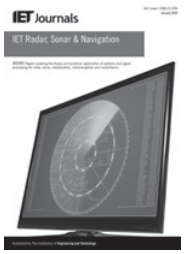
Published in IET Radar, Sonar and Navigation
Received on 26th January 2014
Revised on 6th April 2014
Accepted on 23rd April 2014
doi: 10.1049/iet-rsn.2014.0037



ISSN 1751-8784

Online clutter estimation using a Gaussian kernel density estimator for multitarget tracking

Xin Chen[1], *Ratnasingham Tharmarasa*[1], *Thia Kirubarajan*[1], *Mike McDonald*[2]

[1]*Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada*
[2]*Surveillance Radar Group, Radar Systems, Defence R&D Canada – Ottawa, Ottawa, ON, Canada*
E-mail: chenx73@univmail.cis.mcmaster.ca

Abstract: In this study, the spatial distribution of false alarms is assumed to be a non-homogeneous Poisson point (NHPP) process. Then, a new method is developed under the kernel density estimation (KDE) framework to estimate the spatial intensity of false alarms for the multitarget tracking problem. In the proposed method, the false alarm spatial intensity estimation problem is decomposed into two subproblems: (i) estimating the number of false alarms in one scan and (ii) estimating the variation of the intensity function value in the measurement space. Under the NHPP assumption, the only parameter that needs to be estimated for the first subproblem is the mean of false alarm number, and the empirical mean is used here as the maximum likelihood estimate of that parameter. Then, for the second subproblem, an online multivariate local adaptive Gaussian kernel density estimator is proposed. Furthermore, the proposed estimation method is seamlessly integrated with widely used multitarget trackers, like the joint integrated probabilistic data association algorithm and the multiple hypotheses tracking algorithm. Simulation results show that the proposed KDE-based method can provide a better estimate of the false alarm spatial intensity and help the multitarget trackers yield superior performance in scenarios with spatially non-homogeneous false alarms.

1 Introduction

In many multitarget tracking systems, the false alarms (i.e. clutter points) reported by the sensor are non-uniformly distributed in the measurement space with an unknown distribution. On the other hand, many multitarget tracking systems require the spatial distribution of false alarms for the purpose of measurement-to-track association, track state update and new track initialisation. Usually, the false alarms are modelled by a non-homogeneous Poisson point process (NHPP) in the measurement space [1] and to fully describe the distribution of an NHPP process, only the corresponding spatial intensity function is needed [2]. Thus, multitarget tracking systems need the spatial intensity function of false alarms as a priori information. In addition, because the measurements originating from targets and the false alarms are mixed together and indistinguishable before being processed by the multitarget tracking algorithm [3, 4], the output of the multitarget tracker should be used when the spatial intensity of false alarms is estimated. Furthermore, it would be desirable to integrate the clutter estimation method into existing multitarget trackers, so that current multitarget tracking systems can continue to be used.

In general, there are two different types of methods that are capable of estimating the spatial intensity of false alarms. In the first type, it is assumed that the spatial distribution of false alarms is uniform inside track's validation gate. Then the measurements inside the validation gates, along with the estimates of target states, are sent to the spatial intensity estimator. In [1], the multitarget trackers use the sample spatial intensity obtained from the set of measurement points inside the validation gate as the spatial intensity of false alarms. However, the method in [1] heavily depends on the validation gate size, and in many multitarget trackers, the validation gate is generated solely based on the innovation matrix and the gate probability threshold [1], so it is quite possible that the validation gate is not suitable for the spatial intensity estimation problem. Furthermore, this estimation method is biased, because it does not distinguish target originated measurements from false alarms. In [5], 'track perceivability', the target existence probability conditional on all previous measurements [6], was used to yield the unbiased estimate of the spatial intensity of clutter by probabilistically excluding the measurements originated from targets in the latest measurement set. It should be noted that these methods are only able to estimate the spatial intensity of false alarms for detections inside track validation gates. However, because there are detections that do not fall in any track's validation gate, the values for the spatial intensity of false alarms outside validation gates are also required, especially for the cost calculation of newly initialised tracks [7].

The second type of clutter estimation methods tries to estimate the clutter spatial intensity for any detection point inside the measurement space. One widely used method is the classic clutter map method [8]. This method needs the operator to manually divide the measurement space into bins and assumes that the clutter intensity is constant inside each bin. Then, for bin i in the clutter map, the clutter

spatial intensity is calculated as

$$\hat{\lambda}(z) = \frac{\text{number of false alarms in bin } i \text{ so far}}{A_i \times \text{number of measurement frames}} \quad (1)$$

where A_i is the size of bin i. The classic clutter map is mathematically similar to a multivariate histogram method, and according to [9], the histogram is not suitable for data points with two or more dimensions.

Another method that is widely used to estimate the spatial intensity of false alarms over the measurement space is the clutter spatial intensity estimator using the nearest neighbour detection point [8, 10, 11]. The nearest neighbour-based estimator also needs the operator to manually partition the measurement space into bins and the clutter spatial distribution in each bin is assumed to be Poisson. Then, for bin i, its spatial intensity of false alarms is set as the inverse of a mathematical distance metric from bin i to the nearest detection in the current frame of measurements. After using the latest frame of measurements to estimate the spatial intensity of false alarms, a time-averaging operation will be taken to smooth the estimate over time. The mathematical distance metric used here is usually different from the Euclidean distance. The former is the volume of a well-defined shape, which centres around the corresponding bin and touches the selected detection point. From a probability density function estimation perspective, the nearest-neighbour intensity estimator presented in [8, 10, 11] belongs to category of the K-nearest neighbour density estimator [9]. It should be noted that the K-nearest neighbour density estimator relies on the assumption that the data points in and around bin i are homogeneously and isotropically distributed, otherwise the inverse of the mathematical distance between bin i and the nearest detection point will not be equal to the spatial intensity anymore [12]. In addition, for widely used non-linear-sensors like Doppler radar and two-dimensional (2D) radar, it is unclear what shape should be used to calculate the mathematical distance in their measurement space. Furthermore, the output of this clutter intensity estimator has a discontinuous 'block' nature, just like the multivariate histogram.

In [13], an approximate Bayesian estimation method, which also tries to provide the estimate of false alarm spatial intensity over the whole measurement space, is proposed to handle the non-homogeneous clutter background using the random finite set (RFS) theory. In that paper, besides the set of measurement points, $\boldsymbol{Z} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_m\}$, and the set of targets, $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, the set of clutter generators, $\boldsymbol{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_l\}$, is defined in a space $\mathfrak{C}$, just like in [14, 15]. Here $\boldsymbol{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_l\}$ is assumed to produce the non-homogeneous false alarms and $\mathfrak{C}$ is separate from $\mathfrak{X}$ and $\mathfrak{Z}$, which represents the space of target state and the space of measurement, respectively. In addition, by assuming that the clutter generator $\boldsymbol{c}_i$ produces at most one clutter point $\boldsymbol{z}_i^{(C)}$ and $\boldsymbol{C}$ is an NHPP process, the estimation of false alarm spatial intensity is transformed into an estimation problem for the spatial intensity of clutter generator, and a probability hypothesis density (PHD) filter [16] is derived for clutter generator. To derive a closed-form expression to iteratively update the PHD function of clutter generator, it is further assumed in [13] that:

- The measurement model connecting $\boldsymbol{c}_i$ and $\boldsymbol{z}_i^{(C)}$ has a likelihood function in the form of a Gaussian function, whose mean and variance are both unknown and need to be estimated.
- For RFS $\boldsymbol{C}$, its PHD function can be expressed as a mixture of the normal-Wishart probability density function.

