Consistent Cell Tracking in Multi-frames with Spatio-Temporal Context by Object-Level Warping Loss

Junya Hayashida Kazuya Nishimura Ryoma Bise
Kyushu University Fukuoka, Japan
bise@ait.kyushu-u.ac.jp

Abstract

Multi-object tracking is essential in biomedical image analysis. Most methods follow a tracking-by-detection approach that involves using object detectors and learning the appearance feature models of the detected regions for association. Although these methods can learn the appearance similarity features to identify the same objects among frames, they have difficulties identifying the same cells because cells have a similar appearance and their shapes change as they migrate. In addition, cells often partially overlap for several frames. In this case, even an expert biologist would require knowledge of the spatial-temporal context in order to identify individual cells. To tackle such difficult situations, we propose a cell-tracking method that can effectively use the spatial-temporal context in multiple frames by using long-term motion estimation and an object-level warping loss. We conducted experiments showing that the proposed method outperformed state-of-the-art methods under various conditions on real biological images.

1. Introduction

Multi-object cell tracking is essential in biomedical image analysis for time-lapse images, where hundreds of cells in a population are individually tracked over thousands of frames. Automatic cell tracking enables us to obtain cell-behavior metrics including cell-migration speed and cell-lineage information.

There are three issues with cell tracking (Fig. 1). First, cells have a similar appearance and their shapes change as they migrate. This makes it difficult to identify the same cells at different times on the basis of shape similarity. Second, cells often touch and have blurry intercellular boundaries (call a cell cluster), as shown in the 3rd and 4th frames in Fig. 1. In such cases, even experts often fail to identify individual cells from one image. Third, a cell may divide into two cells (cell mitosis); conventional general-object tracking methods have difficulty tracking such cells. To distinguish mitosis and separation from a cluster, it is required to observe before and after the mitosis event.

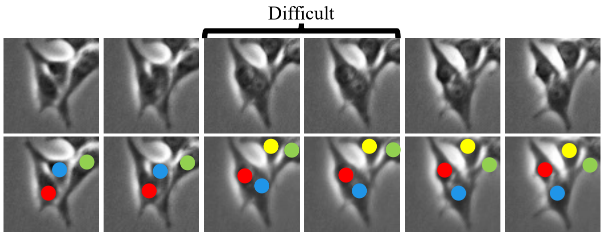

Figure 1. Example of difficult case. Top: original images, Bottom: ground-truth of trajectories. In the 3rd frame, the blue and red cells severely touch and form a cluster in several frames. It is difficult to identify individual cells from only the 3rd and 4th frames. If we observe the entire frame, we can identify the individual cells.

Tracking-by-detection is one of the most common approaches to multi-object tracking. A recent trend is to incorporate existing (bounding box) detectors into trackers and simultaneously train a detector, which detects bounding boxes, and an appearance model, which measures the appearance similarity of detected bounding boxes [15, 18, 46, 4, 11, 39]. LSTM [22] has often been used to learn for measuring the similarity among the time series of detected bounding boxes [35, 12, 10, 14, 26]. However, these approaches do not use the spatial context outside from the bounding boxes (*e.g.*, the positional relationship of nearby objects), which is important information for cell tracking since they have similar appearances and their shapes change during the migration. To effectively use the spatial context, point-based methods for detection and motion estimation recently have attracted attention [21, 55, 59]; these methods directly extract the common image features from the entire image by using a single network to estimate the position and motion map. However, they have two drawbacks; 1) they independently perform detection and association. Objects are first detected, and the detected objects between successive frames are then associated using one-by-one matching. This means the detection errors directly propagate to the association steps. 2) they use only the local temporal context in two frames; *i.e.*, they cannot extract the long-term spatial-temporal context from multiple frames. Under high-density conditions, cells often touch and have blurry intercellular boundaries (forming a cluster region) in several frames. For example in Fig. 1, the red and blue cells move and form a cluster region at the 3rd and 4th frames; we can not identify individual cells from only one image. Such conditions make it difficult to detect cells and estimate cell motion by using only two frames, and due to the assignment constraint (one-to-one matching), the current methods often mistakenly associate a cluster consisting two cells as one cell. If we can observe the cells in multiple frames, we can observe the individual cells moving and forming the cluster , and then separating. This indicates that the long-term spatial-temporal context is important for multi-object tracking.

There are two challenges facing the extraction of tracking information (position and motion) from multiple frames. First, it is required to extract the information from distant frames. For example in Fig. 1, to distinguish mitosis from separation of a cell from a cluster, we need the information when cells forming a cluster region. Second, the consistency of the motions and positions must be preserved among multiple frames. The estimated maps may have inconsistencies, *e.g.,* if a cell is detected at $t-1$ and its motion is toward a specific position, no cell is detected at that position at t. Such inconsistencies often affect tracking performance. In simultaneous estimation in multiple frames, the consistency of the motions and positions among the frames is important for a stable estimation. Indeed, in our preliminary study, when we simply inputted multiple frames to a 3D CNN and tries to simultaneously estimate the positions and motions in every successive frame, it did not improve the performance.

In this paper, we propose a cell-tracking method that can effectively use the long-term spatial-temporal context in multiple frames. To use the spatial-temporal context effectively, a 3D CNN simultaneously estimates multi-frame motion and position maps all at once, given multiple frames as inputs, as shown in Fig. 2(a). In this multi-frame estimation, we estimate the motion in a long interval from t to T (long-term motion) in addition to the motion between successive frames (short-term motion), which is directly used for tracking. The long-term motion information facilitates not only training the network to use the spatial-temporal context but also interpolating for false negatives.

In this estimation, it is a key of our research to preserve consistency among estimation results. To preserve consistency, we introduce a warping loss that penalizes any inconsistency between the estimated positions and motions in multiple frames, which enables the model to directly learn the tracking operation. In addition, instead of one-by-one matching, we introduce tracking-by-object-level warping that transforms the detected region of each cell at $t-1$ into the corresponding region at t by using the estimated motion, in which the object-level motion information is directly trained using the warping loss. We conducted experiments to evaluate our method. The results indicate that it outperformed the state-of-the-art methods under various conditions on real biological images.

Our main contributions are summarized as follows:

- We propose a cell tracking method for simultaneously estimating the trajectories of multiple objects in multiple frames effectively using the spatial-temporal context in multiple frames. We perform long-term motion estimation from t to T to facilitates not only training the network to use the spatial-temporal context but also interpolating for false negatives.
- We propose an object-level warping loss that penalizes any inconsistency between the estimated position and motion in multiple frames. The individual trajectories in multiple frames can be directly obtained using this warping operation. In contrast to tracking-by-detection, our tracking-by-warping method can separate a cluster region into multiple cells even if a cluster consisting of multiple cells was detected as one object.
- Experiments on real biological images demonstrated the effectiveness of the proposed method under various conditions. Our method outperformed the state-of-the-art methods under all conditions.

