Consensus protocols for distributed tracking in wireless camera networks

Sandeep Katragadda, Juan C. SanMiguel, Andrea Cavallaro
Centre for Intelligent Sensing
Queen Mary University of London
Email: {s.katragadda, juan.carlos.sanmiguel, a.cavallaro}@qmul.ac.uk

***Abstract*—Consensus-based target tracking in camera networks faces three major problems: non-linearity in the measurement model, temporary lack of measurements (naivety) due to the limited field of view (FOV) and redundancy in the iterative exchange of information. In this paper we propose two consensus-based distributed algorithms for non-linear systems using the Extended Information Filter as underlying filter to handle the non-linearity in the camera measurement model. The first algorithm is an Extended Information Consensus Filter (EICF) that overcomes the effect of naivety and non-linearity without requiring knowledge of other nodes in the network. The second algorithm is an Extended Information Weighted Consensus Filter (EIWCF) that overcomes all the three major problems (naivety, redundancy and non-linearity) but requires knowledge of the number of cameras (N_c) in the network. The basic principle of these algorithms is weighting node estimates based on their covariance information. When N_c is not available, EICF can be used at the cost of not handling the redundancy problem. Simulations with highly maneuvering targets show that the two proposed distributed non-linear consensus filters outperform the related state of the art by achieving higher accuracy and faster convergence to the centralised estimates computed by simultaneously considering the information from all the nodes.**

I. Introduction

Distributed algorithms aim to achieve scalability and comparable performance to centralised algorithms, where global communications are assumed [1]. Recently, distributed state estimation problems have attracted interest for wireless camera networks applications such as target tracking [2] and action recognition [3]. Consensus-based approaches are distributed estimators where nodes communicate only with neighbours and exchange data locally [4]. The nodes do not require prior knowledge of the network and produce the same state estimations in all nodes of the network. The advantage of consensus approaches is that the global decision is available at every node so each one can take decisions independently. Fully connected networks can perform consensus in one single iteration, being equivalent to centralised approaches. However, in general networks are partially connected and consensus requires a number of iterations to converge.

The performance of distributed trackers depends on the shared data and the type of filter employed. The Kalman Consensus Filter (KCF) [3] and Information Consensus Filter (ICF) [5] are widely used average consensus filters for distributed tracking. KCF and ICF are based on the Kalman Filter (KF) [6] and Information Filter (IF), respectively.

There exist three main problems in achieving consensus for target tracking: non-linearity, naivety and redundancy. The target state dynamics (e.g. location on a reference plane) and the corresponding target measurements provided by the cameras (e.g. target coordinates in the image plane of each camera) are *non-linearly* related to each other [7]. Hence, the tracking algorithms that depend on linear filters, such as KF and IF, can not be applied to camera networks. Other filters such as the Extended Kalman Filter (EKF) [8] or Extended Information Filter (EIF) [9] address this non-linearity. Although the Extended Kalman Consensus Filter (EKCF) [10] handles non-linearities, it does not address naivety and redundancy. Due to the limited field of view of the cameras, there are *naive* nodes that do not provide target measurements. In average consensus, such naive nodes decrease the accuracy of the generated estimates such as in KCF and EKCF, where the local averaging does not consider the accuracy or reliability of the information. Such homogeneous weighting of local contributions leads to sub-optimal results. Finally, iterative data exchange in consensus approaches causes *redundancy* of information in the network, correlates the node estimates and thereby delays the convergence to the optimal result. The Information Weighted Consensus Filter (IWCF) [11] handles both naivety and redundancy, but works only for linear systems. The ideal distributed approach should handle non-linearity and naivety while avoiding communication redundancy.

This paper proposes two new consensus-based distributed tracking algorithms for non-linear systems such as camera networks namely, the Extended Information Consensus Filter (EICF) and the Extended Information Weighted Consensus Filter (EIWCF). The EICF weights the estimates of each node based on its accuracy and solves the problem of naivety. Nodes viewing the target get higher weights as their error covariance is lower than that of non-viewing nodes. This concept is called weighted averaging [12], [13] and is used in the ICF [5]. The concept of weighted averaging is applied to EIF to handle naivety along with non-linearity. The EIWCF is developed based on the EIF and the IWCF [11], which also performs weighted averaging. In addition to this, the IWCF weights the prior information less than the obtained measurement information. IWCF solves both the redundancy and naivety problems and speeds up the convergence to the centralised result. The software of the proposed methods is available at *http://www.eecs.qmul.ac.uk/~andrea/software.htm*.

EICF solves the problems of non-linearity and naivety without requiring the knowledge of the number of nodes in the network. When the number of nodes is available, EIWCF can be used to solve the redundancy problem along with non-linearity and naivety at almost the same communication and computation cost as EICF. Both algorithms use EIF as the underlying filter to deal with non-linearity and employ consensus as the fusion scheme to support the limited communication range of wireless cameras. EICF and EIWCF weight the prior and measurement information about the target differently. Experimental results show that the two proposed approaches outperform the accuracy of EKCF, which also handles non-linearity, at the expense of higher communication and computational costs, especially when more consensus iterations are used.

The paper is organised as follows. Section II describes the state of the art in distributed tracking in camera networks. Section III presents the EIF and its centralised version. Section IV describes the proposed EICF and EIWCF algorithms. Comparisons of the proposed algorithms with the centralised result and EKCF are discussed in Section V. Finally, Section VI concludes the paper.

II. State of the art

This section discusses consensus-based distributed target tracking in camera networks, focusing on the problems of non-linearity, naivety and redundancy.

In the case of camera-based target tracking, the measurement model of a camera can be represented as a non-linear homography function that converts coordinates from the ground plane to the image plane of the camera [14]. As this measurement model is non-linear, KF or IF-based approaches such as KCF, ICF and IWCF that work only for linear systems are not suitable.

