Abstract—Tracking algorithms are often designed around optimistic assumptions on uncertainty model. Handling with conflicting data, however, requires specific strategies, that consider quality of information sources. To improve performance of tracking systems, the use of reliability, as evaluation of quality of data sources, has been proved to be a promising technique. In this paper we show how to use reliability of information sources to increase performance of tracking methods, using two different strategies: discount and pruning. We apply those two strategies in two different scenarios: landmark based mobile robot localization using Extended Kalman Filter and multi-agent object-tracking using Particle Filter. Experimental results show effectiveness of proposed methodology.

Keywords: Multi-agent tracking, localization, Kalman filtering, Particle filtering, estimation, reliability.

I. INTRODUCTION

One of the most important abilities of an autonomous agent is gathering information from the surrounding environment in order to build a symbolic representation of it. In particular, in many applications it is useful to track moving objects over time. This goal is usually achieved by tracking algorithms.

Tracking algorithms can be divided in two classes: object tracking algorithms and self tracking, or localization, algorithms. The main goal of the former class is to locate the absolute position of moving objects over time. The latter one, instead, uses fixed objects as reference to locate position of a mobile base with on-board sensors.

In this paper we will treat these two classes in the same framework. Therefore a general approach will be presented, and applied to multi-source object-tracking and localization problems.

Despite this paper has precise focus on tracking methods, the same concepts can be applied to a generic data fusion system.

A multi-source tracking architecture uses data coming from various input sources, “a priori” knowledge about the environment and a known model of uncertainty.

Therefore, the success of information fusion depends on how well the knowledge produced by the fusion process represents reality, which depends on how adequate data are, how good and adequate is the uncertainty model, and how accurate, appropriate or applicable is the prior knowledge.

At a first step, the performance of a fusion process can be estimated by the error between real state and estimated one. This is an idealistic assumption, because it is almost impossible to relate true state with the estimated one. However, perfect observations are typically not available, due to low resolution of sensors, random noise in sensed environment, or because the process to be estimated cannot be well modeled. Consider for example observations extracted from image processing routines for a camera mounted on a mobile robot.

The first level of uncertainty knowledge is usually provided by covariance matrices, but this is not sufficient to guarantee good performance. In fact, it is impossible to detect defective sources, or situation wherein the measurements gathered contain errors. In such situations it is thus necessary to develop a second order of fusion that involves reasoning about the conditions wherein measurements are taken. This introduces a higher level of knowledge, in the form of uncertainty of sources.

Many authors proposed to deal with the second level of knowledge, by introducing the concept of reliability [1]. The definition of reliability involves two aspects:

- observation reliability, related to single measurement of a sensor or information source;
- source reliability, related to the quality of a whole source.

In this paper we use reliability of information sources to improve tracking methods for mobile robotic platforms. The main idea is to define reliability of a source, as the result of processing information about the source itself. From this perspective, reliability is not associated with fusion results, but it can be viewed as an “a priori” knowledge about the operating situations.

In the following sections, we will present a brief introduction of the problem, showing an overview of current methods present in literature. Then we will formalize the problem of tracking within a reliability framework, focusing attention on two strategies available for reliability handling: discount and pruning.

To evaluate effectiveness of our proposal we will present experiments in two different scenarios: landmark-based mobile robot localization and multi-agent object tracking.

II. RELATED WORK

The key point of successful tracking algorithms resides in the correct estimation of the system state. This is dependent on adequate parameters chosen per environment settings, correct sensor noise model and accuracy of sensors. The process of modeling beliefs has always some limitations, and models are valid only within a certain range. So, when combining information provided by many sources, we have to take into
account the range and the limitations of the belief models used for each source. An inappropriate choice of metrics or poor estimation of the likelihood functions can provide an inadequate belief model and can lead to conflicting and unreliable beliefs to be combined.

The most part of literature focus its attentions on establishing reliability of the beliefs computed within the framework of the model selected.