In [17], also based on the assumption that one clutter generator produces at most one clutter point, a PHD update equation, which is similar to the one presented in [13], is proposed for the clutter generator, but with a simpler derivation. In [18], an extension of the method presented in [13] is proposed, so it can be used with the classic multitarget trackers [3, 4].

In this paper, a multivariate Gaussian kernel density estimator (KDE), which is capable of handling the measurement origin ambiguity, is proposed to obtain the spatial intensity of false alarms. The proposed KDE is able to estimate the spatial intensity of false alarms for the whole measurement space. The proposed method is locally adaptive, which means that the bandwidth matrix can vary from one kernel to another, and the weights of kernels can be automatically adjusted according to the Bayesian principal. In addition, it is capable of working online and the total number of the kernel components will not increase without a bound. Furthermore, the bandwidth matrix is automatically determined by the proposed estimator from the data set, and that matrix is not constrained to be diagonal. In addition, to accelerate the optimisation of the bandwidth matrix, an expression for the gradient of the cost function, which is used in the cross-validation (CV) technique [19], with respect to the bandwidth matrix is derived, and that expression does not contain any matrix inversion operation.

Compared with the methods that only use the measurements inside the validation gates, the proposed method can provide the clutter spatial intensity estimate for any point in the measurement space. Compared with the classic clutter map method or the nearest neighbour method, the proposed method does not require the measurement space to be manually divided into sectors, and the output of the proposed method is guaranteed to be continuous and does not have the discontinuous 'block' nature. In addition, unlike the nearest neighbour method, the proposed method does not rely on the calculation of the mathematical distance, and can be used for non-linear sensors, like 2D radar and Doppler radar. Compared with the clutter generator method in [13, 18], the proposed method does not rely on the assumed clutter generator and no assumption is made related to the dynamic evolution of the clutter generator. In addition, the proposed method does not rely on the amplitude information of clutter, so it can be used to handle the clutter with Gaussian distributed amplitude as well as the clutter with non-Gaussian distributed amplitude.

Preliminary results were presented in a conference paper [20], whereas the current paper provides more technical details and simulation results. The remaining part of this paper is organised as follows. A general background on the multivariate KDE is given in Section 2. In Section 3, a multivariate KDE is presented to estimate the spatial intensity of false alarms for the multitarget tracking problem. The results of simulation experiments are shown in Section 4. Finally, Section 5 concludes this paper.

2 Background on kernel density estimator

In this section, a general introduction to the multivariate KDE is given.

doi: 10.1049/iet-rsn.2014.0037

For a d-dimensional random sample set $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ that are drawn from a probability density function $f(\boldsymbol{x})$, the KDE for $f(\boldsymbol{x})$ is defined as

$$\hat{f}_H(\boldsymbol{x}) = \sum_{i=1}^{n} w_i K_H(\boldsymbol{x} - \boldsymbol{x}_i) \tag{2}$$

where $K(\boldsymbol{x})$ is the multivariate kernel, $\boldsymbol{H}$ is the bandwidth matrix, $K_H(\boldsymbol{x}) = |\boldsymbol{H}|^{-1/2} K(\boldsymbol{H}^{-1/2}\boldsymbol{x})$, $0 < w_i < 1$ and $\sum_i w_i = 1$. The multivariate kernel $K(\boldsymbol{x})$ is a spherically symmetric probability density function that has the following properties [9]

$$\int K(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = 1 \tag{3}$$

$$\int \boldsymbol{x} K(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = 0 \tag{4}$$

$$\int \boldsymbol{x}\boldsymbol{x}^{\mathrm{T}} K(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} \neq \boldsymbol{0} \tag{5}$$

and $\boldsymbol{H}$ is a symmetric and positive-definite matrix [21]. A good property for $\hat{f}_H(\boldsymbol{x})$ defined in (2) is that all the continuity and differentiability properties of the kernel function K will be inherited. Thus, if K is, for example, the normal density function, then $\hat{f}_H(\boldsymbol{x})$ will be a smooth curve having derivative of all orders. On the contrary, for the histograms or the nearest neighbour density estimator, usually their estimated probability density function $f(\cdot)$ is not smooth.

The performance of the multivariate KDE defined in (2) strongly depends on the choice of the bandwidth matrix $\boldsymbol{H}$, whereas the exact type of the kernel function $K(\cdot)$ is much less important [9]. For example, in a univariate case, where the kernel function can be written as $K[(x–y)/h]$ and the true probability density function is $f(x)$, the expectation and the variance of the probability density function estimate from n samples is [9]

$$E\hat{f}_h(x) = \int \frac{1}{h} K\left(\frac{x-y}{h}\right) f(y)\, \mathrm{d}y \tag{6}$$

$$\begin{aligned} \mathrm{Var}\,\hat{f}_h(x) = \left(\sum_{i=1}^{n} w_i^2\right) \left\{ \int \frac{1}{h^2} K^2\left(\frac{x-y}{h}\right) f(y)\, \mathrm{d}y \right. \\ \left. - \left[\frac{1}{h}\int K\left(\frac{x-y}{h}\right) f(y)\, \mathrm{d}y\right]^2 \right\} \end{aligned} \tag{7}$$

Based on (6), the expectation of $\hat{f}$ is equal to the convolution of f with the kernel K, scaled by the bandwidth h, that is, a smoothed version of f. Furthermore, the expectation of $\hat{f}$ deterministically depends on the choice the bandwidth, but not directly related to the sample size. This characteristic is common for almost all density estimation methods, that is, the estimate is always of the form

$$\begin{aligned} \text{estimate} = \text{smoothed version of the true density} \\ + \text{random estimation error} \end{aligned} \tag{8}$$

and the first term in the above is directly related to parameters used in the kernel density estimation method, but not the sample size. In other words, the bias in $\hat{f}$ does not directly depend on the sample size, and increasing the size of the sample set alone does not guarantee an unbiased estimate [9]. For a univariate KDE, to obtain an asymptotically consistent estimate $\hat{f}$ at a single point x, the kernel function K should have the following characteristics:

- The kernel function K is a bounded Borel function satisfying

$$\int |K(x)|\, \mathrm{d}x < \infty, \quad \int K(x)\, \mathrm{d}x = 1 \tag{9}$$

and

$$|xK(x)| \to 0 \text{ as } |x| \to \infty \tag{10}$$

- h_n is the bandwidth used in the KDE when the sample size is n, satisfies

$$h_n \to 0 \text{ and } nh_n \to \infty \text{ as } n \to \infty \tag{11}$$

Under above assumptions, provided that the true probability density function f is continuous at x, there is

$$\hat{f}(x) \to f(x) \text{ in probability as } n \to \infty \tag{12}$$

In other words, to obtain an asymptotically unbiased estimate, the bandwidth for the kernel must approach 0 as $n \to \infty$, but with a speed slower than n^{-1} [9]. For a d-dimensional multivariate KDE, the same principle can be applied. For example, under the constraint that the bandwidth matrix is diagonal, the optimal bandwidth along each coordinate directions should decrease as $O[n^{-1/(4+d)}]$ and the converging speed for the estimate $\hat{f}$ is $O[n^{-4/(4+d)}]$ if the optimal bandwidth matrix is used [19].

