

# Multiple Mice Tracking using a Combination of Particle Filter and K-Means

Wesley Nunes Gonçalves, João Bosco Oliveira Monteiro,  
Jonathan de Andrade Silva, Bruno Brandoli Machado, Hemerson Pistori  
Dom Bosco Catholic University  
Research Group in Engineering and Computing  
Av. Tamandaré, 6000, Jardim Seminário, 79117-900, Campo Grande, MS, Brazil

Valguima Odakura  
TecSinapse - Tecnologia da Informação Ltda, R&D Department  
Rua Julio Verne, 70, casa 02, Jardim Hípico, 04725-060  
São Paulo, SP, Brazil  
{wnunes, jsilva, bmachado}@acad.ucdb.br, joaobmonteiro@gmail.com,  
valguima.odakura@tecsinapse.com.br, pistori@ucdb.br

## Abstract

*This paper presents a new approach to multiple objects tracking that combines particle filters and k-means. The approach has been tested under an important real world situation, related to pharmacological development, which has also proved to serve as an interesting ground-truth dataset provider for the evaluation of tracking algorithms. The obtained results are then compared to other models. The promising results of these experiments are presented.*

## 1 Introduction

Tracking multiple objects is an essential task in many computer vision applications, such as tracking multiple animals to automate experiments with animals of laboratory [3], social insect tracking application [9, 11], monitoring people for tracking players [14] and identification of 3-d human motion [4]. Basically, tracking multiple objects consists in determining which and how many objects in the scene will be tracked and then locating each one of them in consecutive frames. This task receives a special attention in computer vision; however, it is still an open and challenging problem due to the variation in the conditions of lighting, presence of noise and potentially ambiguous conditions, such as occlusion of multiple similar objects.

A variety of algorithms for tracking multiple objects has been proposed. Many of them are based on predictive filters, in order to achieve robustness to occlusion and real time performance. Predictive filters use a stochastic model

of the tracked objects dynamics in order to propagate the state of the system, from frame to frame. The predicted state is combined with information derived from an observation model, to estimate the current state of the system [6].

Computer vision has been used to automate scientific experiments in laboratory to test new medicines on mice. Noldus *et al* [17] describes how video-based tracking allows researchers to study the animal behaviour in a reliable and consistent way for a long period of time. The researcher is interested in how the behaviour patterns change during exposure to pharmacological agents. In general, these activities are applied to large scale, using multiple doses and different animals, in order to ensure reliable statistics.

Tracking multiple mice is an interesting task because mice are deformable and, in some cases, indistinguishable objects. The animal behaviour along these experiments may be recorded, automatically or semi-automatically, in video. During the experiment, the researcher observes the animal and records information about actions related to specific behaviour of interest. Video-based tracking can aggregate automatic pattern recognition, applied to the captured images, to extract measurements from animal behaviour.

In particular, for measuring motion behaviour (e.g. track length, velocity, acceleration), unusual movements that happen after long periods of inactivity and the ones that occur during many hours or days (daily behaviour analysis, for example), video-based tracking is appropriated once a human observer is unable to perform these tasks efficiently [17]. In addition, automatic tracking does not suffer from fatigue or distraction and this approach eliminates the subjectivity when various observers classify the same action performed

by the animal.

In this paper, we propose an efficient approach based on particle filter and k-means to track multiple objects, which has been applied to the problem of mice tracking during pharmacological experiments. The paper presents a dynamic model, inspired in the random walk [1] motion model. The observation model was based on blobs of the segmented image. In order to deal with situations where mice are in contact or partial occlusions occur the k-means algorithm [7, 10] is used. The tracker was analyzed using image shots in situations where the mice are with and without contact among themselves. The particle filter performance was compared to that of human specialists, in the open-field experiment. This experiment proved to be an interesting way to compare tracking algorithms, as it provides ground truth data related to the mice spatial position over a five minutes observation section. Our proposal demonstrated to be robust and efficient, when compared to human performance.

The paper is organized as follows. Section 2 presents related work that applies the particle filter to multiple objects tracking. Section 3 briefly reviews particle filters. In Section 4 the dynamics model used in this work is explained. Section 5 explains how the k-means algorithm was applied to the observation model. The experiments performed and the results of the particle filter applied to mice tracking are described in the Section 6. Finally, the conclusion and future work are discussed.

