2018 24th International Conference on Pattern Recognition (ICPR)
Beijing, China, August 20-24, 2018

Multiple Mice Tracking: Occlusions Disentanglement using a Gaussian Mixture Model

Ario Sadafi[1], Vasiliki-Maria Katsageorgiou[1], Huiping Huang[2], Francesco Papaleo[2], Vittorio Murino[1,3], Diego Sona[1,4]

[1]Pattern Analysis and Computer Vision (PAVIS), Istituto Italiano di Tecnologia, Genova, Italy.
[2]Neuroscience and Brain Technologies (NBT), Istituto Italiano di Tecnologia, Genova, Italy.
[3]Department of Computer Science, University of Verona, Verona, Italy.
[4]Neuroinformatics Lab. (NILab), Fondazione Bruno Kessler, Trento, Italy.
Contact: name.surname@iit.it

***Abstract*—Mouse models play an important role in preclinical research and drug discovery for human diseases. The fact that mice are a social species partaking in social interactions of high degree facilitates the study of diseases characterized by social alterations. Hence, robust animal tracking is of great importance in order to build tools capable of automatically analyzing social behavioral interactions of multiple mice. However, the presence of occlusions is a major problem in multiple mice tracking. To deal with this problem, we present here a tracking algorithm based on Kalman filter and Gaussian Mixture Modeling. Specifically, Kalman tracking is used to track the mice and when occlusions happen, we fit 2D Gaussian distributions to separate mouse blobs. This helps us to prevent mice identity swaps as it is an important feature for accurate behavior analysis. As the results of our experiments show, the proposed algorithm results in much fewer identity swaps than other state of the art algorithms.**

I. Introduction

Mice are widely used by biologists due to their low cost, and because their biologic, genetic and behavior traits are highly similar to those of humans. This feature, along with the easiness in breeding, encourage scientists to create various models of different neurological diseases characterized by behavioral alterations. Hence, genetically modified mice are extensively used as a mean to reveal links between genes and behavior.

To study rodents' behavior, biologists usually rely on the manual inspection and annotation of video recordings in order to define relevant behavioral modules. This is, however, a very time-consuming task. Due to this, experiments are usually limited to few minutes (usually less than 15 minutes), while many behavioral patterns require much longer timespan to reveal.

The interest in automatic rodent behavior analysis is witnessed by the increasing number of publications appearing in the field, with tracking of mice being the first step in any proposed automatic system. Robust tracking throughout long experiments is vital for reliable analysis of mouse behavior, especially when mouse identity preservation is relevant because influenced by drugs or with different genetic background or alterations affecting their behavior. In this case identity swaps between mice can lead to incorrect behavior analysis and conclusions.

Mice are social species that tend to exhibit repetitive complex interactions among each other, resulting in frequent occlusions, hence making tracking a difficult task. Considering that in many experiments mice are not labeled in any way (RFID, color, etc.), to preserve as far as possible their natural behavior, reliable tracking is even more complicated.

Various tracking systems have been developed by the computer vision community to support scientists in monitoring mice behavior, employing different approaches ranging from Markov random fields [4] and Bayesian approaches [5] to Sift-Flow analysis [8]. Off the shelf software products are also available but they impose many restrictions. For instance EthoVision [7] is helpful for behavior analysis but requires the animals to have different fur colors. Thus, such a system cannot be used when animals are identical, unless color labeling on the animal's fur is used. This in turn might result in undesired affects in mice behavior.

An example of a label-less algorithm for multiple mice tracking was proposed by Giancardo et al. [2]. In this system, a watershed algorithm was extended with expectation-maximization to manage temporal consistency into multiple frames. Another approach to label-less tracking was proposed by Sadafi et al. [8]. This algorithm utilizes Sift-Flow analysis to address the problem of identity swaps in case of occlusions, but it can fail in longer and more complicated occlusion events.

