General Constraints for Batch Multiple-Target Tracking Applied to Large-Scale Videomicroscopy

Kevin Smith[1,2] Vincent Lepetit[1] Alan Carleton[2]
[1]Computer Vision Laboratory [2]Flavour Perception Group, Brain Mind Institute
École Polytechnique Fédérale de Lausanne (EPFL) 1015 Lausanne, Switzerland
Email: {kevin.smith, vincent.lepetit, alan.carleton}@epfl.ch

Abstract

While there is a large class of Multiple-Target Tracking (MTT) problems for which batch processing is possible and desirable, batch MTT remains relatively unexplored in comparison to sequential approaches. In this paper, we give a principled probabilistic formalization of batch MTT in which we introduce two new, very general constraints that considerably help us in reaching the correct solution. First, we exploit the correlation between the appearance of a target and its motion. Second, entrances and departures of targets are encouraged to occur at the boundaries of the scene. We show how to implement these constraints in a formal and efficient manner.

Our approach is applied to challenging 3-D biomedical imaging data where the number of targets is unknown and may vary, and numerous challenging tracking events occur. We demonstrate the ability of our model to simultaneously track the nuclei of over one hundred migrating neuron precursor cells collected from a 2-photon microscope.

1. Introduction

Multiple-target tracking has historically focused on sequential processing approaches. This approach is natural for real-time applications, yet real-time processing is unnecessary for many types of problems such as biomedical image analysis. More importantly, sequential methods cannot revisit poor or erroneous past estimations in light of new information. Therefore, when possible, one should consider batch processing as it optimizes globally over time and does not suffer from this problem. Modern computation power makes batch processing more practical than in the past, but batch MTT remains mostly unexplored and unused compared to sequential MTT.

As outlined in Fig. 1 featuring a sequence of migrating neurons, this work makes two main contributions to batch processing MTT, though our problem formulation may be regarded as a contribution as well, since we did not find a fully satisfactory one in the literature. Our contributions are two very general and intuitive constraints. These constraints fit naturally into the formulation, and improve the results of state-of-the-art batch tracking models.

The first constraint exploits the often ignored correlation between the appearance of a target and its motion. In tracking, it is conventional to assume a target's motion and appearance are independent, but for a wide range of objects including people, vehicles, and animals, this assumption is not valid. We avoid this independence assumption in our problem formulation, and propose a joint motion-appearance model which encourages target motion that agrees with appearance. In this work, our model reflects the fact that migrating neuron nuclei usually elongate in the direction they travel. While our joint motion-appearance model used for neuron nuclei is relatively straightforward, the correlation property is still valid for objects requiring more complex models.

Our second constraint is so natural it is surprising it does not appear in the literature, at least to the best of our knowledge. It simply states that targets tracks are more likely to begin and end at the boundaries of the scene in time or space (or both). Many authors have found *ad hoc* variations of this constraint useful in the past [9, 2], but to our knowledge, our formulation is the first to incorporate this constraint into the problem formulation in a principled manner.

We demonstrate our approach on complex sequences of migrating neurons in 2-photon 3-D videomicroscopy, in which over one hundred individual cells are present. The number of targets we are able to successfully track is noteworthy, as state-of-the-art methods typically show results on less than ten simultaneous targets. We also show that our approach compares favorably against [12], a recent and very powerful Markov Chain Monte Carlo (MCMC) batch processing method.

In the space that remains, we first review related work, then give a general formulation of the problem, detailing our two constraints. We then describe our 2-photon microscopy

