CTMC: Cell Tracking with Mitosis Detection Dataset Challenge

Samreen Anjum and Danna Gurari
University of Texas at Austin

Abstract

While significant developments have been made in cell tracking algorithms, current datasets are still limited in size and diversity, especially for data-hungry generalized deep learning models. We introduce a new larger and more diverse cell tracking dataset in terms of number of sequences, length of sequences, and cell lines, accompanied with a public evaluation server and leaderboard to accelerate progress on this new challenging dataset. Our benchmarking of four top performing tracking algorithms highlights new challenges and opportunities to improve the state-of-the-art in cell tracking.

1. Introduction

Studying cell migration and mitosis (cell division) is crucial for better understanding fundamental biological processes such as proper tissue formation and repair [14], wound healing [28], treatment and prevention of cancer and tumoregenesis [11], as well as development and analysis of drug and immune responses [13]. To facilitate this investigation, researchers generate and record live-cell imaging videos showing cell behavior over time. They then use these videos to study two critical tasks: (1) detecting and following individual cells over time, also referred to as *cell tracking* and (2) detecting proliferation of each cell via mitosis events, also referred to as *lineage tracing*. Researchers typically complete this analysis by either manually annotating the videos or reviewing algorithms' results and then correcting mistakes, since existing automated methods are unreliable for general-purpose use.

Towards making progress in creating general-purpose algorithms that consistently work well, numerous large, diverse datasets have been created over the past decade. That is because a general belief is that deep learning algorithms (i.e., neural networks) will perform well if trained on many diverse examples. In parallel with publicly-releasing such datasets, the authors of such work have also created public evaluation servers with leaderboards to encourage an international community to compete on improving algorithms and so accelerate progress.

The aforementioned general approach has emerged across different sub-communities which independently work on Multiple Object Tracking (MOT). The *mainstream* community predominantly focuses on tracking pedestrians or vehicles through everyday scenes [12, 20, 33, 24], including for the following dataset challenges: MOT [5], KITTI [2] and DETRAC [7]. They aim to localize and track all objects using a bounding box around each object. In contrast, a very limited number of datasets focus on *cell tracking* [31, 26]. Moreover, only one such dataset comes paired with an evaluation server and public leaderboard [23]. Progress on this dataset is disconnected from that of the mainstream tracking community because the infrastructure requires a more stringent segmentation around each object (rather than a coarser bounding box).

Our aim is to introduce a new large, diverse cell tracking dataset with public evaluation server and leaderboard, while encouraging collaboration from the mainstream community with the cell tracking community. We introduce a new human-annotated live-cell imaging dataset for cell tracking and lineage tracing, that offers greater diversity than the previous state-of-art cell tracking dataset [23] in terms of number of sequences, total length of sequences, and number of cell lines. It consists of 86 live-cell imaging videos that represent 14 different cell lines (exemplified in Figure 1). For each video, we manually detected and tracked each cell with bounding boxes. We also annotated all mitosis events. Next, we analyze the dataset and compare it with other existing cell tracking datasets. Then, we benchmark modern algorithms from both the cell tracking and mainstream communities on this new dataset, and offer insights on the challenges and opportunities to leverage their respective advantages. Finally, we set-up an evaluation server that supports immediate involvement from those in both the mainstream and cell tracking communities.

More generally, we expect our work will contribute to the design of more generalized MOT algorithms. To facilitate future progress, we are publicly-sharing our dataset with evaluation server and leaderboard.

