Data Fusion with Split Covariance Intersection for Cooperative Perception

Antoine Lima, Philippe Bonnifait, Veronique Cherfaoui, Joelle Al Hage

Data Fusion with Split Covariance Intersection for Cooperative Perception

Antoine Lima[1], Philippe Bonnifait[1], Véronique Cherfaoui[1] and Joelle Al Hage[1]

***Abstract*— Cooperative Perception is an emergent technology that profits from the exchanged perception information between vehicles. However, the inconsistency resulting from the reuse of the same information is a main issue that arises. In this paper, we focus on the study of the Split Covariance Intersection Filter (SCIF), a method capable of handling both independent and arbitrarily correlated estimates and observation errors. We are interested in its use in a Cooperative Perception application to incorporate information coming from other vehicles, which may or may not have been tracked, in a generic tracking solution. A simple case study is first presented to build a deep understanding of the filter tuning, then real experiments carried out with three vehicles equipped with GNSS, camera, LiDAR and High Definition (HD) map features are reported to study how a full tracking architecture relying on SCIF behaves in a real-world situation.**

I. Introduction

Perception of the environment is a key component for intelligent vehicles in the field of Intelligent Transportation Systems (ITS). It can provide information to Advanced Driver-Assistance System (ADAS) or to control and decision making algorithms for autonomous vehicles. In those safety-critical systems, a complete view of the environment and redundant perception modalities are essential. That is why the trend in the last couple of years has been to add more and more exteroceptive sensors on vehicles. However, the inherent physical limitations of sensors coupled with an economic and ecologic desire to reduce the number of sensors have pushed research towards the sharing of perceptive information in what is now called Cooperative Perception (CP). The advantages of CP are numerous: extended field of views, perception beyond physical boundaries such as static or moving objects and reduction of the number of sensors required to get a complete and redundant perception.

However, multiple issues have to be solved before this technology can be used at large scale: information exchange loops, communication delays, incorporation of foreign data that can be untrustworthy or inaccurate... In this paper, we are interested to the first issue, that can introduce *data incest*, the fact that a same piece of information is fused multiple times and lead to an over-confident estimate. This is a classical data fusion issue that can be managed with adapted approaches, such as the Covariance Intersection (CI). It is capable of handling unknown correlation between observations and thus avoid overconfidence by realizing a convex combination of two estimates. The main issue with this family of approaches is its pessimism [1]. To alleviate this issue, approaches such as in [2] have been implemented, where local data is carefully separated from shared data in order to apply a Kalman Filter (KF) on the local data and CI to the shared one. This showed good performance, especially in terms of consistency of the estimates.

In this study we compare this approach to an automatic handling of the unknown cross-correlation using the Split Covariance Intersection Filter (SCIF) in order to build a generic Track-To-Track Fusion (T2TF) system. The SCIF combines the resiliency to unknown degrees of cross-correlation between estimates inherited from the Covariance Intersection Filter (CIF) with the optimality of the KF when some noises are known to be independent from others. However, the filter raises some challenges, such as the necessity to maintain a split estimate of the error covariance, making the filter harder to tune. To get a good understanding of the different parameters involved in the modeling of the system, we propose a basic example described in Section III. Even if this case study is conducted with mobiles evolving in one dimension (1D) to obtain the simplest possible state space models, it captures all types of problems that arise, particularly in terms of information exchange cycles. Thanks to the good understanding of the tuning parameters provided by these simulations, we study a complete real-life situation that implements a complete tracking architecture across three vehicles in Section IV using the SCIF.

II. State of the Art

The field of CP deals with the exchange of localization and perception information between multiple agents and the infrastructure. CP has been achieved by exchange of raw sensor information in [3], [4], of intermediary representations in [5], and more prominently of processed information to lighten the communications [1], [6], [7]. In general, the estimations are realized on each agent independently to avoid having a centralized entity that would hinder scalability. For the same reason, the estimates are generally disjoint, meaning that each object is considered independent from others. That processed information can be at the feature level or at the track level, in which case the local and remote tracks can become correlated. To counteract this, some approaches attempt to estimate the cross-correlation [8] by keeping track of all exchanges but this rapidly becomes unpractical due to the number of agents and exchanges. A simpler solution is to use a fusion operator resilient to unknown correlation. While the classical KF [9] works on the assumption that all estimates are independent, and thus cannot be used in such scenarios, other approaches have been developed in the field of T2TF [10]. In [11], Vasic *et. al* proposed an extension on

[1]Université de Technologie de Compiègne, CNRS UMR 7253, Heudiasyc, France

the Probability Hypothesis Density (PHD) to collaborative perception. The Information Matrix Fusion (IMF) [12], in which the common information between two estimates is explicitly removed before fusing the estimates in informational form. The CI [13] and its derivatives are also common solutions. CI is designed such that two consistent estimates are fused into a consistent estimate, independently of the cross-correlation that might exist between them. The method works by realizing a convex combination of the two estimates based on the minimization of the resulting estimation error covariance. It can be interpreted geometrically as finding the intersection between the estimates covariance ellipses. Variations on the CI have also been proposed, such as the Improved Fast CI [14] or Information Theoretic CI [15] that replace the optimization step by an analytic expression, or the Inverse CI that takes inspiration from the IMF to provide a less pessimistic estimate. These methods have been compared and applied to CP [1], [16], [17], [18], where the CI has generally been deemed pessimistic, motivating the choice for other methods. The Split CI [19] is one of these variations that proposes to separate the covariances into an independent and dependent parts. This way, it is possible to take advantage of knowledge about partial independence. The SCIF has been applied to cooperative localization situation problems in [20] and [21].

