Position Estimation on Image-Based Heat Map Input using Particle Filters in Cartesian Space

1st Niklas Fiedler
TAMS, Hamburg Bit-Bots
Universität Hamburg
Hamburg, Germany
5fiedler@informatik.uni-hamburg.de

2nd Marc Bestmann
TAMS, Hamburg Bit-Bots
Universität Hamburg
Hamburg, Germany
bestmann@informatik.uni-hamburg.de

3rd Jianwei Zhang
TAMS
Universität Hamburg
Hamburg, Germany
zhang@informatik.uni-hamburg.de

***Abstract*—This paper presents an approach for using an image-based heat map as measurement input of a particle filter. Pixels of the heat map are transformed into Cartesian space relative to the robot and regarded as single measurements. The approach uses a novel observation model to weight the particles accordingly to the heat map pixels. While this paper focuses on handling FCNN output, the method is also applicable to other feature recognition methods like saliency approaches. The proposed method shows similar performance in standard cases and huge improvements on erroneous input compared to the conventional approach.**

***Index Terms*—filtering, state estimation, image processing**

I. Introduction

Since the success of AlexNet [1] in the ImageNet challenge [2], neural networks are widely regarded as state of the art in object detection methods. In recent years, fully convolutional neural networks (FCNNs) were used in image segmentation [3] and later also in object localization [4], [5].

The output of these networks is usually a two-dimensional heat map representation of the network activation in immediate positional correlation to the input image. In the conventional approach, clustering is applied to the FCNN output to get the position of the detected object [6]. In the case of mobile robots, the filtering of this position is then performed in Cartesian space to take the robot's odometry into account. This is commonly done by using Kalman [7] or particle filters [8].

This paper proposes a novel filtering approach by directly transforming the raw image-based heat map output onto the ground plane and then applying a particle filter directly on the transformed pixels. While doing this, each pixel of the heat map is regarded as a single measurement. Afterward, clustering is performed on these particles, resulting in the filtered position estimation of the object. The approach aims to reduce the information loss between the FCNN and the filtering process by removing the heat map clustering step in the image space and propagating all information available in the heat map into the particle filter. Thereby, all observations are taken over into the filter, improving the state prediction, especially in edge cases and when data from a secondary source, e.g. other agents, is fused.

This research was partially funded by the German Research Foundation (DFG) and the National Science Foundation of China (NSFC) in project Crossmodal Learning, TRR-169.

In this paper, we evaluate the performance of this approach using the RoboCup Humanoid Soccer domain [9] as a case study. In this domain, it is necessary to track the position of the ball on a soccer field, while handling challenges like fast movements and occlusion from other robots.

Fig. 1 gives an overview of the approach. The FCNN is applied onto the input image (Fig. 1a) and produces a heat map (Fig. 1b). Instead of clustering the heat map in the image space, it is projected into Cartesian space relative to the robot. Fig. 1c depicts particles of the filter representing estimations of the object position and the measurements resulting from the heat map input.

The paper is structured as follows: Firstly, an overview of the current work in this field is given in Section II. Secondly, the approach is presented in Section III. It is then evaluated in Section IV. Finally, the paper concludes with Section V.

II. Related Work

There are two main scenarios for position estimation in robotics. Firstly, the self-localization of a robot which weights the state estimates based on the ability to map measurements onto known facts about the environment [10]. Secondly, the tracking of objects in a relative position to a robot. In this case, objects are tracked directly in the image space and the result is transformed to a Cartesian position.

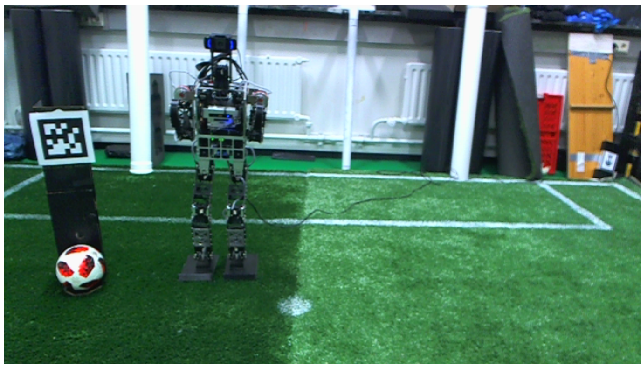
While the paper at hand focuses on object tracking, both scenarios are relevant and will be discussed in the following subsections.