2. Related work

Cell tracking: Many cell tracking methods have been proposed; they use particle filters [36, 47], active contour [31, 51, 56, 61], and detection-and-association [57, 41, 1, 33, 2]. The most common approach is tracking-by-detection that first detects all cells in individual frames [57, 41, 1, 33, 2] and then performs tracking by solving a data-association problem for finding the set of object pairs from many association hypotheses. The major difference from general object tracking methods is the handling of cell mitosis events. Many cell tracking methods address this problem by using linear programming [23, 5, 6, 61, 53] and graph-based optimization for global data association [7, 45, 48, 17]. In such methods, the detection is separate from the tracking; that is, they do not use the spatial-temporal context. Several methods have been proposed to use the spatial-temporal context. Payer *et al.* [37] proposed ConvGRU, which not only extracts local features but also memorizes inter-frame information by embedding cell IDs in each cell region. However, this method requires annotations for all cell regions (as training data) and does not perform well when the cells are densely distributed, because the embedded IDs are not well learned for such cases. It does not expressly learn the motion of each cell. Hayashida *et al.* [20, 21] proposed tracking methods for jointly estimating the position and motion between two frames. These methods have been shown to outperform those that use a detector separately from a tracker. Unlike these methods, our cell tracking method si-

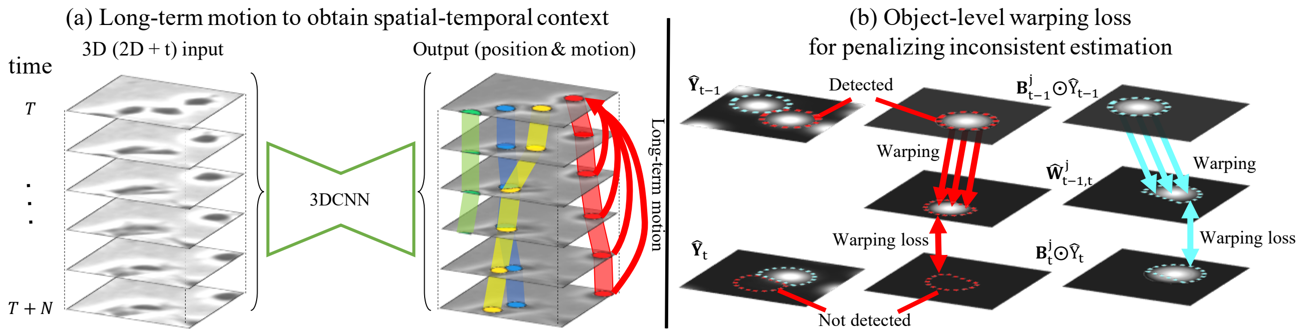
(a) Long-term motion to obtain spatial-temporal context
3D (2D + t) input
time
T
T + N
3DCNN
Output (position & motion)
Long-term motion
(b) Object-level warping loss
for penalizing inconsistent estimation
$\hat{Y}_{t-1}$
Detected
$B^j_{t-1}\odot\hat{Y}_{t-1}$
Warping
$\hat{W}^j_{t-1,t}$
Warping
$\hat{Y}_t$
Warping loss
$B^j_t\odot\hat{Y}_t$
Warping loss
Not detected

Figure 2. Overview of proposed method. (a) Given time-lapse images as inputs, a single network simultaneously estimates multi-frame motion and position maps for every frame. To extract the long-term spatial-temporal context of multiple frames, we estimate long-term motion addition to short-term motion. Here, for visualization purpose, we only show the long-term motion for the red cell but actually these are estimated for every cell. (b) Example of our object-level warping loss from $t-1$ to t, which penalize inconsistent estimation. Left: source image $\hat{\mathbf{Y}}_{t-1}$ and target image $\hat{\mathbf{Y}}_t$. Right: warping image $\hat{\mathbf{W}}^j_{t-1,t}$ was obtained by warping the masked estimated position heat-map $\mathbf{B}^j_{t-1} \odot \hat{\mathbf{Y}}_{t-1}$ using estimated motion $\hat{\mathbf{F}}_{t,t-1}$ (Eq.10), for red and blue cells, respectively. $\mathbf{B}^j_{t-1}$ is the mask image that has 1 on the target cell (red/blue region). For the red cell, warping loss has a high value since the cell was detected at $t-1$ but not at t (inconsistent). For the blue cell, warping loss has a low value since the cell was detected at both frames (consistent).

multaneously estimates the position and motion using the spatial-temporal context in multiple frames.

Tracking-by-detection for general multi-object tracking: In general multi-object tracking (MOT), tracking-by-detection is a standard approach [26, 15, 18, 46, 4, 11, 39]. Most MOT methods use bounding box detectors [32, 42, 40, 52], such as Faster R-CNN [42], and learn the appearance-feature models of the bounding boxes for association. Person re-identification [16, 44, 49] is often used to compute the similarity in appearance of detected persons. LSTM [22] is also often used for learning the long-term appearance model in multiple frames [35, 12, 10, 14, 26]. Kim *et al.* [26] uses a bilinear LSTM to learn long-term appearance models from the sequence of bounding boxes. A recent trend in multi-object tracking is to incorporate existing detectors into trackers and simultaneously train the appearance models and the association in multiple frames [15, 18, 46, 4, 11, 39]. Feichtenhofer *et al.* [15] predicted motions between bounding boxes in successive frames and propagated the loss to the detectors. Sun *et al.* [46] proposed the Deep Affinity Network (DAN) that jointly learns a feature representation for identification and association. Chu *et al.* [11] proposed FAMNet, which incorporates an optimization function for the data-association problem into a network and simultaneously learns the feature extraction for object similarity and data association. These methods use a bounding box detector, and the tracking part only uses the detected bounding boxes. They learn the appearance feature models and association using an end-to-end manner. However, they do not exploit the spatial context of the outer regions of the bounding boxes. We here note that although some methods [19, 34] use multiple frames for learning similarity measurements of detected bounding boxes, they did not use the 'long-term motion' for 'motion' estimation. It is very different from our method. In addition, it is the first attempt to use the object-level warping loss for motion estimation.

Point-based tracking using spatial context: Point-based methods for jointly estimating the center positions of objects and their attributes as points have recently attracted attention [20, 21, 60, 55, 59]. Given an input image, CenterNet [60] simultaneously estimates the center-position heatmap and the size map that stores the height and width of the object's bounding box. Similarly, given two successive frames, CenterTrack [59] also estimates the position offset (motion) of objects between two successive frames. These methods produce multiple maps for the center position heat-map and other attributes. MPM [21] simultaneously represents detection and association in a single map. The advantage of these methods is their effective use of the spatial context, because they directly extract common image features for the motion and position estimation by using a single network. However, cells often partially overlap, and this situation typically continues for several frames, as shown in Fig. 1. Even experts have difficulty identifying each cell from only two frames. To tackle such difficult situations, we propose a cell-tracking method that can effectively use the long-term spatial-temporal context in multiple frames with preserving consistency.

