

Decentralized Tracking in Sensor Networks with Varying Coverage

Jonas Nygårds, Viktor Deleskog and Gustaf Hendeby

The self-archived postprint version of this journal article is available at Linköping University Institutional Repository (DiVA):

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-149920>

N.B.: When citing this work, cite the original publication.

Nygårds, J., Deleskog, V., Hendeby, G., (2018), Decentralized Tracking in Sensor Networks with Varying Coverage, *2018 21st International Conference on Information Fusion (FUSION)*, , 1661-1667. <https://doi.org/10.23919/ICIF.2018.8455669>

Original publication available at:

<https://doi.org/10.23919/ICIF.2018.8455669>

Copyright: IEEE

<http://www.ieee.org/>

©2018 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.



Decentralized Tracking in Sensor Networks with Varying Coverage

Jonas Nygård^{*}, Viktor Deleskog^{*}, and Gustaf Hendeby[†]

^{*} Div. of C4ISR, Swedish Defence Research Agency (FOI), Linköping, Sweden
e-mail: `firstname.lastname@foi.se`

[†] Dept. of Electrical Engineering, Linköping University, Linköping, Sweden
e-mail: `gustaf.hendeby@liu.se`

Abstract—The number of sensors used in tracking scenarios is constantly increasing, this puts high demands on the tracking methods to handle these data streams. Central processing (ideally optimal) puts high demands on the central node, is sensitive to inaccurate sensor parameters, and suffers from the single point of failure problem. Decentralizing the tracking can improve this, but may give considerable performance loss. The newly presented inverse covariance intersection method, proven to be consistent, even under unknown track cross-correlations, is benchmarked against alternatives. Different track-to-track methods, including smoothed association over a window, are compared. A scenario with objects tracked in multiple cameras, not necessarily optimized for tracking, are used to give realism to the evaluations.

I. INTRODUCTION

Target tracking is an important component in surveillance applications, where it is used to provide situational awareness around public arrangements, critical infrastructure, etc. This way, a increasing amount of sensor data, mainly video, is becoming available to support the situation awareness. As the number of sensors increase, traditional target tracking methods that rely on all available measurements being processed in a central node, a *fusion center* (FC), become infeasible. Reasons for this include the high bandwidth needed to transmit all measurements through the network, the sensitivity to signal loss especially in wireless networks, and the risks associated with having a single point of failure. The alternative is to use decentralized fusion, where tracking is performed in direct connection to the sensors, providing tracks which are further refined in one or more *track-to-track* (T2T) fusion centers. This way, the bandwidth is reduced as only relevant tracks must be transmitted, and robustness is increased as loss of communication gracefully degrades the performance to what is obtainable with the single sensor.

Exact decentralized *track-to-track fusion* (T2TF) methods, as presented in [19], rely on full knowledge of correlations between the fused estimates. This requires either detailed information about the processing performed in each node of the system, or the communication of weight matrices. Not only does this lead to increasingly complex algorithms, when self-contained sensor nodes are used the necessary information is often simply not available, rendering the method infeasible. Therefore, methods insensitive to correlation between the estimates can be of use. The sliding window approach using

the approximate algorithm in [20] is one alternative. Another is *generalized information matrix filter* (GIMF, [21]). GIMF extracts the new information from each estimate, utilizing previous estimates and the target model used in the nodes.

Several methods for approximate T2TF that require neither knowledge of the algorithms in the nodes nor memory exist. The wellknown *covariance intersection* (CI, [8]) method provides the optimal consistent estimate¹, but it is often overly conservative. A less conservative method, without consistency guarantees, is *safe fusion* (SF, [6]). SF was introduced to a wider audience in [12], as an attractive alternative for T2TF. It is algorithmically identical to the *ellipsoidal intersection* (IE, [9, 17]) method, but is derived from a different perspective. It also provides the same covariance matrix as the *largest ellipsoid* method [1], but gives a different mean estimate. A recent and promising method is the *inverse covariance intersection* (ICI, [10, 11]) method, which provides a middle way, providing a less conservative estimate than CI, while being able to guarantee consistency under quite general assumptions.

Before any T2TF methods can be used, the tracks from the different sensor nodes must be associated, which in itself is a nontrivial problem. When reduced rate reporting is applied in the sensor network, promising results have been shown [4] using smoothing based association.

The contribution of this paper is an evaluation of different decentralized tracking solutions. Using realistic simulations and experimental data CI, SF, ICI, and GIMF are compared, accounting for also for the association step. The association step is inspired by [4], whereas a different fusion strategy is used. A centralized fusion node provides the baseline. The sensor network considered is a network of cameras directed towards an urban square with only limited overview of the surrounding block. Thus, the sensors coverage and resolution is not necessarily optimal for the task at hand, as is often the case in real life surveillance situations.

