A Unified Model for Context-Based Behavioural Modelling and Classification

Joel Janek Dabrowski[a,*], Johan Pieter de Villiers[b,a,**]

[a]Department of Electrical, Electronic and Computer Engineering, University of Pretoria, 2 Lynnwood Rd, Pretoria, South Africa
[b]Council for Scientific and Industrial Research, Meiring Naudé Rd, Lynnwood, Pretoria, South Africa

Abstract

A unified Bayesian model that simultaneously performs behavioural modelling, information fusion and classification is presented. The model is expressed in the form of a dynamic Bayesian network (DBN). Behavioural modelling is performed by tracking the continuous dynamics of a entity and incorporating various contextual elements that influence behaviour. The entity is classified according to its behaviour. Classification is expressed as a conditional probability of the entity class given its tracked trajectory and the contextual elements. Inference in the DBN is performed using a derived Gaussian sum filter. The model is applied to classify vessels, according to their behaviour, in a maritime piracy situation. The novel aspects of this work include the unified approach to behaviour modelling and classification, the way in which contextual information is fused, the unique approach to classification according to behaviour and the associated derived Gaussian sum filter inference algorithm.

Keywords: Dynamic Bayesian Network, Switching Linear Dynamical System, Information Fusion, Behavior Modelling, Activity Recognition, Maritime Piracy.

1. Introduction

Several dynamical models have been proposed in expert system literature for behaviour modelling, information fusion and classification of time series data. Methods include machine learning algorithms, fuzzy time series models, Box-Jenkins models, state space models, Bayesian networks and dynamic Bayesian networks. The models are often formulated with layers of different methods to achieve the desired goal. The layers or steps involve operations such as data preprocessing, feature extraction, information fusion and training. Preprocessing methods often involve partitioning and clustering of time series data resulting in loss of information. The complexity and variation increase with the number of methods and layers included in such models. Furthermore, behaviour is often a complex temporal entity. Many methods are not able to naturally consider causal relationships of variables over time. Methods that are able to consider temporal relationships are however often limited to application or task.

In this study a novel model that performs behaviour modelling, information fusion and classification simultaneously is proposed. Behaviour modelling is performed by modelling the continuous dynamics of a target. Information fusion involves the integration of contextual elements that influence behaviour. Classification is performed based on behaviour. The model is initialised by defining probabilistic relationships between variables. Once initialised, no preprocessing or feature extraction is required during its on-line operation. Only sequential tracked locations of a target and the contextual element data are required. The Markovian component of the model allows for the modelling of temporal relationships in a causal manner.

In the context of this study, behaviour may be described according to the activities a system engages and how the system transitions between these activities. The behaviour is modelled by defining a set of behavioural activities and ascribing Markovian transition probabilities between the activities. Each activity is modelled according to predefined dynamics. The transitions between activities are influenced by contextual information. By tracking the behaviour of the system, a class may be inferred.

The proposed model is in the form of a dynamic Bayesian network (DBN). It consists of several functional components that include a linear dynamic system (LDS), a switching state variable, a set of contextual element variables and a class variable. The LDS is included to model the continuous dynamics of behavioural activities. The Markov based switching state variable is included to model the transitions between various behavioural activities. The LDS and switching state variable together form a switching linear dynamical system (SLDS). The set of contextual element variables are included for the fusion of contextual information that influences behaviour. The class variable is included for classification. The Gaussian sum filtering (GSF) algorithm is applied to perform inference on the DBN for the purpose of classification.

The model is applied to the problem of classifying vessels in a maritime piracy situation. The problem of identifying maritime pirate vessels is scarcely addressed in literature. A dataset

*Corresponding author.
**Alternative contact details via co-author (other than email): tel: +27 12 841 2590; fax: +27 12 362 5000; Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Private Bag X20 Hatfield, 0028, Pretoria, South Africa.
Email addresses: dabrowski.joel@gmail.com (Joel Janek Dabrowski), jdvilliers1@csir.co.za (Johan Pieter de Villiers)

of simulated vessels operating within a maritime piracy situation is utilised. The simulated data consists of tracked location data, ocean conditions, weather conditions, time of day, season of year, regional location data and vessel class ground truth. Each vessel in an environment is classified as either a pirate, transport or fishing vessel. The results presented demonstrate a classification accuracy of between 71.6% and 99.5% depending on the differences in kinematics of the vessel classes.