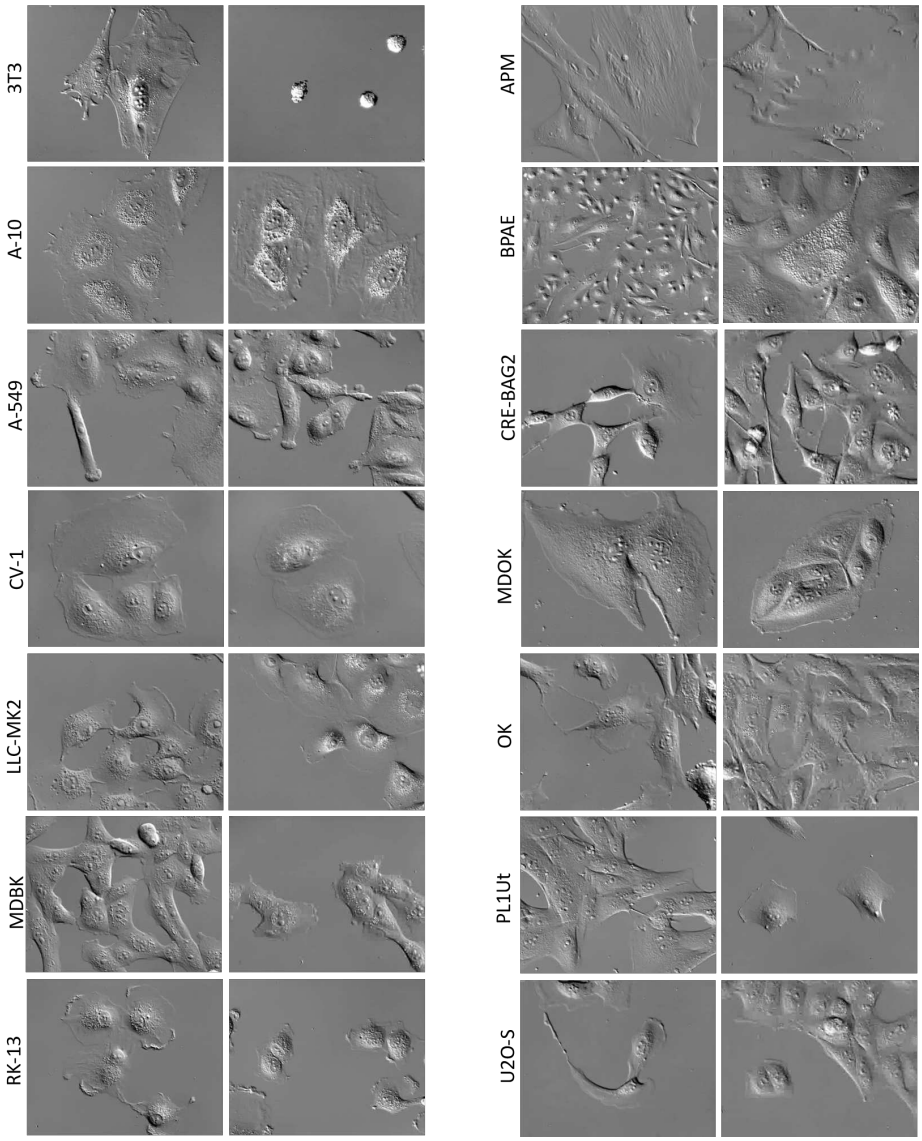

Figure 1. Examples of frames for each of the 14 cell lines in our new dataset that depicts the diversity in cell morphology and frame density.

2. Related Work

Cell Tracking Datasets To enable large scale algorithm training as well as to standardize evaluation, the vision community has established centralized benchmarks for several object tracking tasks [12, 20, 33, 24]. Within the cell tracking domain, the ISBI Cell Tracking Challenge [31] is currently the only standardized benchmark, featuring 52 video sequences, of which 44 are real and the remaining are synthetic. While this benchmark with online evaluation server has played a significant role in enhancing the state-of-art of the cell tracking algorithms, it is limited in size and diversity. Complementing this benchmark, we introduce the largest and most diverse live-cell imaging dataset for cell tracking in terms of number of sequences, total length of sequences and cell lines. Our dataset fills a gap in prior work by focusing on videos collected using differential interference contrast (DIC) microscopy. Prior work only included 4 of 52 videos representing this important video modality [31]. Our work also fills a gap in prior work by standardizing this dataset to be seamlessly accessible to the mainstream community while still meeting the unique needs for cell tracking, by including support and evaluation of lineage tracing (i.e., capturing mitosis events). This is achieved through both the design of the dataset as well as the design of our publicly-hosted evaluation server.

MOT Algorithms Many algorithms have been proposed over the years to support MOT. Most have been introduced for tracking pedestrians and cars (i.e., mainstream algo-

Code	Cell Line	Cell Type	Application
3T3	Albino Swiss Mouse Embryo	Fibroblasts	DNA transfection, transformation studies
A-10	Embryonic Rat Thoracic Aorta Medial Layer	Myoblasts	In vitro wounds
A-549	Male Human Lung Carcinoma	Epithelial	Respiratory ailments, asthma, tissue damage due to asbestos exposure, smoking-related emphysema
APM	African Water Mongoose Skin	Fibroblasts	Wound healing
BPAE	Bovine Pulmonary Artery	Epithelial	Hypertension, atherosclerosis, coronary heart disease
CV-1	Normal African Green Monkey Kidney	Fibroblasts	Cancer research, AIDS, Simian Virus 40 (SV40)
CRE-BAG2	Albino Swiss Mouse Embryo Moloney Murine Leukemia Virus Transfected Cells	Fibroblasts	Virus research
LLC-MK2	Rhesus Monkey Kidney	Epithelial	Production of mumps vaccines, isolation of parainfluenza viruses
MDBK	Madin-Darby Bovine Kidney	Epithelial	Vaccine production
MDOK	Madin-Darby Ovine Kidney	Epithelial	Vesicular stomatitis, infectious bovine rhinotracheitis, sheep bluetongue virus
OK	Opossum Kidney Cortex Proximal Tubule	Epithelial	Chromosome X inactivation research
PL1Ut	Raccoon Uterus	Fibroblasts	Feline and canine viral diseases, study of viruses (herpes simplex virus, reovirus 3, and vesicular stomatitis)
RK-13	Normal Rabbit Kidney	Epithelial	B virus, herpes simplex, pseudorabies virus, vaccinia, rabbitpox, myxoma, simian adenoviruses, rubellavirus
U2O-S	Human Bone Osteosarcoma	Epithelial	Insulin-like growth factor I (IGF-I) and insulin-like growth factor II (IGF II) receptors

Table 1. Description of each cell line included in the CTMC dataset

rithms), both online [9, 29, 34] and offline [30, 25, 36]. Several have been developed to track cells [27, 22, 8, 18, 15, 26, 35], with a few specifically designed to support videos collected using DIC microscopy [17, 10]. We benchmark four modern, popular algorithms for both mainstream and cell tracking aims and conduct a thorough analysis to identify their strengths and weaknesses. We observe that existing cell tracking algorithms generalize poorly to our new dataset, highlighting the difficulty of this dataset. Interestingly, we observe that mainstream algorithms harbor great promise if improved algorithms that can accurately localize cells with bounding boxes. We suggest opportunities to leverage the strengths of both sub-communities to push the limits of algorithms that automatically track cells.

3. CTMC Dataset

We now introduce our dataset, which consists of annotated real live-imaging cell videos for the purposes of cell tracking and lineage tracing. We first describe the videos in our dataset and then explain the annotation procedure.