III. Case Study

A. System modelling

Let's consider an elementary example of mobile vehicles in mutual interaction moving in parallel along a one dimensional path. As each vehicle uses variables seen from multiple point of views, let the notation ${}^k a_j$ be the variable a of j as seen by k. They compose the system $S = \left\{\mathbf{x}_j\right\}_{j\in N}$ and have the ability to measure their absolute position s as well as that of others thanks to exteroceptive sensors. Their goal is to estimate the state of all vehicles with the best possible accuracy while maintaining consistency of estimates. For this, the state of a vehicle k is modeled as a position s_k and constant velocity v_k:

$$\mathbf{x}_k = \left[s_k, v_k\right]^T \tag{1}$$

which is assumed to have a constant velocity affected by a random model scalar error α. Assuming that the current velocity is the average of two samples, the model can be written as:

$$\mathbf{x}_k(t+\Delta t|t) = \mathbf{F}\,\mathbf{x}_k(t|t) + \mathbf{G}\alpha(t) \tag{2}$$

with

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \Delta t/2 \\ 1 \end{bmatrix}, \tag{3}$$

where $\mathbf{F}$ is the transition matrix of the evolution model and $\mathbf{G}$ input matrix of the model error.

The model evolution error α is described by the covariance matrix $\mathbf{Q}$. It is split in two parts such that $\mathbf{Q} = \mathbf{Q}_d + \mathbf{Q}_i$ with $\mathbf{Q}_i$ an *independent part* that describes model errors as in a classical Kalman filter and $\mathbf{Q}_d$ a *dependent part* that models the part of the estimation error that has an unknown degree of correlation with other agents in cooperation. The estimation ${}^k\mathbf{x}_j$ is associated with a covariance ${}^k\mathbf{P}_j$ that characterizes the variance of the estimation error ${}^k e_j$ which is split into a dependent and independent components (themselves independent of each other) such that ${}^k e_j = {}^k e_{j,d} + {}^k e_{j,i}$ and ${}^k\mathbf{P}_j = {}^k\mathbf{P}_{j,d} + {}^k\mathbf{P}_{j,i}$ for all j.

Let us define two observation models that are typical of the problems we encounter in CP. The first one corresponds to a GNSS receiver, which gives to a vehicle its absolute position affected by a random noise $\beta_{\text{GNSS},i}$, modeled to be white and independent:

$$\begin{gathered} {}^k\mathbf{y}_k = s_k + \beta_{\text{GNSS},i} \\ \mathbf{H}_{\text{GNSS}} = \begin{bmatrix} 1 & 0 \end{bmatrix}, \ \mathbf{R}_{\text{GNSS},d} = 0, \ \mathbf{R}_{\text{GNSS},i} = \mathbf{\Sigma}_{\text{GNSS}} \end{gathered} \tag{4}$$

with $\mathbf{H}$ the observation model such that $\mathbf{y} = \mathbf{H}\mathbf{x}$ and $\mathbf{\Sigma}_{\text{GNSS}}$ the known variance of the GNSS noise.

The second corresponds to an embedded perception sensor (e.g. a LiDAR) that gives to a vehicle the relative position of other vehicles. The estimated position of the ego vehicle ${}^k s_k$ is then used to transform this measurement into an absolute position observation as well:

$$\begin{aligned} {}^k\mathbf{y}_j &= {}^k s_k + (s_j - s_k) + \beta_{\text{LiDAR},i} \\ &= s_k + {}^k e_k + (s_j - s_k) + \beta_{\text{LiDAR},i} \\ &= s_j + {}^k e_{k,d} + {}^k e_{k,i} + \beta_{\text{LiDAR},i} \end{aligned} \tag{5}$$

$$\begin{gathered} \mathbf{H}_{\text{LiDAR}} = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad {}^k\mathbf{R}_{\text{LiDAR},d} = {}^k\mathbf{P}_{k,s,d} \\ {}^k\mathbf{R}_{\text{LiDAR},i} = {}^k\mathbf{P}_{k,s,i} + \mathbf{\Sigma}_{\text{LiDAR}} \end{gathered}$$

with $\mathbf{\Sigma}_{\text{LiDAR}}$ the known variance of the LiDAR noise. Note that for this kind of observation, the position estimation error of the measuring vehicle with covariance ${}^k\mathbf{P}_{k,s}$ is added to the uncertainty of the sensor measurement.

Finally, the vehicles exchange their estimates of the system ${}^k S = \left\{{}^k\mathbf{x}_j, {}^k\mathbf{P}_j\right\}_{j\in S}$ with each other. The states estimated by others locally become observations and the covariance matrices of the estimation errors become covariance matrices of the observations. For this, we define a third observation model, where vehicle c expresses itself on vehicle j:

$$\begin{gathered} {}^c\mathbf{y}_{\text{Com},j} = {}^c\mathbf{x}_j, \ \mathbf{H}_{\text{Com}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \\ {}^c\mathbf{R}_{\text{Com},j,d} = {}^c\mathbf{P}_{j,d}, \quad {}^c\mathbf{R}_{\text{Com},j,i} = {}^c\mathbf{P}_{j,i} \end{gathered} \tag{6}$$

This process is described by Algorithm 1 for each vehicle k.

