On Features and Attributes in Multisensor, Multitarget Tracking

Oliver E. Drummond, Ph. D., P. E.
Independent Consulting Engineer
10705 Cranks Road, Culver City, CA 90230 USA
E-mail: Drummond@Worldnet.ATT.Net

Abstract
An obvious use for feature and attribute data is for target typing (discrimination, classification, identification, or recognition) and in combat identification. Another use is in the data (or track) association process. The data association function is often decomposed into two steps. The first step is a preliminary threshold process to eliminate unlikely measurement-track pairs. This is followed by the second step, the process of selecting measurement-track pairs or assigning weights to measurement-track pairs so that the tracks can be updated by a filter. The primary concern of this paper is the use of feature and attribute data in the data association process for tracking small targets with data from one or more sensors.

1. Introduction
Target tracking problems can be broadly categorized into four generic classes [1], as follows: 1. sensor tracking of a single (bright) target, 2. tracking of targets that are large, 3. tracking of targets that are medium sized, and 4. tracking of targets that are small. These four classes are described in more detail in [2]. Note that the size indicated in this list is in terms of the number of resolution elements or pixels. The algorithms used in the signal, image, and track processing for each of these problems differ. A major concern in tracking small targets is the data association function.

Since each class of tracking problem poses different algorithm development issues, this paper will concentrate on only one class of tracking, namely, tracking of small targets using multiple target tracking methods. Multiple target tracking is a relatively new field. The first book dedicated exclusively to multiple target tracking was published in 1986 [3] and a number of recent books are available [4,5,6]. In addition to the numerous papers and reports in the open literature (too numerous to be listed here), there is an on-going series of annual SPIE conferences concerned exclusively with signal and data processing of small targets that started in 1989 [7]. This paper freely extracts and paraphrases material from some of the author’s prior documents [1,8,9]

For this paper, a small target is characterized as one that does not provide enough data for traditional automatic target recognition (ATR) using a single frame of data [8]. In contrast, a target large enough for ATR typically extends beyond a diameter of about 15 resolution elements, for example, larger than 10 by 10 pixels square. Note that it is not uncommon to refer to all objects as targets whether they are of interest or not. Small targets of concern in this paper include point source targets and small extended targets including unresolved closely spaced objects.

A number of different theories could be used for developing algorithms for processing features and attributes. This paper uses Bayesian probability methods and addresses only track maintenance in order to limit the paper length. The primary tracking function of interest is data association and neither target typing or combat identification is addressed.

2. Features and Attributes
Although tracking small targets is a relatively new field, processing methods developed for tracking a target's trajectory, i.e., kinematic tracking, is fairly mature compared to the processing methods developed for using feature and attribute data in tracking. The term measurement (return, report, observation, or signal processing threshold exceedance) refers to all the data obtained by the signal processor or simply the measurement vector and its error covariance matrix, depending on the context.

As used here the term feature refers to characteristics of a target that are from continuous sample space and
are obtained from sensor data that are other than the simple variables of position and its derivatives that are used for kinematic tracking. Examples of features include estimated target dimensions, radar cross section, and other target signature data. Note that it may be that features are not measured directly but are computed based on a number of measured quantities. Whether the features are measured directly from the data of a signal-processing threshold exceedance or computed based on a number of measured quantities of a signal-processing threshold exceedance, in both cases the resulting feature vector and its error covariance matrix will be referred to as the measured feature.

By comparison, the term attribute will be used here to refer to characteristics of a target based on sensor data that are from discrete sample space, for example, literal, categorical, or integer parameters. Examples of attributes include target type, type of radar systems used by a target, and number of engines on an airplane. These particular definitions were chosen because, as defined, feature data and attribute data are processed differently because their uncertainties are treated differently. Using Bayesian probability methods, features can be processed based on their probability density while attributes can be processed based on their discrete probabilities or point masses.

There are a number of forms that attribute information can take and the form depends in part on how the attributes are to be processed. Due to space limitations, the approach taken for this paper is limited to using probabilistic attribute vectors of a specific type for both the processed attribute state and the measured attributes. In this form, a probabilistic attribute vector contains the probability or likelihood of each of the possible attributes.