Medeiros *et al.* [14] proposed a dynamic clustering technique to create multiple single-hop clusters of viewing nodes. As the target moves, the clusters modify their members so that only viewing nodes are members of a cluster. EKF is used as the underlying filter for estimating the target state to handle non-linearity in the measurement model. Here, inter cluster data fusion is not supported. Each cluster head sends the fused estimates to a base station where the final fusion step is performed. The formation and maintenance of clusters adds communication and computation overhead in the network. Nastasi and Cavallaro [15] applied the particle-filter-based token passing approach proposed by Hlinka *et al.* [16] to camera networks. The authors assumed that all nodes in the network can communicate with each other via single-hop or multi-hop links (routing tables are provided). Both [14] and [15] need information about neighbouring nodes.

The KCF algorithm proposed by Saber *et al.* [17] can be applied for camera networks [2], [3], [1], [10] in which nodes do not require FOV information of other cameras. The authors assumed that overlapping cameras are connected (directly or via multiple hops) so each camera can receive knowledge of the target before it enters their FOV. Simonetto *et al.* [13] proposed a weighted average consensus where local estimates are weighted based on their error covariance values. Nodes not viewing the target get lower weights than viewing nodes. A similar concept is applied to the consensus-based information filter [5]. Weighted average consensus algorithms such as Generalized Kalman Consensus Filter (GKCF) [18] and weighted information consensus [13], [5] produce more accurate tracking performance than KCF in the presence of naive nodes.

The cross-covariance resulted from iterative information exchange among nodes is handled by IWCF by weighting prior and measurement information [11]. As the redundancy is present only in the prior information (i.e. the result of consensus in the previous step), the prior term is given less weight than the measurement information. IWCF handles both naivety and cross-correlation among nodes' estimates and it outperforms KCF and GKCF. Similarly to ICF [5], GKCF and IWCF allow multiple consensus iterations between time steps to improve performance. All these consensus filters do not require any hand-off protocols or the knowledge of the FOVs. However, IWCF requires knowledge of the number of nodes present in the network. EKF can be used for distributed target tracking for handling non-linear systems [19], [10]. However, limited target observability and cross-covariance of error among nodes are not handled.

Table I compares the main distributed tracking algorithms for wireless camera networks. These algorithms cannot simultaneously handle the three problems (non-linearity, naivety and redundancy).

III. Extended information filter

Consider a target whose motion model is given by $\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{w}_{k-1})$, where $\mathbf{x}_k$ is the target state at time k and $\mathbf{w}_{k-1}$ is a non-additive Gaussian noise with zero mean and covariance $\mathbf{Q}$. The measurement model of camera c_i is defined as $\mathbf{z}_k^i = h^i(\mathbf{x}_k) + \mathbf{v}_k^i$ where, $\mathbf{z}_k^i$ is the measurement corresponding to the target state and $\mathbf{v}_k^i$ is an additive Gaussian noise with zero mean and covariance $\mathbf{R}$. In the presence of non-linearity in $f(\cdot)$ or $h^i(\cdot)$, EKF [8] can be used. EKF assumes linearity at the best available state using the Taylor series approximation. EKF handles mild non-linearities where first-order approximations of Jacobians are available. The IF [5] is an alternative form of KF that uses the Fisher information matrix (the inverse of the covariance, $\mathbf{P}$, of the state estimate error). In IF, the local estimates of the nodes are represented as an information matrix $\mathbf{Y}_{k|k}^i = {\mathbf{P}_{k|k}^i}^{-1}$ and an information vector $\mathbf{y}_{k|k}^i = {\mathbf{P}_{k|k}^i}^{-1}\mathbf{x}_{k|k}^i$ where $\mathbf{x}_{k|k}^i$ and $\mathbf{P}_{k|k}^i$ are, respectively, the estimated state and associated error covariance of camera c_i for time step k. The concept of IF can be applied to EKF, resulting in EIF [9].

According to [9], the prediction step of EIF is:

$$\begin{aligned}\mathbf{Y}_{k|k-1}^i &= \mathbf{M}_k - \mathbf{M}_k[\mathbf{M}_k + \mathbf{W}_{f,k}^{-T}\mathbf{Q}^{-1}\mathbf{W}_{f,k}^{-1}]^{-1}\mathbf{M}_k \\ \mathbf{y}_{k|k-1}^i &= \mathbf{Y}_{k|k-1}^i f({\mathbf{Y}_{k|k-1}^i}^{-1}\mathbf{y}_{k-1|k-1}^i),\end{aligned} \quad (1)$$

TABLE I: Algorithms for distributed tracking in wireless camera networks

Acronym	Algorithm	Reference	Filtering scheme	Fusion scheme	Challenges		
					Non-linearity	Naivety	Redundancy
DC-EKF	Dynamic Clustering based Extended Kalman Filter	[14]	Extended Kalman Filter	dynamic clustering	✓		
KCF	Kalman Consensus Filter	[3]	Kalman Filter	consensus			
TP-PF	Token Passing based Particle Filtering	[15]	Particle Filter	token passing	✓		
EKCF	Extended Kalman Consensus Filter	[10]	Extended Kalman Filter	consensus	✓		
GKCF	Generalized Kalman Consensus Filter	[18]	Kalman Filter	consensus		✓	
IWCF	Information Weighted Consensus Filter	[11]	Kalman Filter	consensus		✓	✓
EICF	Extended Information Consensus Filter	[This paper]	Extended Information Filter	consensus	✓	✓	
EIWCF	Extended Information Weighted Consensus Filter	[This paper]	Extended Information Filter	consensus	✓	✓	✓

where $\mathbf{M}_k = \mathbf{J}_{f,k}^{-T}\mathbf{Y}_{k-1|k-1}^i\mathbf{J}_{f,k}^{-1}$ and $\mathbf{J}_{f,k}$ is the Jacobian of $f(\cdot)$ with respect to $\mathbf{x}_k$ and $\mathbf{W}_{f,k}$ is the Jacobian of $f(\cdot)$ with respect to $\mathbf{w}_k$. The prediction in KF is computationally less complex than in IF. The prediction operation can be done with EKF equations and the results can be converted to the information form [9] as follows:

$$\begin{aligned} \mathbf{x}_{k|k-1}^i &= f(\mathbf{x}_{k-1|k-1}^i) \\ \mathbf{P}_{k|k-1}^i &= \mathbf{J}_{f,k}\mathbf{P}^i\mathbf{J}_{f,k}^T + \mathbf{W}_{f,k}\mathbf{Q}\mathbf{W}_{f,k}^T. \end{aligned} \tag{2}$$