3.1. Video Collection

We collect the videos from Nikon microscopy's website, in part because they are freely-available. Our resulting dataset contains 86 videos belonging to 14 different cell lines, with a total of 152,584 frames. The shortest and longest videos contain 294 and 4,438 frames, respectively. The average number of frames in a video is 1,774. The videos contain 30 frames per second with a resolution of 320 X 400 pixels. All videos were collected using the Nikon TE2000 DIC imaging modality, with a time interval of 30 seconds before collecting each next frame.

We describe each cell line in Table 1, and show an example from each cell line in Figure 1. Cell lines are taken from a variety of animals including humans, mice, rats, raccoons, and rabbits (Table 1, column 2). Several families of cell types are included: fibroblasts, myoblasts, and epithelial (Table 1, column 3). The particular cell lines are valuable for supporting a wide range of studies including around wound healing, cancer research, and vaccine development (Table 1, column 4).

3.2. Annotation Procedure

Annotation Tool: We designed our tool to capture two key types of information related to live-cell imaging datasets. First, it supports tracking, by representing each cell with a bounding box and a unique id across all relevant frames of the video. Second, it also supports detecting

mitosis events, by capturing the frame at which the cell division occurs paired with the ids of the parent and resulting children cells from the mitosis event.

Since manually *tracking objects* across all frames is a tedious procedure, we developed our in-house video annotation tool to reduce the workload using interpolation (following the design of VATIC [32]). To annotate a cell, the user draws a bounding box around the cell at its first occurrence in the video. Then, the user can readjust the bounding box to encapsulate the cell tightly at future frames of the user's choice. The bounding box representing the cell in the intermediary frames are interpolated using linear interpolation to reduce human workload. This process can be continued till the last frame of the cell for all cells in the video.

To annotate a *mitosis event*, the tool provides a specific flag 'split' which the user can mark when a cell division occurs. This flag records the frame number along with the cell's unique id and visually divides the current bounding box into two separate bounding boxes. These two new boxes can then be used to represent the new children cells. The annotator can then continue to annotate children cells' tracks following the same aforementioned procedure for tracking objects.

Annotation Collection Process: Two in-house experts annotated all the cells in the 86 videos using the annotation tool. They were asked to draw a bounding box around each cell such that they cover all the pixels of the cell while keeping the boundaries of the box as tight as possible. This results in overlapping cells having overlapping bounding boxes. For the cells that lie partially inside the frame, bounding boxes represent only the visible region. The intermediary frames were interpolated by the tool and these interpolated annotations were verified or corrected by the annotators. Figure 2 exemplifies the resulting annotations.

Both annotators recorded the total time they took to annotate the videos as well as their perceived annotation difficulty level of the video. Cumulatively, they spent over 164 hours to annotate all videos. Details regarding both difficulty levels and time taken are reported in Table 2, columns 9-12. We use one set annotator's results as ground truth and use the second set to evaluate human performance (discussed in the algorithm benchmarking section).

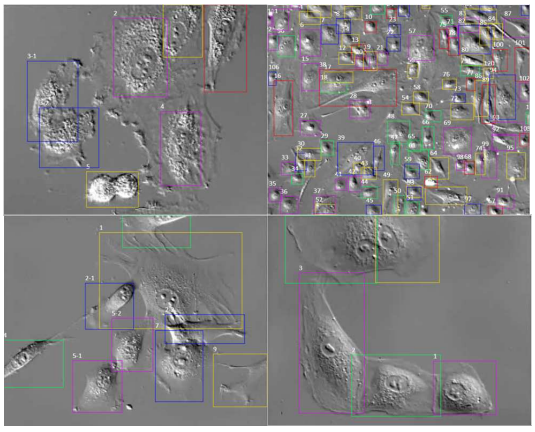

Figure 2. Examples of the human annotations for video frames in the CTMC dataset.

3.3. Dataset Analysis

We analysed the following characteristics per each cell line (i.e., all videos containing that cell line) in the dataset: number of cells (*tracks*), length of sequences, number of mitosis events, total number of detections (*boxes*) and the average density of a frame. Results are shown in Table 2, columns 2-8.

Number of sequences: In total, there are 86 sequences in this dataset, with an average of 6 per cell line. The range of sequences per cell line varies between 4 and 10.

Number of cells (tracks): The dataset consists of a total of 2,900 objects (including children cells). Two of the 86 sequences contain only one cell. In sequences with more than one cell, the smallest number of cells in a video is 2, while the largest number of cells in a video is 185.

Length of sequences: In total, the dataset consists of 152,584 frames which represents 5,084 seconds of videos. The shortest video is 10 seconds long (in the CV-1 cell line) and the longest one is 148 seconds long (in the U2O-S cell line). Given that the frames were collected with 30-second time intervals, the dataset represents 152,520 seconds in real time, with the shortest and longest videos corresponding to 300 and 4,440 seconds respectively in real time.

Mitosis: In total, there are 457 mitosis events with the lowest being 2 (in the CV-1 cell line) and highest being 115 (in the MDBK cell line).

Detections (boxes): The dataset offers a total of 2,045,834 bounding boxes. The range varies from 1,196 to 172,074 across the 86 sequences.

Density: The average density is approximately 13 cells in each frame. Across the 14 cell lines, CV-1 has the lowest density with 3.42 while BPAE has the highest at 27.88.

3.4. Dataset Comparison

We report the differences between our dataset and three modern cell tracking datasets in Table 3. One of the related datasets is the state-of-art cell tracking challenge (CTC) dataset [31].[1] The second related dataset is provided by the authors of DeepCell [26]. The third dataset consists of phase contrast microscopy images (C2C12) [19].