II. METHODS

This section provides information about all the different steps in the decentralized target tracking, assuming point detection of the target are available from all the cameras.

¹Consistent in this context means that the true covariance of the estimate is never larger than the provided covariance estimate.

A. Trackers

All nodes in the network, as well as, the centralized reference filter implement an ordinary *global nearest neighbor* (GNN) multi-target tracker [3]. The scenes are not considered to be complex enough to require more advanced tracking, such as represented by, *e.g.*, a *multi-hypothesis tracker* (MHT). The process and observation models used are provided next.

In the horizontal plane, the process model describes a decoupled damped *constant velocity* (CV) motion. The velocity is damped to avoid unreasonably large velocities, essentially by adopting a zero-mean first-order stationary Markov process as in the Singer process [18], limited to velocity due to the nature of the tracked targets. The model in each direction becomes

$$\mathbf{x}^i(t+T) = \begin{bmatrix} 1 & (1 - e^{-\alpha T})/\alpha \\ 0 & e^{-\alpha T} \end{bmatrix} \mathbf{x}^i(t) + w^i(t), \quad (1a)$$

where $\mathbf{x}^i(t)$ comprises position and velocity for $i \in \{x, y\}$, and T is the sample time, α is the inverse of the maneuver time constant, and $w^i(t)$ is white process noise. The process noise covariance is approximated as

$$\text{cov}(w^i(t)) = Q^i = q \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix}, \quad (1b)$$

where q represents the process noise intensity.

The process model for the z -direction is a terrain map supported *constant position* (CP) model,

$$\mathbf{x}^z(t+T) = \mathbf{x}^z(t) + Tw^z(t), \quad (1c)$$

with process noise $\text{cov}(w^z(t)) = Q^z$.

The combined model has the state vector $\mathbf{x}(t) = [(\mathbf{x}^x(t))^T, (\mathbf{x}^y(t))^T, \mathbf{x}^z(t)]^T$, and $w(t) \sim \mathcal{N}(0, Q)$.

The observation model is provided by the standard pin-hole-camera model [7],

$$z(t) = h(\mathbf{x}(t)) + v(t) = \frac{f}{\zeta(t)} \begin{bmatrix} \xi(t) \\ \eta(t) \end{bmatrix} + v(t), \quad (2)$$

where $(\xi(t), \eta(t), \zeta(t))$ is the position of the target expressed in the camera coordinate system, f is the focal length, and $v(t) \sim \mathcal{N}(0, R)$. Note that the use of the terrain map makes it possible to track the target in the local cameras.

Using the given setup, node n does, given a confirmed track k , produce the track information $\hat{z}_k(t)^n = (\hat{x}_k(t), P_k(t))^n$, where $\hat{x}_k(t)$ is the estimated state vector and $P_k(t)$ is the associated covariance. This information is sent to the fusion node.

B. Track-to-Track Association

Let $\hat{z}_k(t) = (\hat{x}_k(t), P_k(t))$ represent an estimated track of a target k at time t . Define

$$\hat{Z}_k(t_i:t_j) = \{\hat{z}_k(t_i), \hat{z}_k(t_{i+1}), \dots, \hat{z}_k(t_j)\} \quad (3)$$

as the estimated target trajectory of x_k from time t_i to t_j . Consider N individual fusion nodes, which each maintains a set of K_n target tracks

$$\mathcal{Z}^n(t) = \{\hat{Z}_1^n(t_1^n:t), \hat{Z}_2^n(t_2^n:t), \dots, \hat{Z}_{K_n}^n(t_{K_n}^n:t)\}_{n=1}^N, \quad (4)$$

and a FC that maintains K_c target tracks

$$\mathcal{Z}^c(t) = \{\hat{Z}_1^c(t_1^c:t), \hat{Z}_2^c(t_2^c:t), \dots, \hat{Z}_{K_c}^c(t_{K_c}^c:t)\}. \quad (5)$$

Note, the target identities do not necessarily match between the nodes and the FC. At fixed rates, the local fusion nodes communicate their set of their trajectories in (4). The last N_w samples of these are smoothed in the FC, using a *Rauch-Tung-Striebel* (RTS, [14]) smoother, producing the smoothed set of tracklets

$$\bar{\mathcal{Z}}^n(t - (N_w - 1)T : t). \quad (6)$$

The processing at the FC follows three steps: time update, association, and finally track fusion. The association step is repeated in a sequential manner for each local fusion node to compute the global association.