## 2 Previous Work

Particle filter has been extensively used in tracking multiple objects [8, 12]. In this context, the particle filter provides the development of robust tools for tracking because this technique is not limited to linear systems and does not require Gaussian noise, as it happens with the Kalman filter in its basic formulation. The particle filter usually employs visual (e.g. color, texture), geometric (e.g. contours, shape) and motion features for tracking many objects [12, 14, 8].

In [12] a robust framework for tracking rigid and non-rigid objects was developed. The particle filter implementation was based on visual and geometric features. The framework was evaluated in two experiments, a book boundary tracking and a moving leave, in situations that other algorithms may fail.

Branson and Belongie [3] propose an algorithm for tracking the contours of multiple mice. The experiments used frames captured from a side view of the cage where the mice are inserted and stay in constant occlusion. Their algorithm that combines multiple blob detection and a contour tracker demonstrates an acceptable performance, mainly in occlusion. In [11], a particle filter tracker is applied to track fishes. This system is capable of tracking a quantity of fish

modeled as ellipses that may vary over time. Tracking is based on the multitarget likelihood function and a set with 2000 particles. The tracker showed good performance in real world situations, obtaining correct classification rate of 81%.

A feature commonly explored in particle filters is the object colour. In [13] the colour distribution is integrated with the particle filter and takes place in the observation model, providing rotation and scale invariance and robustness to partial occlusion. The implemented algorithm is compared with mean shift algorithm. Many new extensions for the particle filter are being developed, as in [16], who proposed a new algorithm called mean shift embedded particle filter. This new extension takes advantages of both algorithms: particle filter and mean shift. Shan, Wei and Ojardias applied the new extension to hand tracking. The hand tracking problem has also been used in [2] to validate a new approach to handle very high dimensional search spaces that, in the basic approach, would require a prohibitive number of particles to be appropriately modeled.

## 3 Particle Filter

Various algorithms for tracking multiple objects present some important limitations, like considering that random variables are Gaussians and the dynamics and observation model are linear. In some applications, the linearity restriction does not apply and the algorithms cannot estimate, correctly, the object states over time. A possible approach for dealing with this limitation is to employ a non parametric representation based on samples or particles.

In the particle filter framework, the state of the system, in time  $k$ , is usually represented as a multidimensional variable  $\vec{x}_k$  (the particle or sample), with an associated weight  $\omega_k$ . The particle filter performs three steps: prediction, updating and resampling. In the prediction step, the dynamics model  $f$  is applied to each particle:

$$\vec{x}_k = f(\vec{x}_{k-1}) + \vec{r}_{k-1}, \quad (1)$$

where  $\vec{r}_{k-1}$  models the prediction noise.

After prediction, an observation model is used:

$$\vec{y}_k = h(\vec{x}_k) + \vec{v}_k, \quad (2)$$

where  $h(\vec{x}_k)$  is the function that extracts indirect measure of the system and  $\vec{v}_k$  represents the noise associated with the observation model.

In computer vision, this step is usually associated with a segmentation and feature extraction module. The observation is then used to update the current state of the system. It means to update the particle weight  $\omega_k$ . The central idea consists in giving higher weights for particles that are closer to what was observed. In this article, the weight is

calculated using the Euclidean distance from the predicted to observed particle. It is assumed that particles with higher weight are more important for the distribution and better represent the system state.

A common problem with particle filter is that after some iteration the weights of most particles are very small, almost negligible. In order to eliminate particles which have light weights and to concentrate on particles with heavy weights, the final step, resampling method, is used. In the resampling, each particle is mapped in an interval  $[0, 1]$ . A number  $s$  is randomly chosen from a uniform distribution. Finally, the particles that correspond to the interval  $s$  are propagated to the next iteration. Heavier weight particles fit in a larger interval, raising their chance to survive for next interaction. For optimization, the resampling step may be performed just when a significant number of particles has its weight below a certain threshold [15].

After the resampling process, the filter provides a set of particles that represents the possible state of the system. Different strategies may be used to extract useful information from this set of particles [15]. In this work, the heaviest weight particle is chosen as the best representation of the system state.

#### 4 Dynamics Model

To obtain a robust particle filter, it is absolutely important to model the dynamic behaviour of the objects in the system. The mouse movement can be fit by random walk models that are probabilistic discrete step models that involve strong simplifications of a real mouse movement. It consists of a discrete series of displacement events separated by reorientation events [1].