A. Vision-Based Robot Self-Localization

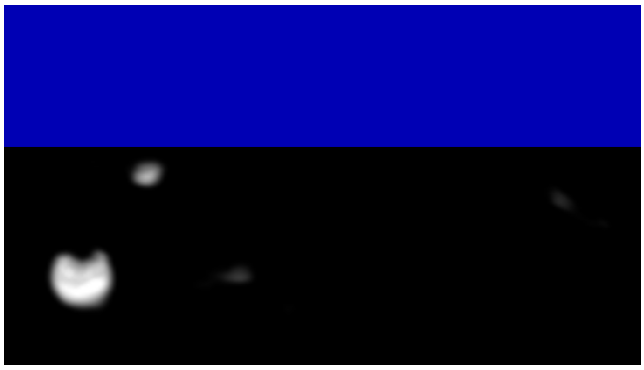
Anati et al. [11] present a way to localize based on the heat maps resulting from soft object detection methods. These heat maps are generated using a histogram of gradient energies and histograms of quantized colors to classify the input and then matched onto the existing map, resulting in a rating of a position estimation of the robot. To the best of our knowledge, this is the only previous work which transforms a heat map output directly into Cartesian space and then filters it there.

B. Object Tracking

Liu et al. [6] used a sliding window CNN approach on regions of interest (ROIs) to compute a heat map of likelihood.



(a) The input image of the vision pipeline showing the typical domain of a soccer field with other robots, field markings, goals and background objects. The AprilTag is added in this setup to provide ground truth for the evaluation. Before the image is processed in the FCNN, it is resized to 200×150 pixels using interpolation.



(b) The heat map generated by the FCNN given the input image from (a). Everything above the green field is cut off and marked blue to show the area which is removed during post-processing, as it is not possible to compute the positions of these pixels in Cartesian space. Besides a large activation for the ball, smaller false positives are visible for one of the goal posts and the penalty mark. The heat map was resized to allow an undistorted image and to keep the aspect ratio of the original input image.

(c) View from above showing the transformed heat map from (b) (**grey**), the particles (**yellow**), the position of the robot (**brown**) and the position of the AprilTag. The particles are accumulated around the area with a high activation of the FCNN while ignoring the false positives due to the previous belief. Due to a high diffusion and a large number of particles, the particle cloud is large but even so has a high density in the center close to the position of the AprilTag representing the ground truth. As only pixels with an activation over a certain threshold are transformed, the image areas without any activation cannot be seen. The explorer particles (see Sect. III-E) are visible in a larger distance from the cluster.

Fig. 1. The input image 1a of the FCNN, the resulting heat map 1b and the transformed measurements mapped into Cartesian space 1c.

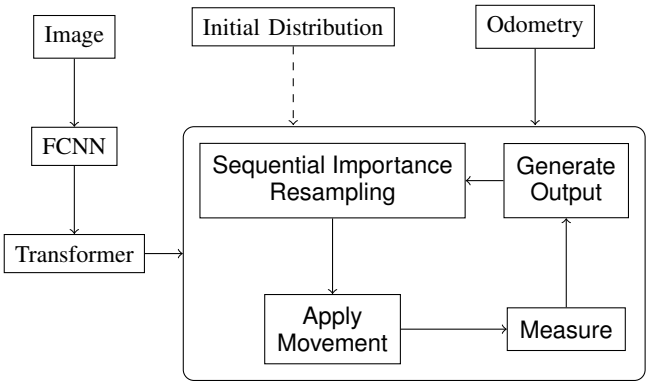


Fig. 2. Schematic representation of the filtering process. The particles are initialized following an initial distribution. Then, the filter runs at a fixed rate independent to the measurement acquisition. The FCNN generates a heat map based on the input image. Each pixel is regarded as a measurement which is transformed into Cartesian space and then applied on the particle filter in the measurement update step. Based on the odometry input the movement is applied onto the particles.

Based on this, the position of the highest value in the heat map and the distance to the last one (used as velocity) are fed into the particle filter.

In [12] and [13], Buyer et al. improved the performance of object tracking in an image with a particle filter by using a multi-layer approach. Each particle is assigned to an area in the binary input image and its weight is adapted accordingly.