3 Estimating clutter spatial intensity with a kernel density estimator

In this section, a multivariate kernel density estimation algorithm is proposed to estimate the spatial intensity of false alarms. To use the KDE, the original false alarm spatial intensity estimation problem is decomposed into two sub-problems: (i) estimating the probability distribution of the number of false alarms and (ii) estimating the spatial variation of clutter intensity. In addition, because a measurement can be generated by a target or a clutter, the output of the multitarget tracker is integrated into the proposed clutter spatial intensity estimator. Furthermore, because the sensor continuously provides measurements, a modification is made to the KDE to handle the continuously arriving measurements and to find the optimal bandwidth matrix numerically.

3.1 Decompose the problem of intensity estimation

In this paper, the false alarms are assumed to be an NHPP, and the problem to estimate spatial intensity of false alarms is decomposed into two independent sub-problems:

(1) Estimating the probability mass function for the number of false alarms per scan.
(2) Estimating the non-homogeneity of the spatial distribution of false alarms.

Since the false alarms are assumed to be an NHPP, the number of false alarms in each scan is mutually independent and follows the same Poisson distribution [1]. Thus, for the first problem listed in the above, only λ, the mean of the number of false alarms per scan, has to be estimated. Assuming that there are n scans, the maximum likelihood estimator for λ is [2]

$$\hat{\lambda}_{\mathrm{MLE}} = \frac{\sum_{i=1}^{n} k_i}{n} \tag{13}$$

where k_i represents the number of false alarms in scan i.

For the non-homogeneity of the spatial distribution of false alarms, from the definition of the intensity function of point process, one has [2]

$$\lambda = \int_{\boldsymbol{S}} c(\boldsymbol{z})\,\mathrm{d}\boldsymbol{z} \tag{14}$$

where $\boldsymbol{S}$ represents the measurement space, $c(\boldsymbol{z})$ is the clutter spatial intensity function and λ is the mean number of false alarms per scan, then the normalised clutter spatial intensity function can be defined as

$$f(\boldsymbol{z}) = \frac{1}{\lambda} c(\boldsymbol{z}) \tag{15}$$

To obtain the spatial variation of the clutter intensity, all we need to know is $f(\boldsymbol{z})$. In addition, $f(\boldsymbol{z})$ can be considered as a probability density function because

$$\int_{\boldsymbol{S}} f(\boldsymbol{z})\,\mathrm{d}\boldsymbol{z} = 1, \quad f(\boldsymbol{z}) \geq 0 \tag{16}$$

Thus, the estimation of the variation of the clutter spatial intensity becomes a probability density estimation problem. That is, given the measurement sets $\boldsymbol{Z}_i = \left\{\boldsymbol{z}_1^{(i)}, \ldots, \boldsymbol{z}_{n_i}^{(i)}\right\}$, $(i = 1, 2, \ldots, n)$, a probability density function $\hat{f}(\boldsymbol{z})$ has to be estimated such that it can best fit the given measurement sets. Once $\hat{f}(\boldsymbol{z})$ and $\hat{\lambda}$ are obtained, the estimate for the clutter spatial intensity function can be calculated as

$$\hat{c}(\boldsymbol{z}) = \hat{\lambda}\hat{f}(\boldsymbol{z}) \tag{17}$$

3.2 Solve measurement origin ambiguity and integrate with JIPDA/MHT tracker

The main difference between the clutter spatial intensity estimator considered in this paper and the classic probability density estimator is that for the former, some samples are generated by targets and they should not be used in the clutter spatial intensity estimator, whereas in the latter all measurements are drawn from the unknown probability density. Furthermore, in the clutter spatial intensity estimation problem, given a set of measurement $\{\boldsymbol{z}_i\}$, $i = 1, 2, \ldots, N$, it is usually impossible to tell whether $\boldsymbol{z}_i$ is a clutter or a target-originated measurement with 100% confidence, even after the multitarget tracking process. Thus, to avoid overestimating the clutter spatial intensity, the output of the multitarget tracker has to be used as an input to the clutter intensity estimator, and the measurement origin ambiguity has to be solved statistically.

To solve the measurement origin ambiguity, many widely used classic multitarget tracking algorithms (e.g. multiple hypotheses tracking (MHT) algorithm and joint integrated probabilistic data association (JIPDA) algorithm) enumerate the association events, and inside each association event, a one-to-one assignment between the measurement and the track is made. The probability for each association event being true is updated recursively by the multitarget tracker. To solve the measurement origin ambiguity, the proposed clutter spatial intensity estimator uses the association event and its corresponding probability calculated by the multitarget tracker, because given the association event χ_i, the source of the measurement is totally determined. In other words, the association event answers the question that whether the measurement $\boldsymbol{z}_k$ is a clutter or generated by a target with a certain probability. For the measurement set $\boldsymbol{Z}^{(k)} = \left\{\boldsymbol{z}_1^{(k)}, \ldots, \boldsymbol{z}_{n_k}^{(k)}\right\}$ at time k, the association event χ_i defines a set $\boldsymbol{Z}_{i,C}^{(k)} \subseteq \boldsymbol{Z}^{(k)}$, where all measurements in $\boldsymbol{Z}_{i,C}^{(k)}$ are clutter. Thus, $\boldsymbol{Z}_{i,C}^{(k)}$ can be used as the input to the multivariate KDE to estimate the normalised clutter intensity conditional on the association event χ_i. Given $\boldsymbol{Z}_{i,C}^{(k)}$, the output of the KDE is $c^{(i)}(\boldsymbol{z})$, the estimate of the normalised spatial intensity function conditional on the association event χ_i.

Furthermore, it should be noted that the following relationship between the conditional expectation and the unconditional expectation always holds

$$E[Z] = E[E[Z|X]] \tag{18}$$

and the clutter intensity $c(\boldsymbol{z})$ is defined as the average number of false detections falling in an infinitesimal area centred at $\boldsymbol{z}$. Thus, one has

$$c(\boldsymbol{z}) = \sum_{i=1}^{n} P(\chi_i) c^{(i)}(\boldsymbol{z}) \tag{19}$$

where $P(\chi_i)$ is the probability of the association event χ_i, which is calculated by the classic multitarget tracker [3, 4]. The integrated procedure of clutter estimation and

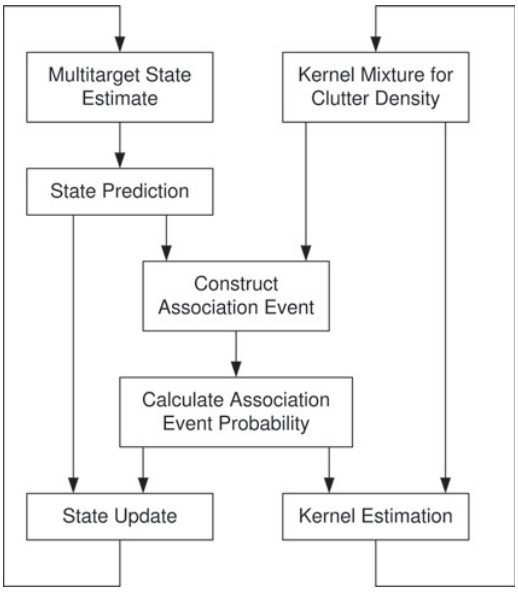


Fig. 1 *Integrated tracking and clutter estimation for JIPDA and MHT trackers*

doi: 10.1049/iet-rsn.2014.0037

multitarget tracking is demonstrated by the flowchart in Fig. 1. Note that, in this method, the total number of measurement-to-clutter events will increase exponentially with the cardinality of the measurement set $\boldsymbol{Z}_k$. Thus, enumerating all measurement-to-clutter events for $\boldsymbol{Z}_k$ is usually infeasible. However, by using each measurement's posterior target originated probability, Murty's K-best algorithm [22] is able to find the most probable K measurement-to-clutter events.