Overall, our CTMC dataset offers a greater number of sequences and greater diversity in terms of cell lines, while containing fewer tracks than CTC, DeepCell, and C2C12. In absolute terms, our CTMC dataset is the largest with 86

[1] CTC consists of 52 live-imaging sequences of which 44 are real and 8 are synthetic. Only 28 out of 52 are 2D sequences while the rest are in 3D.

Code	Seq	Tracks	Frames	Len(s)	Mit.	Boxes	Density	An1(T)	An1(D)	An2(T)	An2(D)
3T3	9	255	17250	574	33	143146	8.03	750	2.1	339	1.9
A-10	7	85	13154	438	17	84274	7.10	150	1.14	103	1.3
A-549	3	106	5760	192	26	102197	17.19	210	1.7	84	2
APM	6	157	11834	395	8	94056	7.79	360	1.8	193	2.3
BPAE	7	344	13491	450	42	343955	27.88	705	1.43	340	1.43
CRE	4	369	7957	265	56	168398	19.62	1080	2.75	407	2.5
CV-1	4	20	4802	160	2	18796	3.42	60	1	12	1
LLC	8	218	13461	448	33	137828	10.28	510	1.4	214	1.4
MDBK	10	533	15437	513	115	323032	21.36	855	1.7	651	1.8
MDOK	9	89	16559	552	10	95738	6.04	210	1	78	1
OK	7	335	10449	348	49	177597	16.66	663	1.7	560	1.3
PL1Ut	5	154	8466	283	21	100738	11.91	294	1.4	218	1.2
RK-13	3	35	3453	115	2	28544	8.78	66	1	26	1
U2O-S	4	200	10511	351	43	227535	18.72	495	2	224	2.5
Overall	86	2900	152584	5084	457	2045834	13.20*	6408	1.58*	3449	1.62*

Table 2. Analysis of each cell line in the dataset. (Len(s) = total length of the sequences in seconds; An = Annotator; T = time taken to complete annotation in minutes; D = average perceived annotation difficulty level by two human annotators where 1=easy, 2=medium, 3=hard; CRE = CRE-BAG2; LLC = LLC-MK2; *average value)

Dataset	Seq	Tracks	CL	Mit	Frames
CTC [31]	52	11,318	10	NP	5,927
DeepCell [26]	81	11,393	4	855	2,610
C2C12 [19]	48	NP	1	NP	49,919
CTMC (Ours)	86	2,900	14	457	152,584

Table 3. Comparison with other datasets. CL = Cell Line; Mit = Mitosis; Seq = Sequences; NP = Not Provided.

sequences and 152,584 frames. Our CTMC dataset contains a greater number of cell lines, with 14 versus 10 cell lines in CTC, 4 in DeepCell and 1 in C212. Our CTMC dataset adds complementary diversity, as it shares only one cell line with DeepCell and none of our 14 cell lines with the CTC or C2C12 datasets.

Importantly, our CTMC dataset provides a complementary set of videos because they were collected using differential interference contrast (DIC) microscopy. CTC contains only four sequences of one cell line obtained using DIC imaging while both DeepCell and C2C12 lack any DIC sequences.

4. Algorithm Benchmarking on CTMC

We now describe our analysis of the tracking performance of state-of-art (1) cell tracking algorithms and (2) mainstream algorithms. We conduct these analyses separately since these methods come from distinct subcommunities that leverage distinct evaluation metrics and algorithm frameworks. For our analysis, we split the CTMC dataset of 86 videos into 47 training and 39 testing videos. In order to represent cells from each cell line in both training and testing phases, we distributed the videos between the two splits with odd numbered sequences in the training split and even numbered sequences in the testing split. All reported results are based on the test split.

4.1. Cell Tracking Algorithms

Methods: We benchmarked two popular state-of-the-art cell tracking algorithms: Viterbi [22] and DeepCell [26].

Viterbi [22], offered as part of the Baxter Algorithms package [1], is the best performing tracker according to the latest Cell Tracking Challenge, achieving a high accuracy on 20 of the 24 available test video sequences. It is an offline tracker which processes multiple frames to determine object associations.

DeepCell's [26] tracking algorithm is a recently published work that combines a popular deep multiple object tracker [29] with a traditional cell event detector [16] to identify important cell life cycle events such as mitosis. For our experiment, we use the pretrained model with default parameters as provided in the package. The model was pretrained on 11,393 cell tracks and 855 mitosis events from 4 different cell lines.

Input to Trackers: Both trackers take as input segmentation masks (rather than bounding boxes). In an attempt to provide a fair comparison between them, we use the same segmented data for both trackers. We employ the top-performing segmentation algorithm according to the latest Cell Tracking Challenge (MU-Lux-CZ) [21], which is designed to process cells obtained using the DIC imaging modality.[2] We use the software package as provided in the challenge website[4] along with the pretrained model

[2]We also tested with the segmentation algorithm provided in the Baxter package, but found it is not suitable for our dataset.

on the DIC-HeLa dataset. We then use this segmented data with both Viterbi and DeepCell, using default parameters for both trackers.