The dynamics model inspired by random walk motion characterizes a random motion and can be described as:

$$f(\vec{x}_k) = \vec{x}_{k-1} + \vec{r}(\sigma) * \vec{V} \quad (3)$$

where  $\vec{r}(\sigma)$  is randomly generated through a Gaussian number generator with standard deviation  $\sigma$  and  $V$  is the velocity of the object.

This model is interesting for objects that do not possess a standard movement, as the mice. However, the parameters chosen should represent the dynamics of the object, as the velocity. These parameters should be carefully chosen once a high or low velocity can diminish the performance of the particle filter.

#### 5 Observation Model

To extract information about the problem in real world it is necessary an observation model. This model extracts the present state of the system based on colors, contours and

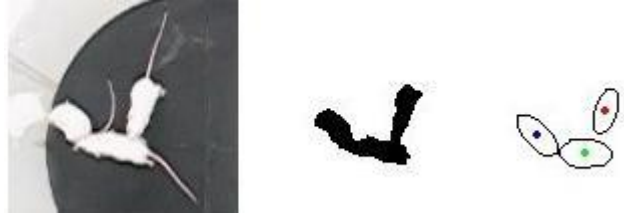


Figure 1. Steps of the K-means observation.

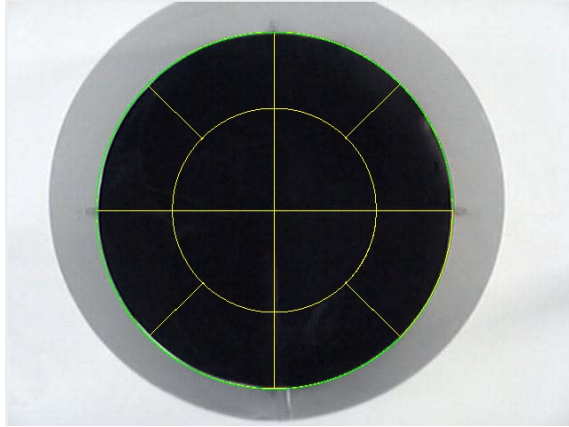
on the center of mass, in most cases. For our problem, the observation model uses the blobs of the segmented image to infer the center of mass and the other parameters of an ellipse that approximates the mice contour. This inference is easily carried out using an observation model based on blobs when the objects have no contact among themselves. The main problem for extracting the present system state is the constant contact between mice, as it is shown in Figure 1.

A solution is the use of the standard k-means algorithm [7, 10] to infer the parameters of the  $n$  objects in scene. To reduce the processing cost, the center of each object is initialized with its position in the previous frame. Each pixel marked as foreground, in the segmented image, is attributed to the closest center using the Euclidean distance from that pixel position to each center. Figure 1 shows an example of a segmented image.

After the attribution step, the center of each pixel group is re-calculated using the mean of the pixels positions. The attribution and the calculation step are repeated until it does not have a significant change,  $\tau$ , between the current center and the previous one. In this application, five iterations are necessary, on average, and 0.1 was experimentally used for  $\tau$ . The parameters of the ellipse are extracted using the pixels attributed to the center of each group.

#### 6 Experimental Results

For the experiments carried out with the particle filter, the state of the system has been treated as a multidimensional variable  $\vec{x}_k = [x y m_e m_i \theta]$ , that represents each object as an ellipse, i.e.,  $x$  and  $y$  are center point,  $m_e$  and  $m_i$  are major and minor axis, respectively, and  $\theta$  is inclination angle. The problem to be investigated is to track four mice in an experiment called open field [5]. This experiment is carried out in a cylindrical arena with acrylic walls with approximately 30 cm of height. The arena of the experiment is divided in twelve regions with the same area. The overhead view of the experiment with the twelve regions is shown in Figure 2. Each mouse has been associated with a different particle filter, all sharing the same observation and dynamic model.