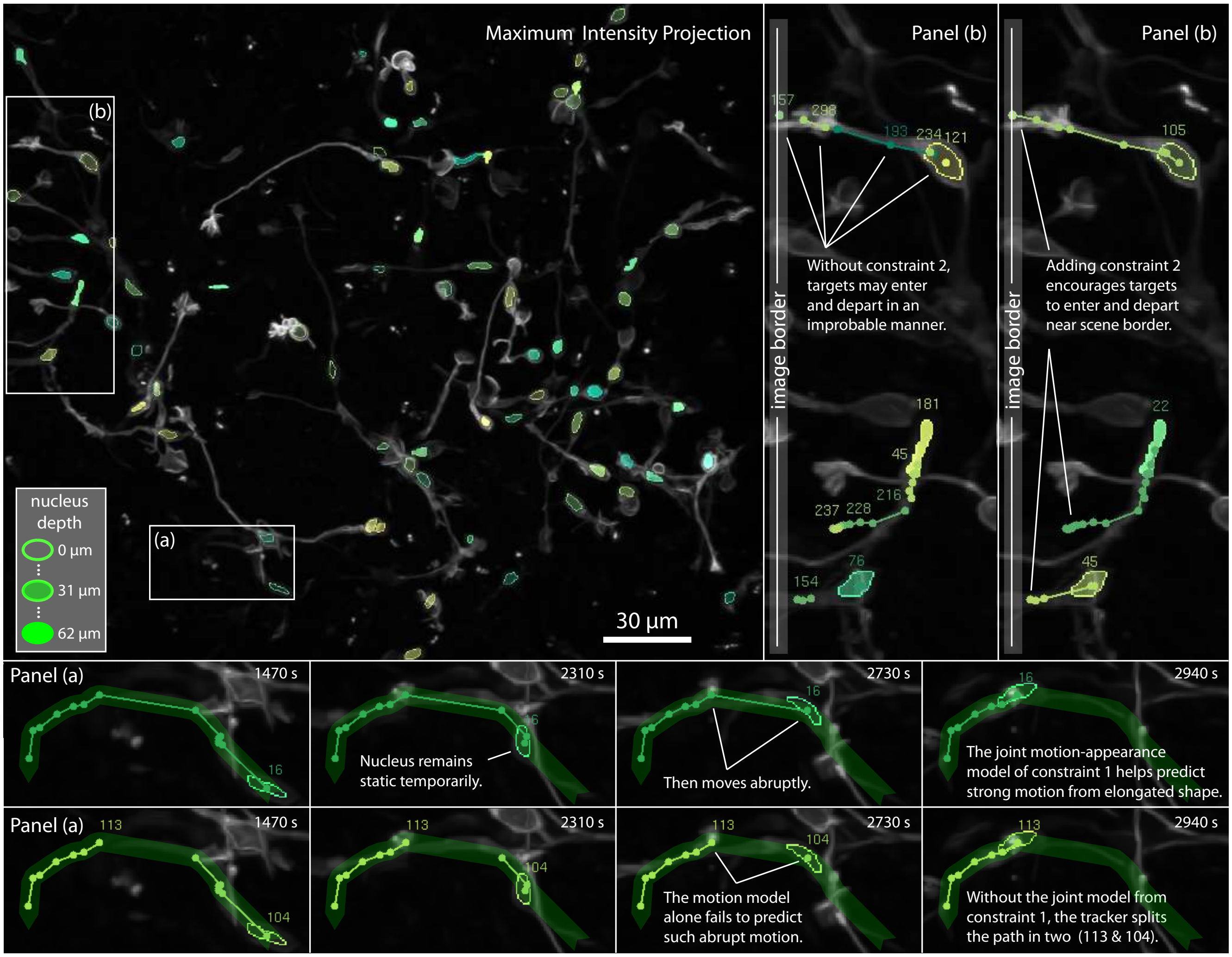

Maximum Intensity Projection
(b)
(a)
nucleus depth
0 µm
31 µm
62 µm
30 µm
Panel (b)
image border
157
298
193
234
121
Without constraint 2, targets may enter and depart in an improbable manner.
181
45
216
237
228
76
154
Panel (b)
image border
105
Adding constraint 2 encourages targets to enter and depart near scene border.
22
45
Panel (a)
1470 s
16
2310 s
16
Nucleus remains static temporarily.
2730 s
16
Then moves abruptly.
2940 s
16
The joint motion-appearance model of constraint 1 helps predict strong motion from elongated shape.
Panel (a)
113
1470 s
104
113
2310 s
104
113
2730 s
104
The motion model alone fails to predict such abrupt motion.
113
2940 s
Without the joint model from constraint 1, the tracker splits the path in two (113 & 104).

Figure 1. The main panel contains an image stack projection of migrating neuron precursor cells, with highlighted nucleus detections. Tracking results are provided in cutouts of panels *(a)* and *(b)*, with target IDs appearing above the tracks. Our 1^{st} constraint, which states that target motion and appearance are dependent, appears in panel *(a)*. In the top row, a joint motion-appearance model imposed by the 1^{st} constraint assists in predicting the correct motion by informing us that an elongated nucleus often corresponds to high velocity in the direction of elongation. Below, results without the joint model are given, where an abrupt change in motion causes tracking failure. Our 2^{nd} constraint, stating targets are more likely to enter and depart near a scene boundary, appears in panel *(b)*. *(left)* Without the constraint in place, target tracks appear and disappear anywhere in the scene. *(right)* Enforcing the constraint discourages this unlikely behavior.

data, the neuron nucleus model, and show how MCMC is used to infer a tracking solution. Finally, we provide results and compare our model with that of [12].

2. Related Work

Multi-target tracking has its roots in radar applications, beginning with the well-known multi-hypothesis tracker (MHT) [3]. The MHT handles ambiguities in data association by propagating many hypotheses until they can be resolved. However, the cost of propagating beliefs grows exponentially, and in practice the number of hypotheses must be pruned. If a correct hypothesis is mistakenly pruned, the MHT cannot recover. The joint probabilistic data association filter [16] is more efficient, as it propagates belief distributions for each target sequentially. More recently, particle filtering methods have been applied to MTT in video, including MTT extensions of Condensation [7], and by sequentially applying MCMC [9]. Sequential approaches are typically more efficient than the MHT, but are prone to failure because erroneous past estimations cannot be revisited when new information becomes available [14].

Recently, batch MTT methods have become increasingly popular, as they search the solution space of all time steps simultaneously. In [11], inference on a Bayesian Network joins path segments into tracks, however a robust procedure for creating path segments is required. Oh *et al.* proposed MCMC Data Association to partition a discrete set of detections into tracks in [14]. In [12], Yu *et al*. extend their work to overcome the one-to-one target to detection assumption, and introduce appearance models. Our work follows [12] in using batch MCMC for inference, but we formulate a new posterior distribution which models our constraints.

Our target application is the tracking of living cells. Previous work in this area has traditionally focused on fitting active shape models —contours in 2-D [13], surfaces in 3-D [6]— to cellular membranes. Modern approaches to estimating a cellular surface often employ level set methods, as in [10] where several rolling white blood cells are tracked. One drawback of active shape models is their lack of accuracy and robustness: a detector designed to find specific cell types such as the one we describe in Section 4.1 can often detect the cell more reliably and retain finer detail [2]. Another drawback is that the complexity of active shape models limits the number of tracked cells to a handful, whereas with a cell detector we are able to track over 100 cells.

More recent approaches to cell tracking include [5], where several types of human cells are tracked using mean-shift, and [15], where particle filters were used to track protein structures. However, sequential MTT methods are less appropriate for cellular tracking, as real-time processing is not required and batch methods can search the solution space over time to find a globally optimal solution.