Evaluation Metrics: To compare the performance, we use the standard tracking metric, TRA, as described by Cell Tracking Challenge [31]. This metric represents the predicted and the ground truth tracks as graphs and computes the number of steps required to match both graphs. It penalizes the following errors: false negatives, false positives, under-detections, missing edges, miss-labeled edges, and false positive edges.

$$TRA = 1 - \frac{min(AOGM, AOGM_0)}{AOGM_0}$$

where AOGM is the Acyclic Oriented Graph Matching measure (weighted sum of the graph operations) [23]. We modified the overlap measure computed in the metric to support that our ground truth was collected in the form of bounding box detections rather than segmentation masks. Instead of computing pixel overlap of the predicted mask with the ground truth mask, we extract the bounding box of each segmented mask from the predicted results and compute its overlap with the ground truth bounding box. Resulting values can range between 0 and 1, with better performance indicated by values lying closer to 1.

Results: Results are shown in Table 4.

Overall, the majority of the sequences were difficult for both trackers. The average TRA scores obtained using Viterbi and DeepCell were 0.39 and 0.32, respectively. For reference, we also computed the inter-annotator agreement using the TRA metric, and found that human performance yields a score of 0.95. Part of the poor performance from the DeepCell algorithm is that it failed to produce results for 20 out of 39 sequences because the algorithm failed to generate meaningful features using the pre-trained model. Altogether, our findings reveal that our new dataset is challenging for existing algorithms.

To better understand what makes this dataset difficult for existing trackers, we next compared performance across cell lines. Both trackers performed the best on all sequences of A-10 and CV-1 cell lines, with a mean score above 0.80. These cell lines have low density frames with a fewer number of total cells (14 in A-10 and 4 in CV-1), and therefore are relatively easy cell lines (exemplified in Figure 3). All the sequences in cell lines such as A-549, APM, CRE-BAG, OK, U2O-S proved to be challenging for both trackers. For these sequences, the average score of Viterbi was below 0.2 while DeepCell failed to produce results. These cell lines have varying shapes and consist of high density frames (exemplified in Figure 4a). In addition, certain sequences of the 3T3, MDBK, MDOK and PL1UT were also challenging for the same reasons.

	Viterbi [22]	**DeepCell [26]**
3T3 (124)	**0.38**	0.15
A-10 (14)	**0.83**	0.82
A-549 (82)	**0.12**	0.00
APM (90)	**0.33**	0.00
BPAE (114)	0.30	**0.46**
CRE-BAG2 (229)	**0.07**	0.00
CV-1 (4)	**0.89**	0.86
LLC-MK2 (56)	**0.59**	0.44
MDBK (205)	0.38	**0.48**
MBOK (27)	**0.51**	0.36
OK (111)	**0.25**	0.00
PL1Ut (71)	0.21	**0.26**
RK-13 (20)	0.53	**0.60**
U2O-S (137)	**0.04**	0.00
Average	**0.39**	0.32

Table 4. Performance of cell tracking algorithms with respect to the TRA metric on each cell line and overall. Numbers in parenthesis represent the total number of cells in the cell line. Overall, Viterbi performs better than DeepCell. A-10 and CV-1 have the lowest number of cells and density are the highest scoring cell lines, while A-549, CRE-BAG2, OK, PL1Ut and U2O-S are the most difficult ones.

While both DeepCell and Viterbi showed similar patterns in most sequences, we also observed a few cases where they displayed contrasting performances. In one such case, Viterbi performed considerably well on two sequences with larger cells (as shown in Figure 4b) while DeepCell failed to generate any features at all suggesting that its pre-trained model did not generalize to cells with larger sizes. We also observed a few sequences like that displayed in Figure 4c where DeepCell tracked more detected cells than Viterbi.

4.2. Mainstream Tracking Algorithms

Methods: We benchmarked two state-of-art mainstream algorithms: Tracktor [9] and DeepSORT [34]. Both were designed for tracking pedestrians.

Tracktor [9] is the top-ranked, publicly-available algorithm on the 2019 MOT Challenge [3]. [3]. It tracks each object by modifying the regression component in the object detector to predict the object's location in the next frame. We train Tracktor's object detector as well as the siamese reidentification network on our training split. For the object detector, we follow the recommendation to use Faster-RCNN with a FPN backbone, relying on the default parameters. This detector is two-class based, where the cells are labeled '1' and the background is labelled '0'. [4]

[3]This algorithm is ranked second, since the first ranked method is not yet published.

[4]Of note, the object detector requires all frames in the sequence contain at least one bounding box. Since our dataset contains several intermediary

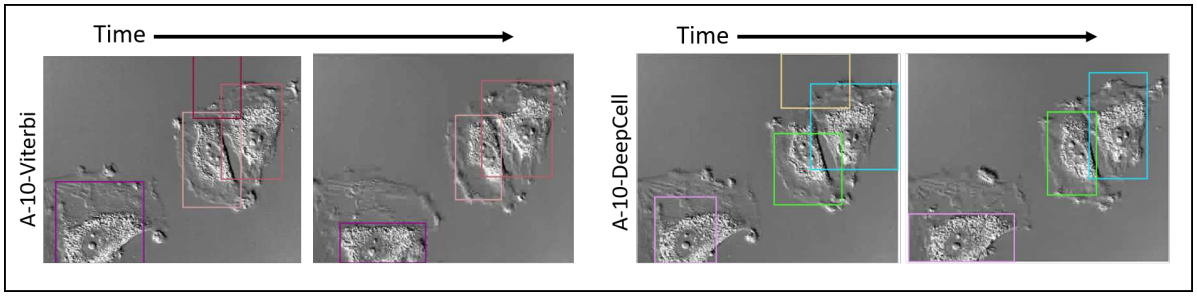

Figure 3. Example showing two frames of a video (A-10) where both Viterbi (left) and DeepCell (right) perform well (TRA score > 0.80).