In this paper we propose a novel label-less multiple mice tracking method based on the interactions between Kalman filter (KF) and Gaussian Mixture Modeling (GMM), which is used to prevent identity swaps in case of occlusions. In particular, GMM fitting is used to separate mouse blobs in occluded frames. Specifically, 2D Gaussian distributions are fitted on mice bodies throughout the series of the occluded frames, helping the tracker to preserve mice identities as well as acceptable estimation of head and tail positions.

All our experiments were conducted in dark environment such that mice were in their normal active condition and mice interactions were recorded using a thermal camera. Due to special settings, most of the common tracking algorithms fail in this scenario. Detection in tracking algorithms is mostly based on features like pattern and color of the objects.

978-1-5386-3788-3/18/$31.00 ©2018 IEEE

Considering that mice present no significant feature other than their deformable shape, a detector that segments them during the occlusion is the crucial part of the tracking. It can be unfeasible, therefore, to adapt works based on features' learning like the solution in [9] where they propose a deep model to learn the features of the subjects. Similarly, applying Multiple Instance Learning like in [10] does not allow to disentangle the occlusion in our specific set-up, due to lack of any significant visible features.

The rest of this paper is organized as follows: In section II the description of the designed method is discussed; Experiments and results are presented in section III and finally conclusion remarks are discussed in section IV.

II. METHODS

The proposed tracking system is based on four main components: a background subtraction module to identify mice blobs; a tracking algorithm based on Kalman filter; a segmentation component that identifies exact mice positions in case of occlusion; and a methodology for the detection and correction of possible head and tail inversions.

A. Background Subtraction & Blob Detection

The experiments were recorded using a thermal camera that measures the temperature instead of light intensity. This somehow simplifies the separation between foreground and background. After converting the thermal images to gray-scale levels and normalizing them according to minimum and maximum temperature, we determined an intensity threshold for the background subtraction. In order to automatize this process, a random set of frames $\{F^1, \ldots, F^K\}$ was selected and a model for foreground and background intensities was determined for each experiment separately as follows:

$$\begin{aligned}
\sigma_{ij}^k &= \begin{cases} 1 & \text{if} \quad F_{ij}^k - \bar{F} > 0 \\ 0 & \text{otherwise} \end{cases} \\
\eta &= \frac{\sum_k \sum_{ij} F_{ij}^k \times \sigma_{ij}^k}{\sum_k \sum_{ij} \sigma_{ij}^k} \\
\beta &= \frac{\sum_k \sum_{ij} F_{ij}^k \left(1 - \sigma_{ij}^k\right)}{\sum_k \sum_{ij} \left(1 - \sigma_{ij}^k\right)}
\end{aligned}$$

where $\bar{F}$ is the average temperature over all K frames over all pixels, and β and η are average background and foreground temperatures respectively. Calculating the threshold with a random set of frames give more robustness to statistics, but practically one frame ($K = 1$) is enough. Using this range of temperatures we defined a practical way to determine a threshold τ separating the foreground from the background based on the following formulation:

$$\tau = \gamma\eta * (1 - \gamma)\beta$$

and we found that a good choice for all experiments is to fix γ in the range $[0.8, 0.85]$.

After thresholding, morphological operations like image opening and image closing with different structure element sizes were performed to obtain smoother and better shaped blobs [8].

B. Kalman-based Tracking

Existing mathematical models can be used to define the trajectory of a moving object. Among them, Kalman filter is a dynamic model widely used to model moving objects. In our case, mice are assumed to be moving with constant acceleration such that a Kalman filter with this constraint is used to predict new mice locations based on their previous positions. Predicted points are then assigned to each mouse's track using the Munkres assignment algorithm [3]. Euclidean distance between the projected positions and the detected blobs is used to define the cost matrix. Finally, Kalman filter is corrected and parameters are fine tuned to fit the actual movement.

For better robust tracking and preservation of mice identities, the proposed algorithm does not rely only on Kalman filter and Munkres assignment but, as will be explained next, during occlusions, mice identities are preserved by fitting Gaussian Mixture Models (GMM).