1) *Time Update*: Each global track maintained by the FC is predicted to current time t from previous time t_- according to

$$\hat{x}_k^c(t) = F(t - t_-)\hat{x}_k^c(t_-) \quad (7a)$$

$$P_k^c(t) = F(t - t_-)P_k^c(t_-)F^T + Q(t - t_-)^T \quad (7b)$$

where $F(T)$ and $Q(T)$ describes the dynamics of the target x_k^c , as described in (1).

2) *Association*: The association step is split in two sub-steps: gating and assignment.

The gating step is based on the all Mahalanobis distances [3] over the smoothing window, with the gating criterion

$$\bar{d}_{ij}^2(t, N_w) = \frac{1}{N_w} \sum_{n=0}^{N_w-1} d_{ij}^2(t - nT) \leq \gamma_G \quad (8a)$$

$$d_{ij}^2(t) = (\hat{x}_i^c(t) - \bar{x}_j^n(t))^T S_{ij}^{-1}(t) (\hat{x}_i^c(t) - \bar{x}_j^n(t)) \quad (8b)$$

$$S_{ij}(t) = P_i^c(t) + \bar{P}_j^n(t) \quad (8c)$$

where $i = 1, 2, \dots, K_c$, $j = 1, 2, \dots, K_n$ and γ_G is the track gating threshold. Note that the possible cross-covariance terms (P_{ij}^{cn} and P_{ji}^{nc}) in (8c) are neglected.

The global assignment is computed by the auction algorithm [2] using the assignment matrices, \mathcal{A}_{cn} , which are computed for each fusion node. The assignment matrix is then populated as

$$\mathcal{A}_{cn} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1K_n} \\ a_{21} & a_{22} & \cdots & a_{2K_n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{K_c 1} & a_{K_c 2} & \cdots & a_{K_c K_n} \\ \log(\beta_E) & -\infty & \cdots & -\infty \\ -\infty & \log(\beta_E) & \cdots & -\infty \\ -\infty & -\infty & \ddots & -\infty \\ -\infty & -\infty & -\infty & \log(\beta_E) \end{bmatrix}, \quad (9a)$$

where a_{ij} is the mean Gaussian log-likelihood associated with the assignment of global track $\hat{z}_i^c(t)$ to local track $\hat{z}_j^n(t)$, if the local track passed the gate in (8a). The log-likelihood is

$$a_{ij} = \begin{cases} \frac{1}{N_w} \sum_{n=0}^{N_w-1} \log(P_T p_{ij}(t - nT)), & \text{if (8a) holds} \\ -\infty, & \text{otherwise} \end{cases} \quad (9b)$$

$$\beta_E = \beta_{NT} + \beta_{FT}, \quad (9c)$$

where

$$p_{ij}(t) = \frac{1}{(2\pi)^{M/2} \sqrt{S_{ij}(t)}} \exp\left(-\frac{1}{2}d_{i,j}(t)\right).$$

Here, M is the dimensionality of the tracked state vector and P_T is the probability that a given target is tracked. The terms $\log(\beta_E)$ allows for not assigning an observation to an existing track, where $\beta_E = \beta_{NT} + \beta_{FT}$ is the density of new targets, β_{NT} , and false targets, β_{FT} . The global association is then computed from the assignment matrices $\mathcal{A}_{c1}, \mathcal{A}_{c2}, \dots, \mathcal{A}_{cN}$.

Local tracks that are not assigned to any global track are passed down to the track initiation logic. A local track, $\hat{z}_j^n(t)$, is confirmed to be a new global track if $\bar{d}_{ij}^2(t, N_w) > \gamma_{NT}$ is fulfilled for every global track $\hat{z}_i^c(t)$.

3) *Track Fusion*: Once the track associations have been obtained, the T2T fusion methods described next are used to obtain a track estimate in the FC.

C. Track-to-Track Fusion

To fuse estimates from different trackers that might produce correlated results requires methods that are insensitive to correlated input. Below, four such methods are described.

1) *Generalized Information Matrix Filter (GIMF, [21])*: GIMF is a T2T fusion algorithm with memory. It attempts to extract only the new information using knowledge of previous estimates, this way avoiding information reuse. The fusion rule becomes

$$\hat{x}_f = P_f(P_1^{-1}\hat{x}_1 + [P_2^{-1}\hat{x}_2 - P_2(t_f|t_1)^{-1}\hat{x}_2(t_f|t_1)]) \quad (10a)$$

$$P_f^{-1} = P_1^{-1} + (P_2^{-1} - P_2(t_f|t_1)^{-1}). \quad (10b)$$

The method can either be performed in the fusion center, or in the local nodes as in the channel filter [5].