Haarnoya et al. presented a method to train discriminative deterministic state estimators [14] which take images as an input. This is achieved by feeding the raw input images into a convolutional feed-forward network which produces an "intermediate observation" and a matrix. The network is connected to a Kalman filter which uses the intermediate observation as measurement and the matrix as covariance matrix to filter the input. While training, the loss of the network is calculated by the difference between the ground truth and the prediction of the Kalman filter. Thus, the network learns to generate the optimal input for the filter from an image. However, due to the restrictions of the Kalman filter, multi-modal and non-linear filtering are not possible without adaptions. Additionally, by the reduction of the measurement to a single state, a lot of information is lost before filtering.

All methods presented in this section apply the filtering step on the image space. This is not applicable to mobile robots because due to camera movement the targets are not trackable as their movement in the image depends on the movement of the object, the camera and the robot itself.

III. Approach

Our approach follows the conventional particle filter principle as depicted in Fig. 2. After initialization based on a given distribution, the particle filter runs independently to the measurement input at a fixed rate. The belief state of the filter models the position of the object in Cartesian space $p = (x, y)$.

A. Filter Input

To use every pixel of the FCNN-output in the particle filter, every pixel is treated as a measurement in the form $m = (r, w)$, r being the position of the transformed particle in the 2-dimensional space (x, y) and $w \in \mathbb{R}[0:1]$ representing the weight of the measurement which is equal to the normalized value of the corresponding pixel in the heat map.

The high amount of measurements can be reduced by applying a minimal activation threshold and by reducing the size of the heat map. Both of these methods can be applied before transforming the pixels, thus significantly improving computational performance. Using a threshold leads only to exclusion of pixels with a low activation, thus reducing the number of measurements with a comparably low impact on the particle weights (see Section III-B). Downscaling of the heat map also reduces the number of transformed pixels but with a higher impact on the particle weights, as all information is reduced independently of their activation level.

B. Observation Model

A novel observation model is necessary to accommodate the high number of measurements. A particle is defined similarly to a measurement consisting of a position r representing the state and a weight w. The distance $\delta_{i,j}^t$ between particle p_i^t and measurement m_j^t is computed for all particles $p \in P^t$ to all measurements $m \in M^t$. During the measurement update step, the particle weights $w(p_i^t)$ for every index i at current time t are computed using $\delta_{i,j}^t$.

With increasing distance to the robot, the density of measurements decreases. Thus, the distance between a particle and surrounding measurements increases.

The weight of measurement m_i^t is denoted as $w(m_i^t)$. To reduce the number of measurements taken into consideration for a particles weight, we use a set of the k measurements with the smallest distance to a particle p_i^t denoted as C_i^t. This results in a consideration of the local environment of a particle. Higher values for k translate to a larger environment taken into account.

$$w(p_i^t) = \sum_{j=1}^{k} \frac{w(m_j^t)}{\delta_{i,j}^t} \text{ with } m_j^t \in C_i^t \tag{1}$$

C. Measurement Aging

The particle filter runs at a fixed rate. The acquisition of measurements consists of multiple steps: the preprocessing, execution of the FCNN, post-processing of the FCNN output and transformation of the pixels into the world. Therefore, the measurement-input-rate can be lower than the resampling rate of the particle filter and can also unpredictably vary over time. Our solution to the problem is *measurement aging*.

By decreasing the weight of each measurement by a constant value v in each step of the particle filter (2), the impact of the measurements decreases.

$$M^{t+1} = \begin{cases} M^{t+1} & |M^{t+1}| > 0 \\ \{w(m_i^t) - v | \forall m_i^t \in M^t\} & |M^{t+1}| = 0 \end{cases} \tag{2}$$

Because of the high computational cost of measuring the distance between all P^t to a m_i^t and the possibility of a $w(m_i^t)$ to sink below 0, a threshold value $l \geq 0$ is used. Every measurement m_i^t with $w(m_i^t) < l$ gets removed from M^t. If no measurement is available, each particle gets the same weight, allowing a completely random resampling.

As the robot moves, the movement model proposed in Section III-D has to be applied to measurements, too.

D. Movement Model

Since the state space of the filter is relative to the mobile robot, a movement model for the particles is necessary when the robot moves. The movement model gets the linear and angular odometry of the robot as input and translates it into a movement of the particles in Cartesian space. To take measurement inaccuracies into consideration in the filter, Gaussian noise is added to the particle movement. Additionally, in each particle filter step, Gaussian noise is added to the position of each particle, causing them to drift apart when no measurement is available and thus to represent the growing uncertainty.