The results of prediction can be converted to the IF form as:

$$\begin{aligned} \mathbf{Y}_{k|k-1}^i &= {\mathbf{P}_{k|k-1}^i}^{-1} \\ \mathbf{y}_{k|k-1}^i &= \mathbf{Y}_{k|k-1}^i\mathbf{x}_{k|k-1}^i. \end{aligned} \tag{3}$$

The update step in EIF [9] is:

$$\begin{aligned} \mathbf{Y}_{k|k}^i &= \mathbf{Y}_{k|k-1}^i + \mathbf{I}_k^i \\ \mathbf{y}_{k|k}^i &= \mathbf{y}_{k|k-1}^i + \mathbf{i}_k^i, \end{aligned} \tag{4}$$

where, $\mathbf{i}_k^i$ and $\mathbf{I}_k^i$ are the information contribution terms defined as:

$$\begin{aligned} \mathbf{i}_k^i &= \mathbf{J}_h^T\mathbf{R}^{-1}[\mathbf{z}_k^i - h^i(\mathbf{x}_{k|k-1}^i) + \mathbf{J}_h\mathbf{x}_{k|k-1}^i] \\ \mathbf{I}_k^i &= \mathbf{J}_h^T\mathbf{R}^{-1}\mathbf{J}_h. \end{aligned} \tag{5}$$

Here, $\mathbf{J}_h$ is the Jacobian of $h^i(\cdot)$ with respect to $\mathbf{x}_k$ (in our case $h^i(\cdot)$ is given by the first term of the R.H.S of Eq. 23).

The use of IF is advantageous compared to KF because the computation of the update step is less complex in IF. Especially in the case of multi-sensor fusion, the update step at the fusion centre is a simple addition of the information contribution terms from all N_c nodes in the network as:

$$\begin{aligned} \mathbf{Y}_{k|k}^g &= \mathbf{Y}_{k|k-1}^g + \sum_{i=1}^{N_c}\mathbf{I}_k^i \\ \mathbf{y}_{k|k}^g &= \mathbf{y}_{k|k-1}^g + \sum_{i=1}^{N_c}\mathbf{i}_k^i \end{aligned} \tag{6}$$

Hence, IF is usually preferred. The global state estimate and corresponding error covariance can be recovered using:

$$\begin{aligned} \mathbf{P}_{k|k}^g &= {\mathbf{Y}_{k|k}^g}^{-1} \\ \mathbf{x}_{k|k}^g &= {\mathbf{Y}_{k|k}^g}^{-1}\mathbf{y}_{k|k}^g. \end{aligned} \tag{7}$$

Eq. 6 represents the centralised version of EIF. The next section will introduce the corresponding distributed version.

IV. Distributed state estimation using the Extended Information Filter

We aim to compute global estimates $\mathbf{y}_{k|k}^g$ and $\mathbf{Y}_{k|k}^g$ (Eq. 6) through local network communications. Achieving average consensus means computing the average of all nodes' information and simultaneously making the result ($\mathbf{x}_{k|k}$) available at all nodes in the network. The average consensus on a random variable $\mathbf{a}$ can be computed as [4]:

$$\mathbf{a}_l^i = \mathbf{a}_{l-1}^i + \epsilon \sum_{j\in C_i^n}(\mathbf{a}_{l-1}^j - \mathbf{a}_{l-1}^i), \tag{8}$$

where l is the index for the consensus iteration, C_i^n is the neighbourhood of node c_i, and $\epsilon \in (0, 1/\Delta_{max})$ guarantees the convergence of the estimates of the nodes of the network to the average estimate of all nodes, i.e. $\hat{\mathbf{a}} = \frac{1}{N_c}\sum_{i=1}^{N_c}\mathbf{a}^i$. Δ_{max} represents the maximum degree of the network. EKCF achieves the average consensus on the local estimates ($\mathbf{x}_{k|k}$) computed by EKF [10].

To overcome the naivety problem, weighted averaging can be applied [12], [13], [5] considering the associated covariance of the state estimate. Such weighted averaging of the target state can be achieved by average consensus on the information terms:

$$\hat{\mathbf{x}}_{k|k} = \frac{\sum_{i=1}^{N_c}{\mathbf{P}_{k|k}^i}^{-1}\mathbf{x}_{k|k}^i}{\sum_{i=1}^{N_c}{\mathbf{P}_{k|k}^i}^{-1}} = \frac{\sum_{i=1}^{N_c}\mathbf{y}_{k|k}^i}{\sum_{i=1}^{N_c}\mathbf{Y}_{k|k}^i} = \frac{\left(\sum_{i=1}^{N_c}\mathbf{y}_{k|k}^i\right)/N_c}{\left(\sum_{i=1}^{N_c}\mathbf{Y}_{k|k}^i\right)/N_c}, \tag{9}$$

where N_c is the total number of nodes in the network and $\hat{\mathbf{x}}$ is the global estimate.

A. Extended information consensus filter

We apply weighted averaging to EIF and propose two distributed filters for tracking targets in wireless camera networks, EICF1 and EICF2, which compute the local information, $\mathbf{y}_{k|k}^i$ and $\mathbf{Y}_{k|k}^i$, differently.

EICF1 runs at each node c_i and computes the local information values, $\mathbf{y}_{k|k}^i$ and $\mathbf{Y}_{k|k}^i$, based on their own respective measurement information, $\mathbf{i}_k^i$ and $\mathbf{I}_k^i$:

$$\begin{aligned} \mathbf{Y}_{k|k}^i &= \mathbf{Y}_{k|k-1}^i + \mathbf{I}_k^i \\ \mathbf{y}_{k|k}^i &= \mathbf{y}_{k|k-1}^i + \mathbf{i}_k^i, \end{aligned} \tag{10}$$

and then exchange the values $\mathbf{y}_{k|k}^i$ and $\mathbf{Y}_{k|k}^i$ with neighbours to achieve average consensus.