Tracking using optical flow: Tracking methods using optical flow have been proposed [58, 27, 29, 54, 38]. They basically estimate the optical flow of an entire image between successive frames using conventional or deep-based methods [13]; then, the optical flow was used to facilitate tracking. In their methods, the flows are estimated from the original images but not from object-level estimation results. In contrast, we perform the object-level warping loss to train a network so that produce consistent results.

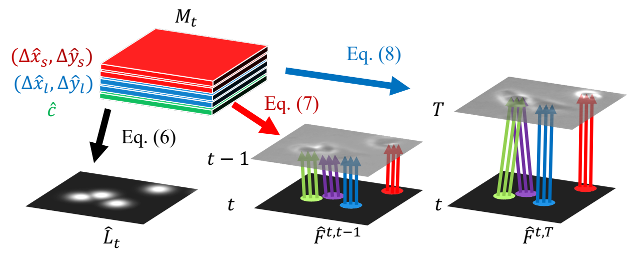

Figure 3. Each map can be converted into short-term motion-vector field $\hat{\mathbf{F}}_{t,t-1}$, long-term motion-vector field $\hat{\mathbf{F}}_{t,T}$ and cell-position heat-map $\hat{\mathbf{Y}}_t$.

3. Tracking using long-term spatial-temporal context

As shown in Fig. 2 (a), our method simultaneously estimates the short-term and long-term motion and position of each cell in multi-frames using 3DCNN, in which long-term motion facilitates extracting the long-term spatial-temporal context and is used for interpolation of false negative. In the training of the network, we perform an object-level warping loss to preserve the consistency among the estimated motion and positions in multi-frames as shown in Fig. 2(b). Our method performs tracking by object-level warping that transfers a region of each cell from $t-1$ to t by using the estimated motions. The individual trajectories in multiple frames can be directly obtained using this warping operation. The details are described as follows.

Encoding long-term motion in heat-map:
Given multiple time-lapse frames $\{\mathbf{I}_t; t \in \{T-1, \dots, T+N\}\}$, a single network simultaneously estimates the two types of motion (short and long) and the position maps at $t \in \{T, ..., T+N\}$ (Fig. 3). To represent this map, we follow the recently proposed MPM [21], which simultaneously represents the position and moving direction between successive frames. The motion vector of each object is encoded on the pixels of the object's center position, and the distribution of the magnitudes of the vector represents the heat-map of the center positions, where the local-maxima of the heat-map indicates the center position. In contrast to MPM, we encode two motion vectors (short-term and long-term motion) and the cell center position to a map.

In our problem setting, the trajectories of the cell-center positions are annotated and used as the training data, where we denote an annotated cell position for the i-th cell at t as $\boldsymbol{a}_t^i = (x_t^i, y_t^i)$. The partial sequence $\{\mathbf{I}_t; t \in \{T-1, ..., T+N\}\}$ is taken as the input images since we cannot input the entire sequence due to the limitation of GPU memory. Note that we treat motion as an inverse direction from t to $t-1$ in order to naturally define cell mitosis when a mother cell divides into two daughter cells in one motion from a daughter cell.

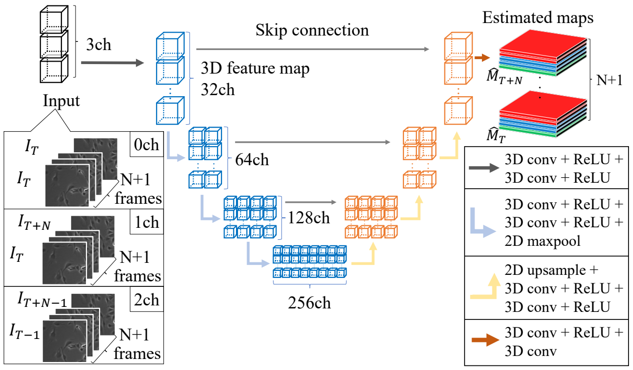

Figure 4. Network architecture.

For each annotated center $\boldsymbol{a}_t^i$, the short-term motion from t to $t-1$ is defined as $(\Delta x_s^i, \Delta y_s^i) = (x_{t-1}^i - x_t^i, y_{t-1}^i - y_t^i)$. The long-term motion from t to T is defined as $(\Delta x_l^i, \Delta y_l^i) = (x_T^i - x_t^i, y_T^i - y_t^i)$. The concatenation of these two motion vectors $\boldsymbol{v}_t^i = (\Delta x_s^i, \Delta y_s^i, \Delta x_l^i, \Delta y_l^i, \mathrm{C})$ is encoded on pixel $\boldsymbol{p}$ around the i-th center $\boldsymbol{a}_t^i$. We introduce a constant value C on an additional dimension to distinguish the following two cases; 'no cells at $\boldsymbol{p}$' is represented as $(0,0,0,0,0)$, and 'no motion' as $(0,0,0,0,\mathrm{C})$. The encoded vector $\mathbf{M}_t^i(\boldsymbol{p})$ is represented as

$$\mathbf{M}_t^i(\boldsymbol{p}_t) = w(\boldsymbol{p}_t)\frac{\boldsymbol{v}_t^i}{||\boldsymbol{v}_t^i||_2}, \tag{1}$$

$$w(\boldsymbol{p}_t) = \exp\left(-\frac{||\boldsymbol{a}_t^i - \boldsymbol{p}_t||_2^2}{\sigma^2}\right), \tag{2}$$

where $w(\boldsymbol{p}_t)$ is a Gaussian function with the annotation coordinate $\mathbf{a}_t^i$ (center point) as its peak, and σ is a hyper-parameter that indicates the standard deviation of the distribution and controls the spread of the peak.

The entire map $\mathbf{M}_t$ at t is defined as follows:

$$\mathbf{M}_t(\boldsymbol{p}) = \mathbf{M}_t^{i'}(\boldsymbol{p}), \tag{3}$$

$$i' = \arg\max_i ||\mathbf{M}_t^i(\boldsymbol{p})||_2. \tag{4}$$

Since the motion vector $\boldsymbol{v}_t^i/||\boldsymbol{v}_t^i||_2$ is a unit vector in Eq.(1), $w(\cdot)$ is a function that represents the magnitude of each vector in the map. The magnitude of $\mathbf{M}_t$ shows the cell-position heat-map in which a local maximum indicates the center position of an object. This simultaneous representation guarantees coherence whereby if a cell is detected, the corresponding two types of motions are always obtained. The set of ground-truth maps is defined as $\{\mathbf{M}_T, \mathbf{M}_{T+1}, ..., \mathbf{M}_{T+N}\}$.