E. Explorer Particles

Depending on the use case and the FCNN, multiple regions with high activations can occur (see Fig. 5 and 7a). This situation requires multi-modal filtering capabilities. While the particle filter is able to model the situation, it can get stuck in local optima because particles can accumulate at a single measurement cluster.

In [15]–[18] several approaches for multi-target tracking are presented.

In our case, only a single target is tracked. Detections outside of the main cluster should not be ignored to prevent convergence on a false positive while a true positive is also detected.

We introduce *explorer particles* to approach this problem. Explorer particles are an adaption of the sequential importance resampling algorithm [8], which spawns a configurable fraction of the particles with the lowest support randomly according to the current probability-distribution of the particle state in Cartesian space. Thus, creating the possibility for the particles to get a high weight resulting in multiple particles getting resampled in the same area.

In Fig. 1c, explorer particles are visible as the sparsely distributed particles distant from the measurements.

F. Output Generation

After each filtering step, a compressed state estimation of the filter can be extracted from the current particle set P^t. The optimal output generation algorithm is highly dependent on the use case of the filter.

In unimodal environments (estimating the position of a single object), a mean of all or a certain share of the highest rated particles is a computationally cheap and easily implementable option.

Multi-modal state distributions and features like a covariance matrix of the generated result require more sophisticated

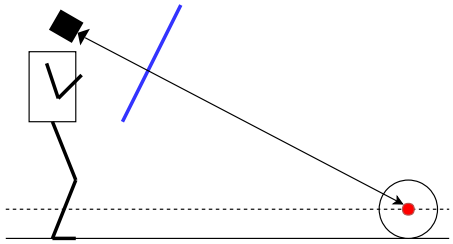


Fig. 3. Graphic representation of the transform method. The camera pose is measured via forward kinematics. A ray is formed from this pose through a pixel in the image plane (**blue**). The intersection of this ray with the ground plane (**red**) is equal to the position of the pixel on the ground. Since the ball is higher than the floor, the intersection plane (**dotted**) is lifted by the half of the ball height to compensate the perspective error.

particle clustering methods. K-means [19] separates the particles into a predefined number of clusters. A soft clustering method based on Gaussian mixture models (GMMs) like the expectation-maximization algorithm (EM-algorithm) [20] also generates a predefined number of clusters but describes them as GMMs. With x-means [21] and extended versions of the EM-algorithm [13] it is also possible to detect the optimal number of clusters.

In our case, the mean of 95% of the particles with the highest support is used.

IV. Evaluation

The evaluation of this approach shows that there is no performance trade-off compared to the conventional approach in trivial cases. Different edge-cases are highlighted in which the immediate inclusion of the FCNN heat map improves the state estimation.

A. Measurement Generation

In our evaluation, we use the FCNN model proposed by Speck et al. [4]. It is trained on publicly available annotated images from ImageTagger [22] to detect a soccer ball in the RoboCup environment and is currently in use as part of the vision system of the Hamburg Bit-Bots [23]. Unedited RGB camera images scaled down to the input size of $200 \times 150 \times 3$ are fed into the net in the vision pipeline. Since no depth information is available from the sensor, the distances of the pixels are computed by the assumption that the ball is lying on the floor and the knowledge of the ball size, as shown in Fig. 3. The pose of the camera is computed using forward kinematics and thus invokes measurement errors especially when walking. Using this method only pixels below the horizon can be transformed. Therefore, the post-processing handles this by cropping the images above the horizon to remove non-transformable pixels.

B. Method

Our approach is compared to a particle filter which takes the center of highest activation from the same FCNN as a single relative measurement. This approach is similar to the measurement acquisition in [6], but with a heat map generated by an FCNN as input. The output of both of these filters is compared to the *ground truth*, measured by an AprilTag detector [24], [25]. The measurement setup can be seen in Fig. 1a. The AprilTag measurements still contain some noise, which is significantly less than the error occurring in the detection methods and therefore negligible. Both methods are configured in the same way and solely differ in the observation model (see Section III-B).

In this unimodal environment (there is always only one ball in the soccer field), the mean of 95% of the highest rated particles is used to generate the output state, as the weakest 5% of the particles are used as explorer particles.