2) *Safe Fusion (SF, [6, 12])*: The method is given in Algorithm 1. The idea with SF is to approximately decouple the fused estimates, and only pick information from one of the estimates in each direction for the fused result. SF has no known consistency guarantees, and examples where it is inconsistent are known [11]. However, it can still be considered an attractive alternative for T2TF [12].

3) *Covariance Intersection (CI, [8])*: CI combines the two estimates in a conservative way, which ensures that information is not double counted. The algorithm is given by

$$\hat{x} = P_f(\omega P_1^{-1}\hat{x}_1 + (1-\omega)P_2^{-1}\hat{x}_2) \quad (11a)$$

$$P_f^{-1} = (\omega P_1^{-1} + (1-\omega)P_2^{-1})^{-1}, \quad (11b)$$

where ω is chosen to minimize a suitable norm of P , e.g., its determinant or trace which corresponds to optimizing the entropy and variance, respectively. In case of low dimensional estimates ($\dim(x) \leq 4$ using the determinant and $\dim(x) \leq 2$ using the trace), closed form expressions can be derived [16]. Analytic approximations are also available for both norms [13].

It has been shown that, without any assumptions about the correlations, CI provides the optimal fused estimate that guarantees that P_f is consistent [15].

Algorithm 1 Safe Fusion [12]

Given two possibly correlated estimates of x , \hat{x}_1 and \hat{x}_2 such that $P_1 = \text{cov}(\hat{x}_1)$, and $P_2 = \text{cov}(\hat{x}_2)$:

- 1) SVD: $P_1 = U_1 D_1 U_1^T$, let $T_1 = D_1^{-1/2} U_1^T$.
- 2) SVD: $T_1 P_2 T_1^T = U_2 D_2 U_2^T$.
- 3) Let $T = U_2^{-1/2} T_1$, $\hat{\hat{x}}_1 = T \hat{x}_1$ and $\hat{\hat{x}}_2 = T \hat{x}_2$, where by construction $\text{cov}(\hat{\hat{x}}_1) = I$ and $\text{cov}(\hat{\hat{x}}_2) = D_2$.
- 4) Select the most informative source for each component $i = 1, 2, \dots, \dim(x)$, let \bar{D} be a diagonal matrix and

$$\begin{aligned} [\hat{\hat{x}}]_i &= [\hat{\hat{x}}_1]_i, & [\bar{D}]_{ii} &= 1 & \text{if } [D_2]_{ii} \geq 1, \\ [\hat{\hat{x}}]_i &= [\hat{\hat{x}}_2]_i, & [\bar{D}]_{ii} &= D_2^{ii} & \text{if } [D_2]_{ii} < 1. \end{aligned}$$

- 5) The final estimate becomes:
 $\hat{x}_f = T^{-1} \hat{\hat{x}}$ and $P_f = T^{-1} \bar{D}^{-1} T^{-T}$.
-

4) *Inverse Covariance Intersection (ICI, [10, 11])*: The relatively recent ICI method is designed to overestimate and remove any common information from the fused estimate. Assumptions are made about correlations, which allows ICI to be less conservative than CI. The resulting algorithm is:

$$\hat{x} = P_f((P_1^{-1} - \omega P_c^{-1})\hat{x}_1 + (P_2^{-1} - (1-\omega)P_c^{-1})\hat{x}_2) \quad (12a)$$

$$P_f^{-1} = P_1^{-1} + P_2^{-1} - P_c^{-1} \quad (12b)$$

$$P_c = \omega P_1 + (1-\omega)P_2, \quad (12c)$$

where again ω is chosen to minimize a norm of P_f , e.g., the trace or determinant, related to the variance and the entropy, respectively, and where P_c is the overestimated common information. Approximate analytical expressions for ω using both norms are provided in [13].

ICI has been shown to be consistent when the two estimates can be represented as

$$\hat{x}_1 = P_1(\Lambda_1^{-1}\lambda_1 + \Gamma_1^{-1}\gamma_1)$$

$$P_1^{-1} = \Lambda_1^{-1} + \Gamma_1^{-1}$$

$$\hat{x}_2 = P_2(\Lambda_2^{-1}\lambda_2 + \Gamma_2^{-1}\gamma_2)$$

$$P_2^{-1} = \Lambda_2^{-1} + \Gamma_2^{-1},$$

where all the components $\lambda_1, \lambda_2, \gamma_1$, and γ_2 are mutually independent, except for the pair γ_1, γ_2 , and an $\alpha > 0$ exists such that $\alpha \Gamma_1^{-1} \preceq P_2^{-1}$ and $\frac{1}{\alpha} \Gamma_2^{-1} \preceq P_1^{-1}$. This includes the cases, but is not limited to, both estimates sharing the same common information and estimates with common process noise [10].