Network model and MSE loss:

Fig. 4 shows an overview of our network trained to estimate the set of maps given multiple frames $\{\mathbf{I}_t \in \mathbb{R}^{W\times H}; t \in$

$\{T-1,...,T+N\}\}$. The network has a U-Net-like architecture [43] and 3D convolution layers. For the input, we create 3ch × 3D data ($(N+1)\times W\times H$), in which the 0ch consists of the duplicate images of the initial frame $\{\mathbf{I}_T,...,\mathbf{I}_T\}$, 1ch consists of time-lapse images $\{\mathbf{I}_T,...,\mathbf{I}_{T+N}\}$, and 2ch consists of the shifted time-lapse images $\{\mathbf{I}_{T-1},...,\mathbf{I}_{T+N-1}\}$. The pair (0ch and 1ch) at time t is $\{\mathbf{I}_t,\mathbf{I}_T\}$ and corresponds to learning the long-term motion from t to T. The pair (1ch and 2ch) at time t is $\{\mathbf{I}_t,\mathbf{I}_{t-1}\}$ and corresponds to learning the short-term motion from t to $t-1$.

This network is trained using the sum of two types of losses $L = L_{mse} + L_w$, where L_{mse} indicates the mean squared error (MSE) between the estimated maps and their ground-truth, and L_w indicates the warping loss to prevent inconsistency among frames. Let us define the estimated maps as $\{\hat{\mathbf{M}}_T,\hat{\mathbf{M}}_{T+1},...,\hat{\mathbf{M}}_{T+N}\}$. The L_{mse} is defined as

$$L_{mse} = \frac{1}{N+1}\sum_{t=T}^{T+N}\sum_{\mathbf{p}}(||\mathbf{M}_t(\mathbf{p})-\hat{\mathbf{M}}_t(\mathbf{p})||_2^2 \\ +(||\mathbf{M}_t(\mathbf{p})||_2 - ||\hat{\mathbf{M}}_t(\mathbf{p})||_2)^2), \quad (5)$$

where the first term $||\mathbf{M}_t(\mathbf{p})-\hat{\mathbf{M}}_\mathbf{t}(\mathbf{p})||_2^2$ is the squared error between the ground-truth map $\mathbf{M}_t$ and estimated map $\hat{\mathbf{M}}_t$. The second term $(||\mathbf{M}_t(\mathbf{p})||_2 - ||\hat{\mathbf{M}}_\mathbf{t}(\mathbf{p})||_2)^2$ is the squared error between the magnitudes of $\mathbf{M}_t$ and $\hat{\mathbf{M}}_t$; it directly reflects the error of the position heat-map (i.e., detection), making training stable.

Object-level warping loss:

The estimated maps may have inconsistencies, *e.g.,* when no cell is detected at that position at t even if a cell is detected at $t-1$ and its motion is toward a specific position. To mitigate this problem, we introduce a warping loss that penalizes such inconsistencies between the estimated position and motion in multiple frames. To find inconsistencies between the frames, we can not directly compare the detection results because cells move and we do not know the association between detected positions. Therefore, we perform a warping operation to find the inconsistencies. An image-level warping loss has often been used for the optical-flow estimation problem to estimate the corresponding pixels from a source image to a target image. However, in multi-object tracking, the image-level warping loss does not penalize switching errors. Therefore, we perform warping for each object (object-level warping) to penalize switching cases. In addition, since cell regions are not provided as the training data, we perform the warping for the estimated heat-map instead of the original images. The loss is computed by the sum of the MSE between the target image and warped image from the source image for each cell. This loss maintains consistency among estimation results.

Fig. 2(b) illustrates our warping loss from $t-1$ to t. In this example, a cell is miss-detected at t, and this miss-detection is inconsistent with the correct detection results at $t-1$. The warping operation enables us to compare the MSE of the heat-map between the different frames by warping the estimated heat-map using the motion information. This loss penalizes the inconsistency.

Let us define the estimated vector at pixel $\boldsymbol{p}$ as $\hat{\mathbf{M}}_t(\boldsymbol{p}) = (\hat{\Delta x}_s, \hat{\Delta y}_s, \hat{\Delta x}_l, \hat{\Delta y}_l, \hat{c})$. $(\hat{\Delta x}_s, \hat{\Delta y}_s)$ corresponds to the weighted short-term motion vector from t to $t-1$, and $(\hat{\Delta x}_l, \hat{\Delta y}_l)$ corresponds to the weighted long-term motion vector from t to T. The $\hat{c}$ stores the weighting information, and the motion vector can be restored by multiplying the vectors by $\mathrm{C}/\hat{\mathrm{c}}$ [1]. The estimated center-position heat-map $\hat{\mathbf{Y}}_t$, and two types of motion vector fields $\hat{\mathbf{F}}_{t,t-1}, \hat{\mathbf{F}}_{t,T}$ are defined as follows:

$$\hat{\mathbf{Y}}_t(\boldsymbol{p}) = ||\hat{\mathbf{M}}_t(\boldsymbol{p})||_2, \quad (6)$$

$$\hat{\mathbf{F}}_{t,t-1}(\boldsymbol{p}) = \frac{\mathrm{C}}{\hat{c}}\left(\hat{\Delta x}_s, \hat{\Delta y}_s\right), \quad (7)$$

$$\hat{\mathbf{F}}_{t,T}(\boldsymbol{p}) = \frac{\mathrm{C}}{\hat{c}}\left(\hat{\Delta x}_l, \hat{\Delta y}_l\right), \quad (8)$$

where positions around the cell centroid only have a non-zero vector, and its magnitude value indicates the value of the position heatmap at $\boldsymbol{p}$ as represented by Eq. 6. Fig. 3 illustrates the transformation from $\hat{\mathbf{M}}_t$ into $\hat{\mathbf{Y}}_t, \hat{\mathbf{F}}_{t,t-1}, \hat{\mathbf{F}}_{t,T}$ when a cell divides into two cells. In this case, the motion vectors from the two daughter cells point to the same position in the mother cell (the green and purple arrows).