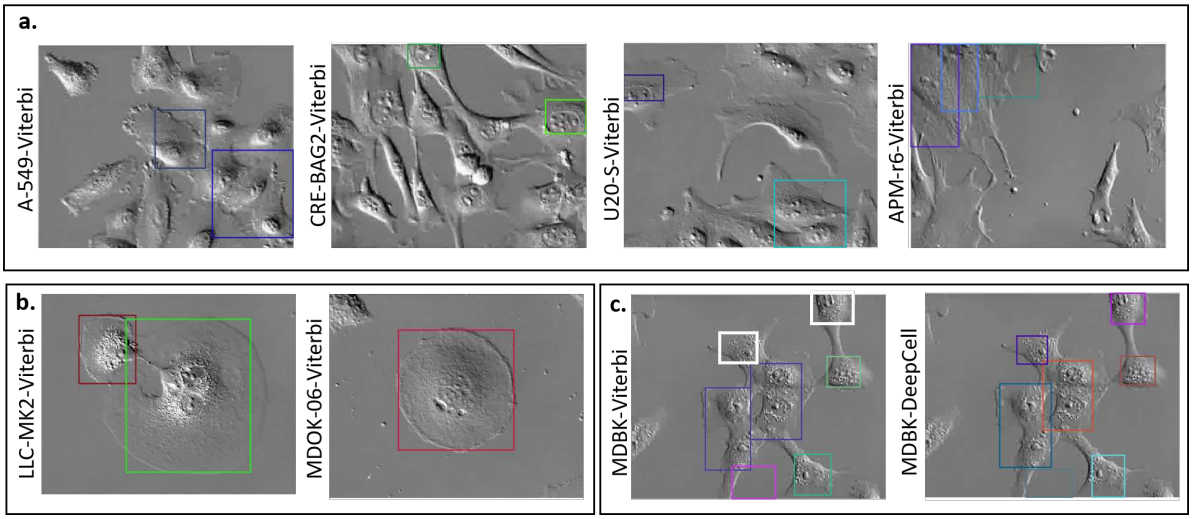

Figure 4. Examples of videos that were challenging to the cell trackers. The first row shows examples where both Viterbi and DeepCell had low performance while the bottom row shows sequences displaying opposing performance. a. Single frames from sequences where the segmentation algorithm fails to detect most cells, hence affecting the tracking performance. b. Single frames from easy videos where DeepCell fails to track, while Viterbi performs very well (Observation: large cells, low density). c. Single frames from a sequence in the MDBK cell line where DeepCell performs better than Viterbi (white boxes represent cells that were segmented but not tracked by Viterbi).

DeepSORT [34] is a popular, highly-cited Kalman Filter based tracker. It relies on good quality detections to perform reidentification of objects. The object is linked across frames using the Hungarian algorithm with a deep association metric. Again, we use default parameters in our experiment and the original feature generation model which was pre-trained on the MARS [37] dataset.

Input to Trackers: We experiment with two types of input detections. First, we use the object detector, Faster-RCNN, trained for Tracktor to generate detections on our sequences following Tracktor's parameters used to achieve the reported best performance. Second, to assess the potential of the trackers in the presence of perfect detectors, we use our ground truth detection bounding boxes.

Evaluation Metrics: We evaluate using two popular metrics in the mainstream tracking community: Multiple Object Tracking Accuracy (MOTA) and ID F1 Score (IDF1) [24]. While MOTA represents object coverage, IDF1 quantifies the object's identity across frames in the sequence. We use the standard evaluation package provided by *py-motmetrics* [6] for this purpose.

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t))}{\sum_t GT_t}$$

where, FP is the total number of bboxes not covering any GT bbox, FN is the total number of GT bboxes not covered by any bbox and IDSW is the Identity Switch, i.e., a bounding box covering a GT bounding box from a different track than in the previous frame. MOTA scores can range between negative values and 100, with better performance indicated by values lying closer to a 100.

$$IDF1 = \frac{2IDTP}{2IDTP + IDFP + IDFN}$$

where IDFN is False Negative ID, IDTP is True Positive ID

frames with no cells and thereby no bounding boxes, we annotate each of these frames with a fake 4X4 bounding box of the background pixels and label those boxes as '0'.

Detection:	**FasterRCNN**				**Ground Truth**			
Tracker:	**Tracktor[9]**		**DeepSORT[34]**		**Tracktor[9]**		**DeepSORT[34]**	
Evaluation Metric:	**IDF1**	**MOTA**	**IDF1**	**MOTA**	**IDF1**	**MOTA**	**IDF1**	**MOTA**
3T3 (124)	**57.58**	**34.33**	34.98	-33.40	67.25	65.08	**96.05**	**99.33**
A-10 (14)	**47.13**	**-18.37**	22.60	-151.33	65.97	52.73	**92.63**	**99.87**
A-549 (82)	**50.90**	**35.50**	37.20	2.70	56.90	54.50	**84.60**	**99.85**
APM (90)	**27.10**	**-31.00**	15.00	-132.80	39.27	30.17	**97.43**	**99.57**
BPAE (114)	**49.37**	**34.47**	38.60	0.87	58.87	53.67	**95.60**	**99.73**
CRE-BAG2 (229)	**48.95**	**25.75**	26.60	-35.80	55.75	50.95	**93.75**	**99.30**
CV-1 (4)	**51.65**	**-14.60**	35.25	-135.90	72.60	49.10	**99.80**	**99.55**
LLC-MK2 (56)	**73.77**	**57.00**	60.23	28.67	77.57	70.60	**96.37**	**99.67**
MDBK (205)	**69.86**	**65.04**	54.34	40.00	72.50	70.86	**90.46**	**99.60**
MBOK (27)	**32.55**	**-38.88**	23.28	-151.05	44.18	34.63	**97.83**	**99.83**
OK (111)	**43.97**	**33.20**	27.83	-13.93	50.83	53.10	**94.30**	**99.53**
PL1Ut (71)	**36.90**	**25.15**	25.20	-14.15	41.55	38.60	**96.05**	**99.55**
RK-13 (20)	**73.80**	**64.20**	64.50	36.70	74.20	72.80	**99.10**	**99.70**
U2O-S (137)	**61.40**	**58.50**	53.10	45.65	62.00	65.25	**85.20**	**95.35**
Average	**51.78**	**23.59**	37.05	-36.70	59.96	54.43	**94.23**	**99.32**