C. Occlusion Segmentation

The most difficult problem in multiple animal tracking is the existence of occlusions. An occlusion occurs when mice are in very close proximity, resulting in detection of a single blob, corresponding to the area covered by more than one mouse. We have to deal with this problem that might present different levels of complexity. For this reason we designed a two-stage algorithm able to adapt to the occlusion complexity, which is activated only for some of the frames where occlusion happens. This allows to properly detect the mice and preserve their correct identities through these frames. The second stage is only activated if after the first stage enough number of blobs are not detected.

1) First stage of occlusion segmentation: The first stage is based on the work of Sadafi et al. [8], which uses mathematical morphology on grayscale images. The occluded image is eroded and used as a mask for the morphological gray-scale image reconstruction, which returns the union of connected components in the original image [1]. Then the reconstruction of the complemented dilation is used to reduce the noise. Finally, the blobs are extracted. Refer to [8] for further details on this method.

With this approach, quick operations are carried out in the occluded images and if the occlusion is not so complex, blobs separation is achieved and there is no need for the second stage of occlusion segmentation that is more computationally demanding.

2) Second stage of occlusion segmentation: In case of complex occlusions, morphological operations are not enough to separate mouse blobs. To deal with this, we introduce an approach based on Gaussian Mixture Model (GMM). Specifically, each mouse body is considered to be formed by a

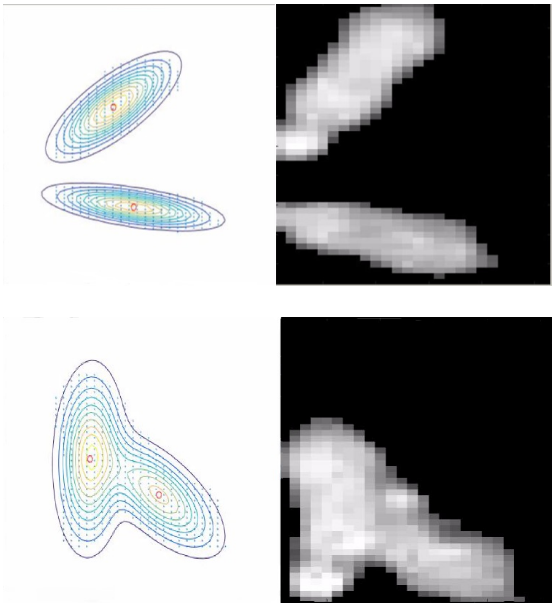

Fig. 1. Examples of two different frames showing how Gaussian distributions are fitted on mice bodies

set of pixels whose gray intensity spatially has a 2D Gaussian distribution. This results in fitting a number of Gaussian distributions equal to the number of mice (see Figure 1 for an illustration).

Fitting is an iterative process, which terminates either by reaching the error rate below a defined threshold or after a fixed number of iterations [6]. In our experiments we set the termination error rate to 10^{-10} and the maximum allowed number of iterations to 200. We observed that when the iterative algorithm was going beyond this amount of iterations the original frame was so complex that a better choice was to drop it as the probability for a wrong segmentation is very high.

Moreover, mean and covariance of the Gaussian distributions of the same mouse in two contiguous frames are assumed to be very similar. Hence, the body centroid of the last known blob estimated in the previous frame is used as the center of mass in the new frame. This allows to converge faster and maintain mice identities. In any case, after a successful convergence, the obtained distributions represent the mouse bodies in the scene, and they are assigned to each mouse track using the Munkres algorithm. As aforementioned, this algorithm requires an allocation cost matrix. In this case, the Kullback–Leibler (KL) divergence is used as a similarity measure to assign the distributions of the previous frame (i.e., the mouse identities) to the newly detected ones. Having two Gaussian distributions f and g, KL divergence is defined in a closed form as follows:

$$\begin{aligned}\delta(f,g) \quad = \quad & \frac{1}{2}\left[\log\frac{|\Sigma_g|}{|\Sigma_f|} + Tr[\Sigma_g^{-1}\Sigma_f] - d + \right.\\ & \left.(\mu_g - \mu_f)^T\Sigma_g^{-1}(\mu_g - \mu_f)\right]\end{aligned}$$

where Σ is the covariance matrix, μ is the mean vector and d is the dimension of the distributions. To make sure that the covariance matrix is invertible, it is regularized adding a small number to the diagonal elements. Since KL divergence is not symmetric, its symmetrized version is considered, computing the mean of the two KL divergences obtained inverting the order of the Gaussians.