The object-level warped image $\hat{\mathbf{W}}^j_{t-1,t}$ for the j-th cell from $t-1$ to t is defined as the image that the masked image $\mathbf{B}^j_{t-1}\odot\hat{\mathbf{Y}}_{t-1}$ is warped by the motion $\hat{\mathbf{F}}_{t,t-1}$, in which $\mathbf{B}^j_t$ indicates a binary mask that has 1 for the foreground region $\mathbf{S}^j_t$, and $\odot$ is Hadamard product. The foreground region of the j-th cell at frame t ($\mathbf{S}^j_t$) is a set of pixels that satisfies $||\mathbf{M}^j_t(\boldsymbol{p})||_2 > th_m$. It is defined as:

$$\hat{\mathbf{W}}^j_{t-1,t}(\boldsymbol{p}) = \{\mathbf{B}^j_{t-1}\odot\hat{\mathbf{Y}}_{t-1}\}(\boldsymbol{p}+\hat{\mathbf{F}}_{t,t-1}(\boldsymbol{p})), \quad (9)$$

where the warping direction is the inverse of the motion vector, similar to that of optical-flow estimation. Similarly, the warped image $\hat{\mathbf{W}}^j_{T,t}$ from T to t (long-term motion) is obtained using $\hat{\mathbf{F}}_{t,T}$:

$$\hat{\mathbf{W}}^j_{T,t}(\boldsymbol{p}) = \{\mathbf{B}^j_T\odot\hat{\mathbf{Y}}_T\}(\boldsymbol{p}+\hat{\mathbf{F}}_{t,T}(\boldsymbol{p})). \quad (10)$$

This warping operation is applied to estimation result from T to $T+N$, and the set of warped images $\{\hat{\mathbf{W}}\}$ and center position heat-maps $\{\hat{\mathbf{Y}}\}$ are obtained.

If the heat-maps $\hat{\mathbf{Y}}_t$ and $\hat{\mathbf{Y}}_{t-1}$ and motion field $\hat{\mathbf{F}}_{t,t-1}$ are accurately estimated, the warped image $\hat{\mathbf{W}}_{t-1,t}$ should

[1] To avoid dividing C by zero value, a very small value was added to $\hat{c}$ in the implementation

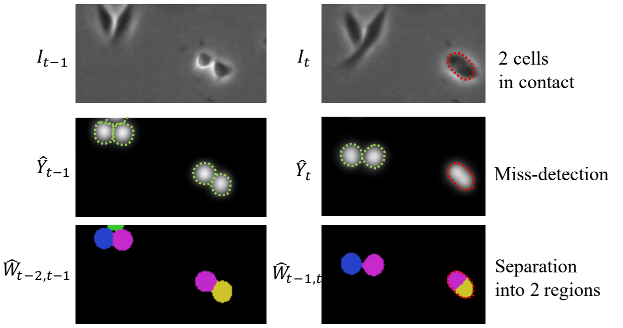

Figure 5. Our tracking using warping recovers from miss-detection. In 2nd frame, two cells were detected as single region. However, these two cells were correctly detected in previous frame; thus, warped regions were separately distributed in region.

be the same as $\hat{\mathbf{Y}}_t$. The warping loss L_w is defined as the sum of the MSEs between $\hat{\mathbf{W}}$ and $\hat{\mathbf{Y}}$:

$$L_w = \frac{1}{\mathrm{N}} \sum_{t=T+1}^{\mathrm{T+N}} \sum_{j=1}^{K^t} (||\hat{\mathbf{W}}_{t-1,t} - \mathbf{B}_t^j \odot \hat{\mathbf{Y}}_t||_2^2 \tag{11}$$

$$+||\hat{\mathbf{W}}_{T,t} - \mathbf{B}_t^j \odot \hat{\mathbf{Y}}_t||_2^2), \tag{12}$$

where K^t is the number of cells at t.

Tracking by object-level warping

We track cells by applying an object-level warping operation, which transforms the detected region of each cell at $t-1$ into the next frame by using the estimated short-term motion $\hat{\mathbf{F}}_{t,t-1}$. First, we initialize the regions around cell centroids at the initial frame by using position heat-map $\hat{\mathbf{Y}}_0$; then, we obtain the trajectory by simply warping the regions of each cell into the next frame (Eq. 9). This operation is iterated for one frame to the next.

Our tracking-by-warping method has three advantages. First, it can handle a mitosis event because $\hat{\mathbf{F}}_{t,t-1}$ can represent the one-to-two matching of the regions. Second, it can recover from a miss-detection. Let us consider a case in which two cells are detected at $t-1$ but overlap at t in a way that one of the cells is not detected at t, as shown in Fig. 5. Here, the warped regions of the two cells are separated into two regions in a single detected region at t. The proposed method can recover from this miss-detection. Third, it can recover from the case in which there are still false negatives after applying our object-level warping operation by interpolating the miss-detection using long-term motion $\hat{\mathbf{F}}_{t,T}$. When a track termination and the beginnings of new tracks are found, our network is applied to the sequence images whose initial frame is the termination time of the track. Our method determines the connection of the tracklets by using the estimated long-term motion from the initial cell position of the new track. The false-negative positions between the tracklets can be simply interpolated. Since the long-term motions relative to the initial frame are estimated for every frame, only hyper-parameter is the maximum number of frames.

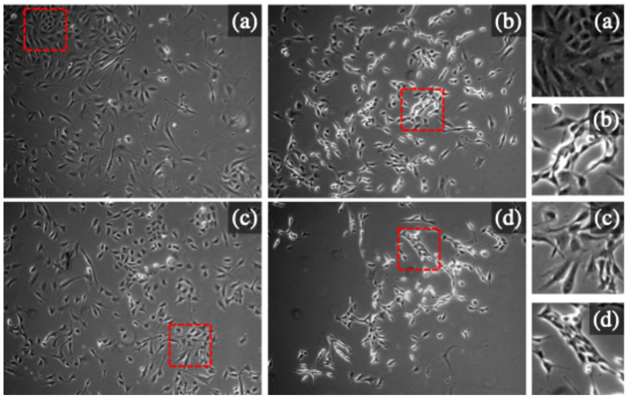

Figure 6. Example images captured under four conditions. a) BMP2, b)BMP2+FGF2, c) Control, d) FGF2, (e)-(h) Enlarged images of red box in (a)-(d). Cell appearance and image contrast differ depending on conditions.

4. Experiments

Data set and experimental setup

For our experiments, we used an open dataset [25] containing time-lapse image sequences captured by phase-contrast microscopy because these data were used in the study most related to ours [21] and the data fit our target[2]; cells often overlap; thus, it is difficult even for an expert to detect individual cells from only one image, as shown in Fig. 1. The data include image sequences captured under four different conditions: (a) BMP2, (b) BMP2+FGF2, (c) Control, (d) FGF2, in which each condition has four image sequences (total: 16 sequences). As shown in Fig. 6, the cell appearance significantly differs depending on the conditions; cells often shrink and partially overlap in FGF2, cell regions tend to be expanded under BMP2, and there are both expanded and shrunken cells under BMP2+FGF. Tracking under FGF2 is particularly difficult.

In each image sequence, the images were captured every 5 minutes for 780 frames with a resolution of 1392×1040 pixels. For the training data, all cells were annotated in 200 frames of the BMP2, BMP2+FGF2. For the test data, all cells were annotated in BMP2; three cells were randomly selected at the beginning of the sequence. Then, the three cell's family trees throughout the entire sequence were annotated in all other 15 sequences, where the number of the annotated cells increased with time due to cell division. The total number of annotated cells was 202851. In the test, Control and FGF2 conditions differed from the conditions of the training data. To train our network, we used the Adam

[2] These data are more challenging compared with the data of the ISBI Cell Tracking Challenge [40, 50], which focuses on segmentation tasks.