B. Filtering

In this section, we remind the equations of filters that will be studied in Section III-C. First, the Kalman Filter (KF) [9], which is the optimal filter for linear systems affected by

Algorithm 1 Cooperative Perception Algorithm

$^k\mathbf{x}_k, {}^k\mathbf{P}_k \leftarrow \text{Predict}_1(^k\mathbf{x}_k, {}^k\mathbf{P}_k)$ ▷ GNSS
$^k\mathbf{x}_k, {}^k\mathbf{P}_k \leftarrow \text{Update}_1(^k\mathbf{x}_k, {}^k\mathbf{P}_k, {}^k\mathbf{y}_k, \mathbf{R}_{\text{GNSS}}, \mathbf{H}_{\text{GNSS}})$

for $j \in S \setminus k$ **do** ▷ LiDAR
  $^k\mathbf{x}_j, {}^k\mathbf{P}_j \leftarrow \text{Predict}_1(^k\mathbf{x}_j, {}^k\mathbf{P}_j)$
  $^k\mathbf{x}_j, {}^k\mathbf{P}_j \leftarrow \text{Update}_1(^k\mathbf{x}_j, {}^k\mathbf{P}_j, {}^k\mathbf{y}_j, {}^k\mathbf{R}_{\text{LiDAR}}, \mathbf{H}_{\text{LiDAR}})$
end for

for $c \in S \setminus k$ **do** ▷ Cooperative Estimation
  for $j \in S$ **do**
    $^k\mathbf{x}_j, {}^k\mathbf{P}_j \leftarrow \text{Update}_2(^k\mathbf{x}_j, {}^k\mathbf{P}_j, {}^c\mathbf{y}_{\text{Com},j}, {}^c\mathbf{R}_{\text{Com},j}, \mathbf{H}_{\text{Com}})$
  end for
end for

white noises:

$$\begin{aligned}
\mathbf{x}^+ &= \mathbf{F}\mathbf{x} \\
\mathbf{P}^+ &= \mathbf{F}\mathbf{P}\mathbf{F}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T \\
\mathbf{K} &= \mathbf{P}^+\mathbf{H}^T\left(\mathbf{H}\mathbf{P}^+\mathbf{H}^T + \mathbf{R}\right)^{-1} \\
\mathbf{x} &= \mathbf{x}^+ + \mathbf{K}\left(\mathbf{y} - \mathbf{H}\mathbf{x}^+\right) \\
\mathbf{P} &= \left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)\mathbf{P}^+\left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)^T + \mathbf{K}\mathbf{R}\mathbf{K}^T
\end{aligned} \tag{7}$$

The Covariance Intersection Filter (CIF) [13] which is usually given in informational form is here given in a Kalman form for the sake of consistency, according to [2].

$$\begin{aligned}
\mathbf{x}^+ &= \mathbf{F}\mathbf{x} \\
\mathbf{P}^+ &= \mathbf{F}\mathbf{P}\mathbf{F}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T \\
\mathbf{K} &= \tfrac{\mathbf{P}^+}{\omega}\mathbf{H}^T\left(\mathbf{H}\tfrac{\mathbf{P}^+}{\omega}\mathbf{H}^T + \tfrac{\mathbf{R}}{1-\omega}\right)^{-1} \\
\mathbf{x} &= \tfrac{\mathbf{x}^+}{\omega} + \mathbf{K}\left(\tfrac{\mathbf{y}}{1-\omega} - \mathbf{H}\tfrac{\mathbf{x}^+}{\omega}\right) \\
\mathbf{P} &= \left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)\tfrac{\mathbf{P}^+}{\omega}
\end{aligned} \tag{8}$$

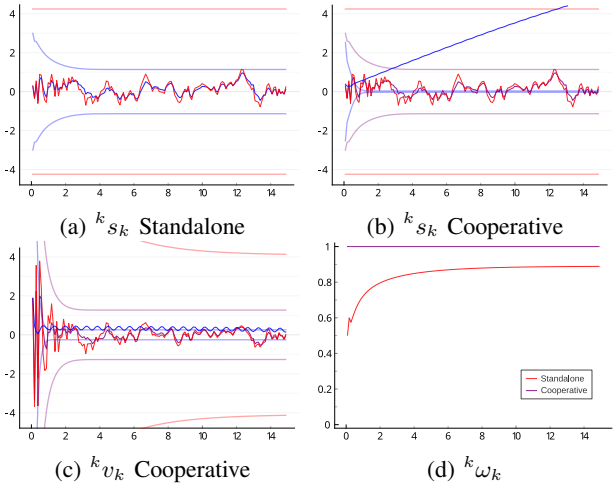
with ω found by minimizing the volume of the resulting covariance matrix:

$$\omega = \underset{0<\omega^+<1}{\arg\min}\, \det(\mathbf{P}) \tag{9}$$

From Equation (8), one might get a sense of how the CIF works compared to a KF: it essentially adds a weighting factor ω to Equation (7), which determines how much of one or the other estimate is taken into account in the convex combination.