This approach fails to produce good estimation in particular situations, wherein it is not possible to detect errors from the inside of fusion process. For example, a fusion process cannot detect ambiguous features, unpredictable systematic error or conflicting data. In such cases standard algorithms fail to handle correctly data from defective sources. As a consequence, the data fusion mechanism blindly merges information without taking care of the quality of resulting estimation.

The convergence proof of some algorithms, e.g. for Kalman filter [2], ensures that eventually bad observations influence will be mitigated, but in many cases it will require some time to achieve recovery. Some works, done in directions of adaptable filtering [3], provide a faster recovery framework, but they still lack the ability to handle conflicting data.

In many cases, however, it is possible to improve algorithm performances using additional information about the sources. For example, it is possible to extract important behavior patterns depending on characteristics of the estimated environment, such as failure rate, unpredictable systematic error, or ambiguous features detection.

This may be achieved by using reliability coefficients, which introduce the second level of uncertainty (uncertainty of evaluation of uncertainty [1] [4]).

Although the concept of reliability has been already introduced before, it allows for various interpretations and representations and is not yet well established [1] [5].

Several authors suggest that an “information source is totally reliable if and only if the information it delivers is true in the real world” [6]. In some papers it has been defined as “pedigree information” [7], “confidence” [8], “multi-criterion feature ranking” [9].

A general model for partially reliable information sources is analyzed in [10], formalizing how to model information sources with different trustworthiness.

As pointed in [1] it is possible to develop several strategies to handle reliability information. The most evident are:

- Discount strategies. Information data are used in the filtering process accordingly a weight, evaluated from reliability information.
- Pruning strategies. Quality of data input is identified and used to eliminate data of poor reliability from fusion process.

Discount strategies are related to standard behavior of filtering algorithm [11]. Observations from sensors or agents are merged using sensor noise model in the form of covariance matrices. These parameters can be adapted in order to decrease influence of low quality information.

Pruning strategies are used in different context. Some authors use reliability to select from which sensor gather information [12] [13]. Sensors selection techniques can be found in [14] [15]: using a kind of fuzzy measure, for each sensor a measure of reliability is estimated and then used to select only those sources of information with higher quality.

The key points of these strategies can be found in many works in literature [16] [17] [18] [19] [20]. However they lack to explicit address the problem of role definition of reliability. In many cases it is hidden under some specific tuning of parameters, or implicit considerations of information sources. Moreover this fact complicates comparison among different methods and it arises in different interpretations of fusion process results.

In this paper we define a specific role for reliability, to show how it affects performances of a fusion process. To proof effectiveness of exposed methods, we performed comparisons among different behavior of the same algorithm. We drive the fusion process of two different filtering methods (a Kalman Filter and a Particle Filter) to expose the differences in handling wrong measurements. More in details we simulated three specific conditions of error wherein a real application could be and gathered the fusion results.

Looking inside fusion algorithms, reliability can be associated to single observation from the source rather than the whole source. The former case is the most addressed by current literature, e.g. [13] [7] [9] [6]. Many works consider reliability, or uncertainty, as random sensor noise error, and typically it is already included by the covariance matrices used for filter update step, but source reliability has not been investigated in depth.

On the other hand, in many applications it is possible to determine some peculiar properties of information source [5]. For example, unpredictable systematic error, due to physical constraints of sensor, or poor calibration and many false positive.

As uncertainty can be modeled in different frameworks [21], there is a lot of works for Possibility Theory and Evidence Theory. Data fusion techniques using reliability established for Fuzzy Theory are illustrated in [5] [15], while it is possible to find methods developed within Dempster-Shafer Theory [10] [11] [12] [14]. As far as we know, a few works are present for Bayesian Theory. The most significant one is explained in [13], where reliability is used to detect false tracks, after the fusion process.

In this paper we consider multi-source tracking to operate within the Probability Theory, provided by the Bayesian Filtering, to show real approach suitable for this framework.