Table 1. Cell-detection performance in terms of precision (Pre.), recall (Rec.) and F1-score (F1). Met. denotes method.

Met.	Bensch [3]	Bise [7]	CMF [20]	MPM [21]	Ours
Pre.	0.583	0.850	0.968	0.964	**0.972**
Rec.	0.623	0.811	0.902	0.932	**0.940**
F1	0.602	0.830	0.934	0.948	**0.956**

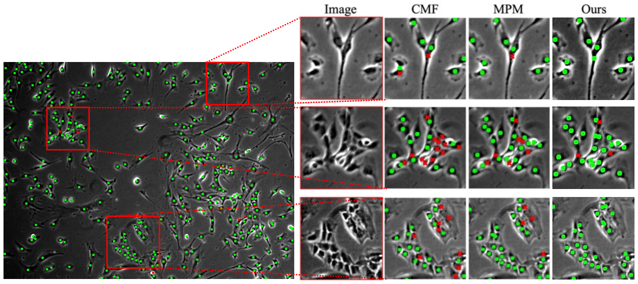

Figure 7. Examples of detection results under the Control. Left: the entire image depicts our results; the total number of cells was 346; Right: enlarged images of detection results from CMF [20], MPM [21], and ours. Green: true positive; Red: false negative.

[28] optimizer with a learning rate of 10^{-4}, epoch$=300$, $\sigma = 6$, C $= 5$, $th_m = 0.03$, $n = 6$ in all experiments. The detection regions are obtained by the thresholding of the heatmap (threshold $= 0.3$). Although a larger number of input images n is the better, we set n to 6 due to the memory limitation of the GPU (NVIDIA TITAN RTX: 24GB).

We compared our method with six other methods: Bensch [3], which uses an asymmetric graph-cut and frame-by-frame association; Chalfoun [9], which first segments cell regions [57] and then performs optimization for the frame-by-frame association; Bise[7], which solves a global data association problem using an entire sequence; CMF [20], which uses a CNN to estimate motion flow; MPM [21], which achieved state-of-the-art (SOTA) performance on this data set); CenterTrack [59], which simultaneously estimates a position map and motion map by using a single network. (The method was developed for general multi-object tracking; thus, it cannot handle mitosis events and tracking under dense conditions. We tested this method to show that a state-of-the-art method for general multi-object tracking does not work for cell tracking). We used the training data for training and tuning the parameters of all the methods.

Cell detection performance

We evaluated the cell-detection performance by using precision, recall, and F1-score. Note that we could not apply learning-based segmentation because the data-set only had point-level annotation. Table 1 lists the cell-detection performances of the methods. The CNN-based methods (CMF, MPM, and Ours) outperformed the other two methods. Our method slightly improved all the metrics compared with MPM (SOTA). The framework using spatial-temporal context in multiple frames probably contributed to this improvement. Fig. 7 shows examples of cell-detection results by CMF [20], MPM [21], and our method. Our method successfully detected cells that were in close contact, while the other methods miss-detected them.

Table 2. Target effectiveness

Method	BMP2*	FGF2+ BMP2*	Cont.	FGF2	Ave.
Bensch [3]	0.543	0.448	0.621	0.465	0.519
Chalfoun [8]	0.691	0.587	0.683	0.604	0.641
†Li [30]	0.630	0.710	0.700	0.570	0.653
†Kanade [23]	0.800	0.790	0.830	0.640	0.765
Bise [7]	0.788	0.633	0.733	0.710	0.716
CenterTrack[59]	0.547	0.501	0.428	0.520	0.499
CMF [20]	0.939	0.841	0.756	0.761	0.822
MPM [21]	0.958	0.911	0.803	0.829	0.875
Ours	**0.978**	**0.955**	**0.909**	**0.884**	**0.931**

Table 3. Target effectiveness (TE) in ablation study. TE is the average of all conditions. 'multi' is simultaneous estimation in multiple frames. 'wl' is the object-level warping loss. 'lt' is long-term motion estimation. 'int.' is interpolation.

method	multi	wl	lt	int.	TE
w/o wl, lt, int,	✓				0.862
w/o lt, int.	✓	✓			0.901
w/o int.	✓	✓	✓		0.903
Ours	✓	✓	✓	✓	**0.931**

Cell tracking performance

We evaluated the tracking performance in terms of target effectiveness [24, 21]. Target effectiveness indicates the number of consecutive frames in which the tracking method correctly tracked a cell continuously. To compute this metric, we first assigned each ground-truth cell position an estimated cell position for each frame and then found the most often assigned estimated track. The target effectiveness is the number of the assigned frames of the estimated track divided by the total number of frames in the ground-truth. This metric is very strict. Even if only one switching error occurs in the middle of the trajectory, the target effectiveness is 0.5.

Table 2 lists the target effectiveness [3]. Note that BMP2 and FGF2+BMP2 (denoted with '*') were the same conditions in the training but Cont. and FGF2 were different conditions from the training conditions. The non-deep learning methods (Bensch, Chalfoun, Li, Bise, and Kanade) were sensitive to the culture conditions, and did not perform well, in particular, under FGF2. We consider that these non-deep learning methods could not capture image features for

[3] The target-effectiveness scores of †Li and †Kanade were evaluated using the same data-set in their papers [30, 24].

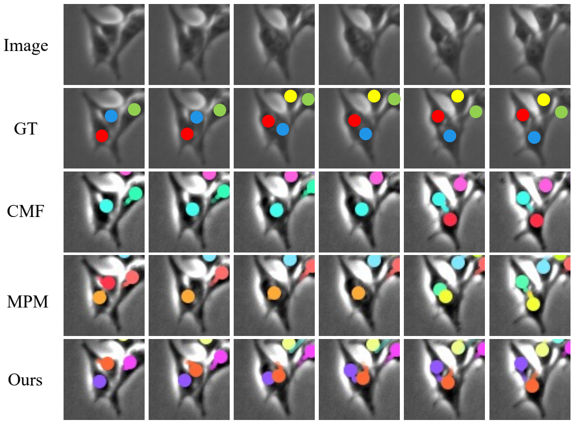

Figure 8. Example of tracking results under FGF2 (Same example as shown in Fig. 1). GT indicates ground-truth. Although conventional methods had miss-tracked significantly touching cells at 3rd and 4th frame, our method correctly tracked them by using spatial-temporal context.

various conditions and that poor detection results affect the tracking performance. The state-of-the-art general multi-object tracking method (CenterTrack) performed the worst since it cannot handle mitosis events and dense conditions. CMF and MPM use the spatial context in two frames by estimating position and motion maps. These methods outperformed the other conventional methods. Compared with (SOTA) MPM, our method reduced the error rate by 45% on average (from 0.125 to 0.069).