2. Related Work

The method presented in this study is unique. To affirm this, a literature study that covers a wide range of applications and methods is thus conducted. The survey has a focus on expert systems and recent publications. Studies that consider behaviour modelling of maritime vessels, humans and animals considered. Time series analysis and forecasting methods are explored. Forecasting methods are relevant as they require a dynamic model of a system to perform forecasting. Related context-based and information fusion applications are discussed. Finally, maritime piracy related methods are covered. It is noted that there are few studies in the literature that attempt to identify maritime pirate vessels.

2.1. Maritime Vessel Behaviour Modelling

Various methods in the literature have been proposed for modelling behaviour of maritime vessels. DBN and kinematic based models are of particular interest. A DBN is applied for abnormal maritime vessel behaviour detection in (Castaldo et al., 2014). An event-based DBN is applied to describe causal relationships of normal ship movements. Normal ship movements are related to specific zones defined by a topographical map. A low level observation layer is defined to analyse trajectories. A high level event layer is defined where zone changes are used to detect abnormal behaviour. The method performs on-line learning of behaviours. Mazzarella et al. (2014) propose a method for discovering maritime vessel activities using automatic identification system (AIS) data. The AIS data is used to discover fishing areas. A 'stops' and 'moves' trajectory partitioning method is used to detect fishing behaviour events. This method essentially utilises the motion dynamics of the vessel to detect activities. For example, fishing is characterised by significant variations in course over ground. Fishing regions are discovered through the clustering of detected fishing events. These studies demonstrate that tracked data and trajectory may be effectively used for behaviour modelling.

2.2. Human Behaviour Modelling

Human behaviour modelling and detection is an application that has been widely studied in the field of computer vision and surveillance. The approach to behaviour modelling proposed in this study may be considered to stem from approaches presented in this field. Furthermore, the model proposed in this study may easily be applied to the human behaviour detection problem. Surveys on human behaviour recognition are presented in (Gowsikhaa et al., 2014), (Turaga et al., 2008) and (Hu et al., 2004). Methods for human behaviour analysis include graphical models (including the DBN), the HMM, rule-based approaches, support vector machines, syntactic approaches, dynamic time warping, finite state machines and neural networks. Of particular relevance to this study is the DBN, the HMM and other state space methods such as the linear dynamical system (LDS).

State space models are the most commonly used methods for modelling temporal dynamics in human behaviour recognition applications (Turaga et al., 2008). The Kalman and particle filters are popular filtering methods for tracking. Hernàndez et al. (2014) propose a method for human activity recognition based on kinematic features. Of particular interest, the humans are tracked using a state space model and a local search particle filter. Walia and Kapoor (2014) propose an evolutionary particle filter for object tracking that is based on the improved cuckoo search. Luo et al. (2014) use the parameters of a robust linear dynamical system as motion features. Human actions are classified using the maximum margin distance learning method by combining the motion features and local appearance features. A recent expert system application using the HMM is presented in (Kodagoda and Sehestedt, 2014). A sampled HMM is used to model pedestrian motion patterns and simultaneously track people with a particle filter. The model provides the means to track pedestrians under long term occlusions. These studies demonstrate the ability for state space methods to track and model complex objects.

The DBN is a parametric method that is well suited to model complex actions (Turaga et al., 2008; Gowsikhaa et al., 2014). Various forms of DBNs have been proposed in literature for human behaviour modelling (Gowsikhaa et al., 2014; Turaga et al., 2008). A recent example of such a system is presented in (Wang and Ji, 2014). The model provides the means to simultaneously incorporate contexts into a unified model. Various layers of the model are compiled. These layers include object detection, tracking and feature selection. The DBN is trained according to the features. The DBN is described as a coupled HMM that captures dynamic interaction between target appearance and motion.

The human behaviour modelling applications described generally use the DBN to model dynamics of the systems. The DBN is not necessarily used for information fusion as well as classification. The method proposed in (Wang and Ji, 2014) does allow for context to be included. However, the DBN is modelled on high level features that have been extracted from the data.