What corresponds to the estimated state vector for kinematic tracking is what will be referred to as the processed attribute state vector that contains the a posteriori probabilities of each possible attribute. What corresponds to the measurement vector in kinematic tracking is what will be referred to as the measured attribute vector and it contains the likelihood of each of the possible attributes based on measured attributes or on attributes computed from sensor measurements of an apparent target. The likelihood for an attribute in this form is the probability of obtaining the phenomena observed (measured) by the sensor given that the apparent target exhibits that specific attribute. Note the term apparent target is used because what appears to be a target may actually be due to false signals, persistent clutter, or sensor phenomena not directly and completely due to a single target.

Note that some sensor processors make a hard decision for the measured attributes and that could be represented by the probability of one for the identified attribute. However, these sensor processor decisions will typically exhibit some decision errors. Assuming that an average value of probability of a decision error can be estimated empirically for a sensor, it can be used to convert a sensor processor’s hard decision into a probabilistic attribute vector that contains the probability of each of the possible attributes. If the probability of a decision error is \( P_e \) then the probability of a correct decision is \( P_a = 1-P_e \). Accordingly, the attribute vector would contain the value \( P_a \) for the attribute identified by the sensor processor based on measurements of an apparent target. The attribute vector would contain a value of \( P_e \) for all other possible attributes.

In addition to features and attributes, there is another class of data that has some characteristics of both attributes and features. The term that will be used here for this type of data is categorical features. Categorical features are from continuous sample space (possibly bounded) but they are based on known characteristics of the targets and sensors that allow classified into a finite number of classes or categories. The continuous sample space is caused by either random measurement errors or by the distribution of the inherent parameters of each type of target that cause the features that are measured, or both. An example of a categorical feature is the estimated wing span of an aircraft given there are only a few types of aircraft in the field of regard, the wing span of each type of aircraft is known a priori, and the sensor obtains measurements with measurement errors from which the wing span of a tracked target can be estimated based on a single look by a sensor.

Note that as with features, it could be that the categorical features are not measured directly but are computed based on a number of measured quantities. Whether the features are measured directly as some of the measured quantities of a signal-processing threshold exceedance or computed based on a number of measured quantities of a signal-processing threshold exceedance, in both cases the resulting feature vector and it covariance matrix will be referred to as a measured categorical feature.

In a real tracking system application, the difference between features (as first defined) and categorical features may be muddied. For example, the characteristics of a feature for most targets might be
known but not known for other targets. In fact, depending on why features are processed, the distinction between features and categorical features may have little meaning. For this paper, the term categorical feature is defined for the purpose of facilitating the discussion of how features are processed. Using Bayesian probability methods, categorical features can be processed using composite estimation that uses multiple models based on a hybrid estimation method that combines probability densities and discrete probabilities [1,5,9,10,11].

3. Preliminary Thresholding: "Gating"

The data association function can be viewed as a two-step process, namely, the preliminary thresholding in tracking is frequently the first step in the data association function in a target tracker and is sometimes referred to as gating or gate processing. This first step is followed by the second step that is the process of selecting measurement-track pairs or assigning weights to measurement-track pairs so that the tracks can be updated by a filter. The filter might be a Kalman filter or extended Kalman filter (or an equivalent filter or maybe even an approximation thereto). Note in target tracking the filter update is typically also decomposed into two steps. The first step is the time update to predict the state and the measurement to the time that the next measurement (or set of measurements) was actually observed. After the two steps of the data association, the filter measurement update is performed and the track data is stored in the track files.

3.1 Background: Gating in Kinematic Tracking

In the discussion that follows, for emphasis and clarity, a measurement used for kinematic tracking is referred to as a kinematic measurement. The elements of a kinematic measurement typically consists of the measurements of one or more of the following: range, azimuth, elevation and range rate plus their error covariance matrix.

In many tracking systems, the only purpose for the preliminary thresholding is to reduce the processing load. In kinematic tracking, a region in measurement space is identified that is centered at the predicted position of a target where the measurement is expected to be for that target. That region is the track gate and the size of the region can be established in a number of ways. The method used to size the gate depends on the type of information available. The size of the gate depends on the variance of both the vector of the measurement errors and the vector of the predicted target state, often just position components. For example, a 99.7% gate would be sized so that the correct measurement for a track would be in its gate with a .997 probability. A more effective gate size could be computed using the formula of Eq. 4.7 of [3]. Only measurements that fall within the track's gate, i.e., within the identified region of measurement space, are used in the subsequent data processing for that track.