III. IMPROVE DATA-FUSION USING RELIABILITY

A. General fusion process

Let $S = \{s_1, \ldots, s_I\}$ be data produced by $I$ information sources at time $t$, and $Z = \{z_1, \ldots, z_N\}$ be a set of measurements about $S$ taken by $N$ sensors in the environment.

Within a fusion framework, a model $M$ is a representation of the world that uses observations $z_i$ and prior knowledge to evaluate $x_i$ as an estimation of $s_i$, and the uncertainty of such estimation within a certain theoretical framework.
The estimation is based on an information fusion process defined by the operator $F(x_1, \ldots, x_t)$ built in the framework of the theory used for belief representation. This operator specifies how observations are combined together to determine the estimated belief $x_t$.

In practical applications the fusion operator is defined as a filtering method.

A generic filtering method is usually performed in two notable steps: predict and update [22]. In the predict phase a prediction of system state is performed, using previous state information and control data. During the update step, predicted state is evaluated against the observations gathered from sensors.

For a Kalman Filter these two steps are distinguished in (see [2] for a complete review of Kalman Filter):

- **Predict:**
  
  $$
  \begin{align*}
  \hat{x}_t' &= E_t \hat{x}_{t-1} + B_t u_t \\
  P_t' &= E_t P_{t-1} E_t^T + Q_t
  \end{align*}
  $$

- **Update:**
  
  $$
  \begin{align*}
  K_t &= P_t' H_t^T (H_t P_t' H_t^T + R_t)^{-1} \\
  \hat{x}_t &= \hat{x}_t' + K_t (z_t - H_t \hat{x}_t') \\
  P_t &= (I - K_t H_t) P_t'
  \end{align*}
  $$

where $P_t$ is the error covariance matrix, $\hat{x}_t$ is the estimation of the state at time $t$, $K_t$ is the gain of Kalman filter, $F_t$ is the state transition model which is applied to the previous state $x_{t-1}$, $B_t$ is the control-input model which is applied to the control vector $u_t$, $z_t$ is observation (or measurement) of the true state $x_t$, $H_t$ is the observation model, $Q_t$ and $R_t$ are respectively the covariance of process noise and the covariance of observation noise.

In a Particle Filter algorithm, it is possible to identify the following steps (see [23] for an in deep presentation of Particle filtering methods):

- draw samples from the proposal distribution:
  
  $$
  x_t^{(L)} = \pi(x_t|z_{0:t-1}, y_{0:t})
  $$

- update the importance weights up to a normalizing constant:
  
  $$
  w_t^{(L)} = \frac{w_t^{(L)} p(y_t|x_t^{(L)}) p(x_t^{(L)}|x_{t-1}^{(L)})}{\pi(x_t^{(L)}|z_{0:t-1}, y_{0:t})}
  $$

- compute the normalized importance weights:
  
  $$
  w_t^{(L)} = \frac{w_t^{(L)}}{\sum_{J=1}^P w_t^{(J)}}
  $$

- Compute an estimate of the effective number of particles as
  
  $$
  \hat{N}_{eff} = \frac{1}{\sum_{L=1}^P w_t^{(L)}^2}
  $$

- If the effective number of particles is less than a given threshold $N_{thr}$, then perform resampling:
  
  - Draw $P$ particles from the current particle set with probabilities proportional to their weights. Replace the current particle set with this new one.
  - set $w_t^{(L)} = 1/P$.

The predict step corresponding to the same of Kalman filter is the drawing of samples from proposal, while the sampling step (here expanded in all sub-step) is the corresponding update phase.

### B. Role of reliability

As previously mentioned, it is not possible to provide a formal definition of reliability. That is because it is dependent upon theory used for fusion operator, as well as the application wherein the fusion process is performed.

Despite that, it is still possible to define a generic concept of reliability, following some practical considerations. Some of them are already described in literature [1] [5] [6] [7] [8] [9]. In this paper, with the term reliability, we will indicate the idea that it is possible to assign a probabilistic value of confidence to a specified information source.