Fig. 8 shows examples of tracking results of CMF, MPM, and ours for the sequence, which is the same sequence as shown in Fig. 1. In this example, the red and blue cells moved and formed a cluster at the 3rd and 4th frames; then, they separated into individual cells at the 5th frame. CMF and MPM misidentified these two cells as a single cell and the separation from the cluster at the 5th frame as a mitosis event because they could not use multi-frame information. In contrast, our method successfully tracked these cells using the spatial-temporal context in multi-frame. Fig. 9 shows the tracking results from our method, in which it correctly tracked the various cells under each condition. In the cell mitosis case (Fig. 9 (b)(c)), our method successfully identified the mitosis and tracked the divided cells. Fig. 10 shows 3D view of estimated cell trajectories. This lineage information enables us to automatically compute various cell-behavior metrics.

Table 3 shows the average of target effectiveness for all conditions in the ablation study. The first row indicates the method that estimates the positions and motions in multi-frame by 3D-Unet without using our main contributions; long-term estimation, object-level warping loss, and interpolation using long-term motion. This could not improve the performance from that of MPM. In contrast, each element of the proposed method improved the tracking performance. In particular, the object-level warping loss significantly improved the performance. These results demonstrate that our method addressed the two challenges to extract tracking information from multiple frames as discussed in the introduction; extraction of the information from distant frames, and preserving the consistencies among motions and positions in multiple frames.

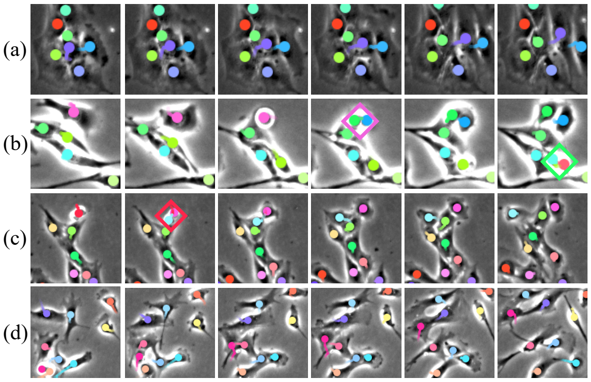

Figure 9. Example of tracking results on various conditions. (a) BMP2, (b) FGF2+BMP2, (c) Control, (d) FGF2. ◇: cell mitosis.

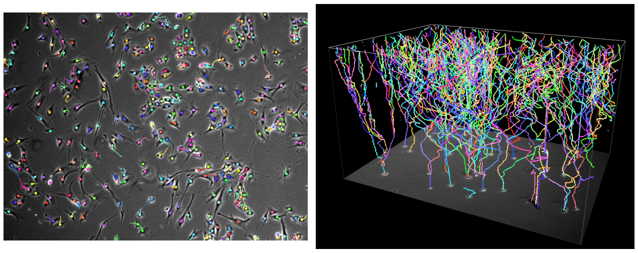

Figure 10. Examples of our tracking results under Control. (a) Entire image. (b) 3D view of estimated cell trajectories. Z-axis indicates time, and each color indicates trajectory of a single cell.

5. Conclusion

We proposed a cell tracking method that effectively uses the long-term temporal context in multi-frames with preserving consistency by introducing long-term motion estimation, object-level warping loss. In our experiments, our method outperformed the compared methods under various conditions in real biological images. A limitation of our method is that may not maintain consistency of results between the separated inputs for the network, in which the maximum number of the input images is limited by GPU memory. If the limitation of GPU memory is mitigated, our method can be easily extended to end-to-end cell tracking that involves inputting a sequence and estimating all trajectories. We believe our method will contribute to general multi-object tracking research in addition to cell tracking.

Acknowledgements: This work was supported by JSPS KAKENHI Grant Number JP20H04211.

References

[1] Saad Ullah Akram, Juho Kannala, Lauri Eklund, and Janne Heikkilä. Joint cell segmentation and tracking using cell proposals. In *ISBI*, pages 920–924, 2016.

[2] Assaf Arbelle and Tammy Riklin Raviv. Microscopy cell segmentation via convolutional lstm networks. In *ISBI*, pages 1008–1012, 2019.

[3] Robert Bensch and Olaf Ronneberger. Cell segmentation and tracking in phase contrast images using graph cut with asymmetric boundary costs. In *ISBI*, pages 1220–1223, 2015.

[4] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *ICCV*, pages 941–951, 2019.

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

[6] Ryoma Bise, Yoshitaka Maeda, Mee-hae Kim, and Masahiro Kino-oka. Cell tracking under high confluency conditions by candidate cell region detection-based-association approach. In *Proceedings of Biomedical Engineering*, pages 1004–1010, 2011.

[7] Ryoma Bise, Zhaozheng Yin, and Takeo Kanade. Reliable cell tracking by global data association. In *ISBI*, pages 1004–1010. IEEE, 2011.

[8] Joe Chalfoun, Michael Majurski, Alden Dima, and *et al*. Lineage mapper: A versatile cell and particle tracker. *Scientific Reports*, 6(1):36984, 11 2016.

[9] Joe Chalfoun, Michael Majurski, Alden Dima, Michael Halter, Kiran Bhadriraju, and Mary Brady. Lineage mapper: A versatile cell and particle tracker. *Scientific reports*, 6:36984, 2016.

[10] Long Chen, Haizhou Ai, Chong Shang, Zijie Zhuang, and Bo Bai. Online multi-object tracking with convolutional neural networks. In *ICIP*, pages 645–649. IEEE, 2017.

[11] Peng Chu and Haibin Ling. Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In *ICCV*, pages 6172–6181, 2019.

[12] Qi Chu, Wanli Ouyang, Hongsheng Li, Xiaogang Wang, Bin Liu, and Nenghai Yu. Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism. In *ICCV*, pages 4836–4845, 2017.

[13] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *ICCV*, December 2015.

[14] Kuan Fang, Yu Xiang, Xiaocheng Li, and Silvio Savarese. Recurrent autoregressive networks for online multi-object tracking. In *WACV*, pages 466–475. IEEE, 2018.

[15] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Detect to track and track to detect. In *ICCV*, pages 3038–3046, 2017.

[16] Yu Fengwei, Li Wenbo, Li Quanquan, Liu Yu, Shi Xiaohua, and Yan Junjie. Multiple object tracking with high performance detection and appearance feature. In *ECCV*, pages 36—42, 2016.

[17] Jan Funke, Lisa Mais, Andrew Champion, Natalie Dye, and Dagmar Kainmueller. A benchmark for epithelial cell tracking. In *ECCVW*, 2018.

[18] Rohit Girdhar, Georgia Gkioxari, Lorenzo Torresani, Manohar Paluri, and Du Tran. Detect-and-track: Efficient pose estimation in videos. In *CVPR*, pages 350–359, 2018.