As an aside, note that there can be important computational considerations in designing the gate processing for a tracker [1]. If there are more than a few targets in the field-of-view, then the process of determining which measurements are in each track's gate (the "gate search" process) can be computationally intensive if simple brute force methods are used. With more than a few targets, simplistic gate search methods should be avoided. In addition, elliptical (ellipsoidal or hyper-ellipsoidal, as appropriate) gates are usually more effective but are also more processor intensive than are rectangular (or hyper-rectangular, as appropriate) gates.

A hyper-ellipsoidal gate process usually involves computing a chi-square statistic (or an approximation of it) of the innovations that is compared to a threshold value. Computing a chi-square statistic typically requires a matrix inversion, multiplies, and additions. In contrast, a hyper-rectangular gate process typically does not require a matrix inversion, and involves only adds, compares, and at most a few multiplies. Thus with more than a few targets, it is advisable to use two gates in series, the first is an oversized hyper-rectangular (or rectangular) gate. The measurements in that track gate, i.e., that pass this first threshold test, are then processed using a second track gate that is a hyper-ellipsoidal (or elliptical or ellipsoidal, as appropriate) gate [1].

3.2 Some Assumptions

The discussion of the gate processing for kinematic processing provides background for the preliminary threshold processing of features, attributes, and categorical features.

To facilitate the discussion, the set of attributes will be assumed to be mutually exclusive and exhaustive. The techniques that are described can be readily adapted to the more general case. The assumption for all random variable from continuous sample space is that any deviation of their probability density function from Gaussian can be neglected. Furthermore, the assumption initially is that either attributes, features or categorical features are obtained with or without kinematic measurements. If there is an attribute
obtained with a kinematic measurement, the assumption is that they are statistically independent. Also, the initial assumption is that features, attributes and categorical features are static, i.e., for a target they do not change over time except for changes due to the errors in measuring them and, if applicable, in estimating them from measurements. It is also assumed initially that the kinematic track filter does not employ multiple models. Many of these various assumptions can be relaxed and these methods adapted to handle the less restricted cases.

Since the track stage of interest is track maintenance, unless indicated otherwise, the assumption is that the a priori information for both the target state and all discrete alternative or hypothesis has already been incorporated into each track, if applicable.

3.3 Preliminary Thresholding of Features

Since features are much like kinematic measurements, features can be processed in much the same way. If the errors of a feature vector are cross-correlated with the errors of an accompanying kinematic measurement vector, then they should be processed as a single vector, including the filtering process. This estimated state vector should be a concatenation of the kinematic states and feature states. If filtered this way, then a properly designed Kalman (or similar) filter should provide consistent covariance matrix of the estimation errors of the estimated kinematic states and the feature parameters.

Computing the vector consisting of the predicted kinematic measurements and predicted features and also computing the covariance matrix provides (along the kinematic measurements and features) the information needed to compute both a hyper-rectangular gate and a hyper-elliptical gate for a track. Thus this processing is identical to gating in measurement space except that it is a higher dimensional space and hence involves more computationally complex processing. The threshold value is computed as discussed in Section 3.1.

If the kinematic measurements and the features are independent, then the processing can be simplified somewhat. The features can be filtered separately from the kinematic measurements. Then for the hyper-rectangular gate processing for a track and a measurement, the magnitude of each element of the kinematic-measurement's innovations vector can be tested in turn against its threshold followed by similar testing of each element of the feature innovations vector. Note that the order of the processing of these two vectors can be reversed or even interleaved, if that ordering is more effective for a tracking system application. If any element of these two vectors fails its test, then that measurement is considered not a potentially valid measurement for that track. The kinematic innovations vector is the difference between the kinematic measurement vector and its predicted vector. The feature innovations vector is the difference between the predicted feature and the measured feature vector. The threshold used for an element of an innovations vector is proportional to the standard deviation of that element based on the innovations covariance matrix. For the hyper-ellipsoidal gate processing for a track and a measurement, two chi-square statistics can be computed separately, one for the kinematic measurements and the other for the features. These two can then be added and compared to the appropriate threshold.

Note that a chi-square statistic is used because in kinematic tracking it is assumed that any deviation of the innovations from exhibiting Gaussian characteristics can be neglected. Furthermore, even if the true probability density of the innovations were known and were not Gaussian, then in most cases it would be too processor intensive to use the proper statistic instead of chi-square. In processing features, however, it may be that the innovations for some features are clearly not Gaussian and the above assumption should be revisited.