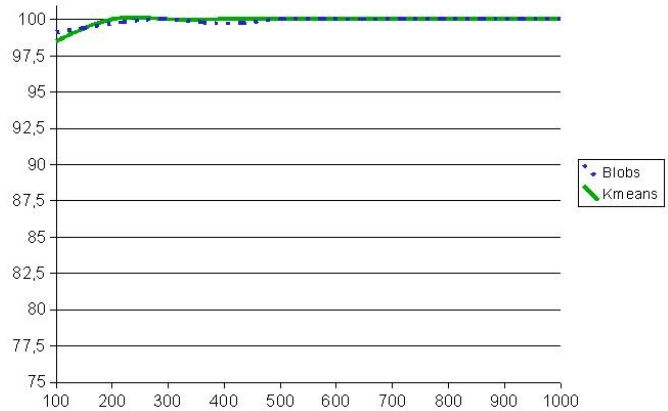


**Figure 2. Regions used in the experiments with mice.**

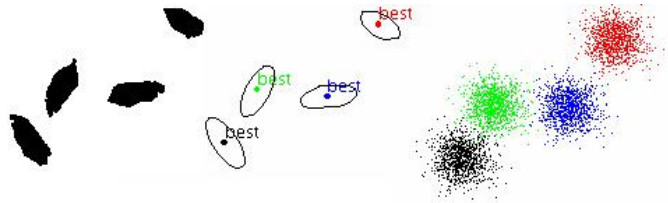
In order to evaluate the efficiency of the implemented particle filter a methodology based on regions is used. This methodology is the same used by human specialists when an automated system is not available. It associates a region to each mouse, in accordance to a visual approximation of its center of mass. The region supplied by tracking algorithm is compared to the ground truth region, thus providing a correct classification rate. A special situation occurs in images where the manual marking is not clearly found, i.e., situations in that the mouse is in a border region and the specialist is not certain about which region the mouse is found. In this situations the correct classification is designated when the region returned by the particle filter and by the manual marking are frontier regions and the difference between the centers of mass is around a threshold. In this study, the threshold used was the size of major axis of the ellipse that represents the mouse.

In our experiment, image shots, taken from a camera placed over the open field, have been used. Shots where the mice are in contact and in partial occlusion have been chosen. This image shots have been extracted using recorded videos from a digital camera Canon Powershot A80, with resolution of 320 x 240 pixels. Later, the frames have been segmented using background subtraction. All the tests have been performed using a computer with a processor P4 2.8GHz, 512MB of memory RAM and operational system Fedora Core 5.

The experiments were carried out with variations in the parameters of the filter. For the random walk model, the parameters tested were the standard deviation  $\sigma$  and the velocity in the X and Y direction. For the observation model, the tested parameters were a model based on blobs, a model based in the k-means algorithm and a combination of the previous models. The filter was evaluated in 54 images in



**Figure 3. Particles Number versus Correct Classification for Mice without Contact.**

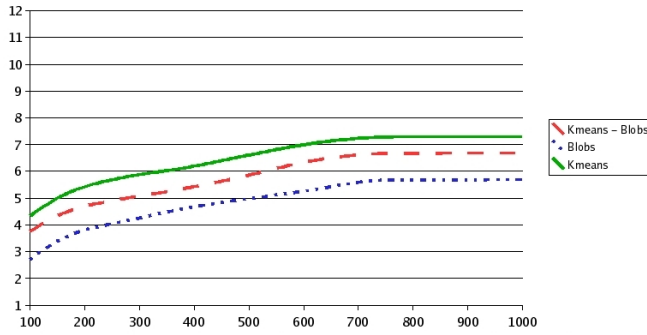


**Figure 4. Example of mice without contact.**

with 4 mice move separately, that is with no contact and 96 images in wich mice move as a cluster, in contact with others.

The results are presented in Figures 3, 5 and 6. In Figure 3, the graphic presents the changing in correct classification rate as the number of particles grows for images shots where the mice are not in contact. Only the mean result for parameters configuration is plotted. Both models, blobs and k-means, present classification rate above 97%, with just 100 particles. The graphic shows that the observation models based on blobs and k-means possess practically the same good performance when the mice move with no contact with others (example in Figure 4).

The relation of the growth in the number of particles to the runtime is presented in Figure 5. The observation model based on blobs has been faster in relation to the model based on k-means, but both presented a constant runtime after 700 particles. Due to the high runtime of the k-means model in Figure 5 and the equality in correct classification rate between the two models in situations where the mice move separately in Figure 3, experiments with a combination between the models were carried out. That combination was carried out in the following way, when the image does not