D. Head and Tail Detection

Along with the center of mass for each mouse body, head and tail positions are important descriptive features used in behavior analysis. These points help neuroscientists distinguish behaviors like "Nose to Nose sniffing", "Nose to Tail", etc., or to detect more complex behavioral patterns. Similarly to mouse body tracking, two scenarios are assumed for updating head and tail positions depending on the existence or the absence of occlusion.

1) Detection in absence of occlusion: In the easiest case, when mice are well separated, head and tail positions are updated using the angle (θ) of the major axis of the corresponding blob and its length (l). Indeed, possible positions of head and tail correspond to the major axis endpoints computed for each mouse as follows:

$$P = \mu \pm \frac{1}{2}\, l \begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}$$

where μ is the column vector containing the x and y coordinates of the center point of the mouse's body. The correct head and tail assignment is then computed according to their distance from the head and tail positions in the previous frame.

2) Detection in case of occlusion: In case of occlusion, the fitted Gaussian distributions help us to identify the head and tail positions. Assuming μ the center of mass, and $\hat{v}$ and $\hat{\lambda}$ the principal eigenvector and the corresponding eigenvalue respectively, computed from the covariance matrix of a given Gaussian distribution associated with a mouse, the head and tail positions are obtained according to the following relation:

$$\begin{aligned}\theta \quad &= \quad \arctan\left(\frac{\hat{v}(y)}{\hat{v}(x)}\right)\\ P \quad &= \quad \mu + \sqrt{\hat{\lambda}}\begin{bmatrix}\sin\left(\theta \pm \frac{\pi}{2}\right)\\ \cos\left(\theta \pm \frac{\pi}{2}\right)\end{bmatrix}\end{aligned}$$

Again, the correct head and tail assignment is determined with the Munkres algorithm, and the Euclidean distance from the head and tail positions in the previous frame is used to define the cost matrix.

E. Head and Tail Inversion Detection

We assume that mice move forward and not backwards, which is plausible since they naturally only go forward and rarely backward in an open arena. To detect a possible inversion of head and tail positions, for every frame and after each position update the following vectors are computed for each mouse

$$\vec{d} = \mu(t) - \mu(t-1)$$
$$\vec{o} = P_{Head}(t) - P_{Tail}(t)$$

where the vector $\vec{d}$ describes the direction of movement of the center of mass in two consecutive frames, while $\vec{o}$ is the orientation of the body. The angle α between these two vectors is then computed and used to prevent head-tail inversions.

This check is performed with a voting mechanism, such that every time α becomes bigger than 90^{o} an alarm is raised. After a number of consecutive alarms (in our experiments we used 10), the possibility for an inversion is high enough so that head and tail positions of the mouse are inverted again by the system to correct the error. Moreover if the magnitude of $\vec{d}$ is smaller than the predefined threshold, the mouse movement is not significant and will not be used for detection of a possible inversion.

III. Experiments & Results

To evaluate the performance of the proposed tracking algorithm, we used six different video recordings (each lasting at least one hour) with two mice interacting in an arena. The tracking results of the algorithm proposed in [2], manually corrected by behavioral scientists, were used as ground truth to evaluate the performance of our tracker.

The proposed algorithm was implemented in Matlab® 2016 and was run on an Intel® Xeon® core at 3.20 GHz.

A. Experimental Setup

Experiments were carried out in an $50cm \times 50cm$ open arena with a thermal camera mounted about $1.5m$ above it. Mice were free to move and interact in the arena while their interactions were recorded with a FLIR A315 thermal camera having a resolution of 320×240 at the rate of 30 frames per second. The recordings were carried out in almost complete darkness since this is less stressful for the animals and they are in their most active phase. Animals were recorded without any tagging to prevent any alteration to their behavior and interactions.