[19] Brasó Guillem and Leal-Taixé Laura. Learning a neural solver for multiple object tracking. In *CVPR*, 2020.

[20] Junya Hayashida and Ryoma Bise. Cell tracking with deep learning for cell detection and motion estimation in low-frame-rate. In *MICCAI*, pages 397–405, 2019.

[21] Junya Hayashida, Kazuya Nishimura, and Ryoma Bise. MPM: Joint representation of motion and position map for cell tracking. In *CVPR*, 2020.

[22] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[23] Takeo Kanade, Zhaozheng Yin, and Ryoma Bise. Cell image analysis: Algorithms, system and applications. In *WACV*, pages 374–381, 2011.

[24] Takeo Kanade, Zhaozheng Yin, Ryoma Bise, Seungil Huh, Sungeun Eom, Michael F. Sandbothe, and Mei Chen. Cell image analysis: Algorithms, system and applications. In *WACV*. IEEE, 2011.

[25] Elmer Ker, Sungeun Eom, Sho Sanami, and *et al*. Phase contrast time-lapse microscopy datasets with automated and manual cell tracking annotations. *Scientific Data*, 5:180237, 11 2018.

[26] Chanho Kim, Fuxin Li, and James M Rehg. Multi-object tracking with neural gating using bilinear lstm. In *ECCV*, pages 200–215, 2018.

[27] Diederik P Kingma and Jimmy Ba. Optical flow-based real-time object tracking using non-prior training active feature model. *Real-Time Imaging*, pages 204–218, 2005.

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *arXiv*, 2014.

[29] Kale Kiran, Pawar Sushant, and Dhulekar Pravin. Moving object tracking using optical flow and motion vector estimation. In *ICRITO*, 2015.

[30] Kang Li and Takeo Kanade. Nonnegative mixed-norm preconditioning for microscopy image segmentation. In *IPMI*, pages 362–373, 2009.

[31] Kang Li, Eric D Miller, Mei Chen, Takeo Kanade, Lee E Weiss, and Phil G Campbell. Cell population tracking and lineage construction with spatiotemporal context. *Medical image analysis*, 12(5):546–566, 2008.

[32] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37. Springer, 2016.

[33] Filip Lux and Petr Matula. Dic image segmentation of dense cell populations by combining deep learning and watershed. In *ISBI*, pages 236–239, 2019.

[34] Keuper Margret, Tang Siyu, Andres Bjoern, Brox Thomas, and Schiele Bernt. Motion segmentation & multiple object tracking by correlation co-clustering. *PAMI*, pages 140–153, 2018.

[35] Anton Milan, S Hamid Rezatofighi, Anthony Dick, Ian Reid, and Konrad Schindler. Online multi-target tracking using recurrent neural networks. In *AAAI*, pages 4225–4232, 2017.

[36] Kenji Okuma, Ali Taleghani, Nando De Freitas, James J Little, and David G Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*, pages 28–39, 2004.

[37] Christian Payer, Darko Štern, Thomas Neff, Horst Bischof, and Martin Urschler. Instance segmentation and tracking with cosine embeddings and recurrent hourglass networks. In *MICCAI*, pages 3–11, 2018.

[38] Wei qiang Li, Jiatong Mu, and G. Liu. Multiple object tracking with motion and appearance cues. In *ICCVW*, pages 161–169, 2019.

[39] Yaadhav Raaj, Haroon Idrees, Gines Hidalgo, and Yaser Sheikh. Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields. In *CVPR*, pages 4620–4628, 2019.

[40] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016.

[41] Markus Rempfler, Valentin Stierle, Konstantin Ditzel, Sanjeev Kumar, Philipp Paulitschke, Bjoern Andres, and Bjoern H Menze. Tracing cell lineages in videos of lens-free microscopy. *Medical image analysis*, 48:147–161, 2018.

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, pages 91–99, 2015.

[43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015.

[44] Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. In *ICCV*, pages 300–311, 2017.

[45] Martin Schiegg, Philipp Hanslovsky, Bernhard X Kausler, Lars Hufnagel, and Fred A Hamprecht. Conservation tracking. In *ICCV*, pages 2928–2935, 2013.

[46] Sun ShiJie, Akhtar Naveed, Song HuanSheng, Mian S.Ajmal, and Shah Mubarak. Deep affinity network for multiple object tracking. *TPAMI*, 2019.

[47] Ihor Smal, Wiro Niessen, and Erik Meijering. Bayesian tracking for fluorescence microscopic imaging. In *ISBI*, pages 550–553, 2006.

[48] Hang Su, Zhaozheng Yin, Seungil Huh, and Takeo Kanade. Cell segmentation in phase contrast microscopy images via semi-supervised classification over optics-related features. *Medical image analysis*, 17(7):746–765, 2013.

[49] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, and Bernt Schiele. Multiple people tracking by lifted multicut and person re-identification. In *CVPR*, pages 3539–3548, 2017.

[50] 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):1141–1152, 2017.

[51] Xiaoxu Wang, Weijun He, Dimitris Metaxas, Robin Mathew, and Eileen White. Cell segmentation and tracking using texture-adaptive snakes. In *ISBI*, pages 101–104, 2007.

[52] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, pages 3645–3649. IEEE, 2017.

[53] Zheng Wu, Danna Gurari, Joyce Wong, and Margrit Betke. Hierarchical partial matching and segmentation of interacting cells. In *MICCAI*, pages 389–396, 2012.

[54] Zhu Xizhou, Wang Yujie, Dai Jifeng, Yuan Lu, and Yichen Wei. Flow-guided feature aggregation for video object detection. 2017.

[55] Zhenbo Xu, Wei Zhang, Xiao Tan, Wei Yang, Huan Huang, Shilei Wen, Errui Ding, and Liusheng Huang. Segment as points for efficient online multi-object tracking and segmentation. In *ECCV*, 2020.

[56] Fuxing Yang, Michael A Mackey, Fiorenza Ianzini, Greg Gallardo, and Milan Sonka. Cell segmentation, tracking, and mitosis detection using temporal context. In *MICCAI*, pages 302–309, 2005.

[57] Zhaozheng Yin, Takeo Kanade, and Mei Chen. Understanding the phase contrast optics to restore artifact-free microscopy images for segmentation. *Medical image analysis*, 16(5):1047–1062, 2012.

[58] Yu Huang, T. S. Huang, and H. Niemann. Segmentation-based object tracking using image warping and kalman filtering. In *ICIP*, volume 3, pages 601–604 vol.3, 2002.

[59] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. *ECCV*, 2020.

[60] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv*, 2019.

[61] Zibin Zhou, Fei Wang, Wenjuan Xi, Huaying Chen, Peng Gao, and Chengkang He. Joint multi-frame detection and segmentation for multi-cell tracking. *ICIG*, 2019.