2.3. Animal Behaviour Modelling

Expert systems for animal behaviour classification is particularly relevant in livestock applications. Classification of behaviour provides a means to manage and monitor livestock. In general, the livestock are monitored using collars containing sensors such as global positioning systems and inertial measurement units (accelerometers). A method that represents different cattle motions using mixture models has been proposed in (Gonzalez et al., 2015). A decision tree is used to classify behaviour according to thresholds defined by the mixture models. In another application, a two stage classifier is proposed for cattle behaviour classification (Dutta et al., 2015). In the first stage, clustering is performed using probabilistic principle component analysis, Fuzzy C-means and self organising maps. In the second phase the clustering results are classified using an ensemble classifier using ensemble methods such as bagging and AdaBoost along with classification methods such as binary tree, linear discriminative analysis and naive Bayes classifiers. Similar approaches have been used for sheep behaviour classification. For example, see (Umstatter et al., 2008).

The dynamics or temporal characteristics of the data are ignored in the above described two-step methods that involve clustering and classification. To address this limitation, temporal information is included using the bag of class posteriors (BOCP) method in (Smith et al., 2015). Temporal information is considered by creating intervals and estimates are fused across time. Features are extracted from each time interval and used for classification.

In the methods presented, only one considers the dynamics of the system. The dynamics are however extracted as features from intervals. Such methods often lead to concerns of selecting correct interval lengths. Continuous models of dynamic systems, such as the linear dynamical system do not encounter such problems.

2.4. Other Behaviour Modelling Related Applications

Other behaviour modelling related expert system applications include alcohol consumption detection (Robinel and Puzenat, 2014), water end use demand forecasting (Nguyen et al., 2014; Bennett et al., 2013), stock market forecasting (Hassan et al., 2007) and dynamic clustering of energy markets (Dias and Ramos, 2014).

2.5. Time Series Analysis

Time series analysis models include stationary process models, spectral models, state space models, and non-linear models (Chatfield, 1996). The autoregressive process (AR) and moving average process (MA) are two traditional stationary process models. These models have been combined and extended to form the Box-Jenkins models. These include the autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) models. Spectral (or frequency domain) methods involve the estimation of spectral density function of a time series. Such methods include the Fourier transform and the wavelet transform (Shumway and Stoffer, 2010). State space models are vector-based first order differential equation models that describe dynamic systems. Various types of non-linear time series models, such as the neural network have been applied to time series analysis.

State space models represent a dynamic system as a set of first order differential equations in the form of a state vector (Franklin et al., 2002). State space methods are considered to be more flexible and capable of handling a wider variety of problems than the Box-Jenkins methods (Durbin and Koopman, 2001). In time series analysis literature, state space models are often associated with the Kalman filtering algorithm. The Kalman filter is applied to the linear dynamical system (LDS) or linear Gaussian state space model (Barber, 2012). A limitation of the LDS is that it models systems that exhibit linear dynamics. The extended Kalman filter and the unscented Kalman filter have been developed to address nonlinear systems (Thrun et al., 2005). In their extensive survey on time series forecasting methods, Gooijer and Hyndman (2006) note that the use of the state space framework using the Kalman filter is limited in literature. Furthermore, it is suggested that some textbook authors are unaware that the Kalman filter can track non-stationary processes stably. This perhaps indicates a general lack of knowledge of the advantages of the state space approach.

Fuzzy time series (FTS) is a method that uses the fuzzy logic for modelling and forecasting time series. The general approach to forecasting involves fuzzification of the time series, establishing fuzzy relations and defuzzification (Yu and Huarng, 2010). Fuzzification involves partitioning of the data into fuzzy sets. The fuzzy relations are defined to describe temporal relationships between fuzzy sets. Forecasting values are computed under defuzzification rules. Selecting appropriate partitions and the fuzzy relations are challenging problems (Yu and Huarng, 2010; Askari and Montazerin, 2015). Askari and Montazerin (2015) address the problem of selecting appropriate partitions by constructing the fuzzy sets using fuzzy clusters. Additionally, problems associated with high order FTS and multiple variable time series are addressed in this study. In particular, high order FTS can be applied to a single variable FTS and multi-variable FTS are applied to single order FTS. Yu and Huarng (2010) address the problem of defining fuzzy relations by using a neural network to calculate them. The FTS methods may thus require additional methods to address challenges encountered. This is likely to result in increased computational complexity of the FTS solution as a whole. Furthermore, through fuzzification the data is discretised into fuzzy sets. Through this process information may potentially be lost.