An elliptical (or hyper-ellipsoidal) gate is used in gate processing because it is obtained mathematically (in addition to some constants) by computing minus the logarithm of the likelihood function that a specific measurement is due to the target of a specific track. Methods for computing an appropriate threshold for hyper-ellipsoidal gates have been studied extensively and are available, although there are some practical limitations [3,12]. The a posteriori probability that a measurement is due to the target of a specific track is not used in gating because it depends on complicated computations that involve all the measurements and tracks and so that would defeat the purpose of the gating process.

3.4 Preliminary Thresholding of Attributes.

The gate processing of attributes appears to be very different from for kinematic measurements or features. Consider a "minus log likelihood" approach to the gate processing of attributes that is analogous to the gate process used for kinematic measurements and features. Devising such an approach raises the issue of what to use for a threshold value. If the purpose of the gate process is to eliminate unlikely track-
measurement pairs then if there is no rational method to compute a threshold for attributes, then there is no purpose to including attributes in the gate process. This paper proposes approaches for computing this threshold value. First the minus log likelihood computation is described and then methods for computing the threshold are addressed.

Given a vector of attribute probabilities (the measured attribute vector) based on a single current (or recent) measurement and a track including its processed attribute state vector obtained from prior measurements, a scalar can be computed for use in the attribute gate process. The scalar envisioned is the inner product of these two vectors, namely, the inner product of the processed attribute state vector of the track and the measured attribute vector. The resulting scalar, which is the attribute likelihood is in effect

\[ p[a_m(n)|A_j(n-1), j-m] \]

where

\[
\begin{align*}
    j &= \text{track index} \\
    n &= \text{time index} \\
    m &= \text{measurement index (for time } n) \\
    a_m(n) &= \text{phenomena of measurement } m \text{ used to compute the measured attribute vector} \\
    A_j(n-1) &= \text{phenomena of all measurements up to time } n-1 \text{ used to compute the processed attribute state vector for track } j
\end{align*}
\]

and \( j-m \) means that measurement \( m \) is from the target of track \( j \). The final computation is to compute minus the logarithm of this scalar to obtain the minus-log likelihood for that measurement-track pair.

An appropriate threshold is to compute a scalar that is computed in the same way as the attribute minus log likelihood except that the vector of attribute \( a \) \textit{a priori} probabilities of false signals is used instead of the processed-attribute state vector of the track. This requires that a reasonable value be obtained for the probability of each attribute for false signals on average.

If it is not practical to obtain a realistic attribute \( a \) \textit{a priori} probabilities of false signals, then there are a number of alternatives that can be considered. One alternative is to use for the attribute \( a \) \textit{a priori} probabilities of false signals a vector with all the alternative attributes equally probable. That is, if there are \( k \) attributes then the \( a \) \textit{priori} probability of each possible attribute for a false signal is assumed to be simply \( 1/k \).

Another more conservative alternative is to use what will be referred to as the complementary probability vector. The \textit{complementary-probability vector} used for the attribute \( a \) \textit{priori} probabilities of false signals is computed as follows. Form a vector of ones with the same number of elements as the processed attribute state vector and subtract the processed attribute state vector from it. Then normalize the resulting vector by computing the sum of its elements and dividing each element of that vector by that sum to obtain the complementary probability vector. This vector could then be used for the attribute \( a \) \textit{priori} probabilities of false signals to compute the threshold.

Yet another alternative is to just not include attributes in the gating process. However, there may be the need for a practical threshold value for attributes in the second step of the data association process after the gating process, so the above methods might be used for that purpose even if attributes are omitted from the gate processing.

How the threshold processing that is used depends on the type of measurement that is obtained. In most cases the measurement that provides attribute data will also provide a kinematic measurement vector. For that case, the attribute minus log likelihood can be added to the kinematic minus log likelihood (the chi-square statistic) and compared to the appropriate threshold. The appropriate threshold would then be the sum of the attribute threshold (as discussed above) and the kinematic threshold. If the sum of the minus-log likelihood functions is larger than the sum of the thresholds, then that track-measurement pair is not included in the second step of the data association processing. The processing just described is analogous to hyper-ellipsoidal gate processing.