Let’s start from the previously introduced definition of a generic fusion operator $F$. To take into account reliability of information sources, it can be modified as $F_R(x_1, \ldots, x_t, R_1, \ldots, R_t)$, where $R_t$ indicates the reliability coefficient related to the source $i$. The reliability coefficients are responsible to control behavior of fusion operator, affecting how respective belief $x_i$ are integrated each other.

Each coefficient $R_t$ is dependent on the selected model, and on global knowledge, that includes the characteristics of the environment. In general a standard method to evaluate $R_t$ is not available, because it depends on specific characteristics of environment or information sources, and computation of reliability is related to the specific application. Moreover it is important to underline that reliability of sources is strictly related to presence of conflicting data that indicate presence of unreliable sources.

The computation of reliability values for the experimental cases considered in this paper is described in Section IV.

### C. Data Fusion work flow with reliability

To use reliability the two steps of a generic fusion algorithm are modified as follows:

1) predict: use previous state to estimate current state of system;
2) evaluate reliability for current measurements. It can be a value per observations or an overall estimation per information source;
3) update: integrate information data (from ego perceptive sensors or from external information sources), considering their reliability.

For a Kalman filter, the use of reliability involves changes in the estimation of gain $K$, and in the matrix $H$ of the observations model. For a particle filter based algorithm, reliability alter the normal processing of measurements, typically weighting particles during the sampling step.
D. Reliability application

As previously said, there are two different methodologies of reliability use: discount and pruning. Both should drive fusion process, altering standard steps performed.

1) Discount strategy: To explicitly use reliability of information sources, a probabilistic distribution function over the observation space is evaluated. Therefore, data coming from specific sources are mapped into a numeric value used as discount factor for update step:

\[ g(R_i) : R_i \rightarrow [0, 1] \]  \hspace{1cm} (1)

The function \( g(R_i) \) is dependent on how reliability is represented. In general, it depends on the specific application, but its meaning is strictly tied with probability concept. A straightforward definition of \( g \) is:

\[ g(R_i) = R_i \]  \hspace{1cm} (2)

where we assume that \( R_i \) is already a probability value. Another example is:

\[ g(R_i) = \frac{\text{ranking}}{\text{maxranking}} \]  \hspace{1cm} (3)

where \( \text{ranking} \) is the result of a checklist feature [9]. The discount factor has the role of weighting measurements in fusion update: it varies from 0 (no inclusion) to 1 (complete inclusion).

For each information data, the discount factor is used to weight their inclusion in the update step:

\[ F_R \equiv F(x_1, \ldots, x_I, g(R_1), \ldots, g(R_I)) \]  \hspace{1cm} (4)

The previous formula specifies that information from reliable sources are dealt together unreliable ones using a uniform methodology. The function \( g(R_i) \) evaluates a discount factor and during the update step, information with higher reliability will drive the estimation.

2) Pruning strategy: The pruning approach is more pragmatic than discount one. The idea is to detect defective sources and just do not consider them in the update step. Reliability considerations are mapped into a probability value used as threshold to detect poor quality data.

\[ F_R \equiv F(\bar{X}_J) \]  \hspace{1cm} (5)

Here \( \bar{X}_J = (x_{i_1}, \ldots, x_{i_J}) \), \( i_J \leq I \) is the resultant set of measurements that satisfy a certain reliability criterion.

This simple approach has been proved to be very effective by many authors. In particular [24] points out that using variable thresholds overcome fixed one. The main problem is how to adapt the threshold accordingly to the reliability. Some authors suggest to use failure rate, or consensus [25].

IV. Experimental evaluation

To evaluate the proposal of this paper two different robotic tasks have been used: landmark-based localization with Extended Kalman filter, and multi-agent object tracking using Particle Filter. The choice is not casual. In the chosen setup measurements are extracted from images taken from a camera. The characteristics of sensor make it a good test-bed, because it is very sensible to noise and clutter from the environment.