3. Batch MTT Problem Formulation

In this section, we formulate our global objective in Section 3.1. Our constraints appear in Sections 3.2 and 3.3.

3.1. Problem Definition and Notations

Our goal is to find the most likely set of target states given the set of all detections, or measurements, for the entire sequence. To formalize this goal, let $\boldsymbol{\mathcal{X}}_t$ be the set of the states of all targets up to time t, and $\boldsymbol{\mathcal{Z}}_t$ be the set of all detections, up to time t. Our goal is to find:

$$\arg\max_{\boldsymbol{\mathcal{X}}_\mathrm{T}} p(\boldsymbol{\mathcal{X}}_\mathrm{T} \mid \boldsymbol{\mathcal{Z}}_\mathrm{T}) \qquad (1)$$

where T is the duration of the sequence. $\boldsymbol{\mathcal{X}}_t$ and $\boldsymbol{\mathcal{Z}}_t$ can be decomposed as follows.

- $\boldsymbol{\mathcal{X}}_t = \{\mathbf{X}_1 \dots \mathbf{X}_t\}$ where $\mathbf{X}_t$ is the set of the states of all targets for time t;
- $\mathbf{X}_t = \{X_t^1 .. X_t^i .. X_t^\mathcal{I}\}$, where X_t^i is the state of the i^th target at time t. For more efficient notation, we set the number of targets $\mathcal{I}$ to a constant but sufficient number (i.e. the total number of detections).
- $X_t^i = (M_t^i, O_t^i, R_t^i)$. Each state X_t^i is made of the target's kinematic parameters M, its appearance O, and a flag R to indicate if a target is present. With $\mathcal{I}$ fixed, switching R allows the number of targets to vary (though many target indexes i will never appear);

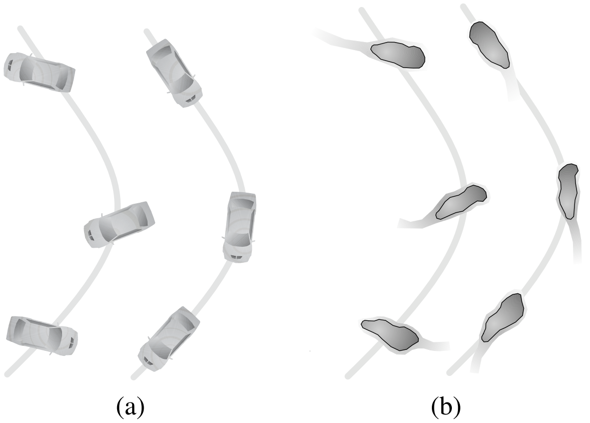


Figure 2. Our first constraint: *(a)* For many everyday objects, appearance and motion are related. *(b)* For the neuron nuclei we consider in our application this is also the case. Neuron nuclei tend to be elongated in the direction of their motion. We learn the joint distribution between appearance and motion, and exploit it in our formulation.

- $\boldsymbol{\mathcal{Z}}_t = \{\mathbf{Z}_1 \dots \mathbf{Z}_t\}$ where $\mathbf{Z}_t$ is the set of detections obtained at time t;
- $\mathbf{Z}_t = \{Z_t^1 .. Z_t^j .. Z_t^{\mathcal{J}_t}\}$, where Z_t^j represents the j^th measurement and $\mathcal{J}_t$ is the total number of measurements at time t;
- $Z_t^j = (L_t^j, A_t^j)$. Each measurement Z_t^j is composed of its location L and its appearance A.

A classic derivation gives:

$$p(\boldsymbol{\mathcal{X}}_\mathrm{T} \mid \boldsymbol{\mathcal{Z}}_\mathrm{T}) \propto \begin{array}{l} p(\mathbf{Z}_1 \mid \mathbf{X}_1)\, p(\mathbf{X}_1) \times \\ \prod_{t=2..\mathrm{T}} p(\mathbf{Z}_t \mid \mathbf{X}_t) p(\mathbf{X}_t \mid \mathbf{X}_{t-1}) \end{array} , \qquad (2)$$

under the standard assumptions that the states $\mathbf{X}_t$ follow a Markov process, and that the measurements $\mathbf{Z}_t$ are dependent only upon the current state $\mathbf{X}_t$, and conditionally independent of the other states given this state $\mathbf{X}_t$.

Given the measurements $\boldsymbol{\mathcal{Z}}_T$ provided by a detector, we want to find the targets $\boldsymbol{\mathcal{X}}_T$ that maximize the product of Eq. (2). In the following, we discuss each term of this product and introduce our two constraints.

3.2. The Observation Model and our First Constraint

By assuming that the detections are independent, the observation model term $p(\mathbf{Z}_t \mid \mathbf{X}_t)$ in Eq. (2) is the product:

$$p(\mathbf{Z}_t \mid \mathbf{X}_t) = \prod_j p(Z_t^j \mid \mathbf{X}_t) \, .$$

By summing over all the possible cases, each term $p(Z_t^j \mid \mathbf{X}_t)$ can be decomposed as:

$$\begin{array}{l} p(Z_t^j \mid \mathbf{X}_t) = p(Z_t^j \text{ is a false alarm}) + \\ \sum_i p(Z_t^j \mid X_t^i) p(\text{target } i \text{ created } Z_t^j) + \\ \sum_{i,i'} p(Z_t^j \mid X_t^i,\ X_t^{i'}) p(\text{targets } i \text{ and } i' \text{ created } Z_t^j) \ + \dots \end{array}$$

This sum can be expanded to consider detections arising from more than two targets. The terms: $p(Z_t^j$ is a false alarm), $p($target i created $Z_t^j)$, and $p($targets i and i' created $Z_t^j)$ are priors reflecting the quality of the target detector. Except for the first term, they appear only if the related targets are present, as defined by their R_t^i flags. In the following, we will only consider the terms where Z_t^j corresponds to zero or one target. A more complex model is required for the higher order terms.