The hyper-ellipsoidal gate can be preceded by a hyper-rectangular gate. For hyper-rectangular gate processing for a measurement-track pair, the order of the processing must first be established. The attributes could be processed before the kinematic measurements of visa versa. The most effective processing order to use and the most effective processing order of the individual measurements in the kinematic measurement vector depends on the specific characteristics of the sensors and targets.

If the kinematic measurements are processed first, then the magnitude of innovation corresponding to each element of the kinematic measurement vector would be processed in turn and compared to its
threshold. If the all these innovation magnitudes are less than their threshold, then the attribute minus-log likelihood function would be tested against its threshold. Any measurement-track pair that passes all these threshold tests would then be processed using the hyper-ellipsoidal gate.

If there are features in addition to kinematic measurements, then they too can be processed along with the kinematic measurements as discussed in Section 3.3. Thus the gate processing to handle kinematic measurements, attributes, and also features would be much like the processing just described for kinematic measurements and attributes. If on the other hand, there are only attributes for a measurement and no features or kinematic measurements, then a single attribute threshold process would serve in place of both types of gates, hyper-ellipsoidal and hyper-rectangular. The extension of the preliminary threshold processing discussed in this section to handle multiple sets of attributes, i.e., multiple attribute vectors that are independent, is straightforward.

3.5 Preliminary Thresholding of Categorical Features

In their simplest form, there are two classes of categorical features. With the simpler of these two classes, call it Class 1, the value for the feature vector (or scalar) is know a priori for each alternative category or hypotheses, i.e., the inherent features for a category, are fixed and deterministic. With the other class, call it Class 2, the values of the feature mean vector (or scalar) and its covariance matrix are know a priori for each alternative category or hypotheses. This covariance matrix is the so called "within class" (in this case "within category") covariance matrix that reflects the variation about the mean of the true feature across targets for a category.

More generally, for Class 3, the mathematical model for the measurement equation and possibly also the dynamic equation might be different for each alternative category or hypothesis. Also the categorical feature state vector need be the same length as the measured feature vector. All three of these classes of categorical feature problems can be processed using non-switching (static) multiple-model methods [1,5,9,10,11] if it is assumed that the feature characteristics do not change over time for a target.

Yet another aspect of processing categorical features is the dependence of the kinematic measurements on the categorical features. There are two distinctly different types of dependencies. First the characteristics of the kinematic measurements may or may not depend on the feature category for a target. Alternatively, the feature category for a target could depend on the kinematic measurements or the kinematic state, but that can be even more complex and will not be discussed here due to page limits. Note that an even more complex relationship is conceptually possible where the dependency of the kinematics and the feature category for a target is in both directions.

The second type of dependency is between both the estimation errors of the kinematic state and kinematic measurement errors and the measurement errors of the measured categorical features for a target. Remember that it may be that the measured categorical features are not measured directly but rather might be computed from data obtained in conjunction with a measurement of an apparent target.

For a particular system application there are four possible combinations of these two types of dependencies and the processing method for one type may not be the best for another. Considering these four possible combinations of dependencies along with the 3 classes of categorical features could lead to 12 different processing methods to be explored.

To simplify this discussion, only three of these combinations will be addressed. First the simpler case of the errors of the measured categorical features independent of both the kinematic measurements and kinematic estimated state for a target and also independence between feature category and both the kinematic measurements and kinematic estimated state for a target will be addressed. This case will be discussed for the two simpler categorical feature classes.

For this simpler case and for all three classes of categorical features, the kinematic measurement is processed in a filter separately from the feature filtering, if applicable, for each category. Also a kinematic chi-squared statistic is computed from the kinematic innovations for a track-measurement pair independently of the categorical feature data.

For the Class 1 categorical features, a filter in not needed because the values for the inherent categorical features for a track are known a priori for each feature category. The difference between the measured categorical feature vector for a measurement and the a priori value for the feature vector for a feature category serves as the innovations vector for a feature category for a track-measurement.
pair. For each track-measurement pair the chi-square statistic is computed for every feature category. Note that for this case the processing can be simplified because the categorical feature chi-square statistic does not depend on the tracks, only on the measurements and the feature categories. Accordingly, the categorical feature chi-squares can be computed for each measurement and all feature categories without using any track data. These chi-square statistics are then used along with the additional constants needed to compute the likelihood function of the features for each feature category.