3.3 Choose bandwidth for kernel density estimator

For the multivariate KDE, the most important design parameter is the bandwidth matrix $\boldsymbol{H}$ and it should be chosen optimally under some criteria. One commonly used criterion to determine an optimal $\boldsymbol{H}$ is the mean integrated squared error (MISE) defined as

$$\mathrm{MISE}\{\hat{f}_H(\boldsymbol{z})\} = E_f\left[\int \left(\hat{f}_H(\boldsymbol{z}) - f(\boldsymbol{z})\right)^2 \mathrm{d}\boldsymbol{z}\right] \tag{20}$$

which can be decomposed into the integrated squared bias and the integrated point-wise variance

$$\mathrm{MISE}\{\hat{f}_H(\boldsymbol{z})\} = \int \left\{E_f\left[\hat{f}_H(\boldsymbol{z})\right] - f(\boldsymbol{z})\right\}^2 \mathrm{d}\boldsymbol{z} + \int \mathrm{Var}_f\left[\hat{f}_H(\boldsymbol{z})\right] \mathrm{d}\boldsymbol{z} \tag{21}$$

The optimal bandwidth matrix is the one that can minimise MISE and can be chosen through CV technique [9, 21].

In the CV approach, instead of using (20) directly, the following score function is minimised with respect to $\boldsymbol{H}$

$$M_0(\boldsymbol{H}) = \int \hat{f}_H(\boldsymbol{z})^2 \mathrm{d}\boldsymbol{z} - 2n^{-1} \sum_{i=1}^{n} \hat{f}_H^{(-i)}(\boldsymbol{z}_i) \tag{22}$$

where $\hat{f}_H^{(-i)}(\cdot)$ is defined as the estimated density function from all available n data points except $\boldsymbol{z}_i$

$$\hat{f}_H^{(-i)}(\boldsymbol{z}) = \frac{1}{1-w_i} \sum_{j=1, j\neq i}^{n} w_j K_H(\boldsymbol{z} - \boldsymbol{z}_j) \tag{23}$$

The reason behind using $M_0(\boldsymbol{H})$ is that the expectation of $M_0(\boldsymbol{H})$ is equal to MISE minus a constant term. If the minimiser of M_0 is not far away from the minimiser of $E(M_0)$, then minimising M_0 should give a bandwidth matrix close to the optimal one [9]. Further define K_H^* as the convolution of the kernel K_H with itself, then (22) becomes

$$\begin{aligned} M_0(\boldsymbol{H}) = &\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j K_H^*(\boldsymbol{z}_i - \boldsymbol{z}_j)\right\} \\ &- 2n^{-1}\left\{\sum_{i=1}^{n} \frac{1}{1-w_i} \sum_{j=1, j\neq i}^{n} w_j K_H(\boldsymbol{z}_i - \boldsymbol{z}_j)\right\} \end{aligned} \tag{24}$$

When K_H is a Gaussian kernel with zero mean and covariance matrix $\boldsymbol{H}$, (24) can be calculated quite easily, because in this case K_H^* is also a Gaussian kernel with zero mean and covariance matrix $2\boldsymbol{H}$. To find the optimal $\boldsymbol{H}$, a numerical minimisation algorithm has to be used on $M_0(\boldsymbol{H})$ with an over-smoothed bandwidth matrix as the starting initial [23].

To achieve a reliable performance, usually the numerical optimisation algorithm requires the gradient of the objective function. In addition, to make the numerical optimisation procedure stable, the matrix inverse operation should be avoided. For Gaussian kernel, defining the precision matrix $\boldsymbol{S}$ as the inverse of $\boldsymbol{H}$, then because there is a one-to-one correspondence between $\boldsymbol{S}$ and $\boldsymbol{H}$, minimising $M_0(\boldsymbol{S})$ over $\boldsymbol{S}$ is equivalent to minimising $M_0(\boldsymbol{H})$ over $\boldsymbol{H}$. With $\boldsymbol{S}$, the Gaussian kernel locating at $\boldsymbol{z}_j$ becomes

$$\begin{aligned} K_{\boldsymbol{S}}(\boldsymbol{z}) &= (2\pi)^{-m/2} |\boldsymbol{S}|^{1/2} \exp\left\{-\frac{1}{2}\boldsymbol{\nu}^{\mathrm{T}} \boldsymbol{S} \boldsymbol{\nu}\right\} \\ &= (2\pi)^{-m/2} |\boldsymbol{S}|^{1/2} \exp\left\{-\frac{1}{2}\mathrm{tr}\left[(\boldsymbol{\nu}\boldsymbol{\nu}^{\mathrm{T}})\boldsymbol{S}\right]\right\} \end{aligned} \tag{25}$$

where $\mathrm{tr}(\cdot)$ means the matrix trace, $\boldsymbol{\nu} = \boldsymbol{z} - \boldsymbol{z}_j$ and the dimension of $\boldsymbol{z}$ is m. In the above equation, there is no matrix inverse anymore.

However, as the inverse of covariance matrix $\boldsymbol{H}$, matrix $\boldsymbol{S}$ also needs to be symmetric. Thus, general matrix derivative rules do not apply to $\boldsymbol{S}$ because of its special structure [24]. To relax the symmetric constraint, in the proposed method, defining $\boldsymbol{S} = \boldsymbol{\Xi}^T\boldsymbol{\Xi}$ and optimise CV score function with respect to $\boldsymbol{\Xi}$ under the constraint that $\boldsymbol{\Xi}$ is invertible [i.e. $\leq \det(\Xi)^2 > 0$] [21]. In addition, in this paper, by using the following facts [24, 25]

$$\begin{aligned} \frac{\mathrm{d}\det(\boldsymbol{\Xi}^{\mathrm{T}}\boldsymbol{\Xi})}{\mathrm{d}\boldsymbol{\Xi}} &= 2[\det(\boldsymbol{\Xi})]^2((\boldsymbol{\Xi})^{-1})^{\mathrm{T}} \\ \frac{\mathrm{d}\mathrm{tr}[(\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}})\boldsymbol{\Xi}^{\mathrm{T}}\boldsymbol{\Xi}]}{\mathrm{d}\boldsymbol{\Xi}} &= 2\boldsymbol{\Xi}(\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}) \end{aligned} \tag{26}$$

an expression for the gradient of the score function with respect to $\boldsymbol{\Xi}$ is derived as

$$\begin{aligned} \nabla_{\boldsymbol{\Xi}} M_0(\boldsymbol{\Xi}) = &\sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j (4\pi)^{-m/2} \exp\left\{-\frac{\mathrm{tr}[\Sigma_{i,j}\boldsymbol{\Xi}^{\mathrm{T}}\boldsymbol{\Xi}]}{4}\right\} \\ &\times \left\{\det(\boldsymbol{\Xi})[\boldsymbol{\Xi}^{-1}]^{\mathrm{T}} - \frac{1}{2}\det(\boldsymbol{\Xi})\boldsymbol{\Xi}\boldsymbol{\Sigma}\right\} \\ &- \sum_{i=1}^{n} \frac{1}{1-w_i} \sum_{j=1, j\neq i}^{n} w_j \exp\left\{-\frac{\mathrm{tr}[\boldsymbol{\Sigma}_{i,j}\boldsymbol{\Xi}^{\mathrm{T}}\boldsymbol{\Xi}]}{2}\right\} \\ &\times \frac{2}{n}(2\pi)^{-m/2}\{\det(\boldsymbol{\Xi})[\boldsymbol{\Xi}^{-1}]^{\mathrm{T}} - \det(\boldsymbol{\Xi})\boldsymbol{\Xi}\boldsymbol{\Sigma}\} \end{aligned} \tag{27}$$

where $\Sigma_{i,j} = (\boldsymbol{z}_i - \boldsymbol{z}_j)(\boldsymbol{z}_i - \boldsymbol{z}_j)^{\mathrm{T}}$.