TABLE I
Overview of the errors measured.

	mean error [m]	max error [m]	standard deviation of the errors [m]
our approach	0.0771	0.54	0.07714
conventional approach	0.0879	0.61	0.08142

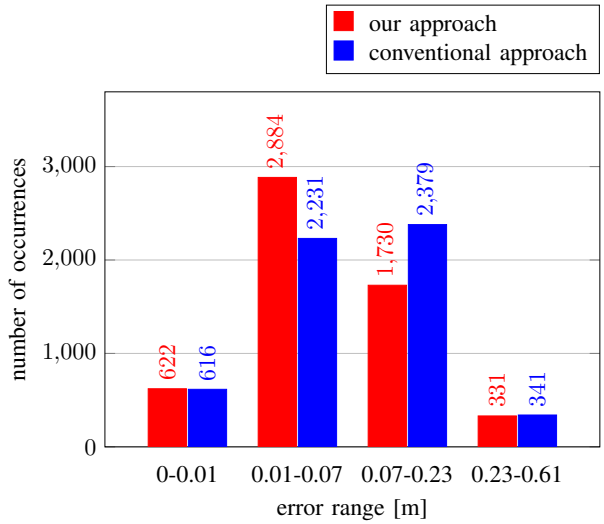
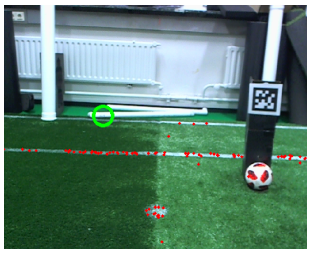


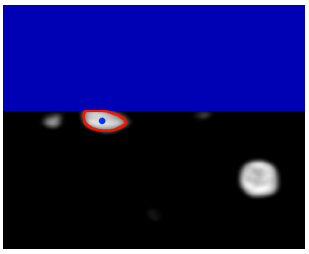
Fig. 4. Bar chart depicting the distribution of the measurement error of our approach (**red**) compared to the error produced by the conventional method (**blue**). The error is measured as the Euclidean distance of the measurement (filter output) to the ground truth in meters. This chart is the result of 5567 measurements for both methods taken in the same scenario. There was no error larger than 0.61 m (see Table I).

C. Results

To evaluate the performance of the proposed method, the output error of the conventional method and our approach was measured for 5567 continuous filter steps. While measuring, the object was moved in sight of the robot with varying velocities and in multiple directions. The robot itself was not moving. The scenario does not include errors resulting from edge cases discussed in Section IV-D. The results listed in Table I shows a better performance of our proposed method compared to the conventional approach. It represents the error measured as the Euclidean distance of the measurement (filter

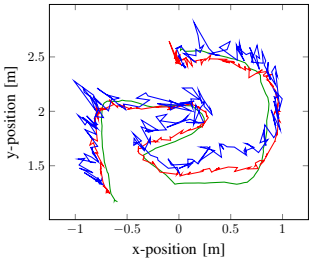


(a) The output of the vision pipeline with the ball detected via clustering in the image frame (**green**) and points of field lines (**red**).

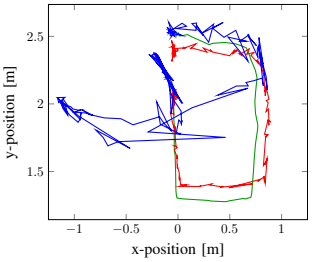


(b) FCNN output with false positive activations from goal posts and white objects near the field border. The cluster detected by cluster detection in the image frame is marked in **red** and its center, which will be the filter input, is marked in **blue**.

Fig. 5. False positive activations in the FCNN output and the resulting error in the post-processing. A detection of these shapes is necessary since a partially concealed ball also results in this form of activation. By feeding all information available in the heat map into the filter, the true positive in the measurement is kept.



(a) Without false-positive detections in the FCNN



(b) With false-positive detections in the FCNN

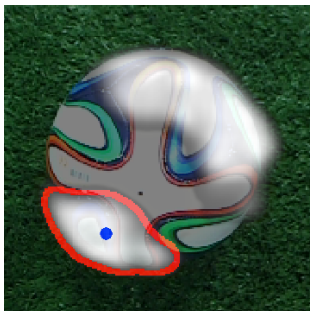
Fig. 6. The traces depicted are the estimate of the ball pose estimated in Cartesian space relative to the robot. The ground truth is marked in **green**, the result of the conventional approach in **blue** and our method in **red**.

output) to the ground truth in meters. The values are the result of 5567 measurements. The compared methods show a significant difference in the distribution of the measurement error (see Fig. 4). Despite the improvements over the conventional method, this work focuses on the performance of the filter in edge-cases in the following section.