**Figure 5. Particle Number versus Runtime.**

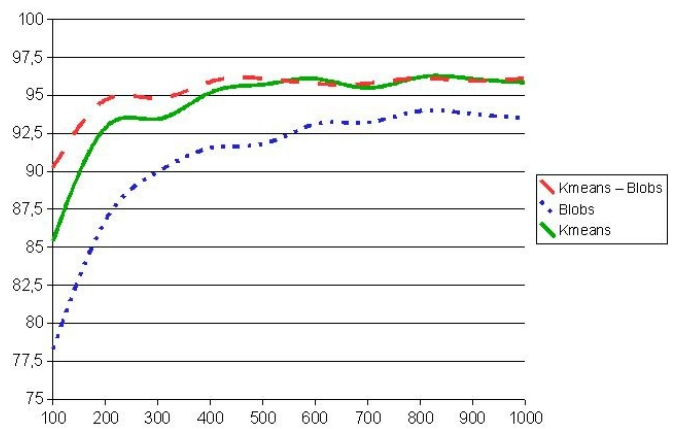
have mice interaction the observation based on blobs is utilized and when the image contains mice in contact the observation based on k-means is used. The system recognizes mice interaction when the quantity of blobs in the image is less than the quantity of mice tracked. That combination reduces runtime, shown in the Figure 5, and maintain the correct classification rate (Figure 6).

In Figure 6, the graphic presents the result for images shots with mice moving as a cluster. Tracking a cluster of mice, as illustrated in the Figure 7, is a difficult task, also for humans. For contact, the observation model based on blobs does not achieve a good performance. However, the model based on k-means and in the combination between the models resolve the problem, achieving great results. The model based on k-means with 400 particles achieved a mean of 95% accurate classification. In contrast to the first model, the model based on combination between k-means and blobs achieved a mean of 96% accurate classification. Although both techniques present similar correct classification rate, the combination of k-means and blobs outperformed the k-means, once it obtained a faster runtime (see Figure 5).

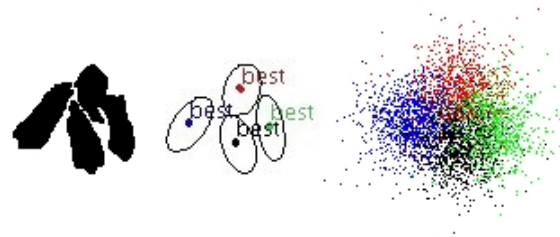
In Figure 8 it is shown the tracking results for three mice in a sequence of images where mice start moving separately, then they cluster together and afterwards they split again. For each situation it is presented the particles and the ellipse representing the mice. It is worth noting that even in the clustering situation presented the combination of particle filter and k-means, as proposed in this article, obtained accurate tracking results.

## 7 Conclusion and Future Work

This article showed an implementation of particle filter applied to the tracking of multiple mice in the open field experiment. Experiments have shown that an observation model based on the combination between k-means



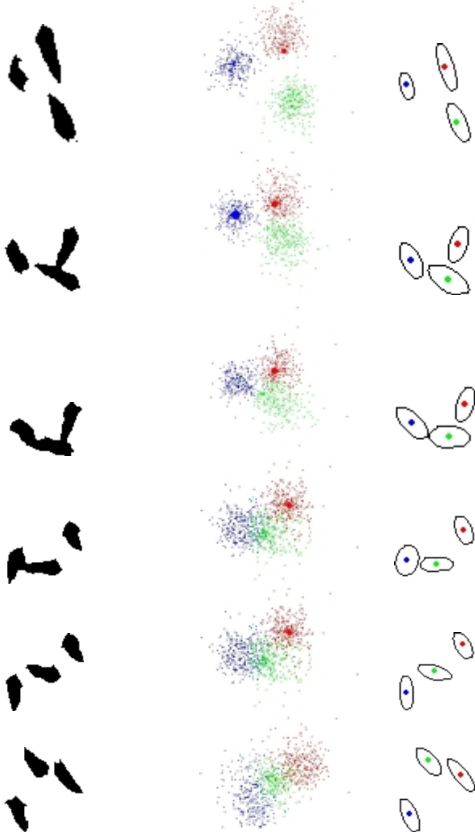
**Figure 6. Particle Number versus Correct Classification for Mice in Contact.**



**Figure 7. Example of mice in contact.**

and blobs can lead to more accurate classification rates than a model based only on blobs or k-means, without a great penalty in processing time. The use of k-means during the observation phase of the filter showed to be important for partial occlusion robustness. However, when mice move separately, the results are basically equal. The use of standard k-means has the drawback that it cannot separate clusters that are non-linearly separable in input space. However, the typical mice clustering situations, as illustrated in Figures 1 and 8, are linearly separable. The problem of optimal locality, associated with the standard k-means formulation, can be handled using good initial values. In the experiments discussed in this paper, the use of previous estimated positions of the mice, as an initial guess, has proven to be a good choice.