These likelihood functions are used to compute the measured categorical feature vector. What corresponds to the measurement vector in kinematic tracking is the measured feature category vector that contains the likelihood of each of the possible feature categories based on measured features for a measurement and the feature a priori feature values for the feature categories. What corresponds to the estimated state vector for kinematic tracking is what will be referred to as the processed feature category state vector that contains the a posteriori probabilities of each possible feature category for a track based on processing all prior measurements.

The process that follows is like the processing of attributes. The scalar is computed, the categorical feature likelihood, that is the inner product of these two vectors, namely, the inner product of the processed feature category state vector of the track and the measured feature category vector. The final computation is to compute minus the logarithm of this scalar to obtain the categorical feature minus-log likelihood for that measurement-track pair.

The threshold used with the categorical feature minus-log likelihood is computed using a method that is similar to that used for attributes. An appropriate threshold value might be using a priori characteristics of false signals.

The threshold is computed in the same way as for the attribute minus log likelihood except for how the vector of feature category a priori probabilities of false signals and the a priori value for the feature vector for false signals are computed. First the difference between the measured categorical feature vector for a measurement and the a priori value for the feature vector for false signals serve as the innovations vector for a false signal. For each measurement the chi-square statistic is computed for every feature category. This needs to be computed only once for each measurement. These chi-square statistics for the feature categories for a measurement are then used along with the additional needed constants to compute the likelihood function for each feature category for false signals.

These likelihood functions are then used to compute the measured false signal categorical feature vector that contains the likelihood of each of the possible feature categories for a measurement-track pair. What corresponds to measured categorical feature vector is the measured false signal categorical feature vector that contains the likelihood of each of the possible feature categories based on the a priori false signal characteristics and a measurement. What corresponds to the processed feature category state vector is what will be referred to as the feature category a priori probabilities vector for false signals that contains the discrete a priori probabilities of each possible feature category for false signals. The threshold is computed by computing the inner product of the measured false signal categorical feature vector and the feature category a priori probabilities vector for false signals. Note that to use this method to compute the threshold requires that a reasonable value be obtained for the probability of each feature category for false signals on average and also the value of the category feature vectors for false signals for each category.

If it is not practical to obtain realistic categorical feature a priori information for false signals, then there are a number of alternatives that can be considered. One alternative is to use for the feature category a priori probabilities of false signals a vector with all the alternative categories equally probable as discussed in Section 3.4. For the chi-square values needed to compute the measured false signal categorical feature vector, the value of chi-square corresponding to cumulative probability of say 0.997 could be used, or what ever other value is appropriate for the track system at hand. Yet another alternative for the feature category a priori probabilities of false signals is to use the a complementary-probability vector for the feature category a priori probabilities of false signals. Thus there are a number of ways of computing the threshold value for the threshold for testing the categorical feature minus-log likelihood.

The categorical feature minus-log likelihood is processed along with the kinematic chi-square statistic (if kinematic measurements are available) in the same way as for attributes as outlined in Section 3.4 to complete the hyper-ellipsoidal gate processing. This gate processing also can be preceded by hyper-rectangular gate processing as outlined for attributes. Thus the preliminary threshold processing step for Class 1 categorical features is the same as for the
processing attributes except for the computation of the measured categorical feature vector and of the threshold which do differ from the computations used for attributes.

The Class 2 category feature processing differs from the Class 1 processing because the values of inherent features for a feature category are not deterministic. Rather, they are assumed to be characterized for each feature category by their mean and covariance matrix. There are a number of methods that can be used to process Class 2 categorical features and some are more efficient than others. The method described here is not necessarily the most efficient but is relative easy to describe so as to convey the concepts that apply. As mentioned previously, the method that follows is applicable to the case in which both the measurements and the measurement errors are independence of both the feature measurement errors and the feature category for a target.

The primary difference between the processing of Class 1 and Class 2 categorical features is that in Class 1 no processed categorical state vectors (one for each category) are maintained for each track but they are computed for each track for Class 2. In both classes, a processed feature category state vector is maintained for each track.

The difference between the measured categorical feature vector for a measurement and the predicted value for the processed categorical feature state vectors for a feature category for a track serves as the innovations vector for a feature category for a track-measurement pair. For each track-measurement pair the chi-square statistic is computed for every feature category. These chi-square statistics are then used along with the additional constants needed to compute the likelihood function of the features for each feature category for a measurement-track pair.