Constraint #1: The movement of a target and its appearance are not independent. Our first constraint appears in the expression of the term $p(Z_t^j \mid X_t^i)$, illustrated in Fig. 2. If $R_t^j =$ present, we take $p(Z_t^j \mid X_t^i)$ to be:

$$\begin{aligned} & p(Z_t^j \mid X_t^i) = p(L_t^j, A_t^j \mid M_t^i, O_t^i, R_t^j) \propto \\ & p(L_t^j \mid \text{pos}(M_t^i))\; p(A_t^j \mid O_t^i)\; p(A_t^j \mid v(M_t^i)) = \\ & \mathcal{N}(\text{pos}(M_t^i)\,; L_t^j, \Sigma_L)\; \mathcal{N}(O_t^i; A_t^j,\ \Sigma_A)\; p(A_t^j | v(M_t^i)), \end{aligned} \tag{3}$$

where $\mathcal{N}(\cdot)$ is a Normal distribution, $\text{pos}(M_t^i)$ the position of target i at time t, and $v(M_t^i)$ is its velocity vector. Traditionally, authors assume independence between appearance and motion in $p(Z_t^j \mid X_t^i)$. Ignoring this assumption gives rise to a term modeling the correlation between detection appearance and target motion, $p(A_t^j \mid v(M_t^i))$.

For complex objects, modeling this correlation can be difficult. But when this correlation is isotropic, as for our neuron nuclei (the full model is given in Section 4.3), we can take

$$p(A_t^j \mid v(M_t^i)) = p(C(A_t^j, v(M_t^i)))\,, \tag{4}$$

where $C(A_t^j, v(M_t^i))$ is a vector made of two parts

$$C(A_t^j, v(M_t^i)) = [\,\widetilde{A_t^j},\ \|v(M_t^i)\|\,]^\top\,. \tag{5}$$

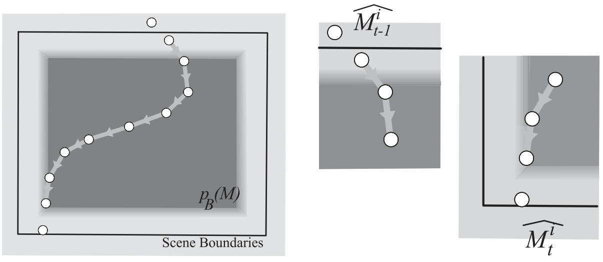
$\widetilde{A_t^j}$ is a shape descriptor of detection j after intensity normalization and a rotation that shifts the velocity vector $v(M_t^i)$ to a fixed direction. The second term is the speed of target i. Modeling a distribution over C is simple to estimate after reorienting all vectors to a unique direction.

3.3. The Motion Model and our Second Constraint

Constraint #2: The entrance and departure of a target should occur near a boundary of the scene. Our second constraint is illustrated Fig. 3 and appears in the motion model $p(\mathbf{X}_t \mid \mathbf{X}_{t-1})$. A rigorous derivation is relatively long but intuitive, and given below.

If we assume the targets move independently from each other, we have:

$$p(\mathbf{X}_t \mid \mathbf{X}_{t-1}) = \prod_i p(X_t^i \mid \mathbf{X}_{t-1}) = \prod_i p(X_t^i \mid X_{t-1}^i)\,.$$



(a) (b)

Figure 3. Our second constraint: (a) Entrances and departures of targets are more likely to occur near a boundary of the scene (a brighter background implies a higher likelihood). (b) To determine the likelihood of target arriving or departing at time t, we consider the target's motion-based prediction, $\widehat{M_{t-1}^i}$ for arrival and $\widehat{M_t^i}$ for departure.

Traditionally, $p(X_t^i \mid X_{t-1}^i)$ is limited to a motion model. In our case, it also depends on the presence of a target at t and $t-1$ as given by R_t^i and R_{t-1}^i,

$$\begin{aligned} & p(X_t^i \mid X_{t-1}^i) = p(M_t^i,\ O_t^i,\ R_t^i \mid M_{t-1}^i\ O_{t-1}^i,\ R_{t-1}^i) = \\ & p(M_t^i|M_{t-1}^i, R_{t-1}^i, R_t^i)\; p(O_t^i|O_{t-1}^i, R_{t-1}^i, R_t^i) \times \\ & p(R_t^i|M_{t-1}^i, R_{t-1}^i)\,, \end{aligned}$$

if we assume that the target kinematics and appearance at time t are independent conditionally on the kinematics and appearance at time $t-1$. Because of this conditionality, this assumption is compatible with our model in Eq. (3).

We must define each of these three terms for the four possible cases of object presence expressed by R_t^i and R_{t-1}^i. Each combination of R_t^i and R_{t-1}^i corresponds to some transition for the target, given in the table below.