The goal of the experiments is to provide an overview of behavior of widely used algorithms for data fusion, under different conditions of error. The focus of these comparisons is to understand in which situations the knowledge of reliability can determine an improvement in estimation quality. To achieve this aim we performed each experiment in three modes: without using reliability, using a discount strategy and using a pruning one. We use the same “calibration” for the filter so the only differences in the results are given by different measurements handling.

A. Kalman Filter localization

The first set of experiments is performed to evaluate the behavior of a self-tracking (or localization) algorithm. The aim of these experiments is to show how reliability can improve localization under these error assumptions: systematic error, which represents the problem of dealing with measurements affected by a constant error; misrecognition, wrong associations among related features; and false positives, perception of features not present.

These issues are very common in robotic applications, and are chosen as the most representative by our experience.
Although ground-truth data are not available in such a dataset, it is possible to evaluate localization error using the position of a set of waypoints provided by data-set authors. We use these waypoints to estimate localization error, as the difference between the estimated position and the reached waypoint. It was not possible to evaluate the angular error because of the absence of real values for the orientation of the robot.

For each experiment we run a standard algorithm, using a Matlab implementation of Extended Kalman Filter (by Tim Bailey and Juan Nieto4), and two variants using discount strategy and pruning strategy as explained in the section III-D. The results show several runs of the same dataset, increasing the error at each run. Moreover they show the comparison of methods with the same amount of error.

**Experiment 1: Systematic error**

Consider a robot with a quite accurate calibration of color segmentation. The field is located inside a pavilion with big windows and colored curtains. During the match, the sun projects some additional light on some landmarks causing incomplete perceptions. For example, the landmark is not fully recognized and bearing and distance estimated are incorrect by a constant error.

To simulate this situation we chose two particular markers in the middle of field and we modified their observations adding linear and angular error. In particular we fixed constant angular error to errθ = 0.0087 rad (5°), while the linear error has been varied errρ ∼ \( U(0.5, 1) \) m:

\[
\text{perception}_{\text{landmark}} < \rho, \theta > = \text{error} < \text{err}_\rho, \text{err}_\theta >
\]

During execution we assume that these two landmarks are unreliable.

Given this knowledge, the Equation 4 evaluates reliability with larger covariance matrix for the unreliable sources:

\[
\begin{align*}
g_{\rho}(R_{\text{perception}_{\text{landmark}}}) &= e_\rho \\
g_{\theta}(R_{\text{perception}_{\text{landmark}}}) &= e_\theta
\end{align*}
\]

where \( e_\rho = 1.5 \) and \( e_\theta = 1.33 \) (the values are chosen by empirical consideration, their meaning is not related to formal evaluation).

Instead in pruning strategy the observations considered unreliable were skipped, so they do not contribute to estimate state.

The experiment results are visible in Figure 2. The results show the execution of data-set with three different settings. The normal execution is the Kalman Filter localization algorithm. The solid red line shows the increasing mean error of normal execution as the increase of amount of systematic error. The discount strategy (the green dashed line with triangle) and the pruning strategy (the blue point line with circle) have better performances, the high systematic error does not affect significantly the performance.

![Figure 2. Experimental results for situation with systematic errors. The x axis is the amount of error added, while along y axis it is shown the linear error.](http://www-personal.acfr.usyd.edu.au/tbailey/software/index.html)

**Experiment 2: Misrecognition**

Consider the same setup of the previous experiment. Two close landmarks have similar features (for example one is orange, and one is red, the colors are close on the color table). The calibration of robot does not allow to fix exact thresholds for these two features. Again an external factor (e.g. the sun) changes some parameters. The result is that classification of the features fails in some case to identify landmark, and perceptions are confused, or misrecognized.

To consider this aspect, we chose two close landmarks and modified the dataset adding errors on the observations of these landmarks. The errors consist in changing the perception of one landmark with the other, so we insert in the data wrong associations between landmarks. This is done accordingly an “a priori” probability prob\(_{\text{misrecognition}} \subset [0, 1]\).