Finally, the Split Covariance Intersection Filter (SCIF) [19] in which the covariance $\mathbf{P}$ is split into two components, one containing the estimation error that is known to be perfectly independent $\mathbf{P}_i$ and one containing the estimation error that might contain correlation $\mathbf{P}_d$. Intuitively, the SCIF combines a KF and a CIF, where the KF handles the independent part and the CIF handles the correlated one. There is however a subtlety: the distinction of a dependent and independent part in the model and observation noises $\mathbf{Q}$ and $\mathbf{R}$.

$$\begin{aligned}
\mathbf{x}^+ &= \mathbf{F}\mathbf{x} \\
\mathbf{P}_d{}^+ &= \mathbf{F}\,\mathbf{P}_d\mathbf{F}^T + \mathbf{G}\,\mathbf{Q}_d\mathbf{G}^T \\
\mathbf{P}_i{}^+ &= \mathbf{F}\,\mathbf{P}_i\mathbf{F}^T + \mathbf{G}\,\mathbf{Q}_i\mathbf{G}^T \\
\mathbf{P}^+ &= \tfrac{\mathbf{P}_d{}^+}{\omega} + \mathbf{P}_i{}^+ \\
\mathbf{R} &= \tfrac{\mathbf{R}_d^+}{1-\omega} + \mathbf{R}_i^+ \\
\mathbf{K} &= \mathbf{P}^+\mathbf{H}^T\left(\mathbf{H}\mathbf{P}^+\mathbf{H}^T + \mathbf{R}\right)^{-1} \\
\mathbf{x} &= \mathbf{x}^+ + \mathbf{K}\left(\mathbf{y} - \mathbf{H}\mathbf{x}^+\right) \\
\mathbf{P} &= \left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)\mathbf{P}^+\left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)^T + \mathbf{K}\mathbf{R}\mathbf{K}^T \\
\mathbf{P}_i &= \left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)\mathbf{P}_i{}^+\left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)^T + \mathbf{K}\,\mathbf{R}_i\mathbf{K}^T \\
\mathbf{P}_d &= \mathbf{P} - \mathbf{P}_i
\end{aligned} \tag{10}$$



(a) $^k s_k$ Standalone (b) $^k s_k$ Cooperative

(c) $^k v_k$ Cooperative (d) $^k \omega_k$

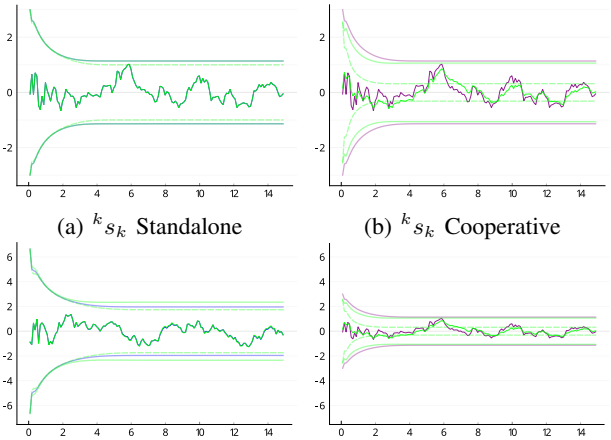
Fig. 1: Estimation errors and $\pm 3\sigma$ confidence domains (meters over time in seconds). KF only updates in blue, CIF only updates in red and K/CIF in purple.

C. Comparative Study

In order to evaluate the performance of different filtering approaches, one with a combination of KF and CIF, one with a SCIF and another with a SCIF in T2TF configuration, Algorithm 1 is applied in simulation. $N = 3$ vehicles move along parallel straight lines with speeds that change over time. Therefore, the constant velocity model has a non-null modelling error. To simplify the analysis of the study, decorrelated white noises have been added to the sensors. The code and parameters to run these simulations can be found on `https://gitlab.utc.fr/-/snippets/52`. In the following comparisons, we study both standalone (with on-board sensors only) and cooperative (with on-board sensors and estimates communicated by others) estimation, respectively represented by Update_1 and Update_2 in Algorithm 1.

1) Manual Kalman and CI Updates

The incorporation of cooperative data introduces information loops and leads to data incest. Figures 1a and 1b illustrates that, while functioning as usual on standalone data, a KF is not able to properly handle cooperative data fusion. The estimate can be seen to over-converge on the wrong value,



(a) ${}^{k}s_k$ Standalone (b) ${}^{k}s_k$ Cooperative

(c) ${}^{c}s_k$ Standalone (d) ${}^{c}s_k$ Cooperative

Fig. 2: Estimation errors and $\pm 3\sigma$ confidence domains. K/CIF in purple and SCIF in green. Continuous lines are the total confidence domains and dashed the independent part.

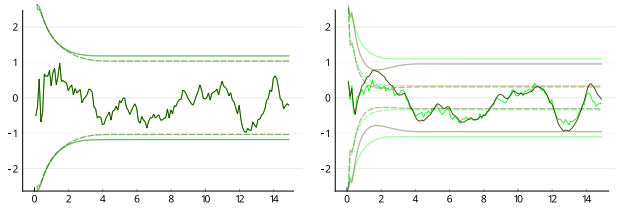
and not being able to correct itself as a consequence.

On the other hand, if the CIF is used for both standalone and cooperative data, such an issue does not arise but the estimation is pessimistic. This is because the filter has a tendency to "jump" on the best estimate during the optimization step when the associated covariances are homogeneously shaped, which is the case in our 1D scenario. Indeed, the intersecting effect of the CI works best when two estimate errors are orthogonal to one another, and the filter displays more of a copying effect otherwise. This effect can be seen more prominently in Figures 1b and 1d, where the cooperative estimate is the same as the standalone because it is the best estimate of all vehicles, as illustrated by the ω that is always found to be 1. We can however observe that indirectly observed quantities such as the velocity can be estimated as well, although slowly in Figure 1c. Indeed, with partial observations, the CIF must deal with infinite uncertainties [20]. It finds an ω that balances the loss of information on the observed part with the small gain on the indirectly observed part introduced by those infinite uncertainties. This can be seen in Figure 1d, where ω starts by taking a lot of the observation, and as the velocity converges, also converges on an higher value.