	$R_{t-1}^i =$ present	$R_{t-1}^i =$ absent
$R_t^i =$ present	Stays in scene	Enters scene
$R_t^i =$ absent	Leaves scene	Absent from scene

For $p(M_t^i \mid M_{t-1}^i,\ R_{t-1}^i,\ R_t^i)$, the prediction term for the target kinematics, we get the following table:

	$R_{t-1}^i =$ present	$R_{t-1}^i =$ absent
$R_t^i =$ present	$\mathcal{N}(M_t^i;\ \widehat{M_t^i},\ \Sigma_M)$	$p_{\text{B}}(\widehat{M_{t-1}^i})$
$R_t^i =$ absent	λ_M	λ_M

In this table, $\widehat{M_t^i}$ denotes the prediction of M_t^i from the position and velocity in M_{t-1}^i, and Σ_M is estimated from training data. Most, if not all, motion models are reversible, allowing us to compute a back-prediction $\widehat{M_{t-1}^i}$ for M_{t-1}^i from the current position and velocity M_t^i. When a target appears, this back-prediction should be close to the scene boundary, and $p_{\text{B}}(M)$ is a distribution over the scene that favors this configuration. We use a simple piecewise uniform distribution to model it. λ_M is a uniform distribution stating that target location is irrelevant when it is absent.

For $p(O_t^i \mid O_{t-1}^i,\ R_{t-1}^i,\ R_t^i)$, the prediction term for the target appearance, we get the following table:

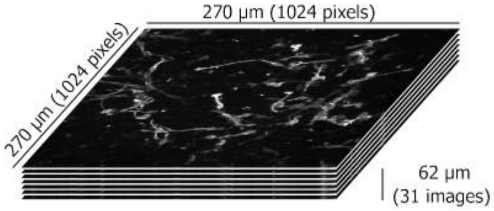


Figure 4. Our videomicroscopy data is composed of a time-series of 24 image stacks acquired from a 2-photon microscope over a 1.5 hour period. Each stack contains 1024 × 1024 × 31 pixels, which corresponds to a 270 × 270 × 62 μm volume of the brain. The data contains over 1700 nucleus detections corresponding to 101 individual neurons.

	$R_{t-1}^i =$ present	$R_{t-1}^i =$ absent
$R_t^i =$ present	$\mathcal{N}(O_t^i;\ \widehat{O_t^i},\ \Sigma_O)$	$\mathcal{N}(O_t^i;\ \overline{O},\ \Sigma_{\overline{O}})$
$R_t^i =$ absent	λ_O	λ_O

$\widehat{O_t^i}$ denotes the prediction for O_t^i. In this work, we simply take $\widehat{O_t^i} = O_{t-1}^i$. The covariance Σ_O is estimated from training data. λ_O is a uniform distribution stating that target appearance does not matter when it is absent. Note that, compared to the previous table, we cannot have a special constraint on the appearance O_t^i of entering targets, and we use a Normal distribution of mean $\overline{O}$ and covariance $\Sigma_{\overline{O}}$ over the appearance O_t^i.

For $p(R_t^i \mid M_{t-1}^i,\ R_{t-1}^i)$, we get the following table:

	$R_{t-1}^i =$ present	$R_{t-1}^i =$ absent
$R_t^i =$ present	$1 - p_{\text{B}}(\widehat{M_t^i})$	p_{e}
$R_t^i =$ absent	$p_{\text{B}}(\widehat{M_t^i})$	$1 - p_{\text{e}}$

where p_{e} is the probability that a target will appear.

This formulation inevitably introduces several parameters, each of which are simple covariances or distributions that can easily be directly computed from training data.

4. Tracking Migrating Neurons

Given the unusual nature of our data, we provide a brief description below. While most neurons are born during the embryonic and postnatal periods, it is now well accepted that some regions of the brain keep producing new neurons throughout adulthood. It is of great interest to understand how processes that govern the birth and development of neurons might be regulated, as this could lead to future treatments of degenerative disorders using adult neural stem cells. The automatic tracking of migrating neurons will help microbiologists quantify useful data such as cell morphology, speed, etc. when studying these processes.

Our collection process [1] begins by constructing a lentivirus vector. In vivo stereotaxic injection of the lentivirus in the subventricular zone of the brain causes newly born neurons to express Green Fluorescent Protein. A slice is then prepared for imaging.

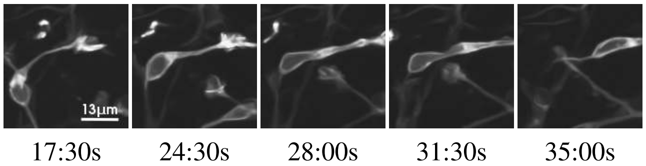


Figure 5. A migrating neuron. The nucleus extends a neurite and growth cone, which it uses as an anchor to pull itself forward. As the nucleus moves it elongates in the direction of motion.

It is then possible to visualize the neurons migrating and developing in their natural environment using two-photon time lapse microscopy (Ultima microscope; Prairie Technologies, Middleton, WI). The specimen is illuminated with $900nm$ light from a tunable pulsed Ti:sapphire femtosecond laser (Mai-Tai^{TM}; Spectra-Physics). Excitation light is focused onto the specimen using a $40x$, NA 0.8 water immersion objective (Olympus). Emitted light is collected in the epifluorescence configuration through a $680nm$ long-pass dichroic mirror and an infrared-blocking emission filter using a photomultiplier tube (Hamamatsu). Scanning and image acquisition were controlled using Prairieview software. A time-series of image stacks, depicted in Fig. 4, is produced and then denoised [4] and stabilized [8].

Migrating neurons move in a characteristic manner, depicted in Fig. 5. The nucleus extends a neurite, at the end of which is a growth cone, which the neuron uses as an anchor to pull itself forward. Because neuron cell bodies are highly deformable and irregular, they are difficult to track. For this reason, we perform tracking on the nucleus, which retains a more consistent shape and can be readily detected.

4.1. Nucleus Detection

We developed a detector which searches for nuclei as compact blobs with a bright surrounding structure based on a Laplacian of Gaussian filter to extract closed contours, as seen in Fig. 6. The observations in $\mathcal{Z}_T$ consist of the processed image stacks along with a set of detected nuclei, which may contain missed detections and false alarms from objects such as growth cones or dead cell matter.