The main contributions of this article are the use of the k-means algorithm in the observation model of a particle filter and the comparison of different observation models. For future research, it would be interesting to include information related to the contour of the mice in the state model, and to use variable velocity in the dynamic model, calculated using optical flow techniques, for instance. It is also im-



**Figure 8. Example tracking three mice with 1000 particles.**

portant to expand the tests using a larger amount of images, with different kind of animals and environments.

## Acknowledgments

This work has been funded by the Dom Bosco Catholic University, UCDB, the Foundation of Teaching, Science and Technology Development of Mato Grosso do Sul State, FUNDECT, and the Brazilian Studies and Projects Funding Body, FINEP. The author of this paper and some of his advisees hold scholarships from the Brazilian National Counsel of Technological and Scientific Development, CNPQ.

## References

- [1] F. Bartumeus, M. G. E. Luz, G. M. Viswanathan, and J. Catalan. Animal search strategies: a quantitative random-walk analysis. In *Ecology*, volume 86, pages 3078–3087, 2005.
- [2] B. C. Brandao, J. Wainer, and S. K. Goldenstein. Subspace hierarchical particle filter. In *SIBGRAP '05: Proceedings of*

- the XVIII Brazilian Symposium on Computer Graphics and Image Processing*, volume 0, pages 194–204, Los Alamitos, CA, USA, 2006. IEEE Computer Society.
- [3] K. Branson and S. Belongie. Tracking multiple mouse contours (without too many samples). In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 1039–1046, Washington, DC, USA, 2005. IEEE Computer Society.
- [4] K. Choo and D. J. Fleet. People tracking using hybrid Monte Carlo filtering. In *International Conference on Computer Vision*, volume 2, pages 321–328. IEEE Computer Society, 2001.
- [5] D. Eilam. Open-field behavior withstands drastic changes in arena size. *Behavioural Brain Research*, 142:53–62, 2003.
- [6] N. Funk. A study of the Kalman filter applied to visual tracking. Technical report, University of Alberta, 2003.
- [7] S. Har-Peled and B. Sadri. How fast is the k-means method? *Algorithmica*, 41(3):185–202, 2005.
- [8] C. Hue, J.-P. L. Cadre, and P. Pérez. A particle filter to track multiple objects. In *IEEE Workshop on Multi-Object Tracking*, pages 61–68, Vancouver, Canada, July 2001.
- [9] Z. Khan, T. Balch, and F. Dellaert. Efficient particle filter-based tracking of multiple interacting targets using an MRF-based motion model. In *IROS*, pages 321–328, Las Vegas, 2003. IEEE Computer Society.
- [10] D. Malyszko and S. T. Wierzhon. Standard and genetic k-means clustering techniques in image segmentation. *cisim*, 0:299–304, 2007.
- [11] E. F. Morais, M. F. M. Campos, F. L. C. Padua, and R. L. Carceroni. Particle filter-based predictive tracking for robust fish counting. In *SIBGRAP '05: Proceedings of the XVIII Brazilian Symposium on Computer Graphics and Image Processing*, page 367, Washington, DC, USA, 2005. IEEE Computer Society.
- [12] F. Moreno-Noguer and A. Sanfeliu. A framework to integrate particle filters for robust tracking in non-stationary environments. In *IbPRIA*, pages 93–101, 2005.
- [13] K. Nummiaro, E. Koller-Meier, and L. V. Gool. A color-based particle filter. *European Conference on Computer Vision*, pages 53–60, 2002.
- [14] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *European Conference on Computer Vision*, pages 28–39, 2004.
- [15] I. M. Rekleitis. *Cooperative Localization and Multi-Robot Exploration*. PhD thesis, School of Computer Science, McGill University, Montreal, Quebec, Canada, February 2003. <http://www.cim.mcgill.ca/~yiannis/Publications/thesis.pdf>.
- [16] C. Shan, Y. Wei, T. Tan, and F. Ojardias. Real time hand tracking by combining particle filtering and mean shift. In *FGR*, pages 669–674, Los Alamitos, CA, USA, 2004. IEEE Computer Society.
- [17] A. J. Spink, R. A. J. Tegelenbosch, M. O. S. Buma, and L. P. J. J. Noldus. The ethovision video tracking system—a tool for behavioral phenotyping of transgenic mice. *Physiology and Behavior*, 73(5):731–744, August 2001.