One weakness of (27) is that the inverse of $\boldsymbol{\Xi}$ is used and this matrix inversion process may bring instability into the numerical optimisation procedure. However, from Cramer's rule, $\det(\boldsymbol{\Xi})[\boldsymbol{\Xi}^{-1}]^{\mathrm{T}}$ is equal to $[\mathrm{adj}(\boldsymbol{\Xi})]^{\mathrm{T}}$, the transpose of the adjugate matrix of $\boldsymbol{\Xi}$, and to obtain $\mathrm{adj}[\boldsymbol{\Xi}]$, only the matrix determinant operation is needed. Thus, in the following implementation, instead of calculating $\det(\boldsymbol{\Xi})[\boldsymbol{\Xi}^{-1}]^{\mathrm{T}}$, the adjugate matrix $[\mathrm{adj}(\boldsymbol{\Xi})]^{\mathrm{T}}$ is used.

Furthermore, besides requiring $\boldsymbol{H}$ to be invertible, it is usually required that the $\det(\boldsymbol{H})$ to be larger than a threshold $\min_{|\boldsymbol{H}|}$ to avoid introducing spurious features. Thus, in our implementation, the following constraint is

also imposed in the bandwidth matrix optimisation procedure

$$(\det\Xi)^2 < \frac{1}{\min_{|\boldsymbol{H}|}} \tag{28}$$

It should be noted that, for many types of sensors, the measurements are not spread equally along all coordinates in the measurement space. For example, for a 2D radar, it is quite common that the maximum detection range can be hundreds of kilometres, whereas the coverage in the azimuth is only from 0 to 90°. Therefore, before optimising the bandwidth matrix, the measurements must be pre-scaled [26], otherwise the numerical optimisation technique will fail. For a d-dimensional random vector $\boldsymbol{X} = [X_1, \ldots, X_d]^{\mathrm{T}}$, after the pre-scaling pre-process, the pre-scaled version of X becomes $\boldsymbol{X}^* = [X_1/\sigma_1, \ldots, X_d/\sigma_d]$, where σ_i is the standard deviation of random variable X_i. By pre-scaling, the data are transformed so that they have unit variance in each coordinate direction.

3.4 Choose weight for each kernel component

In the classic KDE method, the kernel components are usually assigned with equal weights, that is, $w_i = 1/n$ in (2), where n is the total number of sampling points [9]. However, in our proposed KDE method, the kernel components need to have unequal weights, because multiple association events are used to solve the measurement origin ambiguity. Note that, for two different association events χ_i and χ_j, it is quite likely that:

- The number of false alarms in χ_i is different from that number in χ_j.
- The measurement point $\boldsymbol{z}$, which is deemed as a clutter in χ_i, is deemed as being generated by targets in χ_j.

In the following, the classic KDE method is modified, so depending on the association events and their probability, the weight of each kernel component can be unequal.

Assuming that given the association event χ_i, the false alarms in the kth scan's measurement sets $\boldsymbol{Z}^{(k)} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N\}$ are $\boldsymbol{Z}^{(k)}_{i,C} = \left\{\boldsymbol{z}^k_{\chi_i,1}, \ldots, \boldsymbol{z}^k_{\chi_i,N_i}\right\}$, then the estimate for the clutter spatial intensity conditional on χ_i will have the following form

$$\hat{f}(\boldsymbol{z}) = \sum_{j=1}^{N_i} \frac{1}{N_i} K_H(\boldsymbol{z} - \boldsymbol{z}_{\chi_i,j}) \tag{29}$$

Thus, given a single association event, the weights of kernels are equally assigned, just as in the classical KDE method.

Equation (29) can be rewritten as

$$\hat{f}(\boldsymbol{z}) = \sum_{j=1}^{N} \frac{1}{N_i} \mathbb{I}_{\chi_i}(\boldsymbol{z}_j) K_H(\boldsymbol{z} - \boldsymbol{z}_j) \tag{30}$$

where $\mathbb{I}_{\chi_i}(\boldsymbol{z}_j)$ is an indication function such that

$$\mathbb{I}_{\chi_i}(\boldsymbol{z}_j) = \begin{cases} 1, & \text{if } \boldsymbol{z}_j \in \boldsymbol{Z}^{(k)}_{i,C} \\ 0, & \text{otherwise} \end{cases} \tag{31}$$

In other words, conditional on χ_i, the weight for any kernel will be $1/N_i$, if its corresponding measurement is deemed to be a clutter or 0 if its measurement is deemed to be generated by a target. Taking the expectation over all association events, the weight for the kernel component centred at measurement $\boldsymbol{z}_m$ will become

$$w_m = \sum \frac{P_i \mathbb{I}_{\chi_i}(\boldsymbol{z}_m)}{N_i} \tag{32}$$

where P_i is the probability of association event χ_i, N_i is the number of false alarms in association event χ_i and the summation is taken over all association events.

3.5 Processing continuously arriving measurements

The multivariate KDE defined by (2) employs the full data set. On the other hand, in the multitarget tracking problem, the sensor continuously provides measurements to the tracker. Thus, if (2) is used in the kernel clutter spatial intensity estimator without a proper modification, the total number of the kernel components will linearly increase without bound over time. In addition, in most scenarios, the temporal variation of the clutter spatial distribution is much slower than the update rate of the sensor. Thus, it is usually unnecessary to update the kernel components or recalculate the optimal bandwidth matrix at every scan.

To avoid continuously increasing the number of kernel components and to decrease the computational requirement, in our proposed method, the kernel components and the corresponding optimal bandwidth matrix are only updated every K scans, using the measurements and the corresponding association events accumulated in the latest K scans. In addition, to update the bandwidth matrix more efficiently, the old kernel mixture function, which was calculated using the measurements sets in the last $2K$ to $K+1$ scans, is used to help the bandwidth optimisation procedure, based a method modified from the adaptive KDE [9].