The best result is obtained when using a KF on standalone data and CIF on cooperative data, which will be denoted K/CIF. In this case, the KF refines its estimation as in Figure 1a and the CIF copies the best KF estimate over all vehicles.

2) Split CI Filtering

In this second evaluation, we compare the K/CIF to letting a SCIF automatically handle the correlation between tracks. In this case, both standalone and cooperative updates are done with Equation (10). In Figure 2a, one can notice that when standalone, the K/CIF and SCIF yield almost identical results. This is because observation errors are entirely independent in both cases, thus constraining the independent part of the estimate. There is however a pro-



(a) ${}^{k}s_k$ Standalone (b) ${}^{k}s_k$ Cooperative

Fig. 3: Estimation errors and $\pm 3\sigma$ confidence domains. Observation based SCIF in green and track based SCIF in brown. Continuous lines are the total confidence domains and dashed the independent part.

gressive separation of the independent and total covariance due to the incorporation of evolution noise on the dependent part instead of the independent. The copying behavior of the CIF is again noticeable in Figure 2b, while the SCIF slightly reduces errors and uncertainties. This is because the remaining independent noise from the standalone steps is filtered by the KF part, while the dependent uncertainty prevents it from over-converging in its CIF part. This is close to the perfect use case for the SCIF but it is not always so reliable. Indeed, the SCIF does not naturally introduces dependent covariance if observation and model errors are independent. This can lead, in a subsequent filtering step, to the same behavior as Figure 1b because the SCIF resolves to a KF when observations are independent. This can be alleviated by the introduction of dependent model noise, such as in [20], [21] that interpreted $\mathbf{P}_d$ as a way to express the temporal correlation of an error.

When observing another vehicle as standalone, in Figure 2c, the two behave slightly differently. The SCIF provides more pessimistic estimates because the observation noise includes the observer's own covariance (see Equation (5)). This leads to the inclusion of dependent covariance in the observation that is properly handled by the SCIF, but not the KF, that ignores the correlation and provides an over-confident estimate.

3) Track-To-Track Fusion (T2TF)

Until this point, sensor observations from a vehicle were directly used in its cooperative estimation. This is a viable option is such a small system but in more complex situations, a filter might have to deal with many sensors that have different modalities and frequencies. To allow for better separation and adaptativeness, the classical solution is to use a Track-To-Track Fusion (T2TF) architecture [22], in which observations are used in one filter and the resulting estimates are used by other filters. Figure 3b illustrates the results of such an architecture compared to the previous one. While the estimates tend to converge too quickly, they usually give similar results to observation based filters. This means that the introduction of dependent covariance is even harder to tune than before, but the trade-off in flexibility is often sufficient to overcome this limitation. In those cases, a possible way to tune the filters is to first consider all covariance as dependent and to find their power using usual KF tuning methods. In a second time, dependent covariance

	RMSE (m)	3σ CD	Coverage
K/CIF	0.30	1.24	99.94%
SCIF	0.27	1.15	99.11%
T2TF	0.29	0.99	99.62%

TABLE I: Average Root Mean Square Error (RMSE), confidence domain and coverage for the different comparisons over 30 simulations. Coverage measures the consistency.

can be gradually transferred from dependent to independent until inconsistent.

4) Summary

Despite requiring manual separation of standalone and cooperative information when implementing the filter, the K/CIF is a viable solution to fuse shared estimates. This is also the case of the SCIF that provides a generic data fusion method easy to program but harder to tune. The use of dependent covariance is powerful tool in a SCIF in order to represent the temporal correlation of its errors. Its introduction by the model noise $\mathbf{Q}_d$ progressively transfers independent estimation error filtered by the optimistic KF, to a dependent estimation error filtered by the cautious CIF, and makes it an important tuning factor to ensure both consistency and small Confidence Domains (CD) in T2TF scenarios. Indeed, as the independent part can over-converge, it must be compensated by dependent uncertainty, and a compromise must be found between a pessimistic standalone estimate and a consistent cooperative estimate. Finally, even when standalone, the SCIF is able to manage partially correlated model and observation noises better than a KF. Table I provides results averaged over 30 runs of the simulation. We have noticed that the introduction of cooperative information is always beneficial in terms of error and uncertainty reduction, and that while less consistent, the SCIF in observation and T2TF scenarios are similar. In terms of computation performance, the CIF takes half a dozen milliseconds to run, due to the optimization step, which is ten times longer than a KF. The SCIF takes twice as much time as it performs more inversions and more computation than a CIF. In addition, whereas there exist variants of the CIF that replace the optimization step with an analytic expression [14], [15], this is not the case for the SCIF.

IV. Experiments

In this section, we evaluate our simulation conclusions on a complete tracking architecture that uses real data. The fusion scheme is first described in Section IV-A then results are showed in Section IV-B.

A. Description

The experiment, illustrated in Figure 4, is composed of three Renault ZOE experimental vehicles from the Heudiasyc laboratory driving around a roundabout in the city of Compiègne, France. Each vehicle is equipped with a Novatel SPAN-CPT Inertial Measurement Unit with Post-Processed Kinematics corrections for centimeter-level positioning, a Velodyne VLP-32C LiDAR sensor and a Mobileye camera for perception.