Table 5. Performance of MOT algorithms averaged over sequences within each cell line using FasterRCNN and Ground Truth detections. Tracktor performs better with FasterRCNN detections, while DeepSORT works better on Ground Truth detections. Overall, A-10, APM, CRE-BAG2, CV-1, MDOK, PL1Ut are the most challenging cell lines while LLC-MK2, MDBK, RK-13, and U2O-S are tracked better using MOT trackers. Numbers in parenthesis represent total number of unique cells in the cell line.

and IDFP is False Positive ID. Resulting values can range between 0 and 100%, with better performance indicated by values lying closer to 100%.

Results: Results are shown in Table 5.

Overall performance with detections from FasterRCNN: Overall, we observe the dataset is difficult for both algorithms. For instance, both trackers perform considerably worse than human performance. Specifically, using the annotations collected by the two annotators in Section 3, we found that inter-annotator agreement yields a MOTA score of 41.70 and IDF1 score of 73.02. Still, we observe Tracktor outperforms DeepSORT with an average MOTA score of 34.9 over -37 and an average IDF1 score of 51.78 over 37.05. From visually inspecting the results obtained by DeepSORT, we found that its negative MOTA score may be attributed to the higher number of false positives due to the identification of debris as a different track over time.

When comparing the performance across cell lines, we observed that Tracktor performed its best on LLC-MK2, RK-13 and U2O-S, with a MOTA score of above 0.50. This was interesting as U2O-S proved to be the most difficult cell line for the cell trackers in the previous experiment. In contrast, it performed below average on all sequences in A-10, APM, CV-1, and MDOK cell lines. Here again, we note A-10 and CV-1 cell lines were the best scoring sequences with the cell trackers. While the average performance of DeepSORT was poor, like Tracktor, it also performed above average on MDBK, RK-13 and U2O-S cell lines, in addition to two sequences in LLC-MK2 cell line. These findings are interesting in part because they highlight the complementary nature of algorithms emerging from the distinct cell tracking and mainstream communities.

Overall performance with detections from GT: As expected, we observing considerable improvements from both algorithms when using perfect detections as input. However, we were surprised to observe that DeepSORT has the potential to perform exceptionally well; i.e., an average MOTA score of 99.32 and average IDF1 score of 94.23. This is surprising in part because this model was pretrained on a person tracking dataset, yet generalizes well for cell tracking. Overall, we observe similar trends of the performance of the trackers across cell lines when using GT as input as we did when using FasterRCNN as input instead.

5. Conclusion

We introduce a cell tracking dataset and analyse modern algorithms to reveal challenges and opportunities for improving cell tracking algorithms. We publicly share this dataset with an evaluation server to accelerate progress from a larger community on this important problem.

Acknowledgements. This project has been made possible in part by grant number 2018-182764 from the Chan Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation. We thank Tai-Yin Chiu, Klas Magnusson and Tim Meinhardt for their valuable discussions about and suggestions for this research.

References

[1] Baxter algorithms. https://github.com/klasma/BaxterAlgorithms. 5

[2] Kitti vision benchmark suite. http://www.cvlibs.net/datasets/kitti. 1

[3] Mot challenge cvpr 2019. https://motchallenge.net/results/CVPR_2019_Tracking_Challenge/. 6

[4] Mu-lux-cz package. http://public.celltrackingchallenge.net/participants/MU-Lux-CZ.zip. 5

[5] Multiple object tracking benchmark. https://motchallenge.net. 1

[6] py-motmetrics library. https://github.com/cheind/py-motmetrics. 7

[7] Ua-detrac benchmark suite. http://detrac-db.rit.albany.edu. 1

[8] Saad Ullah Akram, Juho Kannala, Lauri Eklund, and Janne Heikkilä. Cell tracking via proposal generation and selection. *arXiv preprint arXiv:1705.03386*, 2017. 3

[9] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 941–951, 2019. 3, 6, 8

[10] Ryoma Bise, Kang Li, Sungeun Eom, and Takeo Kanade. Reliably tracking partially overlapping neural stem cells in dic microscopy image sequences. In *MICCAI Workshop on OPTIMHisE*, volume 5, pages 67–77, 2009. 3

[11] John Condeelis and Jeffrey W Pollard. Macrophages: obligate partners for tumor cell migration, invasion, and metastasis. *Cell*, 124(2):263–266, 2006. 1

[12] Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 304–311. IEEE, 2009. 1, 2

[13] Rachel Evans, Irene Patzak, Lena Svensson, Katia De Filippo, Kristian Jones, Alison McDowall, and Nancy Hogg. Integrins in immunity. *Journal of cell science*, 122(2):215–225, 2009. 1

[14] Clemens M Franz, Gareth E Jones, and Anne J Ridley. Cell migration in development and disease. *Developmental cell*, 2(2):153–158, 2002. 1