In the discount strategy we fixed a probability value prob\(_{\text{threshold}} \) used as threshold for reliability. For each observation of the chosen landmarks we draw a probability prob\(_{\text{wrongness}}\). If prob\(_{\text{wrongness}} > \text{prob}_{\text{threshold}}\), then we enlarged covariance matrix, as for the first experiment.

The same probability threshold is used for pruning strategy, to decide when prune away perceptions of unreliable landmarks.

In Figure 3 it is possible to see the results of the three runs, with the increasing of percentage of error added. In this case performances of discount and pruning strategies are still better than normal execution. However the discount strategy shows better results especially in case of high probability error.

**Experiment 3: False positives**

As before, we have the same match field. As the audience is allowed to watch the event, there is no white wall around the field. For an unfortunate case there is a kid seated near the field with a landmark patterned shirt. The object detection can be confused by this fact and recognize a landmark in an impossible location, causing a false positive detection error.

We simulate this situation by adding observations of a
B. Multi-agent Object Tracking

In this scenario we simulate a multi-agent approach to object tracking. Sources in this case are robotic platforms moving inside a limited area. The main goal is tracking an object, while the robot is patrolling a small portion of the map. Information are sent through the network and shared among agents.

We assume that agents are homogeneous with similar characteristics. Information about tracking objects are provided by a color camera. Because agents are based upon the same platform, correlation of information is straightforward, and reliability of information sources does not require to be adapted.

The extension of the method to heterogeneous platform applications is currently under development.

We used Player/Stage simulation environment [28]. Observations are taken as results of object recognition process and ground truth information are used to evaluate effectiveness of proposed method. Each perception has a zero-mean Gaussian noise added.

Each agent runs a local position estimator based upon their own perception, and a global estimator, taking into account also information from other agents. We assume that clocks on board of agents are synchronized and a common timestamp is available. This can be achieved, for example, by sending a starting event and recording the offset of clocks for each message received by other agents.

We assume agents are provided with a localization system, that allows a good estimation of their own position. This assumption is reasonable for the purpose of the experiment, which is to evaluate effectiveness of reliability in multi-agent scenario.

To be consistent with former experiments, we evaluate two different situations: systematic error and false-positive. The misrecognition test was not performed because of the presence of a single object to be tracked. Extension to multi-agents multi-objects tracking problem is to be considered as future work.

We evaluate performance of estimation from the perspective of an agent that receives data from others. This setup can be viewed as a centralized approach to object tracking: each agent performs all computation of measurements from all agents. We do not consider communication errors, and we implicitly assume that information are synchronized (measurements from agents are related to the same frame). This assumption is not very realistic in a more complex scenario, but it is still reasonable in such context.

Each agent estimates object position using a Particle Filter [23] algorithm. In this scenario we want to consider reliability per information source and we select an agent as “faulty agent”. This selection is done at the beginning. It can be considered as an online reasoning about conflicting data or
false observations rate, or as the result of an “a priori” knowledge of the agent. More practical explanations will be given introducing the experiments.

Observations from the faulty agent are labeled as erroneous and artificially altered before sending them through the network. The receiving agents do not know the amount of alteration.

A perception of an object is a couple \( < \rho, \theta > \), describing the polar coordinates of the object relative to the observer and is the result of the local estimator of each agent.

We generate ten different trajectory of the moving object and then using them to obtain ten value for each increasing error and for each strategy. Then the results are averaged among them.

**Experiment 1: Systematic error**

A common calibration is performed, as agents are based upon the same platform. During training period or previous experiments one agent is detected as defective. For example its camera has some hardware error on color conversion and objects are badly perceived. The amount of systematic error cannot be exactly estimated, but we evaluate that close perceptions are more precise.

Before sending an object position estimation, a systematic error (linear and angular) is added. The systematic error is decomposed in two parameters: linear error \( \text{error}_{\text{linear}} \subset [0.1, 1] \) m, and angular error \( \text{error}_{\text{angular}} \sim (-0.87, 0.87) \) rad (\( \sim (-50^\circ, 50^\circ) \)). The experiment is performed for different values of the linear error, while the angular error is sampled within the specified interval.