Many applications of artificial neural network (ANN) to time series analysis and forecasting may be found in literature. The ANN is generally applied for forecasting using a sliding window technique (Frank et al., 2001). The network consists of a set of *N-tuple* inputs, a set of hidden units and a single output. The *N-tuple* input slides over the complete training set. The output provides the forecast estimate. Various problems and challenges are associated with ANNs. Selecting the initial weights and thresholds is a challenging problem. Incorrectly selected values may lead to a sub-optimal results where the optimisation algorithm selects local optima rather than global optima. Wang et al. (2015) propose the use of an adaptive differential evolution algorithm to select appropriate initial connection weights and thresholds. Kocadağlı and Aşıkgil (2014) use a Bayesian inference approach to train a ANN. An evolutionary Monte Carlo algorithm is proposed. The method is based on Gaussian approximation with recursive hyperparameter. It integrates Monte Carlo simulations with genetic algorithms and fuzzy membership functions. Du et al. (2014) propose a knee point-based nondenominated sorting adaptive differential evolution for multiobjective optimisation in an ANN.

Though the above discussed algorithms may improve training, they are often complex and computationally expensive. Furthermore, ANNs are considered to be a 'black-box' methods. They do not provide insight into the nature of the interactions between variables (Lai et al., 2009).

ANNs may also be combined with other methods to form hybrid ANNs. A recurrent neural network is combined with an autoregressive moving average and an exponential smoothing method for prediction of stock returns (Rather et al., 2015). The optimal weights of the model are determined using a genetic algorithm. A limitation of the method however is that it is data dependent. Saâdaoui and Rabbouch (2014) propose a wavelet based feed forward ANN. The model is nonlinear vector-autoregressive model. The wavelet transforms are used for preprocessing the time series data.

ANNs may also be combined with themselves to form an ensemble of neural networks. Ensemble ANNs have been shown to improve robustness and accuracy when compared to a single ANN. Challenges relating to the ensemble operator that combines the results of the ensemble may be encountered. Kourentzes et al. (2014) propose a mode operator that is based on kernel density estimation. This operator is demonstrated to outperform the mean and median operators.

Hybrid and ensemble ANNs may improve robustness and accuracy, this however comes at the cost of increased complexity. Computation is required for each of the methods included in the model.

Other methods used for time series analysis include granular information (Al-Hmouz et al., 2015), dynamic time warping (Górecki and Łuczak, 2015), wavelets (Joo and Kim, 2015) and other machine learning algorithms. Granular information is often combined with fuzzy time series (Lu et al., 2014; Wang et al., 2014). A general framework for applying machine learning methods for time series prediction is presented in (Wu and Lee, 2015). The framework involves four steps. Firstly, local context of the user query is found using the k-nearest-neighbours method. Secondly, the appropriate number of lags is selected by applying mutual information to measure relevance. Thirdly, a set of training patterns is extracted from the data. Fourthly, the training patterns are fed to the machine learning algorithm. The framework is demonstrated using a ANN, an adaptive neuro-fuzzy inference system and a least squares support vector machine. One drawback of the framework is that the k-nearest-neighbours method does not perform well with noisy data. A second drawback is that the method is computationally expensive.

2.6. Context-Based Applications and Information fusion

Context-based applications model and fuse contextual information. Context may be defined as any information that can be used to characterise the situation of an entity Dey (2001). A context modelling survey has been conducted in (Strang and Linnhoff-Popien, 2004). A more recent survey on context modelling and reasoning in pervasive computing has been conducted in Bettini et al. (2010). Applications of context-based information fusion include human activity classification (Xu et al., 2014), natural language processing (Steinberg and Rogova, 2008), computer vision (Gómez-Romero et al., 2012) and video indexing (Kennedy and Chang, 2007). Context-based information fusion has been used to address various maritime threat and situation assessment problems (Chen et al., 2014; Garcia et al., 2011; Hegde et al., 2009; George et al., 2009). In the more recent application, Chen et al. (2014) propose a context-based expert system that uses a genetic algorithm for knowledge discovery. The genetic algorithm is used to extract knowledge through induction of production rules. The DBN does not feature in any of these applications.

With relevance to the DBN, a location estimation problem has been addressed using a DBN for context-based information fusion (Sekkas et al., 2006). The system is a multi-layered system that uses a discrete valued DBN to fuse information from various sources. Fuzzy logic has been applied to extend this application for imprecise contextual reasoning (Sekkas et al., 2007; Anagnostopoulos et al., 2007). The fuzzy logic is used for the representation of context attributes and for inference based on reliability of the data sources. The referenced study does not necessarily model behaviour of the tracked entities.