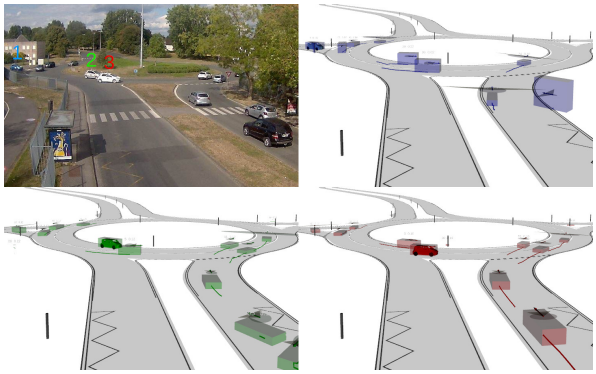


Fig. 4: View of the scene from an infrastructure camera and cyber views of different vehicles. Tracked cars and signs are represented by Oriented Bounding Boxes (OBB) colored in blue for vehicle 1, green for 2 and red for 3. Behind are the surface of the road and signs from an HD map.

Two types of objects are detected in this experiment: cars and road signs. They are easy to detect using existing methods, while representing two important classes of object, moving and static. Cars are represented by a constant yaw rate and acceleration model as described in [23] where $\mathbf{x} = \left[x, y, \theta, v, \omega, a\right]^T$. Signs are represented by static states $\mathbf{x} = \left[x, y, \theta\right]^T$, where θ is the normal to the sign surface. Positions are referenced in an ENU working frame.

1) Observations

In order to study the performance of our tracking solution, a simple perception system that can provide reliable observations has been implemented. The SPAN sensor outputs the ego pose ${}^{Sk}\mathbf{y}_k = \left[x_k, y_k, \theta_k\right]^T$ at 50 Hz. The Mobileye directly outputs a list of objects for many classes and with estimated quantities (such as the velocity) at 35 Hz. Only cars and signs are kept in order to simplify the association process. The observation space is also reduced to oriented bounding boxes (OBB) ${}^{Mk}\mathbf{y}_j = \left[x_j, y_j, w_j, l_j\right]^T$, where x and y are the box centroid, w the width and l the length of an object. Other quantities returned by the Mobileye are estimated in an internal filter whose uncertainties are not given and cannot be modeled, in addition to not always being estimated correctly.

For the Velodyne VLP-32C, two processes take place in parallel on the point cloud. To obtain signs observations, our methodology is similar to [24], in which signs are recognized by their high reflectivity and OBB are fit on them. To obtain cars observations, our methodology is similar to [25], in which non-ground points are clustered and OBB are fit on those belonging to the road surface. Thus, for each point cloud, at 10Hz, those processes output a list of observations ${}^{Lk}\mathbf{y}_j = \left[x_j, y_j, \theta_j, w_j, l_j\right]^T$.

Both sensors have common issues: they do not provide an estimation of their observation uncertainty and provide relative observations. To fix both issues at once, a transformation step is applied to transform observations in the working frame using a vehicle's own pose estimation, similar to Equation (5). The localization error covariance is added the the observation error covariance matrix, which is found

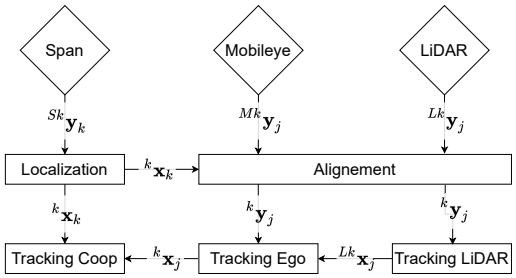


Fig. 5: Tracking Architecture

by empirical study of the innovation. As the camera provides measurements that are tracked in an unknown manner, we define

$$\mathbf{R}_{\text{Mobileye},i} = 0.5\mathbf{\Sigma}_{\text{Mobileye}},\ \mathbf{R}_{\text{Mobileye},d} = 0.5\mathbf{\Sigma}_{\text{Mobileye}}$$
$$\mathbf{\Sigma}_{\text{Mobileye}} = \begin{bmatrix} 4^2 & 0 \\ 0 & 2.25^2 \end{bmatrix} \tag{11}$$

and for the LiDAR, whose processing is snapshot

$$\mathbf{R}_{\text{LiDAR},i} = \mathbf{\Sigma}_{\text{LiDAR}},\ \mathbf{R}_{\text{LiDAR},d} = 0$$
$$\mathbf{\Sigma}_{\text{LiDAR}} = \begin{bmatrix} 0.25^2 & 0 \\ 0 & 0.25^2 \end{bmatrix} \tag{12}$$

2) Tracking

The tracking architecture is described in Figure 5. Each tracking step is composed of a Out-Of-Sequence (OOS) measurement management, that applies the reprocessing algorithm as described in [26]. While this method is heavy, in particular with important delays, it is more optimal than a solution that introduces more model noise. Indeed, when received observations are tracks, they can be predicted to the tracker time to align them temporally [27]. The association between tracks and observations is realized with a gated Mahalanobis distance that includes a penalty when classes are different to avoid matching cars with signs. Finally, the prediction and update steps are realized with Equation (10). In this experiment, all the model noise is considered dependent, and since trackers are cascaded (from sensor, to vehicle, up to cooperation), the power of the model noise is increased at each step.

3) Communication

Vehicles periodically exchange their tracks. As the experiment were realized offline, the communications were simulated following the Cooperative Perception Message (CPM) standard, where communications are limited to 10 Hz. A communication delay was artificially added by waiting a random time $t_{\text{Com}} \sim \mathcal{N}(150, 50)$ ms before reception.