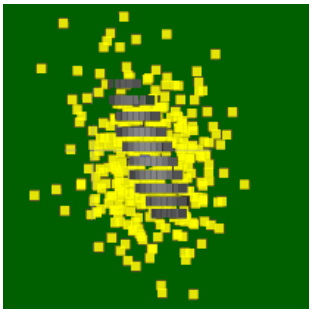
D. Edge Cases

The approach focuses on a stable performance despite significant errors in the FCNN output while maintaining comparable performance to conventional approaches in standard cases.

Fig. 5 shows the detection of false positives in the FCNN output. Especially in a unimodal environment, this issue is highly problematic in the conventional approach, as the clustering algorithm only outputs a single measurement. By feeding the information into the particle filter directly, the error resulting from the detection of false positives can be reduced. As the particles accumulate in the area of the correctly detected object, they do not drift away from it immediately when another wrong measurement is added (see Fig. 6b).



(a) Exemplary detection of multiple clusters in a single object overlaying the input image. The result of a cluster detection by threshold is drawn into the heat map (shape: **red**, center: **blue**). The issue occurs frequently while detecting objects which are partly concealed or close to the robot.



(b) Particles following uncertainties in the measurement. In our case, the situation occurs while detecting objects far away from the sensor. The high distance is the reason for the gaps in the measurement pixels which were spread out due to the transformation.

Fig. 7. Examples of edge cases in the FCNN output

A video taken while recording the traces in Fig. 6 demonstrating the advantages of the proposed method is available online [26].

Especially while detecting objects very close to the robot, a resulting heat map could look like the one in Fig. 7a. By clustering directly in the heat map, only a fraction of the object is detected. This leads to a wrongly located ball center on the image and therefore to a wrong position on the ground plane. The immediate transformation of the heat map into Cartesian space allows the filter to take the distance to the object into consideration. Thus, the two clusters in the heat map are located closely together in Cartesian space, particles accumulate in both of them with their mean in the center of the actual object.

Measurement uncertainties in the heat map arising from fast object movement or distortions due to the transformation are represented in the particle distribution (see Fig. 7b). In the conventional approach, this information is lost in the clustering process of the heat map.

E. Discussion

Depending on the amount of measurements ($|M|$) and particles ($|P|$) and k (the number of the closest measurements taken into account), the runtime of the approach grows in $\Theta(|P|(|M| + |M|log(|M|) + k))$ (based on the method presented in Section III-B). As the input size $|M|$ is highly dependent on the heat map, the threshold, the applied compression of the heat map (see Section III-A) and the number of particles, the computational effort is scalable to the requirements and resources of the environment. Another option to limit the required computational power would be the reduction of measurements after transforming and thresholding to set a maximum amount of measurements. While the application of the observation model is costly, computational effort is spared in the vision pipeline by removing the necessity to detect clusters in the heat map.

As depth cameras are not allowed in RoboCup [9], we do not use them in the approach presented here. In other less restricted environments, the transformation can be changed to use RGB-D. Thereby, errors can be reduced and the detection is not limited to an area below the horizon.

V. Conclusion & Further Work

In this paper, we presented a novel approach on state estimation of objects on the basis of heat maps. The results in the standard case are comparable to the conventional approach while the performance in edge case situations is more stable and reliable (see Table I and Fig. 4).

The drastically increased amount of information used to measure the weight of the particles has a lot of potential for further use and analysis.

The option to handle heat maps representing the likelihood distribution for the object to detect could be evaluated for other detection methods aside from FCNNs like [27] or [28].

As mentioned in III-F, the EM-algorithm or an adaption of it for multi-modal environments can be used to cluster the particles. The output in form of a Gaussian mixture model represents the distribution of the particles in the filter. It can be used for further processing and will represent the measurement uncertainties which originated in the vision system in a compact form. It has to be evaluated in the future, whether in a multi-robot system, the inference of the resulting GMMs could result in more precise measurements.

As this method was developed and tested in the RoboCup environment, its applicability to other domains remains to be evaluated further in the future.

Acknowledgment

Thanks to Michael Görner, Norman Hendrich, Florens Wasserfall and the Hamburg Bit-Bots, especially to Jasper Güldenstein and Jonas Hagge, for their help and discussions.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[2] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[4] D. Speck, M. Bestmann, and P. Barros, "Towards real-time ball localization using cnns," in *Robot World Cup XXII*. Springer, 2018.