B. Results

We compared our method versus two other previously proposed tracking systems, one based on temporal watershed [2], and another one based on Sift-Flow analysis [8]. All three trackers were run on the same datasets and their results were evaluated on the ground truth by counting the number of mice-inversions observed by each one of them.

Behavior analysis, which is a next step to mice tracking, strongly relies on correct mouse identities across the video so counting the total number of mice-inversions is a good measurement of the accuracy of the tracker. Each inversion introduces an error and requires manual correction by the scientists proofreading the tracking result. The lower these inversions get, the faster and accurate the correction of results become.

All trackers were run for a total of 6 hours 52 minutes 8 seconds of video recordings. The proposed algorithm resulted in only 44 identity swaps in total. In comparison, state-of-the-art trackers resulted in many more swaps. For example, the tracker proposed in [8] had a total of 126 identity swaps, the one proposed here resulted in about 65% less mice identity swaps. The detailed results are shown in Table I. On average, using the proposed tracking algorithm, we observe an identity swap every 9 minutes which is dramatically less than in the aforementioned previously proposed tracking systems. Additionally, accurate head and tail detection and auto-recovery of inversions makes the manual correction less cumbersome.

The tracker runs in real-time but during execution of the designed two stage algorithm for occlusions segmentation it requires more time and frame rate drops. After disentanglement of the occlusion tracking speeds up to real-time again.

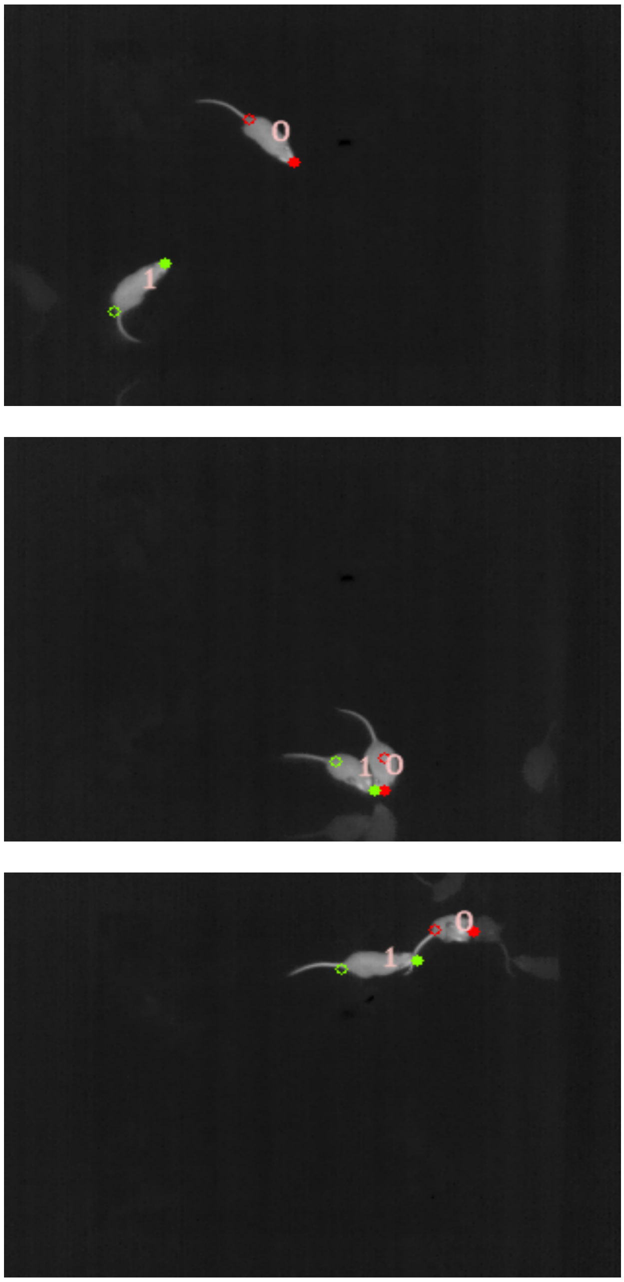

Fig. 2. Examples visualizing the proposed tracking algorithm's output.