EICF2 computes local information values, $\mathbf{y}^i_{k|k}$ and $\mathbf{Y}^i_{k|k}$, based on its own measurement information and also that of neighbouring nodes:

$$\begin{aligned}\mathbf{Y}^i_{k|k} &= \mathbf{Y}^i_{k|k-1} + \sum_{j \in \{c_i \cup C^n_i\}} \mathbf{I}^j_k \\ \mathbf{y}^i_{k|k} &= \mathbf{y}^i_{k|k-1} + \sum_{j \in \{c_i \cup C^n_i\}} \mathbf{i}^j_k.\end{aligned} \tag{11}$$

EICF2 reaches convergence faster than EICF1, at the cost of additional communication to send the measurement information terms. Hence, EICF2 is recommended when sufficient communication resources are available.

The iterative information exchange between neighbours results in redundancy which causes correlation among the nodes' estimates. Hence, the EICF results are sub-optimal because of such correlation among the individual node estimates. In the update step (see Eq. 4) of a filter, the two terms involved are the priors, $\mathbf{y}^i_{k|k-1}$ and $\mathbf{Y}^i_{k|k-1}$, and the measurement information about the target, $\mathbf{i}^i_k$ and $\mathbf{I}^i_k$. The prior information is the result of the prediction on previous estimates, $\mathbf{y}^i_{k|k}$ and $\mathbf{Y}^i_{k|k}$, which are computed after consensus. Hence, the redundancy always lies in the prior information terms, $\mathbf{y}^i_{k|k-1}$ and $\mathbf{Y}^i_{k|k-1}$.

B. Extended information weighted consensus filter

Via proper weighting of prior and measurement information, IWCF mitigates the problem of redundancy [11]. By applying the concept of IWCF to EIF, we propose a non-linear distributed filter called the Extended Information Weighted Consensus Filter (EIWCF). Here the prior information is weighted by $1/N_c$ and the consensus proposals are prepared as:

$$\begin{aligned}\mathbf{v}^i_{k|k} &= \frac{1}{N_c}\mathbf{y}^i_{k|k-1} + \mathbf{i}^i_k \\ \mathbf{V}^i_{k|k} &= \frac{1}{N_c}\mathbf{Y}^i_{k|k-1} + \mathbf{I}^i_k.\end{aligned} \tag{12}$$

After achieving consensus on the $\mathbf{v}$ and $\mathbf{V}$ terms, the results are multiplied by N_c:

$$\begin{aligned}\mathbf{y}^i_{k|k} &= N_c \mathbf{v}^i_{k|k} \\ \mathbf{Y}^i_{k|k} &= N_c \mathbf{V}^i_{k|k}.\end{aligned} \tag{13}$$

These estimates are not affected by non-linearity, naivety and redundancy. However, EIWCF requires the knowledge of the number of nodes in the network (see Eqs. 12 and 13). Thus, EIWCF cannot be applied when such knowledge is not available whereas EICF1 or EICF2 can be used. If sufficient communication resources to receive neighbours' measurement information, $\mathbf{i}^j_k$ and $\mathbf{I}^j_k$, is available, EICF2 achieves faster convergence than EICF1. Hence the choice depends on the available communication resources.

Algorithm 1 summarises the three filters (EICF1, EICF2 and EIWCF) using a switch operator.

Algorithm 1 Distributed Extended Information Filters running at camera node c_i

Input: C^n_i : neighbourhood of c_i
$k, k-1$: time steps
$\mathbf{y}^i_{k-1|k-1}$: posterior information state vector from $k-1$
$\mathbf{Y}^i_{k-1|k-1}$: posterior information matrix from $k-1$
N_c : number of nodes in the network
ϵ : consensus parameter $\epsilon \in (0, 1/\Delta_{max})$
Δ_{max} : maximum degree of the network

Compute prior information terms, $\mathbf{y}^i_{k|k-1}$ and $\mathbf{Y}^i_{k|k-1}$, through prediction (see Eq. 3)
Collect measurement $\mathbf{z}^i_k$
Compute $(\mathbf{i}^i_k, \mathbf{I}^i_k)$ using Eq. 5
If no target detected, set $\mathbf{i}^i_k = 0$ and $\mathbf{I}^i_k = 0$
Switch(Algorithm **)**
Case EICF1 **:**
$N = 1$
Prepare consensus terms

$$\begin{aligned}\mathbf{v}^i_0 &= \mathbf{y}^i_{k|k-1} + \mathbf{i}^i_k \\ \mathbf{V}^i_0 &= \mathbf{Y}^i_{k|k-1} + \mathbf{I}^i_k\end{aligned}$$

Case EICF2 **:**
Send message $(\mathbf{i}^i_k, \mathbf{I}^i_k)$ to C^n_i
Receive messages $(\mathbf{i}^j_k, \mathbf{I}^j_k)$ from all $j \in C^n_i$
$N = 1$
Prepare consensus terms

$$\begin{aligned}\mathbf{v}^i_0 &= \mathbf{y}^i_{k|k-1} + \sum_{j \in \{c_i \cup C^n_i\}} \mathbf{i}^j_k \\ \mathbf{V}^i_0 &= \mathbf{Y}^i_{k|k-1} + \sum_{j \in \{c_i \cup C^n_i\}} \mathbf{I}^j_k\end{aligned}$$

Case EIWCF **:**
$N = N_c$
Prepare consensus terms

$$\begin{aligned}\mathbf{v}^i_0 &= \frac{1}{N_c}\mathbf{y}^i_{k|k-1} + \mathbf{i}^i_k \\ \mathbf{V}^i_0 &= \frac{1}{N_c}\mathbf{Y}^i_{k|k-1} + \mathbf{I}^i_k\end{aligned}$$

end Switch
Perform consensus step
for $l = 1$ to L **do**
Send message $(\mathbf{v}^i_{l-1}, \mathbf{V}^i_{l-1})$ to C^n_i
Receive messages $(\mathbf{v}^j_{l-1}, \mathbf{V}^j_{l-1})$ from all $j \in C^n_i$
Update consensus terms

$$\begin{aligned}\mathbf{v}^i_l &= \mathbf{v}^i_{l-1} + \epsilon \sum_{j \in \{c_i \bigcup C^n_i\}} (\mathbf{v}^j_{l-1} - \mathbf{v}^i_{l-1}) \\ \mathbf{V}^i_l &= \mathbf{V}^i_{l-1} + \epsilon \sum_{j \in \{c_i \bigcup C^n_i\}} (\mathbf{V}^j_{l-1} - \mathbf{V}^i_{l-1})\end{aligned}$$

end for
Compute the posterior at k

$$\begin{aligned}\mathbf{Y}^i_{k|k} &= N\mathbf{V}^i_L \\ \mathbf{y}^i_{k|k} &= N\mathbf{v}^i_L\end{aligned}$$