The DBN model proposed in this study is intended to fuse data from multiple sources in a maritime environment. In literature, information fusion methods have been utilised in various maritime surveillance applications. The ability to include multiple variables and causal relationships between variables makes the BN highly suitable for information fusion. The BN has thus been widely utilised in information fusion applications (Das, 2008). The BN is commonly used for decision making and analysis. In the maritime domain, the BN has been utilised in maritime safety and security (Hanninen et al., 2014; Hanninen and Kujala, 2014; Kruger et al., 2012; Fooladvandi et al., 2009) and maritime domain awareness applications (Costa et al., 2012; Carvalho et al., 2011). Of these references, Costa et al. (2012) is the only study that proposes a BN consisting of both discrete and continuous variables.

Applications of the DBN in information fusion are far scarcer in literature than applications of the BN. A multi-sensor information fusion framework is proposed for fusing dynamically available sensor information for decision making (Zhang and Ji, 2006). The framework includes a method to perform sensor selection. Behavioural modelling is not the purpose of this framework. An application for driver fatigue recognition is proposed (Yang et al., 2010). A discrete variable DBN is used for inferring levels of fatigue. A fuzzy method is used to determine the discrete variables and their state values. The proposed DBN is application specific and does not model kinematic behaviour. The DBN has been used for information fusion in human-computer interfaces (Pavlovic, 1999). Various forms of DBNs, including a hybrid DBN are proposed. The hybrid DBN is proposed for hand gesture recognition and is extended for visual tracking in images. The structure of this DBN is similar to that of the SLDS. The proposed model is application specific and would not extend well to the maritime pirate surveillance problem. In contrast to these referenced studies, the DBN proposed in this study consists of a hybrid DBN that models kinematic behaviour in the context of a particular situation.

2.7. Maritime Piracy Applications

Maritime piracy poses an economic, humanitarian and environmental threat (Middleton, 2008). The problem of maritime piracy is a global concern. Three counter-piracy missions were deployed in late 2008. These include NATO's Operation 'Ocean Shield', EU's Operation 'Atlanta' and the US-led Combined Task Force-151 (Bueger et al., 2011). War ships have been deployed to assist maritime piracy victims and patrol high risk regions. Patrolling efforts are only partially successful due to the vast region that requires patrolling. The application of technology for assisting in the counter piracy endeavour has been proposed (Heger et al., 2009). An advanced study institute (ASI) was held in Salamanca, Spain in September, 2011. The ASI was held to discuss the maritime piracy problem. The objective of the discussions was to assist in deterring, predicting and recognising maritime piracy using information systems (Bossé et al., 2013). Topics such as situation assessment methods, surveillance, information fusion methods and challenges associated with the collaboration between humans and information systems were discussed.

Various methods for counter piracy have been proposed in literature. Spatial and temporal patterns of pirate incidents have been modelled (Marchione and Johnson, 2013; Marchione et al., 2014). The role of contextual knowledge and information fusion in the context of maritime piracy has been discussed in (Rogova and Garcia, 2013). Several methods for maritime pirate simulation have been developed (Jakob et al., 2011; Esher et al., 2010; Varol and Gunal, 2013). The application of game theory has been used to optimise strategies in counter piracy applications. Game theoretic applications that suggest transiting routes that avoid hostile pirate encounters have been proposed (Vaněk et al., 2010, 2011). A game theoretic approach has been developed to optimise patrolling strategies to combat maritime piracy (Marsh, 2009). Various risk analysis applications have been presented for countering maritime piracy (Sevillano et al., 2012; Liwang et al., 2013). Risk analysis is commonly used to assist captains and ship owners in managing risk during a pirate attack. An application of particular relevance is a BN approach to risk management in offshore oil fields presented in Bouejla et al. (2014).

Few pirate detection approaches have been documented in literature. A satellite communication monitoring system for detecting maritime pirates has been proposed (Baldini et al., 2010). Other methods that detect pirate vessels by small craft classification in various forms of imagery have been proposed (Teutsch and Kruger, 2010), (Sanderson et al., 1999). A computer vision method of identifying and tracking vessels, such as pirate vessels, is presented in (Szpak and Tapamo, 2011). In contrast to these applications, the DBN proposed in this study provides a complete framework that considers kinematic behaviour in context of an environment and situation. Through inference, vessels in a pirate situation may be classified. The classification provides the means to identify pirate vessels. The ability to identify maritime pirate vessels provides a contribution to the counter piracy endeavour.