Assuming that at the current time, it is K scans after the last update of the kernel components and bandwidth matrix, and the Gaussian kernel mixture function, which is calculated based on the measurements in the last $2K$ to $K+1$ scans, is

$$\hat{c}_{-2K:-(K+1)}(\boldsymbol{z}) = \sum_{i=1}^{N_{-2K:-(K+1)}} w_i^{(-2K:-(K+1))} N(\boldsymbol{z};\ \boldsymbol{m}_i,\ \boldsymbol{H}_i) \tag{33}$$

To use the information contained in the kernel mixture function $\hat{c}_{-2K:-(K+1)}(\boldsymbol{z})$ to accelerate the bandwidth matrix optimisation, the following procedure is adopted in the proposed kernel clutter spatial intensity estimator:

(1) Assuming that in the last K scans, the sensor sent a sequence of measurement sets $\boldsymbol{Z}^{(-K:-1)} = \{\boldsymbol{Z}^{(-K)}, \ldots, \boldsymbol{Z}^{(-1)}\}$ to the multitarget tracker and the kernel clutter spatial intensity estimator. Define $\boldsymbol{a}_j$ as the jth measurement point in set $\boldsymbol{Z}^{(-K:-1)}$. Calculate the pilot normalised clutter spatial intensity estimation at $\boldsymbol{a}_j$ as $\hat{c}_{-2K:-(K+1)}(\boldsymbol{a}_j)$.
(2) For $\boldsymbol{a}_j$, define its local bandwidth factor as

$$\eta_j = \left\{\hat{c}_{k-1}(\boldsymbol{a}_j)/g\right\}^{-\alpha} \tag{34}$$

in which g is the geometric mean of the pilot normalised clutter spatial intensity estimation of all elements in $\boldsymbol{Z}^{(-K:-1)}$ and α is the sensitivity parameter, a number between 0 and 2. Following the suggestion given in [9], α is set as 1 in the proposed method.

doi: 10.1049/iet-rsn.2014.0037

(3) Set the functional form for the kernel mixture function, which will be calculated using measurements sets $\boldsymbol{Z}^{(-K:-1)}$, as

$$\hat{c}_{-K:-1}(\boldsymbol{z}) = \sum_{j=1}^{N_{-K:-1}} w_j^{(-K:-1)} N(\boldsymbol{z};\ \boldsymbol{a}_j,\ \eta_j \boldsymbol{H}) \tag{35}$$

where $\boldsymbol{H}$ is the optimal bandwidth matrix that will be determined by the CV technique presented in Section 2 and w_j is calculated using the method presented in Section 3.4.

The above steps guarantee that: (i) the mixture function $\hat{c}_{-K:-1}(\boldsymbol{z})$ is a continuous probability density function and (ii) each kernel has a different bandwidth matrix. Furthermore, the above procedure matches the common sense that in the low-intensity area, a kernel with a larger bandwidth matrix will be used to cover that area [9].

A numerical method, which is almost the same as the one presented in Section 3.3, can be used to find the optimal bandwidth matrix $\boldsymbol{\Xi}$. The only difference is that, after considering η_j, the equation for the jth kernel becomes

$$K^{(j)}(\boldsymbol{z}_i) = (2\pi)^{-m/2} \det\left(\eta_j^{-1/2}\boldsymbol{\Xi}\right) \exp\left\{-\frac{\eta_j^{-1}\mathrm{tr}[\boldsymbol{\Sigma}_{i,j}\boldsymbol{\Xi}^{\mathrm{T}}\boldsymbol{\Xi}]}{2}\right\} \tag{36}$$

The gradient of the above equation with respect to $\boldsymbol{\Xi}$ is

$$\frac{\mathrm{d}K^{(j)}(\boldsymbol{z}_i)}{\mathrm{d}\boldsymbol{\Xi}} = (2\pi\eta_j)^{-m/2} \exp\left\{-\frac{\eta_j^{-1}\mathrm{tr}[\boldsymbol{\Sigma}_{i,j}\boldsymbol{\Xi}^{\mathrm{T}}\boldsymbol{\Xi}]}{2}\right\} \times \left\{[\mathrm{adj}(\boldsymbol{\Xi})]^{\mathrm{T}} - \eta_j^{-1}\det(\boldsymbol{\Xi})\boldsymbol{\Xi}\boldsymbol{\Sigma}_{i,j}\right\} \tag{37}$$

4 Simulations

To demonstrate the performance of the proposed method, the spatial intensity estimation method presented in Section 3 is integrated with the JIPDA tracker [3], and then tested on a simulated scenario. In the simulation, whose setting is similar to the one used in the first experiment in [18], a linear 2D position-only sensor is deployed with 100 Monte Carlo trials. Here, besides the method proposed in this paper, two other methods are implemented for comparison. One of them is the clutter spatial intensity estimator using the nearest neighbour distance, which was identified in [8] as the best one among the three estimators presented there. The other one is the spatial intensity estimator proposed in [18].

In this simulation, there are four targets in a 2D region of interest (RoI) covering [0 m, 1000 m] × [0 m, 1000 m]. All targets follow the nearly constant velocity model with white Gaussian process noise. The target velocity process noise variance is 0.001 $\mathrm{m}^2/\mathrm{s}^2$ along each coordinate. All four targets become detectable at time $k = 31$ s and stay inside the RoI during the rest of the Monte Carlo trial. The states of each target at $k = 31$ s are shown in Table 1.

Table 1 Target initial position and velocity in the first simulation

	Positions, m	Velocities, m/s
target 1	(700, 200)	(2, 12)
target 2	(790, 500)	(−1.5, 0)
target 3	(465, 780)	(−0.75, −10)
target 4	(420, 500)	(1.5, 0)

Table 2 x–y Coordinates of dense clutter area vertices in the first simulation

Dense clutter area 1, m	Dense clutter area 2, m
[440, 300]	[740, 300]
[440, 700]	[740, 700]
[460, 700]	[760, 700]
[460, 300]	[760, 300]

A stationary linear position-only sensor, which stays at [0 m, 0 m], is used in this simulation. The measurement noise, which is white and independently-and-identically distributed, follows a Gaussian distribution with zero mean and standard deviation $\sigma_\omega = 1$ m along each coordinate. The probability of detection, $p_{\mathrm{D}} = 0.95$, for all targets. One Monte Carlo trial has 70 scans of measurements reported by the sensor and the sensor reports every 1 s. Two dense clutter areas exist in the measurement space and both areas have a rectangular shape. On average, each of them have eight false alarms per scan. In addition, there are two false alarms in each scan outside the dense clutter areas. For each rectangular area with dense false alarms, Table 2 gives the X–Y coordinates of its four vertices. The trajectories of all four targets and all detections reported by the sensor in the first Monte Carlo trial are given in Fig. 2.

To give a better comparison, there are five combinations of spatial intensity estimators and the JIPDA multitarget tracking algorithm in this subsection. For all combinations, the parameters for the scenario and the JIPDA tracker are the same. The values for the variance of the process noise, the measurement noise and the detection probability match those used in the simulation generation. A new track is initialised using the two-point initialisation method [1] with the maximum speed of 25 m/s. The track existence probability for each newly initialised track is set to 0.1. The initial track will become confirmed when its existence probability value grows to ≥0.98. The track will be ceased once the value of its existence probability becomes smaller

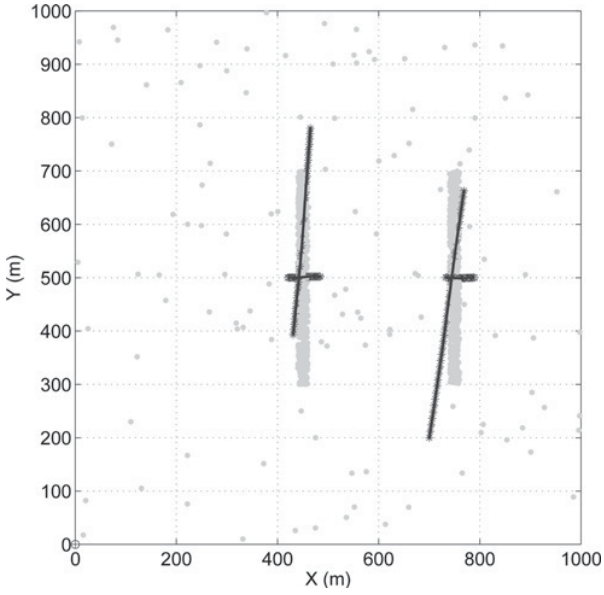


Fig. 2 *Target trajectories and all measurements in the first trial*

than 0.04. To update the probability of track existence, Markov chain I model [3] is used with $P(\text{dead}|\text{exist}) = 0.02$ and $P(\text{exist}|\text{dead}) = 0$.