B. Results

To establish those results, the tracks coming from the cooperative tracker are associated and compared to a ground truth when possible. For cars, the SPAN positions of the three vehicles are used and for signs, their positions as stored in an HD map are used. It is to be noted that a vehicle's own estimation is not communicated. This is because the position accuracy is significantly better than the perception, and with all vehicles communicating their own position the impact of perception would not be perceivable.

The 2D errors and uncertainty bounds of one vehicle about the three vehicles and signs are illustrated in Figure 6. Those curves are computed using a similar method to [28], where the error bound is computed along the direction of the error. That way, the estimate can be considered consistent as long as it is within its bound, while also hinting at the scale of the uncertainty. For example, on the third curve, one can notice that although the error does not vary significantly, its uncertainty changes from high to low at times $t = 12$, 18, 28, 38 and 48. This corresponds to when vehicle 3 is directly observed or simply communicated. In Table II, the Root Mean Square Error (RMSE), confidence domains and coverage of estimates from all vehicles are synthesized. The coverage is consistent in all cases but two, when vehicles 2 and 3 observe vehicle 1. This is because for the first half of the experiment, vehicle 1 is static and the used evolution model (constant acceleration) cannot efficiently converge, resulting in an erroneous position estimate. Except for those two cases, this cooperative perception system is likely consistent with both dynamic and static objects. As a comparison, the KF/CI method performed in a similar fashion, if not marginally better, yielding an RMSE of 0.704, a 3σ CD of 3.475 and a coverage of 99.288% for example for Vehicle 2 as seen from Vehicle 1.

V. Conclusion

In this paper, different filtering solutions based on the Covariance Intersection Filter (CIF) or Split Covariance Intersection Filter (SCIF) have been studied. Through a case study on a simple example with several vehicles, we have shown how measurement, model and estimation errors are involved in a cooperative perception problem. Through this study, we have implemented a SCIF that has shown a good ability to estimate quantities not measured directly while maintaining the consistency of the estimates even when this filter is used to perform track-to-track data fusion in a cooperative framework in which there are information exchange cycles. Thanks to the good knowledge we gained in simulation on both modeling and tuning, we were able to easily implement a SCIF on real data in a realistic scenario with 3 vehicles exchanging their perceptions of each others and of fixed signs. This work has thus shown that, for a SCIF to work properly, great importance must be given to the modeling of the sources of uncertainty and that it is necessary to inject a dependent model noise covariance, especially in T2TF architectures. In future work, this filter will be used as a base for developing new approaches for cooperative perception.

Acknowledgment

This work was carried out within SIVALab, a shared laboratory between Renault and Heudiasyc (UTC/CNRS).

References

[1] F. Seeliger and K. Dietmayer, "Inter-vehicle information-fusion with shared perception information," in *IEEE Intelligent Transportation Systems Conference*, 2014.

	Vehicle 1			Vehicle 2			Vehicle 3			Signs		
	RMSE	3σ CD	Coverage	RMSE	3σ CD	Coverage	RMSE	3σ CD	Coverage	RMSE	3σ CD	Coverage
Vehicle 1	0.031	0.518	100.000%	0.674	3.727	99.526%	0.739	4.004	99.525%	0.264	0.738	99.360%
Vehicle 2	1.113	2.206	88.285%	0.071	0.682	100.000%	0.856	3.370	99.534%	0.261	0.729	99.303%
Vehicle 3	0.947	2.066	94.305%	0.671	3.931	100.000%	0.065	0.609	100.000%	0.245	0.710	99.709%

TABLE II: Root Mean Square Error (RMSE) (in meters), CD and coverage of the three vehicles cooperative estimates line-wise.

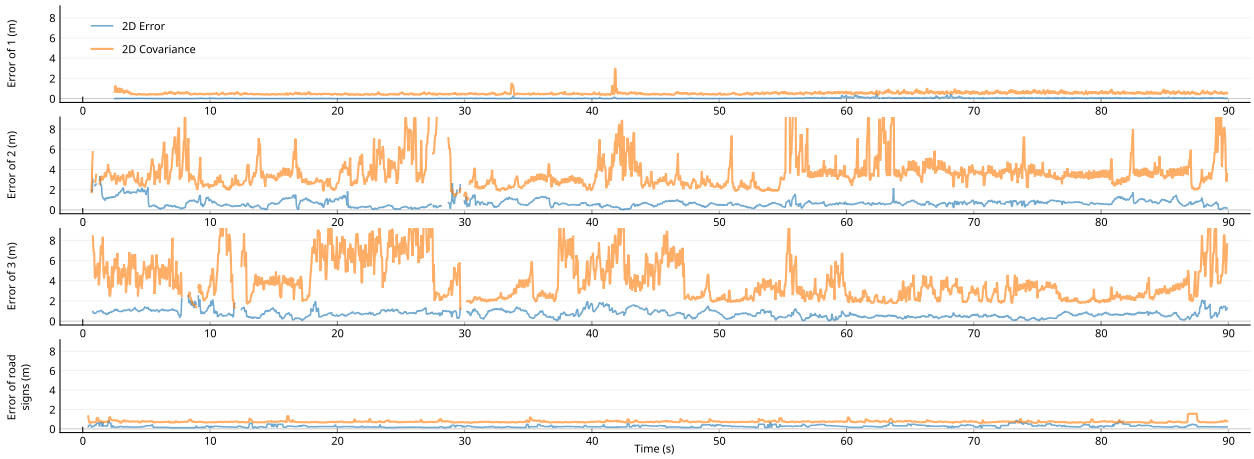


Fig. 6: Estimation 2D errors and 3σ confidence domains from vehicle 1 about other vehicles and road signs

[2] E. Héry, P. Xu, and P. Bonnifait, "Consistent Decentralized Cooperative Localization for Autonomous Vehicles using LiDAR, GNSS and HD maps," *Journal of Field Robotics*, 2021.