[15] Junya Hayashida and Ryoma Bise. Cell tracking with deep learning for cell detection and motion estimation in low-frame-rate. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 397–405. Springer, 2019. 3

[16] Khuloud Jaqaman, Dinah Loerke, Marcel Mettlen, Hirotaka Kuwata, Sergio Grinstein, Sandra L Schmid, and Gaudenz Danuser. Robust single-particle tracking in live-cell time-lapse sequences. *Nature methods*, 5(8):695, 2008. 5

[17] Richard M Jiang, Danny Crookes, Nie Luo, and Michael W Davidson. Live-cell tracking using sift features in dic microscopic videos. *IEEE Transactions on Biomedical Engineering*, 57(9):2219–2228, 2010. 3

[18] Takeo Kanade, Zhaozheng Yin, Ryoma Bise, Seungil Huh, Sungeun Eom, Michael F Sandbothe, and Mei Chen. Cell image analysis: Algorithms, system and applications. In *2011 IEEE Workshop on Applications of Computer Vision (WACV)*, pages 374–381. IEEE, 2011. 3

[19] Dai Fei Elmer Ker, Sungeun Eom, Sho Sanami, Ryoma Bise, Corinne Pascale, Zhaozheng Yin, Seung il Huh, Elvira Osuna-Highley, Silvina Junkers, Casey Helfrich, Peter Liang, Jiyan Pan, Soojin Jeong, Steven S. Kang, Jinyu Liu, Ritchie Nicholson, Michael F. Sandbothe, Phu T. Van, Anan Liu, Mei Chen, Takeo Kanade, Lee E. Weiss, and Phil G. Campbell. Phase contrast time-lapse microscopy datasets with automated and manual cell tracking annotations. *Scientific Data*, 5, 2018. 4, 5

[20] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, Roman Pflugfelder, Luka Cehovin Zajc, Tomas Vojir, Gustav Hager, Alan Lukezic, Abdelrahman Eldesokey, et al. The visual object tracking vot2017 challenge results. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1949–1972, 2017. 1, 2

[21] Filip Lux and Petr Matula. Dic image segmentation of dense cell populations by combining deep learning and watershed. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 236–239. IEEE, 2019. 5

[22] Klas EG Magnusson, Joakim Jaldén, Penney M Gilbert, and Helen M Blau. Global linking of cell tracks using the viterbi algorithm. *IEEE transactions on medical imaging*, 34(4):911–929, 2014. 3, 5, 6

[23] Martin Maška, Vladimír Ulman, David Svoboda, Pavel Matula, Petr Matula, Cristina Ederra, Ainhoa Urbiola, Tomás España, Subramanian Venkatesan, Deepak MW Balak, et al. A benchmark for comparison of cell tracking algorithms. *Bioinformatics*, 30(11):1609–1617, 2014. 1, 6

[24] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 1, 2, 7

[25] Anton Milan, Laura Leal-Taixé, Konrad Schindler, and Ian Reid. Joint tracking and segmentation of multiple targets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5397–5406, 2015. 3

[26] Erick Moen, Enrico Borba, Geneva Miller, Morgan Schwartz, Dylan Bannon, Nora Koe, Isabella Camplisson, Daniel Kyme, Cole Pavelchek, Tyler Price, et al. Accurate cell tracking and lineage construction in live-cell imaging experiments with deep learning. *bioRxiv*, page 803205, 2019. 1, 3, 4, 5, 6

[27] Christian Payer, Darko Štern, Marlies Feiner, Horst Bischof, and Martin Urschler. Segmenting and tracking cell instances with cosine embeddings and recurrent hourglass networks. *Medical image analysis*, 57:106–119, 2019. 3

[28] Luis G Rodriguez, Xiaoyang Wu, and Jun-Lin Guan. Wound-healing assay. In *Cell Migration*, pages 23–29. Springer, 2005. 1

[29] Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 300–311, 2017. 3, 5

[30] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, and Bernt Schiele. Multiple people tracking by lifted multicut and person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3539–3548, 2017. 3

[31] Vladimír Ulman, Martin Maška, Klas EG Magnusson, Olaf Ronneberger, Carsten Haubold, Nathalie Harder, Pavel Matula, Petr Matula, David Svoboda, Miroslav Radojevic, et al. An objective comparison of cell-tracking algorithms. *Nature methods*, 14(12):1141, 2017. 1, 2, 4, 5, 6

[32] Carl Vondrick, Donald Patterson, and Deva Ramanan. Efficiently scaling up crowdsourced video annotation. *International journal of computer vision*, 101(1):184–204, 2013. 4

[33] Longyin Wen, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. Ua-detrac: A new benchmark and protocol for multi-object detection and tracking. *arXiv preprint arXiv:1511.04136*, 2015. 1, 2

[34] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017. 3, 6, 7, 8

[35] Zheng Wu, Danna Gurari, Joyce Y Wong, and Margrit Betke. Hierarchical partial matching and segmentation of interacting cells. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 389–396. Springer, 2012. 3

[36] Zheng Wu, Thomas H Kunz, and Margrit Betke. Efficient track linking methods for track graphs using network-flow and set-cover techniques. In *CVPR 2011*, pages 1185–1192. IEEE, 2011. 3

[37] Liang Zheng, Zhi Bie, Yifan Sun, Jingdong Wang, Chi Su, Shengjin Wang, and Qi Tian. Mars: A video benchmark for large-scale person re-identification. In *European Conference on Computer Vision*, pages 868–884. Springer, 2016. 7