4.2. Appearance and Kinematic Models

It is difficult to define a realistic motion model for the nuclei, as they often change their direction or speed abruptly. However, we use a simple constant velocity model, which seems sufficient in practice as our first constraint usually recovers abrupt changes in motion. Motion predictions ($\widehat{M_t^i}$) are made using forward and backward Kalman filters. Parameters of the Kalman filter are learned from labeled training data in a standard manner.

To model the appearance of the nucleus, we construct a descriptor vector consisting of two components: a shape descriptor and an intensity descriptor. The shape descriptor is a vector of spoke-lengths from the nucleus centroid to the detected contour taken at regular angular intervals,

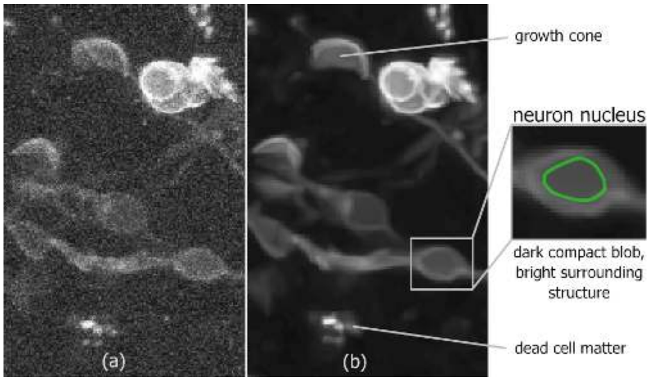


Figure 6. Preprocessing and nucleus detection. Raw image stacks *(a)* are stabilized and denoised *(b)*. A nucleus detector designed to find dark, compact structures surrounded by a bright region generates a set of detections. False detections may be generated by dead cell matter, neurites, and growth cones.

as depicted in Fig. 7. The intensity descriptor is simply a histogram of intensity values taken from the image patch defined by the detection contour.

4.3. Joint Appearance-Motion Model

To jointly represent motion and appearance of a nucleus, we define a joint descriptor $C(A_t^j,\ v(M_t^i))$ as given in Eq. (5). It is composed of the norm of the velocity $v(M_t^i)$ and $\overset{\frown}{A_t^j}$, a spoke-length shape descriptor after realigning A_t^j in the direction of motion, as seen in Fig. 7. We drop the intensity histogram from A_t^j in the joint descriptor, as it is not correlated with the motion. The spoke-lengths are normalized to correspond to a unit area. Since the shape descriptor is designed so that it may be quickly realigned, forming the joint descriptor is very efficient.

A Gaussian Mixture Model (GMM) is trained over the joint descriptor using shape and motion of labeled nuclei. In Fig. 8 we show how the GMM captures the interdependence of shape and motion by constructing nucleus prototypes from the means of the 5 mixture components; $mean_1$ is very round and nearly static, while $mean_5$ is elongated and moves rapidly.

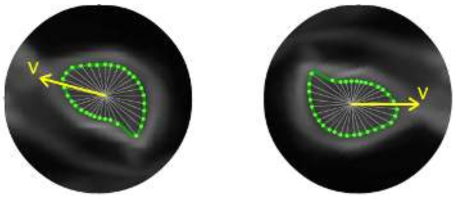


Figure 7. *(left)* Nucleus shape is modeled using spoke-length, defined as a vector of distances from the centroid to the contour taken at regular intervals. *(right)* The joint shape-motion model normalizes the spoke vector and aligns it in the 0° direction.

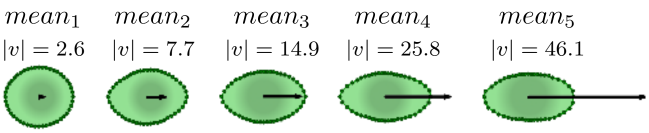


Figure 8. Prototype nuclei capture the interdependence of shape and motion. Above, nucleus prototypes were built from the centers of the 5 GMM mixture components trained to jointly model shape and motion. Note how the mixture components progress from a round, slow nucleus to an elongated, fast nucleus.

5. MCMC Inference

Estimating the *maximum a posteriori* (MAP) of Eq. (2) is an optimization problem over a very large solution space due to the size of our data. To efficiently search this space, we adopt an MCMC approach. As MCMC is well documented in a tracking context [12, 14, 9], we limit our discussion to a summary of our implementation.

MCMC is a general method to sample from an unknown distribution by constructing a Markov chain. We initialize the chain to an empty state, and generate new samples by proposing changes to the previous state via a randomly selected *mcmc move*. We define seven types of moves:

1. *birth* – add a new target to the scene,
2. *death* – remove an existing target from the scene,
3. *associate* – associate a new detection to a target,
4. *dissociate* – dissociate a detection from a target,
5. *merge* – merge two targets into a single target,
6. *split* – split one target into two targets,
7. *swap* – trade a target's existing detection for another.

A proposed state $\boldsymbol{\mathcal{X}}'_T$ is added to the chain according to an acceptance probability, otherwise the previous state $\boldsymbol{\mathcal{X}}_T$ is added. We can make the algorithm even more efficient by storing only the current state $\boldsymbol{\mathcal{X}}_T$ and state with the highest posterior $\overline{\boldsymbol{\mathcal{X}}_T}$. Fixing $\mathcal{I}$ (Sec. 3.1) fixes the number of terms in Eq. (2), and proposed posteriors can be quickly computed by updating only the terms which have changed. After the chain has N samples, the MAP state is given by $\overline{\boldsymbol{\mathcal{X}}_T}$.