In the first and the second combinations, the JIPDA multitarget tracker uses the clutter spatial intensity value reported by the nearest neighbour-based estimator [8]. In the first combination, the RoI is divided into a 10×10 grid and all cells have the equal size for the purpose of the clutter estimation, whereas in the second combination, the RoI is divided into a 50×50 grid. For both spatial intensity estimators, the mathematical distance metric is calculated based on the square shape and an auto regressive filter based on the following equation [8], is used to obtain the time-smoothed estimates of spatial intensity

$$N_k(c) = a_k N_{k-1}(c) + (1 - a_k)\mu_k(c)$$
$$a_k = \frac{k-1}{k},\ k \le M_L;\ a_k = \frac{M_L - 1}{M_L},\ k > M_L \tag{38}$$

In (37), $\mu_k(c)$ is the reciprocal of the estimate of the clutter spatial intensity in cell c using the latest measurement set at time k, whereas $N_k(c)$ is the time-smoothed version of $\mu_k(c)$ and $M_L = 15$ is the smoothing time length.

The third combination uses the estimator proposed in [18], which relies on the recursive update of the clutter generator PHD function, to obtain the spatial intensity of false alarms. The parameters used here are the same as those used in the first simulation in [18].

In the fourth combination, the KDE-based estimation algorithm proposed here is used. The kernel mixture function and the bandwidth matrix are updated every 15 scans. At the 15th scan, that is, the first time of the kernel mixture function update and bandwidth matrix optimisation, the initial bandwidth matrix $\boldsymbol{\Xi}_0$ is set as

$$\boldsymbol{\Xi}_0\boldsymbol{\Xi}_0 = [4/(2d+1)]^{-2/(d+4)}\boldsymbol{S} \tag{39}$$

where $\boldsymbol{S}$ is the empirical precision matrix obtained from the measurement set accumulated in the first 15 scans and d is equal to 2, that is, the dimension of the measurement space. In the following updates, the initial matrix in bandwidth matrix optimisation is equal to the optimal bandwidth matrix obtained in the previous update. From the simulation experiment, it can be observed that, for a background with slowly varying clutter spatial intensity, using the previous optimal bandwidth matrix as the initial start in the bandwidth matrix optimisation procedure can greatly reduce the time for optimisation.

To provide a base line, the true spatial intensity of false alarms is given to the tracker in the last combination.

The definitions of a validate track and a false track follow those given in [27] and the association gate, which is necessary for excluding the ground truth to the track, is 35 m.

Figs. 3 and 4 show the Monte Carlo mean number of valid confirmed tracks and the Monte Carlo mean number of false confirmed tracks, respectively. Fig. 3 indicates that the proposed KDE method has significantly reduced the number of confirmed false tracks such that the performance of the JIPDA tracker is comparable with the situation where the true clutter spatial intensity is known exactly. In addition, from Fig. 4, it can be observed that the proposed estimator has faster true track initiation compared with the nearest neighbour-based estimator and the clutter generator-based estimator. In other words, without specifically tuning any setting in the JIPDA multitarget tracker, the proposed KDE clutter spatial intensity estimator is able to obtain faster track initiation speed and lower false track rate at the same time, compared with the nearest neighbour estimator [8] and the clutter generator estimator [18].

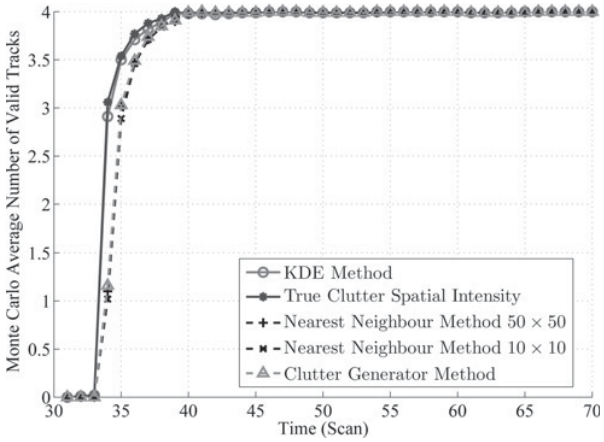


Fig. 4 *Monte Carlo mean of confirmed valid tracks*

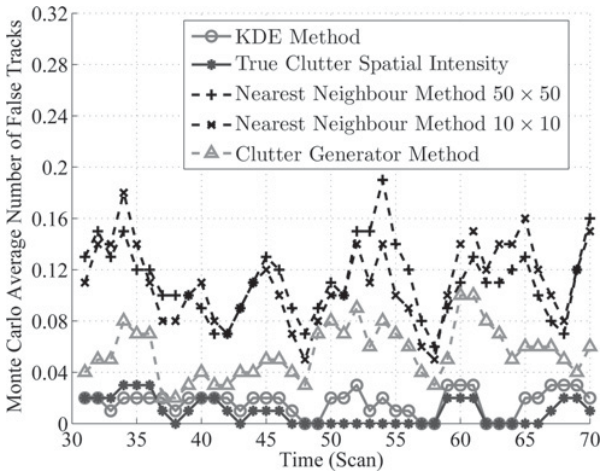


Fig. 3 *Monte Carlo mean of confirmed false tracks*

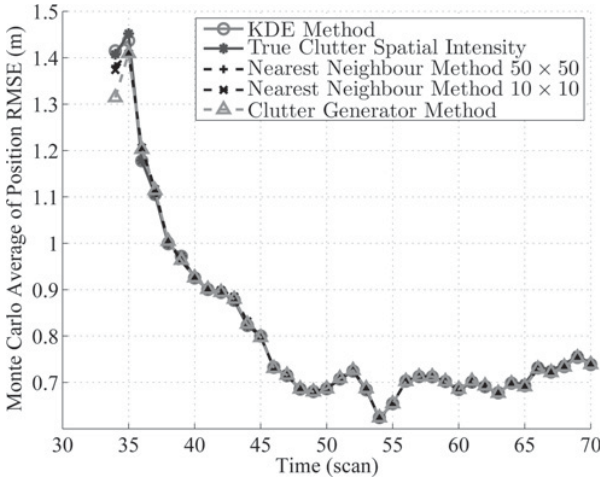


Fig. 5 *Monte Carlo average RMSE for Target 1*

doi: 10.1049/iet-rsn.2014.0037

Table 3 Monte Carlo average for the computational time for a single trial

Algorithms	Time, s
KDE method	251.61
nearest neighbour 50 × 50 method	60.12
nearest neighbour 10 × 10 method	33.39
clutter generator method	94.05
true clutter spatial intensity	27.03

Fig. 5 shows the Monte Carlo mean of the root mean squared error (RMSE) for the position estimates of target 1. The RMSE results match the statement presented in [28], which said that the RMSE improvement because of clutter estimation is usually insignificant (all clutter spatial intensity estimators yielded nearly the same RMSE in [28]). The same RMSE trend was observed for target 2 to target 4.