These likelihood functions are then used to compute the measured categorical feature vector that contains the likelihood of each of the possible feature categories for a measurement-track pair. The categorical feature likelihood is then computed as for Class 1 and the final computation is to compute minus the logarithm of this scalar to obtain the categorical feature minus-log likelihood for that measurement-track pair. The threshold value is computed and the hyper-ellipsoidal gate process is completed as with the Class 1 processing and can be proceeded by hyper-rectangular gate processing as for Class 1.

Finally, consider processing Class 3 categorical features with both types of dependency. That is, characteristics of the kinematic measurements depend on the feature category for a target and also both the estimation errors of the kinematic state and kinematic measurement errors depend on the measurement errors of the measured categorical features for a target. For this class of categorical feature problem the gate processing is as just described for Class 2 except that there is a single filter for each feature category for a target for both the kinematic and the categorical feature data. The kinematic and categorical features are processed simultaneously as a single vector for both the hyper-rectangular and hyper-ellipsoidal gate processes. The processing described for Class 2 applies by using the categorical feature processing that was described for both the kinematic and categorical feature measurements and similarly for the estimated states. Accordingly, this particular Class 3 problem could be processed as a non-switching multiple model problem in which the state is composed of both the kinematic state elements and the categorical feature state elements.

For gate processing, a few of the many kinds of problems that can involve kinematic, features, attribute, and categorical feature have been discussed. Of course there are more combinations that deserve attention than those discussed. Also the probabilistic derivations that are the basis for the processing methods described have not been presented due to space limitations. Finally, some of the simplifications that could further reduce the processing have not been discussed nor have the adaptation of these methods to less restricted problems in which some of the assumptions are relaxed.

4. Data Association, Step 2

The second step of the data association function is to select measurement-track pairs or assign weights to measurement-track pairs so that the tracks can be updated by a filter. There are a variety of algorithms for this process [1,2,3,5,6,7], including both single frame and the more complex multiple frame processing, such as multiple hypothesis tracking. In addition, there are hard decision approaches, such as (independent) nearest neighbor and most probable hypothesis tracking and there are soft decision approaches that are also called probabilistic or Bayesian method. While these approaches are all different, they can be classified for the purpose of this discussion into two groups. In their data association decision or weighting process, one group uses the minus-log likelihood function for each track-measurement pair. For the other group, the likelihood
functions are used. Many single frame methods, for example, that make hard decisions use the minus log likelihood function. By contrast, soft decision methods use the likelihood functions for each track-measurement pair that survives the gate processing.

Given the minus-log likelihood for a measurement-track pair and the appropriate accompanying constants, then the likelihood for that measurement-track pair can be computed. Accordingly, all the information needed for the second step of the data association process has already been addressed in Section 3 for the specific types of problems that were discussed. In many cases the hyper-elliptical gate threshold values discussed in Section 3 are also needed in the second step of the data association processing.

In addition to the data association function in track maintenance is the filtering function. If the processing included either attributes or categorical features, then an additional process function is needed to supplement the filter process. That additional function is to update the processed attribute state vector and the processed feature category state vector, if applicable. This processing can be accomplished using a straight forward application of Bayes rule. The other track maintenance functions are track promotion-demotion logic and track management which should not require any major modifications (at least conceptually) beyond those used for processing kinematic data to accommodate features, attributes, and/or categorical features.

5. Conclusions

In this paper, the types of measurement data used for multiple target tracking with data from multiple sensors has been classified into four types, namely, kinematic, feature, attribute, and categorical features. The motivation for this classification scheme was to partition the types of data according to how it might be process in a tracker because different processing methods are required depending on the characteristics of the data. Processing approaches have been outlined that illustrate how the processing might differ if features, attributes or categorical data were available in addition to kinematic data. The form of the state that corresponds to each of these data types was also shown to depend on the data type.

The paper introduces methods for computing the threshold for the gate processing for attributes and categorical features that are substantially different from methods used for kinematic and feature measurements. Material left for subsequent documentation include the derived equations for the processing methods presented, identified methods to further simplify the processing, describing the processing for other combinations of the types of data, and extension of the processing methods to accommodate relaxation of the assumptions used for the purpose of this paper.

Acknowledgements

Discussions with Dr. Rabinder Madan were very helpful in establishing the direction for this study.

References


11. Sims, F. L. and D. G. Lainiotis, Recursive algorithm for the calculation of the adaptive Kalman