[3] H. Li and F. Nashashibi, "Multi-vehicle cooperative perception and augmented reality for driver assistance: A possibility to 'see' through front vehicle," in *IEEE Conference on Intelligent Transportation Systems*, 2011.

[4] S. Sridhar and A. Eskandarian, "Cooperative perception in autonomous ground vehicles using a mobile-robot testbed," *IET Intelligent Transport Systems*, 2019.

[5] T.-H. Wang, S. Manivasagam, M. Liang, B. Yang, W. Zeng, and R. Urtasun, "V2VNet: Vehicle-to-Vehicle Communication for Joint Perception and Prediction," in *European Conference on Computer Vision*, 2020.

[6] H.-J. Günther, B. Mennenga, O. Trauer, R. Riebl, and L. Wolf, "Realizing collective perception in a vehicle," in *IEEE Vehicular Networking Conference*, 2016.

[7] M. Gabb, H. Digel, T. Müller, and R.-W. Henn, "Infrastructure-supported Perception and Track-level Fusion using Edge Computing," in *IEEE Intelligent Vehicles Symposium*, 2019.

[8] Y. Bar-Shalom and L. Campo, "The Effect of the Common Process Noise on the Two-Sensor Fused-Track Covariance," *IEEE Transactions on Aerospace and Electronic Systems*, 1986.

[9] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Trans. of the ASME–Journal of Basic Engineering*, 1960.

[10] S. Nilsson and A. Klekamp, "A Comparison of Architectures for Track Fusion," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015.

[11] M. Vasic and A. Martinoli, "A Collaborative Sensor Fusion Algorithm for Multi-object Tracking Using a Gaussian Mixture Probability Hypothesis Density Filter," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015.

[12] K. Yang, Y. Bar-Shalom, and K. Chang, "Information Matrix Fusion for Nonlinear, Asynchronous and Heterogeneous Systems," in *International Conference on Information Fusion*, 2019.

[13] S. Julier and J. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *American Control Conf.*, 1997.

[14] D. Franken and A. Hupper, "Improved fast covariance intersection for distributed data fusion," in *International Conference on Information Fusion*, 2005.

[15] M. Hurley, "An information theoretic justification for covariance intersection and its generalization," in *International Conference on Information Fusion*, 2002.

[16] H. Li, F. Nashashibi, B. Lefaudeux, and E. Pollard, "Track-to-track fusion using split covariance intersection filter-information matrix filter (SCIF-IMF) for vehicle surrounding environment perception," in *IEEE Conference on Intelligent Transportation Systems*, 2013.

[17] B. Duraisamy, T. Schwarz, and C. Wöhler, "Track level fusion algorithms for automotive safety applications," in *International Conf. on Signal Processing, Image Processing and Pattern Recognition*, 2013.

[18] M. Ambrosin, I. J. Alvarez, C. Buerkle, L. L. Yang, F. Oboril, M. R. Sastry, and K. Sivanesan, "Object-level Perception Sharing Among Connected Vehicles," in *IEEE Intelligent Transportation Systems Conference*, 2019.

[19] S. Julier and J. Uhlmann, "General Decentralized Data Fusion with Covariance Intersection (CI)," in *Handbook of Multisensor Data Fusion, Theroy and Practice*, 2001.

[20] H. Li, F. Nashashibi, and M. Yang, "Split Covariance Intersection Filter: Theory and Its Application to Vehicle Localization," *IEEE Transactions on Intelligent Transportation Systems*, 2013.

[21] C. Pierre, R. Chapuis, R. Aufrère, J. Laneurit, and C. Debain, "Range-Only Based Cooperative Localization for Mobile Robots," in *International Conference on Information Fusion*, 2018.

[22] M. Aeberhard, S. Schlichtharle, N. Kaempchen, and T. Bertram, "Track-to-Track Fusion With Asynchronous Sensors Using Information Matrix Fusion for Surround Environment Perception," *IEEE Transactions on Intelligent Transportation Systems*, 2012.

[23] M. A. Khan, "Comparison of Track to Track fusion methods for nonlinear process and measurement models," in *Sensor Data Fusion: Trends, Solutions, Applications*, 2019.

[24] A. Welte, P. Xu, P. Bonnifait, and C. Zinoune, "Improved Data Association Using Buffered Pose Adjustment for Map-Aided Localization," *IEEE Robotics and Automation Letters*, 2020.

[25] E. Bernardi, S. Masi, P. Xu, and P. Bonnifait, "High Integrity Lane-level Occupancy Estimation of Road Obstacles Through LiDAR and HD Map Data Fusion," in *IEEE Intelligent Vehicles Symposium*, 2020.

[26] M. Muntzinger, M. Aeberhard, S. Zuther, M. Mählisch, M. Schmid, J. Dickmann, and K. Dietmayer, "Reliable automotive pre-crash system with out-of-sequence measurement processing," in *IEEE Intelligent Vehicles Symposium*, 2010.

[27] C. Allig and G. Wanielik, "Alignment of Perception Information for Cooperative Perception," in *IEEE Intelligent Vehicles Symp.*, 2019.

[28] Z. Tao and P. Bonnifait, "Sequential Data Fusion of GNSS Pseudoranges and Dopplers With Map-Based Vision Systems," *IEEE Transactions on Intelligent Vehicles*, 2016.