III. METHOD EVALUATION

The performance of the described methods will first be exemplified using real experimental data. Then, based on the experimental data, a simulation study is performed to more controlled evaluation using *Monte Carlo* (MC) simulations with ground truth. The parameter ω is optimized with respect to the trace of the resulting covariance matrix in both the CI and ICI method.

For computational efficiency in the MC simulations all methods except CKF share the same global association based



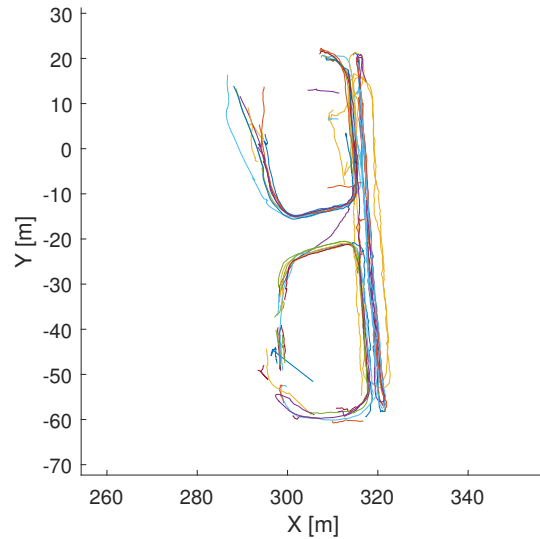
Fig. 1: The experiment area, the street square marked by a star is covered by multiple cameras while the surrounding block is just covered by a couple of cameras per street as illustrated by the sketched field of views of the cameras.

on a SF filter. Note that, using the respective fusion method will influence the associations, and hence the end result.

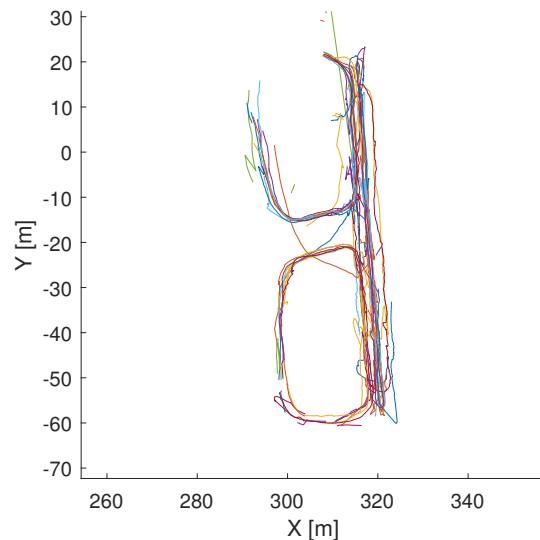
A. Experimental Data

The evaluation is based around a surveillance scenario where a set of cameras is used to persistently monitor a city-square, as depicted in Fig. 1. The experiments are based on a set of cameras used for after-action review in military training, giving access to a network of cameras in an environment where controlled experiments can be performed. The position of the cameras are well calibrated and a terrain model of the area is available allowing tracks in 3D from each camera given the assumption of motion on the ground. Detections of persons obtained from a people detector trained with differences from the background image, allowing shape-based detection of people even with low pixel-counts. The experiments were performed to collect data for anomaly detection and for tracking in *ad hoc* camera networks.

In Fig. 2 the result of a central tracker using eight cameras focused on a T-crossing is illustrated together with the results of a fusion node fusing reports from eight trackers working on the individual cameras. The data is taken from an experiment used to provide a “normal flow” of people for anomaly detection in later sequences from the same scene. No ground truth is available for the experiment other than that there are seven persons in the scene, and a coarse ground truth of the cardinality is constructed by inspecting the images for times when people are occluded by buildings or time-stamp indications in the images. It is impossible to make any conclusive observation from the results in Fig. 2, but it seems



(a) Centralized filter



(b) Decentralized track fusion

Fig. 2: Illustration of the “flow” experiment. A group of seven persons are “simulating” a normal flow for anomaly detection by walking along three loops.

the decentralized solution produces more tracks. Furthermore, the centralized solution works better than the decentralized in some parts of the scene, *e.g.*, the main path to the right, whereas the decentralized solutions appears to be better at keeping the tracks in the lower left corner.

The observations above are corroborated by the cardinality plot in Fig. 3. It is clear that the decentralized fusion suffers from a few more false tracks than the centralized tracker.