Output: $\mathbf{y}^i_{k|k}$ and $\mathbf{Y}^i_{k|k}$

C. Computation cost

We measure the computational cost of EICF1, EICF2 and EIWCF as the number of scalar operations performed [20]. The computational cost, S, of EKCF is:

$$S_{EKCF} = 11n^3 + 6n^2 + 4n + 17 + 3nm + 2nm^2 + mn^2 + 2m + d\left(\frac{n^2}{2} + \frac{5n}{2}\right) + Ln(2+d), \tag{14}$$

where L and d represent the number of consensus iterations involved and degree of the node, respectively; n is the size of the state vector $\mathbf{x}$ and m is the size of the measurement vector $\mathbf{z}$. The computation cost of EICF1 is:

$$S_{EICF1} = \frac{25n^3}{3} + 4n^2 + \frac{5n}{3} + 2nm^2 + mn^2 + 5nm + m + n + 17 + L\left(d(n^2+3n) + \frac{3n^2}{2} + \frac{n}{2}\right). \tag{15}$$

The computation cost of EICF2 is:

$$S_{EICF2} = \frac{25n^3}{3} + 4n^2 + \frac{5n}{3} + 2nm^2 + mn^2 + 5nm + m + n + 17 + L\left(d(n^2+3n) + \frac{3n^2}{2} + \frac{n}{2}\right) + d\left(\frac{3n}{2} + \frac{n^2}{2}\right). \tag{16}$$

Similarly, the computation cost of EIWCF is:

$$S_{EIWCF} = \frac{25n^3}{3} + 6n^2 + \frac{11n}{3} + 2nm^2 + mn^2 + 5nm + m + n + 17 + L\left(d(n^2+3n) + \frac{3n^2}{2} + \frac{n}{2}\right). \tag{17}$$

EICF2 has $d \cdot (n + n^2/2 + n/2)$ more computations than EICF1 because of the $\sum \mathbf{i}$ and $\sum \mathbf{I}$ operations (see Eq. 11). EIWCF has $2(n+n^2)$ more operations than EICF1 because of the additional multiplications with N and $1/N$ ($\mathbf{y}/N$, $\mathbf{Y}/N$, $N\mathbf{v}$ and $N\mathbf{V}$ operations) as shown in Eq. 12 and Eq. 13.

Figure 1 presents the number of computations for each filter with respect to the number of consensus iterations for a state vector of size 5 and degree 2. EKCF has a higher computation cost at lower number of consensus iterations because of its more complex update step. By increasing the number of consensus iterations, the computations required for handling the covariance information of new information filters significantly increase and surpass the cost of EKCF.

D. Communication cost

The number of scalars transmitted by a node is taken as its communication cost T. For EKCF, this cost is relatively low because only the state vector is exchanged in each iteration. In the first iteration of EKCF, each node sends the measurement information terms, $\mathbf{i}_k^i$ and $\mathbf{I}_k^i$, and the prior state vector to its neighbours. Hence the number of scalars, T, transmitted by each node is:

$$T_{EKCF} = n + \frac{n(n+1)}{2} + Ln, \tag{18}$$

where L is the number of consensus iterations considered. EICF1 exchanges the local estimate, the information vector

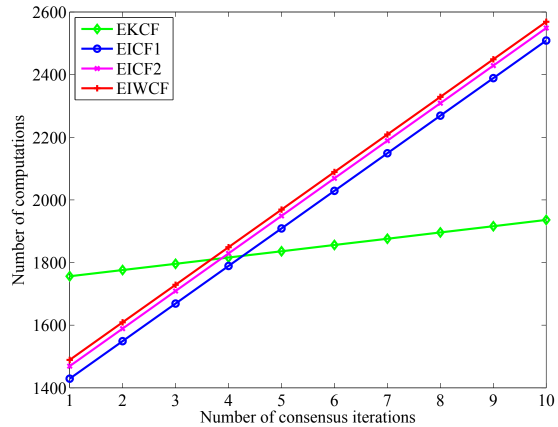


Fig. 1: Computation cost at a camera node c_i vs. number of consensus iterations. Comparison of the computation cost of the Extended Kalman Consensus Filter (EKCF) [10], the Extended Information Consensus Filter1 (EICF1), the Extended Information Consensus Filter2 (EICF2) and the Extended Information Weighted Consensus Filter (EIWCF).

and the corresponding uncertainty matrix, the information matrix, in each iteration. Hence the communication cost is:

$$T_{EICF1} = L\left(n + \frac{n(n+1)}{2}\right). \tag{19}$$

EICF2 exchanges the same information as EICF1 but during the first iteration the measurement information, $\mathbf{i}_k^i$ and $\mathbf{I}_k^i$, is also sent to neighbours. Hence the communication cost is:

$$T_{EICF2} = n + \frac{n(n+1)}{2} + L\left(n + \frac{n(n+1)}{2}\right). \tag{20}$$

Because of the additional $\mathbf{i}_k^i$ and (the upper triangle of) $\mathbf{I}_k^i$ terms, it has an additional communication cost of $n + n(n+1)/2$ scalars. EIWCF exchanges the local estimate, information vector, and the corresponding uncertainty matrix, information matrix, in each iteration. Hence the communication cost is:

$$T_{EIWCF} = L\left(n + \frac{n(n+1)}{2}\right). \tag{21}$$

As the covariance matrices are symmetric only the upper triangular matrix elements are transferred while sending information matrices $\mathbf{I}$ and Y, the cost is $n(n+1)/2$ instead of n^2.