Though few pirate detection approaches have been documented in literature, several methods exist for maritime vessel classification. An obvious method is to identify a vessel class according to AIS data transmitted by the vessel (Tetreault, 2005). However, as discussed in Section 1, AIS data is often unreliable. The data sources that are commonly used for classifying maritime vessels include radar, infra-red and optical imagery. Synthetic aperture radar (SAR) and inverse synthetic aperture radar (ISAR) are commonly used radar imagery methods (Bon et al., 2008). A particularly large body of literature exists for classification using SAR imagery (Greidanus, 2008, 2005; Arnesen and Olsen, 2004). Classification using low resolution radar has also been considered in (Gibbins et al., 1999). Several approaches to vessel classification using infra-red imagery have been published (Teutsch and Kruger, 2010; Alves et al., 2004; Li and Wang, 2008). Vessel classification using land based and ship based optical imagery in harbours has been proposed (Zia-ur Rahman, 2008; Lam, 2013). High resolution satellite optical imagery has also been successfully applied to vessel classification (Corbane et al., 2008). Satellite optical imagery provides more detail than that of radar imagery. Optical imagery however has lower region coverage than that of radar imagery and is more susceptible to weather conditions (Corbane et al., 2008; Greidanus, 2005). Moreover, optical imagery classification is not applicable during night time; though infrared imagery is. Vessel classification methods based on imagery are based on physical features of the vessels rather than the vessel behaviour. Furthermore, imagery methods are often not suitable for vessel sizes under $10 - 15$ meters (Greidanus, 2005). The classification method presented in this study is not limited by vessel size and does not require any physical features of the vessel. The only information required is the tracked trajectory of the vessel.

3. Background

3.1. Methods

The Bayesian network (BN) (Pearl, 1988) is extended through time to form the DBN (Dean and Kanazawa, 1989). By extending the BN through time, sequential data may be modelled (Dean and Kanazawa, 1989; Barber, 2012; Murphy, 2002; Koller and Friedman, 2009; Verner and Nielsen, 2007). The BN and the DBN consist of a set of nodes and links that, together, describe a joint probability distribution. Each node represents some random variable. The links between nodes describe a causal relationship between the variables. The causal relationships are represented by discrete or continuous conditional probability distributions. The DBN provides a powerful and flexible framework that may be used for modelling complex systems. The DBN has been proposed for use in a wide variety of applications. Applications include vehicle detection and tracking (Petrovskaya and Thrun, 2009) human motion analysis in computer vision (Luo et al., 2003), learning from observation (nón et al., 2014) and situation awareness (Wiggers et al., 2011).

The switching linear dynamic system (SLDS) (Bar-Shalom and Li, 1993; Barber, 2012; Murphy, 2002) forms a foundational part of the structure for the DBN presented in this study. Various problems have been successfully solved using the SLDS. These include speech recognition (Mesot and Barber, 2007), econometric (Kim, 1994) and human figure tracking problems (Pavlovic et al., 1999). Several other human behaviour detection applications are discussed in Turaga et al. (2008). The SLDS is an extension of the linear dynamic system (LDS). The LDS is extended by including a discrete switching state variable that defines a particular kinematic state of the modelled system. The SLDS is generally applied to model a piecewise linear dynamical system.

By nature, the SLDS does not provide a direct means for classification. The model proposed in this study may be argued as an extension of the SLDS into a larger DBN as to provide the means for information fusion and classification. No other proposals in literature have been found in which the SLDS has been extended in this manner.

3.2. Inference Algorithms

A vast variety of methods for inference for the LDS and HMM have been documented in literature. The graphical model structure of the LDS is identical to the HMM (Minka, 1999). Exact inference is possible for both of these models. The Kalman filter (Kalman, 1960) is an example of a particularly well-known method for inference in the LDS. The forward-backward and the Baum-Welch algorithms are well known methods for inference in the HMM (Barber, 2012). Inference in the SLDS is NP-hard (Murphy, 2002; Barber, 2012) (See Section 5 for further discussion). Exact inference and simple inference methods such as those applied to the HMM and LDS cannot be applied to the SLDS. More complex approximate inference methods have been proposed for the SLDS. These include the assumed density filtering (ADF) algorithm (Alspach and Sorenson, 1972; Boyen and Koller, 1998; Minka, 2001), the approximate Viterbi inference algorithm (Pavlovic et al., 1999), the approximate variational inference algorithm (Pavlovic and Rehg, 2000; Ghahramani and Hinton, 2000; Oh et al., 2005a), the MCMC algorithm Oh et al. (2005b), the generalised pseudo Bayesian (GPB) approximation algorithm (Bar-Shalom and Li, 1993) and the Gaussian sum filtering (GSF) algorithm (Barber, 2006, 2012). The GSF algorithm is derived for the DBN proposed in this study for the purpose of inferring behaviour and class.