In Fig. 4 tracking with sparse camera coverage is illustrated. The majority of the cameras are still focused on the same T-crossing as in the “flow” experiment but there are one–two

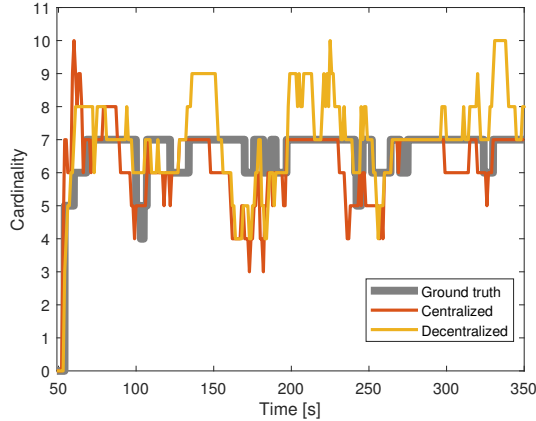


Fig. 3: Cardinality results of the “flow” experiment. The ground truth is a coarse ground truth for when people in the experiment are occluded by a building or the time-stamp in the lower corner of the camera. Note that the detection results are also worse around the buildings but this is not accounted for in the ground truth.

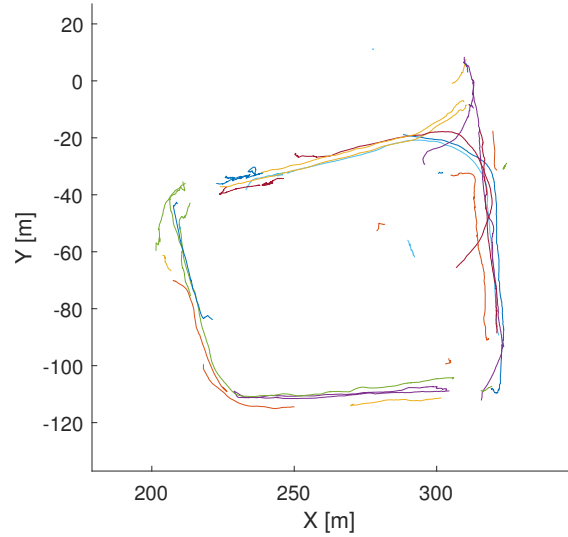
cameras along each street around the block, too, for a total of eleven cameras. The decentralized tracking is still suffering from more false tracks but due to the more robust association it manages to get better track continuity in the lower corner and left street where the centralized filter is more fragmented.

B. Simulated Data

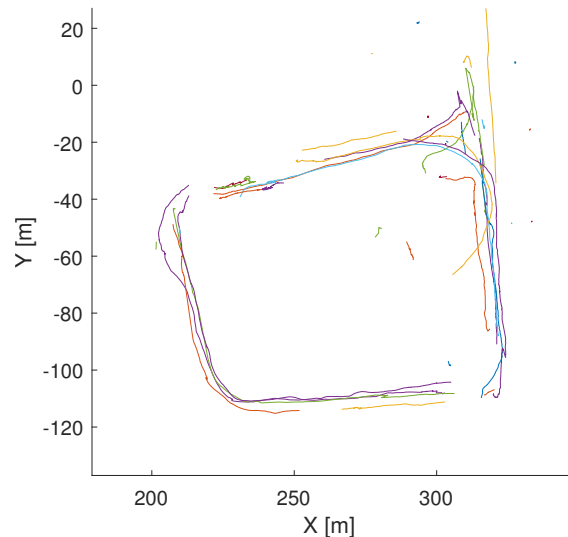
Based on the experiment scenarios, a geometry level simulation as in Fig. 5, is performed. Using the known geometry of the scene combined with the camera calibrations, ideal camera detections with ground truth are simulated. The ideal detections are then disturbed by realistic sensor noise and a Markov process is adapted to give similar detection performance as the real detector, *i.e.*, detections are sporadically missed and false alarm clutter is added to the detections to allow for MC evaluation of the estimators. The *root mean square error* (RMSE) and the *normalized estimation error square* (NEES) of the position part of the state-vector is based on 50 MC simulations and provided in Fig. 6 and 7, respectively.

As seen in Fig. 6, none of the evaluated methods differs considerably in RMSE performance from the others. SF is slightly worse around the time 159–163 s, whereas CI does not fully match the other two methods around time 165–167 s. Overall the results from the CKF is better in the beginning and from 162 s onwards probably because it has access to all measurements and does not need to wait for tracks to start in the individual camera-nodes.