6. Results

To test the performance of our proposed model, we ran experiments comparing with [12], and isolated each of our constraints to measure its influence. Our test sequence, presented in Fig. 1, contains over 1700 detections over 24 time steps, corresponding to 101 neurons. A manually annotated ground truth is used for evaluation. To objectively evaluate performance, we propose a very strict metric, similar to one given in [14]. The Association Recognition Rate is given by $ARR = |CA|/|AA| \times 100\%$. CA is the set of correct associations defined for each actual target path as the detections belonging to the estimated path which best matches. AA is the set of total associations. Estimated paths may only be matched to a single ground truth path.

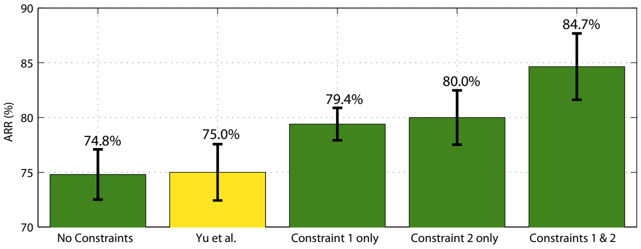


Figure 9. Comparisons of our approach with different combinations of the constraints and the method of Yu *et al.* [12] over 20 trials. Each constraint yields a ≈5% improvement.

Fig. 9 shows the mean ARR results over 100 trials, 20 trials per method. Our implementation of [12] uses appearance and motion models given in Section 3 instead of those in [12] to control factors influencing performance. The method of [12] performs similar to our approach with the constraints relaxed. Enabling each constraint gives a ≈5% gain. With both constraints together, we see an ≈10% improvement over our baseline and [12]. In Fig. 10, we plot the evolution of the ARR measure as MCMC converges, showing each method's best results. Note that the ARR settles higher with enforced constraints.

In Fig. 11, we show the extracted target tracks from a trial of our method comparing enforced and relaxed constraints. Our method suffered far fewer errors with constraints enforced. The majority of the remaining errors result from a few disjoint but otherwise correct tracks.

7. Conclusion

We have introduced two general tracking constraints into a probabilistic MTT tracking formulation. Our results show that modeling the motion-appearance correlation and border constraints on videomicroscopy data of migrating neurons yields a ≈ 10% performance increase. However, due to their simple structure, it is relatively easy to model the motion-appearance correlation of neuron nuclei. Future work may explore models for more complex objects.

Acknowledgements

We thank A. Beyeler, C. Herzberg, and L. Lenherr for help with 2-photon microscopy, stereotaxic injection, and annotation. This work was supported by the SystemsX.ch consortium as an Interdisciplinary Pilot Project, the Brain Mind Institute (EPFL), and the Leenaards Foundation.

References

[1] A. Carleton, L. Petreanu, R. Lansford, A. Alarez-Buylla, and P.M. Lledo. Becoming a New Neuron in the Adult Olfactory Bulb. *Nature Neuroscience*, 6:507–518, 2003. 5

[2] B. Leibe, K. Schindler, and L. Van Gool. Coupled Detection and Trajectory Estimation for Multi-Object Tracking. In *ICCV'07*. 1, 3

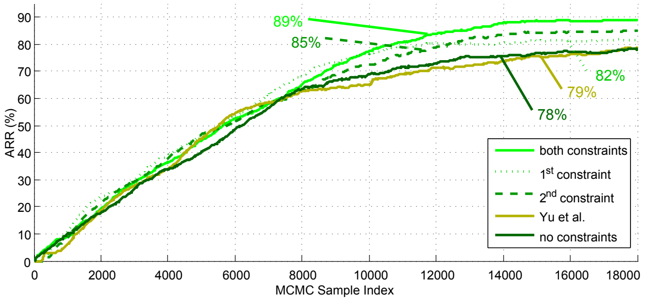


Figure 10. The evolution of ARR during MCMC optimization for the best trials of each method. Our model converges to a solution with a 14% performance gain when the constraints are enforced. The maximum ARR value is indicated for each trial.

[3] D. Reid. An Algorithm for Tracking Multiple Targets. *IEEE Trans. on Automatic Control*, 24(6):843–854, Dec. 1979. 2

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image Denoising by Sparse 3D Transform-Domain Collaborative Filtering. *IEEE Trans. on I. A.*, 16(8):2080–2095, 2007. 5

[5] O. Debeir, P. Van Ham, R. Kiss, and C. Decaestecker. Tracking of Migrating Cells Under Phase-Contrast Video Microscopy With Combined Mean-Shift Processes. *IEEE Trans. on Medical Imaging*, 24(6):697–711, 2005. 3

[6] A. Dufour, V. Shinin, S. Tajbakhsh, and N. Guillen-Aghion. Segmenting and Tracking Fluorescent Cells in Dynamic 3-D Microscopy with Coupled Active Surfaces. *IEEE Trans. on Image Processing*, 14(9):1396–1410, 2005. 3

[7] J. MacCormick and A. Blake. A Probabilistic Exclusion Principle for Tracking Multiple Objects. *IJCV*, 39(1):57–71, 2000. 2

[8] J.M. Odobez and P. Bouthemy. Robust Multiresolution Estimation of Parametric Motion Models. *JVCI*, 6(4):348–365, 1995. 5

[9] K. Smith, D. Gatica-Perez, and J.M. Odobez. Using Particles to Track Varying Numbers of Objects. In *CVPR'05*. 1, 2, 6

[10] D. Mukherjee, N. Ray, and S. Acton. Level Set Analysis for Leukocyte Detection and Tracking. *IEEE Trans. on Image Processing*, 13(4):562–572, 2004. 3