Figure 2 presents the number of scalars needed to be transmitted by each node for each filter with respect to the number of consensus iterations considered.

V. Experiments

We compare the new distributed non-linear filters EICF1, EICF2 and EIWCF with the state of the art filter EKCF [10] and the Centralised Extended Information Filter (CEIF) that uses measurement information from all nodes and estimates the target state according to Eq. 6. As a performance measure, we use the mean error defined as the $L2-$norm between the

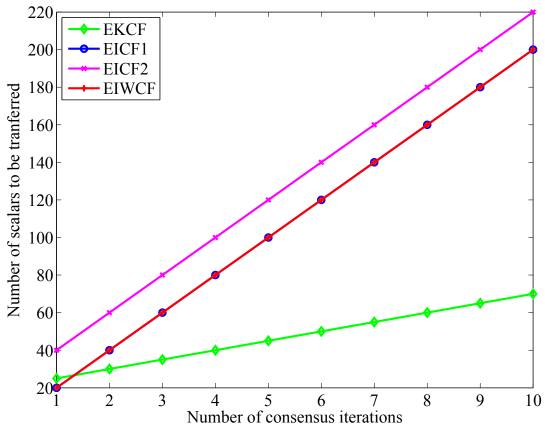


Fig. 2: Communication cost at a camera node c_i vs. number of consensus iterations. Comparison of the computation cost of the Extended Kalman Consensus Filter (EKCF) [10], the Extended Information Consensus Filter1 (EICF1), the Extended Information Consensus Filter2 (EICF2) and the Extended Information Weighted Consensus Filter (EIWCF).

true and estimated target position after a given number of consensus iterations [11]. We analyse the faster convergence to the CEIF results.

A simulated target moving in a 500m×500m area which is under observation of 9 cameras with overlapping FOVs is considered. The FOV of each camera is assumed to be a square region of 200m×200m around the camera. The state vector of the target is given by $\mathbf{x}_k = \begin{bmatrix} x_k & y_k & v_x & v_y & \delta_k \end{bmatrix}^T$. The motion model of the target is as follows [14]:

$$\mathbf{x}_{k+1} = \begin{bmatrix} x_k + v_{x,k}\delta_k + a_x \dfrac{\delta_k^2}{2} \\ y_k + v_{y,k}\delta_k + a_y \dfrac{\delta_k^2}{2} \\ v_{x,k} + a_x\delta_k \\ v_{y,k} + a_y\delta_k \\ \delta_k + e \end{bmatrix}, \quad (22)$$

where (x_k, y_k) is the target position on the ground plane; $(v_{x,k}, v_{y,k})$ is the target velocity and δ_k is the time step between two consecutive measurements. The target acceleration, (a_x, a_y), is modelled as Gaussian noise. To account for synchronisation errors among cameras we consider a time uncertainty e, which is also assumed to be a Gaussian variable. We consider the vector $\mathbf{w} = (a_x, a_y, e)$ as the Gaussian noise vector with zero mean and covariance $Q = diag(\begin{bmatrix} 1 & 1 & 0.001 \end{bmatrix})$.

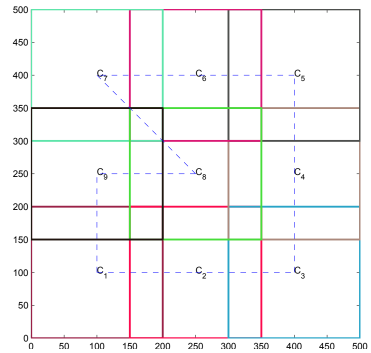
The measurement model of the cameras is:

$$\mathbf{z}_k^i = \begin{bmatrix} u_k^i \\ v_k^i \end{bmatrix} = \begin{bmatrix} \dfrac{H_{11}^i x_k + H_{12}^i y_k + H_{13}^i}{H_{31}^i x_k + H_{32}^i y_k + H_{33}^i} \\ \dfrac{H_{21}^i x_k + H_{22}^i y_k + H_{23}^i}{H_{31}^i x_k + H_{32}^i y_k + H_{33}^i} \end{bmatrix} + \mathbf{v}_k, \quad (23)$$

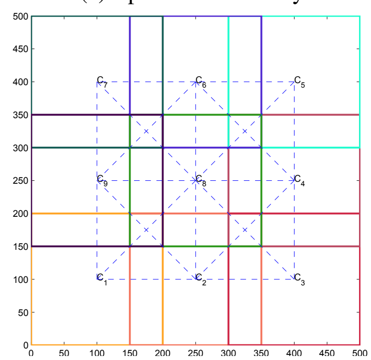
where (u_k^i, v_k^i) is the pixel coordinates of the target in the image plane of camera c_i at time k. The values $H_{11}^i, ...H_{33}^i$ are the elements of the homography matrix and $\mathbf{v}_k$ is the measurement noise, which is considered to be Gaussian with zero mean and variance $\mathbf{R} = diag(\begin{bmatrix} 5 & 5 \end{bmatrix})$. The homography matrix values of each camera are taken from one of the cameras of the APIDIS dataset[1] whose values are:

$$H^i = \begin{bmatrix} 397.2508 & 95.2020 & 287280 \\ 51.7437 & 396.9189 & 139100 \\ 0.0927 & 0.1118 & 605.2481 \end{bmatrix}.$$

We perform the experiments for two types of network connectivity. For the first network, the nodes are sparsely connected with a low average network degree equal to 2 (see Figure 3a). We assume a very limited communication range for the nodes. The second network considers dense connectivity where the degree is higher than the previous case (see Figure 3b). The communication range is larger and each node communicates with more than two nodes. This setup assumes direct communication between overlapping cameras. For each network, 20 target trajectories are generated (see Figure 4a and 4b), where each track is estimated using 20 simulation runs. These two cases are necessary to analyse accuracy or speed of convergence with a varying degree of connectivity. The experiments are conducted assuming no