For the given 2D tracking scenario the expected NEES is 2. As seen in Fig. 7, all the evaluated methods yields considerably lower values, indicating the methods are overly conservative. The trend is that SF is the least conservative method and CI the most conservative. The results are expected, the nature of the decentralized methods evaluated is that they should be



(a) Centralized filter



(b) Decentralized track fusion

Fig. 4: Illustration of the “tracking” experiment. A group of five people are moving freely around the block.

conservative to be able to handle worst case correlations. The CKF is the least conservative but still too conservative due to the overestimated sensor covariance.

IV. RESULTS AND DISCUSSION

The real world experiments indicate that association over a smoothed sequence is a useful method to achieve track-to-track fusion. Since track fusion requires initialized tracks, information from multiple cameras becomes delayed initially, also, sparse detection sequences cannot be fully exploited leading to noisier tracks than for the centralized filter. With the current tuning of the algorithms, the T2T fusion gives better track continuity at the price of more false tracks.

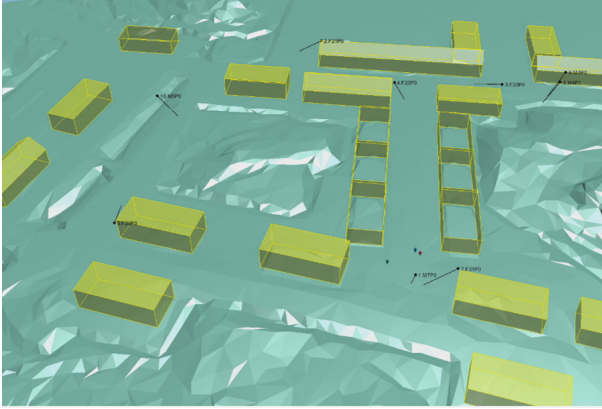


Fig. 5: Simulation of the experiment area, based on the actual geometry.

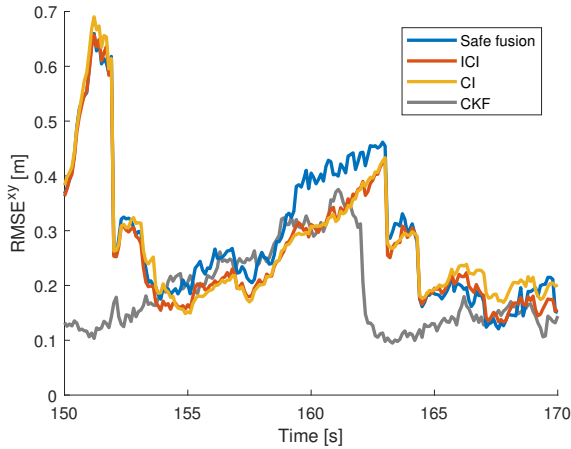


Fig. 6: RMSE in position based on 50 MC runs.

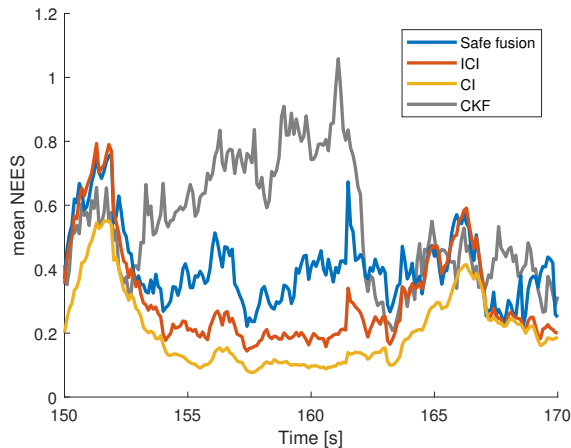


Fig. 7: NEES in position based on 50 MC runs.

Based on the experimental data, the simulations were designed to be faithful to the real-world-experiments to the degree of overestimating the sensor noise covariance making the NEES results inconclusive. For the given scenario the higher covariance also compensates for errors in the ground model so a model with reduced covariance resulted in worse associations.

The GIMF estimates could not be evaluated in the Monte Carlo simulations due to instability in some runs, indicating the sensitivity of this method to deducing the correct correlated information. One reason for this could be that the GIMF method assumes all measurement noises to be independent, whereas this is obviously not case as all trackers are affected by the same ground model.

For the given scenario the SF gives slightly worse RMSE than ICI, while no significant differences can be seen between CI and ICI.

The studied scenario is a good example of a scenario where algorithms such as SF, ICI, and CI are useful since the errors in the ground model will be hard to model faithfully and data from the ground model risks to introduce correlations between the decentralized track estimates. All in all, the resulting correlations between the local trackers will be extremely difficult to model correctly.