[11] P. Nillius, J. Sullivan, and S. Carlsson. Multi-Target Tracking - Linking Identities using Bayesian Network Inference. In *CVPR'06*. 2

[12] Q. Yu, G. Medioni, and I. Cohen. Multiple Target Tracking Using Spatio-Temporal Markov Chain Monte Carlo Data Association. In *CVPR'07*. 1, 2, 3, 6, 7

[13] N. Ray, S. Acton, and K. Ley. Tracking Leukocytes In Vivo with Shape and Size Constrained Active Contours. *IEEE Trans. on Medical Imaging*, 21(10):1222–1235, 2002. 3

[14] S. Oh, S. Russell, and S. Sastry. Markov Chain Monte Carlo Data Association for General Multiple-Target Tracking Problems. In *CDC'04*. 2, 3, 6

[15] G. Wen, J. Gao, and K. Luby-Phelps. Multiple Interacting Subcellular Structure Tracking by Sequential Monte Carlo Method. In *BIBM'07*. 3

[16] Y. Bar-Shalom and T. Fortmann. *Tracking and Data Association*. Academic Press, San Diego, CA, 1988. 2

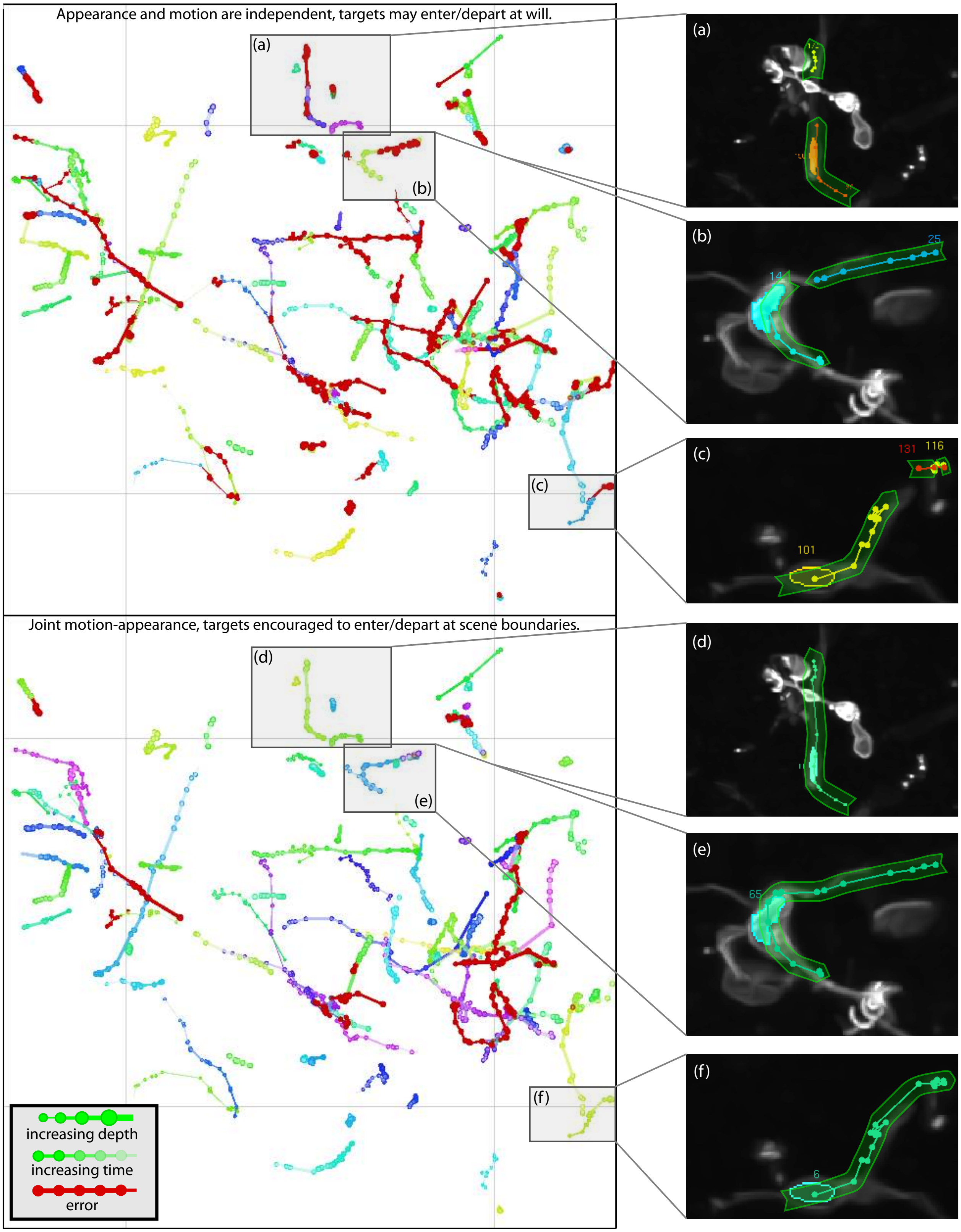


Figure 11. Top: Recovered tracks without constraints. Bottom: Recovered tracks using our two constraints. Errors are highlighted in red. *(a)* A nucleus accelerating quickly causes the unconstrained tracker to split the path. *(d)* Enforcing the constraints results in a correctly estimated track. *(b)* An unpredicted turn causes the tracker to fail. *(e)* Constraint 1 helps predict the turn based on the elongation of the nucleus. *(c)* A target track is segmented by unlikely entrances and exits. *(f)* Constraint 2 forces the target to exit near the scene boundary. More results, including video, are available at `http://cvlab.epfl.ch/~ksmith/`.