Table 3 gives the Monte Carlo mean of the computational time for a single trial in this simulation. All estimators and the JIPDA trackers are coded in MATLAB® 7.12 and tested on a computer with an Intel® Core™2 Duo E6550 central processing unit (CPU) and 2 GB of random access memory. The CPU usage by MATLAB® is around 50% during the execution. From Table 3, it can be observed that the proposed method needs a longer computational time compared with the other methods. However, it should be noted that, in many mission critical scenarios, like the ballistic missile defence, torpedo defence and anti-submarine warfare, the computational power is not a bottleneck, but it is more important to achieve faster track initialisation and lower false track rate.

5 Conclusions

In this paper, a KDE based method is proposed to estimate the spatial intensity of false alarms for multitarget tracking systems. In this estimator, the clutter spatial intensity estimation problem is first decomposed into two separate subproblems:

- Estimating the probability mass function of the false detection number per scan.
- Estimating the non-homogeneity of the spatial distribution of false alarms.

Then, based on the NHPP assumption, a multivariate kernel intensity estimator was proposed to estimate the spatial intensity of false alarms. In the proposed estimator, the output of the multitarget tracker is used to solve the measurement origin ambiguity. The proposed KDE is locally adaptive, which means that the bandwidth matrix can vary from one kernel to another, and the weights of kernels can be automatically adjusted according to the Bayesian principal. In addition, it is capable of working online and the total number of the kernel components will not increase without a bound. Furthermore, the bandwidth matrix is automatically determined by the proposed estimator from the data set, and that matrix is not constrained to be diagonal. In addition, to accelerate the optimisation of the bandwidth matrix, an expression for the gradient of the cost function with respect to the bandwidth matrix is derived, and that expression does not contain any matrix inversion operation. Simulations have shown that the proposed KDE-based algorithm is able to make the JIPDA tracker perform better over non-homogeneous clutter background, when compared with classic approach, the nearest neighbour approach and the clutter generator approach for clutter estimation.

6 References

1 Bar-Shalom, Y., Li, X.R.: 'Multitarget-multisensor tracking: principles and techniques' (YBS Publishing, Storrs, CT, 1995)
2 Daley, D.J., Vere-Jones, D.: 'An introduction to the theory of point processes volume I: elementary theory and methods' (Springer, New York, NY, 2002, 2nd edn.)
3 Mušicki, D., Evans, R.: 'Joint integrated probabilistic data association: JIPDA', *IEEE Trans. Aerosp. Electron. Syst.*, 2004, **43**, (3), pp. 1093–1099
4 Reid, D.B.: 'An algorithm for tracking multiple targets', *IEEE Trans. Autom. Control*, 1979, **24**, (6), pp. 843–854
5 Li, X.R., Li, N.: 'Integrated real-time estimation of clutter density for tracking', *IEEE Trans. Signal Process.*, 2000, **48**, (10), pp. 2797–2805
6 Li, N., Li, X.R.: 'Target perceivability and its applications', *IEEE Trans. Signal Process.*, 2001, **49**, (11), pp. 2588–2604
7 Bar-Shalom, Y., Blackman, S.S., Fitzgerald, R.J.: 'Dimensionless score function for multiple hypothesis tracking', *IEEE Trans. Aerosp. Electron. Syst.*, 2007, **43**, (1), pp. 392–400
8 Mušicki, D., Suvorova, S., Morelande, M., Mora, B.: 'Clutter map and target tracking'. Proc. Eighth Int. Conf. Information Fusion, Wyndham Philadelphia, PA, USA, July 2005, pp. 69–76
9 Silverman, B.W.: 'Density estimation for statistics and data analysis' (Capman and Hall, London, UK, 1986)
10 Hanselmann, T., Musicki, D., Palaniswami, M.: 'Adaptive target tracking in slowly changing clutter'. Proc. Ninth Int. Conf. Information Fusion, Florence, Italy, July 2006
11 Lachlan, K.H.: 'Clutter-based test statistics for automatic track initiation', *Acta Autom. Sin.*, 2008, **34**, (3), pp. 266–273
12 Møller, J., Waagepetersen, R.P.: 'Statistical inference and simulation for spatial point processes' (Chapman & Hall/CRC Press, Boca Raton, FL, 2003)
13 Chen, X., Tharmarasa, R., Pelletier, M., Kirubarajan, T.: 'Integrated clutter estimation and target tracking using Poisson point processes', *IEEE Trans. Aerosp. Electron. Syst.*, 2012, **48**, (2), pp. 1210–1235
14 Mahler, R.: 'CPHD and PHD filters for unknown backgrounds, I: dynamic data clustering'. Proc. SPIE Sensors and Systems for Space Applications III, Orlando, FL, USA, 2009, vol. 7330
15 Mahler, R.: 'CPHD and PHD filters for unknown backgrounds, II: multitarget filtering in dynamic clutter'. Proc. SPIE Sensors and Systems for Space Applications III, Orlando, FL, USA, 2009, vol. 7330
16 Mahler, R.: 'Multitarget Bayes filtering via first-order multitarget moments', *IEEE Trans. Aerosp. Electron. Syst.*, 2003, **39**, (4), pp. 1152–1178
17 Mahler, R.: 'CPHD and PHD filters for unknown backgrounds, III: tractable multitarget filtering in dynamic clutter'. SPIE Proc. Signal and Data Processing of Small Targets 2010, Orlando, FL, USA, 2010, vol. 7698
18 Chen, X., Tharmarasa, R., Pelletier, M., Kirubarajan, T.: 'Integrated Bayesian clutter estimation with JIPDA/MHT trackers', *IEEE Trans. Aerosp. Electron. Syst.*, 2013, **49**, (1), pp. 395–414
19 Wasserman, L.: 'All of nonparametric statistics' (Springer-Verlag, NY, USA, 2005)
20 Chen, X., Tharmarasa, R., Kirubarajan, T., Pelletier, M.: 'Online clutter estimation using a Gaussian kernel density estimator for target tracking'. Proc. 14th Conf. Information Fusion, Chicago, IL, July 2011
21 Duong, T., Hazelton, M.L.: 'Cross-validation bandwidth matrices for multivariate kernel density estimation', *Scand. J. Stat.*, 2005, **32**, (3), pp. 485–506
22 Murty, K.G.: 'An algorithm for ranking all the assignments in order of increasing cost', *Oper. Res.*, 1968, **16**, (3), pp. 682–687
23 Terrell, G.R.: 'The maximal smoothing principle in density estimation', *J. Am. Stat. Assoc.*, 1990, **85**, (410), pp. 470–477
24 Petersen, K.B., Pedersen, M.S.: 'The matrix cookbook' (Technical University of Denmark, Denmark, November 14, 2008)
25 Brookes, M.: 'The Matrix Reference Manual', [online]. Available at http://www.ee.ic.ac.uk/hp/staff/dmb/matrix/intro.html, 2005
26 Duong, T.: 'Bandwidth selectors for multivariate kernel density estimation'. PhD Thesis, School of Mathematics and Statistics. University of Western Australia, Perth, Australia, 2004
27 Rothrock, R., Drummond, O.E.: 'Performance metrics for multiple-sensor, multiple-target tracking'. Proc. SPIE Conf. Data and Signal Processing of Small Targets, Orlando, USA, July 2000
28 Mušicki, D., Evans, R.: 'Clutter map information for data association and track initialization', *IEEE Trans. Aerosp. Electron. Syst.*, 2004, **40**, (2), pp. 387–397