V. CONCLUSION

This paper has presented experimental results for a scenario where distributed tracking is expected to be of future value. In the scenario, a ground model was used to enable tracking targets in single cameras, and sometimes only a few cameras see the target under not optimal geometries. As such, the scenario is a good example of a setup where methods such as *safe fusion* (SF), *inverse covariance intersection* (ICI), and *covariance intersection* (CI) are needed, as the use of a priori data, such as ground models, introduce difficulties to compute correlations between the decentralized track estimates. Furthermore, track-to-track association using a smoothed sequence of track estimates provided a well working approach for the scenario. By trying to achieve high similarity to the real-world experiment, the simulations were unfortunately not as conclusive as would have been preferred when it comes to excite differences between the studied track-to-track fusion methods.

REFERENCES

- [1] A. R. Benaskeur. Consistent fusion of correlated data sources. In *28th Annual Conference of the Industrial Electronics Society*, volume 4, pages 2652–2656, Nov. 2002.
- [2] D. P. Bertsekas. The auction algorithm: A distributed relaxation method for the assignment problem. *Annals of Operations Research*, 14(1): 105–123, Dec. 1988.
- [3] S. Blackman and R. Popoli. *Modern tracking systems*. Artech House, 1, 1999.
- [4] F. Govaers, C.-Y. Chong, S. Mori, and W. Koch. Comparison of augmented state track fusion methods for non-full-rate communication. In *Proc. 18th IEEE Int. Conf. Inform. Fusion*, pages 862–869, 2015.
- [5] S. Grime and H. F. Durrant-Whyte. Data fusion in decentralized sensor networks. *Control Engineering Practice*, 2(5):849–863, 1994.
- [6] F. Gustafsson. *Statistical Sensorfusion*. Studentlitteratur, 2. edition, 2013.

- [7] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1106–1112, 1997.
- [8] S. J. Julier and J. K. Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proc. American Contr. Conf.*, pages 2369–2373, Albuquerque, NM, USA, June 1997.
- [9] B. Noack, J. Sijs, M. Reinhardt, and U. D. Hanebeck. Treatment of dependent information in multisensor Kalman filtering and data fusion. In H. Fourtati, editor, *Multisensor Data Fusion: From Algorithms and Architectural Design to Applications*, page 169–192. CRC Press, 2015.
- [10] B. Noak, J. Sijs, and U. D. Hanebeck. Inverse covariance intersection: New insights and properties. In *Proc. 20th IEEE Int. Conf. Inform. Fusion*, Xi’an, China, July 10–13 2017.
- [11] B. Noak, J. Sijs, M. Reinhardt, and U. D. Hanebeck. Decentralized data fusion with inverse covariance intersection. *Automatica*, 79:35–41, May 2017.
- [12] J. Nygård, V. Deleskog, and G. Hendeby. Safe fusion compared to established distributed fusion methods. In *Proc. IEEE Int. Conf. on Multisensor Fusion*, Baden-Baden, Germany, Sept. 2016.
- [13] U. Orguner. Approximate analytical solutions for the weight optimization problems of ci and ici. In *Sensor Data Fusion: Trends, Solutions, Applications*, Bonn, Germany, Oct. 2017.
- [14] H. E. Rauch, F. Tung, and C. T. Striebel. Maximum likelihood estimates of linear dynamic systems. *American Institute of Aeronautics and Astronautics Journal*, 3(8):1445–1450, Aug. 1965.
- [15] M. Reinhardt, B. Noack, P. O. Arambei, and U. D. Hanebeck. Minimum covariance bounds for the fusion under unknown correlations. *IEEE Signal Processing Letters*, 22(9):1210–1214, Jan. 2015.
- [16] M. Reinhardt, B. Noack, and U. D. Hanebeck. Closed-form optimization of covariance intersection for low-dimensional matrices. In *Proc. 15th IEEE Int. Conf. Inform. Fusion*, Singapore, July 9–12 2017.
- [17] J. Sijs and M. Lazar. State fusion with unknown correlation: Ellipsoidal intersection. *Automatica*, 48:1847–1878, Aug. 2012.
- [18] R. A. Singer. Estimating optimal tracking filter performance for manned maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems*, pages 473–483, 1970.
- [19] X. Tian and Y. Bar-Shalom. Exact algorithms for four track-to-track fusion configurations: All you wanted to know but were afraid to ask. In *Proc. 12th IEEE Int. Conf. Inform. Fusion*, pages 537–544, 2009.
- [20] X. Tian and Y. Bar-Shalom. Track-to-track fusion configurations and association in a sliding window. *Journal of Advances in Information Fusion*, 4(2):146–164, 2009.
- [21] X. Tian and Y. Bar-Shalom. On algorithms for asynchronous track-to-track fusion. In *Proc. 13th IEEE Int. Conf. Inform. Fusion*, pages 1–8, 2010.