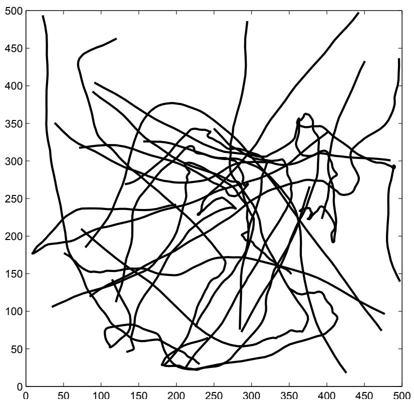
(a) Sparse connectivity



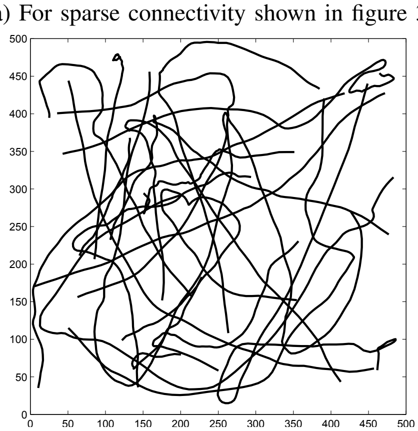
(b) Dense connectivity

Fig. 3: Connectivity types considered in the experiments. Cameras are located in a 500m×500m area and are represented as $c_1, .., c_9$. Their FOVs are represented by $200m \times 200m$ square regions and, the connectivity among the nodes is shown using the blue lines between nodes.

[1] http://www.apidis.org/Dataset/

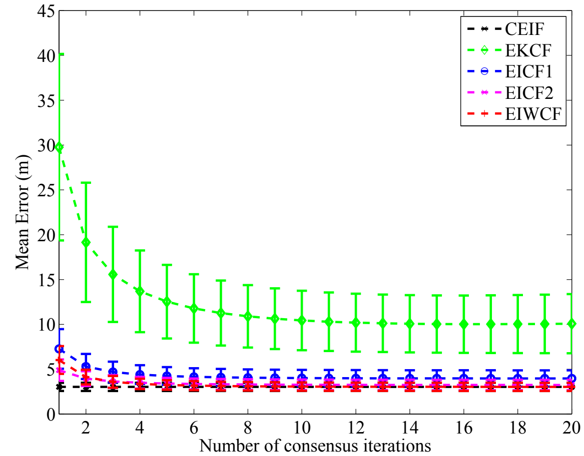


(a) For sparse connectivity shown in figure 3a.

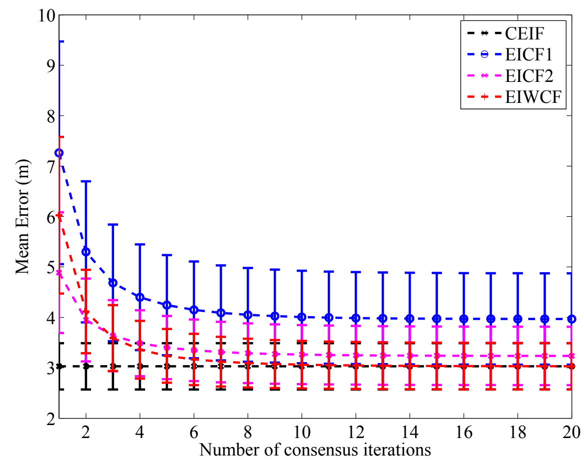


(b) For dense connectivity shown in figure 3b.

Fig. 4: Sample trajectories used in the experiments.



(a)



(b)

Fig. 5: Comparison of accuracy for sparse connectivity shown in Figure 3a. The compared filters are the Extended Kalman Consensus Filter (EKCF) [10], the Extended Information Consensus Filter1 (EICF1), the Extended Information Consensus Filter2 (EICF2) and the Extended Information Weighted Consensus Filter (EIWCF). Figure 5b is zoom of Figure 5a with focus on the proposed filters EICF1, EICF2 and EIWCF.

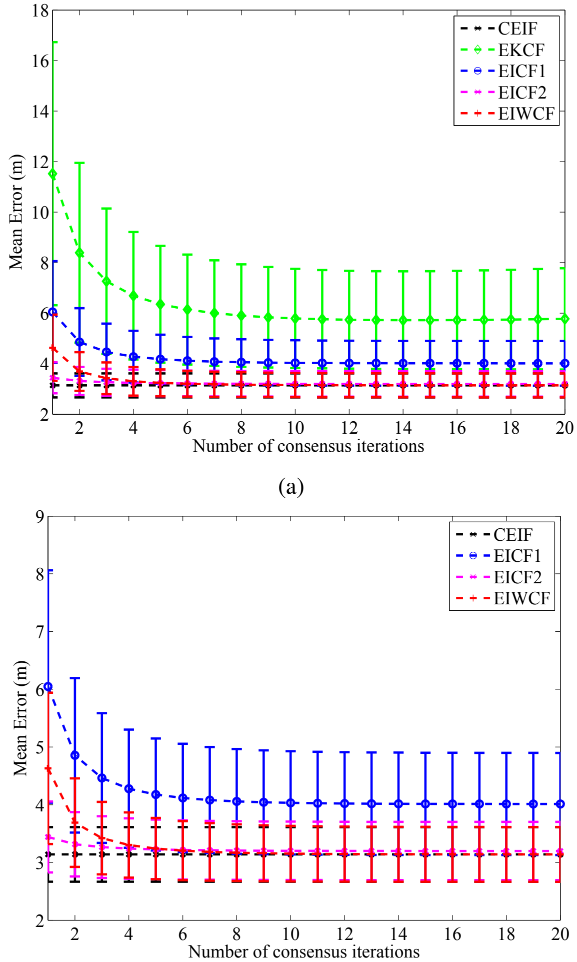
packet losses or link failures in the network.

Figure 5a presents the mean tracking error of the filters for the network shown in Figure 3a. Figure 5b shows the same results with focus on our new filters. Figure 6a presents the mean tracking error of the filters with the setup shown in Figure 3b. Figure 6b shows the same results with focus on our new filters. By analysing the tracking error with different filters (Figure 5a and Figure 6a), it can be observed that the newly developed distributed filters perform better than EKCF. EICF2 converges to the optimal centralised estimate faster than EICF1 because EICF2 considers neighbours information also. Both EICF1 and EICF2 outperform EKCF, but still the performance of EIWCF is higher compared to EICF1 and EICF2, which do not take the redundancy into account. Only EIWCF converges to the optimal centralised estimate because the prior information (which holds redundant data) and the measurement information are weighted properly to avoid the effect of redundancy. It can also be observed that, with the increase in degree of connectivity, the speed of convergence increases for all filters (see Figure 5a and Figure 6a).