We assume that observations with low angle (in the center of image) and low distance (near the agent) are more reliable, thus the reliability is proportional to those assumptions.

For the discount strategy, we evaluate reliability of data coming from defective agent as follow:

\[
g(R_i) = \exp \left( -\frac{\Delta_{\text{perception}}}{\sigma} \right)
\]

\( \Delta \) is distance between received perception and the “reliability threshold” (in this case, perception with \( \rho < \rho_{\text{threshold}} = 0.5 \) m, \( |\theta| < \theta_{\text{threshold}} = 0.087 \) rad (\( \sim 5^\circ \))). The values of \( \rho_{\text{threshold}} \) and \( \theta_{\text{threshold}} \) are based upon statistical experiments independently executed on the same environmental setup. \( \sigma \) is normalization factor to keep \( g(R_i) \subset [0, 1] \).

In the pruning strategy, only higher reliability (inside the “reliability threshold” cone) data are used.

The result shown in Figure 6 are the average of ten different runs (different trajectory of tracking objects).

**Experiment 2: False positives**

As before in the previous experiment, in the same setup, we have a defective agent that has false perceptions. For example, it could be in an area with some strange light conditions and perceive the object in wrong place. The other agents can detect this malfunction by consensus and reasoning about conflicting data. The percentage of wrong perceptions is dependent on many factors and can be estimated as probability of receiving wrong data.

Again perceptions are artificially preprocessed before they are included in the particle filter. A probability value is used to decide when to add false perception. The experiment is performed with \( \text{prob} \subset [0.1, 1] \).

When receiving observations from defective agent, a probability is drawn and checked with a probability threshold. A measurement is declared “unreliable” if \( \text{prob}_{\text{falsepositive}} > \text{prob}_{\text{threshold}} = 0.8 \). As before the value of \( \text{prob}_{\text{threshold}} \) is based on empirical results.

In the discount strategy this influences the weight of particles

\[
g(R_i) = \frac{\Delta_{\text{prob}}}{\sigma_{\text{prob}}}
\]

where \( \Delta_{\text{prob}} = \text{prob}_{\text{threshold}} - \text{prob}_{\text{falsepositive}} \) and \( \sigma_{\text{prob}} \) is normalization factor to keep \( g(R_i) \subset [0, 1] \).

In the pruning strategy, the same probability is used and observations with \( \text{prob}_{\text{falsepositive}} > \text{prob}_{\text{threshold}} \) are eliminated.

As in the previous experiment, the results shown in Figure 7 are the overall results of ten executions. Despite some noticeable spikes, the discount and pruning show the good behavior of algorithm using reliability information.

**V. Conclusions**

In this paper we illustrated an initial framework for improving tracking methods using reliability, within the Bayesian Theory.

We shown how it is possible to improve performances of a vision based framework for object tracking and self tracking algorithms when reliability of information sources is taken into account.

In particular, we analyze two different scenarios: a Kalman Filter based self tracking algorithm and Particle Filter based multi-agent object tracking. In both systems we assume that

**Figure 6.** Experimental results for systematic errors situation. The results are obtain by the average of ten different runs (different trajectory of tracking objects).
information of objects are taken from noisy sensors like a vision camera.

Moreover, we focused our attention on specific problems that could commonly arise: systematic errors and false measurements (misrecognition and false positives).

Despite the promising results more work has to be done: better methodologies to evaluate reliability coefficients have to be developed. This problem can be easily managed by offline calibration and expert knowledge of estimated domain. A more in depth analysis of using learning algorithms is currently under investigation.

More interesting is the problem of understanding the model of reliability using information from environment, as contextual high level reasoning.

Finally, the problem of multi-agent systems has to be investigated more in detail and many other challenging information fusion processes within mobile robots applications can benefit from the proposed approach.

REFERENCES


Figure 7. Experimental results for false positive error situation. As before they show the average of ten distinctive runs.