TABLE I

TRACKING RESULTS OF THE PROPOSED METHOD COMPARED TO THOSE OF THE TRACKER PROPOSED BY GIANCARDO ET AL. [2] AND THE ONE BY SADAFI ET AL. [8]. TOTAL NUMBER OF OBSERVED IDENTITY SWAPS IS REPORTED FOR EACH EXPERIMENT AND TRACKING ALGORITHM.

	Video Length	[2]	[8]	Proposed Method
Exp. 1	1:00:54	27	17	5
Exp. 2	1:09:07	45	23	7
Exp. 3	1:11:45	54	29	12
Exp. 4	1:15:32	49	21	8
Exp. 5	1:09:11	75	28	9
Exp. 6	1:05:39	14	8	3
TOTAL	6:52:08	264	126	**44**

IV. CONCLUSIONS

Reliable localization of mouse bodies during an experiment is the first step for mouse behavior analysis. In this paper we addressed all issues related to mice tracking in an open arena without any labeling of their fur. The similarity between the mice makes the tracking problem complex due to the many occlusions that appear in an experiment. Moreover, mice are deformable, making even more complex the task. As shown by the results of our experiments, the proposed tracking algorithm is robust enough and can be used in long lasting experiments for mouse behavior analysis. Fitting GMMs during a series of occluded frames resulted in significantly decreased number of identity swaps, which in turn eliminates the time required by the scientists to manually correct the data for accurate behavior analysis. Finally, the tracker works with thermal camera which nowadays is widely produced and can be purchased at an affordable price.

The designed method will be made public and distributed in a software package to be used for various behavior analysis experiments. In future, by developing more complicated background subtraction methods, the designed method will be extended to work with various types of modalities such as infrared or normal light cameras.

REFERENCES

[1] L. Vincent, "Morphological grayscale reconstruction in image analysis: applications and efficient algorithms," *in IEEE Transactions on Image Processing*, vol. 2, no. 2, pp. 176-201, Apr 1993.

[2] L. Giancardo, D. Sona, H. Huang, S. Sannino, F. Manago, D. Scheggia, F. Papaleo, and V. Murino, "Automatic Visual Tracking and Social Behaviour Analysis with Multiple Mice", *PLoS ONE*, 8(9), September 2013.

[3] J. Munkres, "Algorithms for the assignment and transportation problems", *Journal of the society for industrial and applied mathematics*, 5.1 (1957): 32-38.

[4] K. Branson and S. Belongie, "Tracking multiple mouse contours (without too many samples)", *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, pp. 1039-1046 vol. 1.

[5] C. Kervrann and F. Heitz, "A hierarchical Markov modeling approach for the segmentation and tracking of deformable shapes", *Graphical Models and Image Processing*, 60.3 (1998): 173-195.

[6] G. McLachlan and D. Peel, "Finite Mixture Models", *Wiley Series in Probability and Statistics*, New York, 2000.

[7] L.P.J.J. Noldus, A.J. Spink, and R.A.J. Tegelenbosch, "EthoVision: a versatile video tracking system for automation of behavioral experiments", *Behavior Research Methods, Instruments, & Computers*, (2001) 33: 398.

[8] A. Sadafi, V.-M. Katsageorgiou, H. Huang, F. Papaleo, V. Murino, and D. Sona, "Multiple Mice Tracking and Segmentation through SIFT Flow Analysis", *Visual observation and analysis of Vertebrate And Insect Behavior 2016 in 23rd International Conference on Pattern Recognition (ICPR 2016)*, 2016.

[9] N. Wang and D. Y. Yeung, "Learning a Deep Compact Image Representation for Visual Tracking", *Advances in Neural Information Processing Systems 26*, 2013

[10] B. Babenko, M. Yang, and S. Belongie, "Visual tracking with online multiple instance learning", *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2009