R., Yin, Z., Kanade, T.: Reliable cell tracking by global data association. In: IEEE International Symposium on Biomedical Imaging (ISBI). pp. 1004–1010. IEEE (2011)
3. Eom, S., Sanami, S., Bise, R., Pascale, C., Yin, Z., Huh, S., Osuna-Highley, E., Junkers, S.N., Helfrich, C.J., Liang, P.Y., et al.: Phase contrast time-lapse microscopy datasets with automated and manual cell tracking annotations. Scientific Data **5**(1), 1–12 (2018)
4. Fey, M., Lenssen, J.E.: Fast graph representation learning with PyTorch Geometric. In: ICLR Workshop on Representation Learning on Graphs and Manifolds (2019)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)
6. Hoffer, E., Ailon, N.: Deep metric learning using triplet network. In: International workshop on similarity-based pattern recognition. pp. 84–92. Springer (2015)
7. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (ICLR) (2015)
8. Maška, M., Ulman, V., Svoboda, D., Matula, P., Matula, P., Ederra, C., Urbiola, A., España, T., Venkatesan, S., Balak, D.M., Karas, P., Bolcková, T., Štreitová, M., Carthel, C., Coraluppi, S., Harder, N., Rohr, K., Magnusson, K.E.G., Jaldén, J., Blau, H.M., Dzyubachyk, O., Křížek, P., Hagen, G.M., Pastor-Escuredo, D., Jimenez-Carretero, D., Ledesma-Carbayo, M.J., Muñoz-Barrutia, A., Meijering, E., Kozubek, M., Ortiz-de Solorzano, C.: A benchmark for comparison of cell tracking algorithms. Bioinformatics **30**(11), 1609–1617 (2014)
9. Musgrave, K., Belongie, S., Lim, S.N.: A metric learning reality check. In: European Conference on Computer Vision (ECCV). pp. 681–699. Springer (2020)
10. Musgrave, K., Belongie, S., Lim, S.N.: PyTorch metric learning (2020)
11. Padfield, D., Rittscher, J., Roysam, B.: Coupled minimum-cost flow cell tracking for high-throughput quantitative analysis. Medical Image Analysis **15**(4), 650–668 (2011)
12. Scherr, T., Löffler, K., Böhland, M., Mikut, R.: Cell segmentation and tracking using CNN-based distance predictions and a graph-based matching strategy. Plos One **15**(12), e0243219 (2020)
13. Sun, Y., Cheng, C., Zhang, Y., Zhang, C., Zheng, L., Wang, Z., Wei, Y.: Circle loss: A unified perspective of pair similarity optimization. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6398–6407 (2020)
14. Ulman, V., Maška, M., Magnusson, K.E., Ronneberger, O., Haubold, C., Harder, N., Matula, P., Matula, P., Svoboda, D., Radojevic, M., et al.: An objective comparison of cell-tracking algorithms. Nature methods **14**(12), 1141–1152 (2017)
15. Wang, X., Han, X., Huang, W., Dong, D., Scott, M.R.: Multi-similarity loss with general pair weighting for deep metric learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5022–5030 (2019)