[5] F. Schnekenburger, M. Scharffenberg, M. Wülker, U. Hochberg, and K. Dorer, "Detection and localization of features on a soccer field with feedforward fully convolutional neural networks (FCNN) for the adult-size humanoid robot Sweaty," in *Proceedings of the 12th Workshop on Humanoid Soccer Robots, IEEE-RAS International Conference on Humanoid Robots, Birmingham*, 2017.

[6] H. Liu, D. P. Moeys, G. Das, D. Neil, S.-C. Liu, and T. Delbrück, "Combined frame-and event-based detection and tracking," in *Circuits and Systems (ISCAS), 2016 IEEE International Symposium on*. IEEE, 2016, pp. 2511–2514.

[7] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[8] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," in *IEE Proceedings F-radar and signal processing*, vol. 140, no. 2. IET, 1993, pp. 107–113.

[9] Robocup soccer humanoid league laws of the game 2017/2018. [Online]. Available: http://www.robocuphumanoid.org/wp-content/uploads/RCHL-2018-Rules-Proposal_final.pdf

[10] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.

[11] R. Anati, D. Scaramuzza, K. G. Derpanis, and K. Daniilidis, "Robot localization using soft object detection," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4992–4999.

[12] J. Buyer, M. Vollert, A. Haas, M. Kocsis, and R. D. Zöllner, "An adaptive multi-layer particle filter for tracking of traffic participants in a roundabout," in *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*. IEEE, 2016, pp. 2625–2631.

[13] J. Buyer, M. Vollert, M. Kocsis, N. Susmann, and R. Zöllner, "Image-based multi-target tracking using a multi-layer particle filter and extended EM clustering," in *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, MFI 2017, Daegu, Korea (South), November 16-18, 2017*, 2017, pp. 620–625. [Online]. Available: https://doi.org/10.1109/MFI.2017.8170391

[14] T. Haarnoja, A. Ajay, S. Levine, and P. Abbeel, "Backprop kf: Learning discriminative deterministic state estimators," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 4376–4384. [Online]. Available: http://papers.nips.cc/paper/6090-backprop-kf-learning-discriminative-deterministic-state-estimators.pdf

[15] E. B. Meier and F. Ade, "Using the condensation algorithm to implement tracking for mobile robots," in *1999 Third European Workshop on Advanced Mobile Robots (Eurobot'99). Proceedings (Cat. No.99EX355)*, Sept 1999, pp. 73–80.

[16] E. B. Koller-Meier and F. Ade, "Tracking multiple objects using the condensation algorithm," *Robotics and Autonomous Systems*, vol. 34, no. 2-3, pp. 93–105, 2001.

[17] M. R. Morelande, C. M. Kreucher, and K. Kastella, "A bayesian approach to multiple target detection and tracking," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 1589–1604, May 2007.

[18] C. Kreucher, K. Kastella, and A. O. Hero, "Multitarget tracking using the joint multitarget probability density," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1396–1414, Oct 2005.

[19] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*. Berkeley, Calif.: University of California Press, pp. 281–297.

[20] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[21] D. Pelleg, A. W. Moore *et al.*, "X-means: Extending k-means with efficient estimation of the number of clusters." in *Icml*, vol. 1, 2000, pp. 727–734.

[22] N. Fiedler, M. Bestmann, and N. Hendrich, "ImageTagger: An open source online platform for collaborative image labeling," in *RoboCup 2018: Robot World Cup XXII*. Springer, 2018.

[23] M. B. et al., "Wf wolves & hamburg bit-bots team description for robocup 2018 – humanoid teensize."

[24] D. Malyuta, "Guidance, Navigation, Control and Mission Logic for Quadrotor Full-cycle Autonomy," Master thesis, Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena, CA 91109, USA, Dec. 2017.

[25] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, oct 2016, pp. 4193–4198.

[26] Position estimation on image-based heat map input demo. YouTube. Accessed 28.12.2018. [Online]. Available: https://youtu.be/gVa_Wc-jQcU

[27] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li, "Automatic salient object segmentation based on context and shape prior." in *BMVC*, vol. 6, no. 7, 2011, p. 9.

[28] D. A. Klein and S. Frintrop, "Salient pattern detection using W_2 on multivariate normal distributions," in *Joint DAGM (German Association for Pattern Recognition) and OAGM Symposium*. Springer, 2012, pp. 246–255.