VI. Conclusions and Future Work

Three major problems in consensus-based distributed tracking for camera networks are non-linearity, naivety and redundancy. Algorithms such as the Information Weighted Consensus Filter (IWCF) take into account the presence of naive nodes and redundant information in the network by weighing the prior and the measurement information of a target under consideration and perform better than the Kalman Consensus Filter (KCF). However, these algorithms are only suitable for linear systems. As the measurement model of a camera is non-linear, Kalman Filter-based algorithms cannot be used. Non-linear filters such as the Extended Kalman Consensus Filter (EKCF) do not address naivety and redundancy.

In this work, we firstly proposed the Extended Information Consensus Filter (EICF) by combining the Extended Information Filter (EIF) and Information Consensus Filter (ICF). The proposed filter performs weighted averaging while addressing the problem of naive nodes and non-linearity. To overcome the redundancy problem, we also proposed the Extended



(a)

(b)

Fig. 6: Comparison of accuracy for dense connectivity shown in Figure 3b. The compared filters are the Extended Kalman Consensus Filter (EKCF) [10], the Extended Information Consensus Filter1 (EICF1), the Extended Information Consensus Filter2 (EICF2) and the Extended Information Weighted Consensus Filter (EIWCF). Figure 6b is zoom of Figure 6a with focus on the proposed filters EICF1, EICF2 and EIWCF.

Information Weighted Consensus Filter (EIWCF) by combining the Extended Information Filter (EIF) and Information Weighted Consensus Filter (IWCF). EIWCF handles naivety, redundancy and non-linearity, and achieves faster convergence by properly weighting prior and measurement information. However, it requires the knowledge of the number of nodes in the network. The performance of the EICF and the EIWCF is analysed via simulated wireless camera networks. The results show that the proposed approaches outperform the EKCF and, with additional communication and computation cost, they converge to the centralised result.

As future work, we will explore reducing the communication and the computation overhead required by the average consensus. Handling dynamic link structure and asynchronous networks are some other possible future works.

Acknowledgement

Sandeep Katragadda is supported by the Erasmus Mundus Joint Doctorate in Interactive and Cognitive Environments, which is funded by the EACEA Agency of the European Commission under EMJD ICE FPA 2010-0012. Juan C. SanMiguel is supported by the EU Crowded Environments monitoring for Activity Understanding and Recognition (CENTAUR, FP7-PEOPLE-2012-IAPP) project under GA number 324359.

References

[1] M. Taj and A. Cavallaro, "Distributed and decentralized multicamera tracking," *IEEE Signal Processsing Magazine*, vol. 28(3), pp. 46–58, 2011.

[2] C. Soto, B. Song, and A. Roy-Chowdhury, "Distributed multi-target tracking in a self-configuring camera network," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 1486–1493.

[3] B. Song, A. T. Kamal, C. Soto, C. Ding, J. A. Farrell, and A. K. Roy-Chowdhury, "Tracking and activity recognition through consensus in distributed camera networks," *IEEE Trans. on Image Processing*, vol. 19(10), pp. 2564–2579, 2010.

[4] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. of the IEEE*, vol. 95(1), pp. 215–233, 2007.

[5] D. Casbeer and R. Beard, "Distributed information filtering using consensus filters," in *American Control Conf. (ACC)*, 2009, pp. 1882–1887.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. of the ASME Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[7] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[8] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Prentice-Hall, 1979.

[9] A. G. O. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*, 1st ed. CRC Press, Inc., 1998.

[10] C. Ding, B. Song, A. A. Morye, J. A. Farrell, and A. K. R. Chowdhury, "Collaborative sensing in a distributed ptz camera network." *IEEE Trans. on Image Processing*, vol. 21(7), pp. 3282–3295, 2012.

[11] A. Kamal, J. Farrell, and A. Roy-Chowdhury, "Information weighted consensus filters and their application in distributed camera networks," *IEEE Trans. on Automatic Control*, vol. 58(12), pp. 3112–3125, 2013.

[12] L. Xiao, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. of the Int. Conference on Information Processing in Sensor Networks (IPSN)*, 2005, pp. 63–70.

[13] A. Simonetto, T. Keviczky, and R. Babuska, "Distributed nonlinear estimation for robot localization using weighted consensus," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2010, pp. 3026–3031.

[14] H. Medeiros, J. Park, and A. C. Kak, "Distributed object tracking using a cluster-based kalman filter in wireless camera networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2(4), pp. 448–463, 2008.

[15] C. Nastasi and A. Cavallaro, "Distributed target tracking under realistic network conditions," in *Sensor Signal Processing for Defence (SSPD)*, 2011, pp. 1–5.

[16] O. Hlinka, P. Djuric, and F. Hlawatsch, "Time-space-sequential distributed particle filtering with low-rate communications," in *Asilomar Conf. on Signals, Systems and Computers*, 2009, pp. 196–200.

[17] R. Olfati-Saber and N. F. Sandell, "Distributed tracking in sensor networks with limited sensing range," *American Control Conf. (ACC)*, pp. 3157– 3162, 2008.

[18] A. T. Kamal, C. Ding, B. Song, J. A. Farrell, and A. Roy-Chowdhury, "A generalized kalman consensus filter for wide-area video networks," in *IEEE Conf. on Decision and Control and European Control Conf. (CDC-ECC)*, 2011, pp. 7863–7869.

[19] H. Long, Z. Qu, X. Fan, and S. Liu, "Distributed extended kalman filter based on consensus filter for wireless sensor network," in *World Congress on Intelligent Control and Automation (WCICA)*, 2012, pp. 4315–4319.

[20] N. Assimakis, M. Adam, and A. Douladiris, "Information filter and kalman filter comparison: selection of the faster filter," *Int. Journal of Information Engineering*, vol. 2(1), pp. 1–5, 2012.