The GPB and GSF algorithms may be generalized into a category of 'merging Gaussians'. Merging Gaussian algorithms assume the approach of representing a DBN's joint probability distribution with a mixture of Gaussians. At each time step, the Gaussian mixture is merged into a smaller mixture with a prespecified number of Gaussians (Ghahramani and Hinton, 2000). The algorithm is related to the particle filter algorithm where the particles are replaced with Gaussian distributions (Barber, 2012). Resampling is performed by collapsing the Gaussian mixture at each time step. These algorithms are extensions of the ADF algorithm where the ADF algorithm uses a single Gaussian distribution rather than a mixture of Gaussians (Barber, 2006). The mixture of Gaussians captures various hypotheses associated with different classes and switching states in the proposed model. This is not possible with a unimodal distribution such as that represented in the Kalman filter. The GSF algorithm is selected for application in this study as it uses the mixture of Gaussians rather than a single Gaussian. Furthermore, the GSF algorithm is proven to be adaptable to the proposed model.

Nonparametric algorithms for inference such as the particle filter and Markov chain Monte Carlo algorithms may also be used for inference. The Rao-Blackwellised particle filter has been applied for filtering in Dynamic Bayesian Networks (Doucet et al., 2000). The Sigma-Point Kalman Filter has been utilised for inference in the LDS (Van Der Merwe, 2004). Deterministic Annealing may be applied for inference in the SLDS (Ghahramani and Hinton, 2000). Such nonparametric methods are reserved for future research where a smoothing algorithm is to be applied to the model proposed in this study.

4. Proposed Model

The structure of the proposed DBN is illustrated in Figure 1. The model consists of several elements. A linear dynamical system (LDS) is included with the variables v_t and h_t. The LDS provides the means to model the continuous kinematic dynamics of the system. The visible state v_t describes the observations of the dynamical system. These are the represented as tracked locations of a target. The hidden state h_t describes the unobservable states of the linear dynamical system that generate the observations v_t. Unobserved states may include acceleration and velocity.

A switching state is provided through the variable s_t. The switching state s_t encodes the behaviour of the system. Through different switching states, the model can alternate between various LDS models. Each LDS model may describe some form of kinematic activity. The s_t, h_t and v_t nodes form an switching linear dynamical system (SLDS) or switching Kalman filter.

A set of nodes, a_t^n, $n = \{1, ..., N\}$ describe a set of N discrete contextual elements at each time step t. Contextual elements influence behaviour. The nodes associated with the contextual elements are considered to be observable. That is, at each timestep, the contextual information is assumed to be known. The contextual elements may be argued to operate as prior information under the Bayesian filtering framework as described in Stone et al. (1999) and Thrun et al. (2005). Finally, the model includes a static class node, c to describe the class of the target.

The LDS in the proposed model is represented by following state space equations (Barber, 2012; Murphy, 2002):

$$h_t = \mathbf{A}(s_t, c, \bar{a}_t)h_{t-1} + \eta_t^h(s_t, c, \bar{a}_t), \tag{1}$$

$$v_t = \mathbf{B}(s_t, c, \bar{a}_t)h_t + \eta_t^v(s_t, c, \bar{a}_t). \tag{2}$$

Variable $\bar{a}_t$ refers to all a_t^n, $n = \{1, ..., N\}$ discrete contextual element nodes at time t. Variables η_t^h and η_t^v describe the state noise process and the measurement noise respectively. These variables are assumed to be white Gaussian noise. The state transition matrix and measurement matrix are described by $\mathbf{A}$ and $\mathbf{B}$ respectively. Equation (1) describes the transition probability distribution $p(h_t|h_{t-1}, s_t, c, \bar{a}_{1:t})$. Equation (2) describes the emission probability distribution $p(v_t|h_t, s_t, c, \bar{a}_{1:t})$.

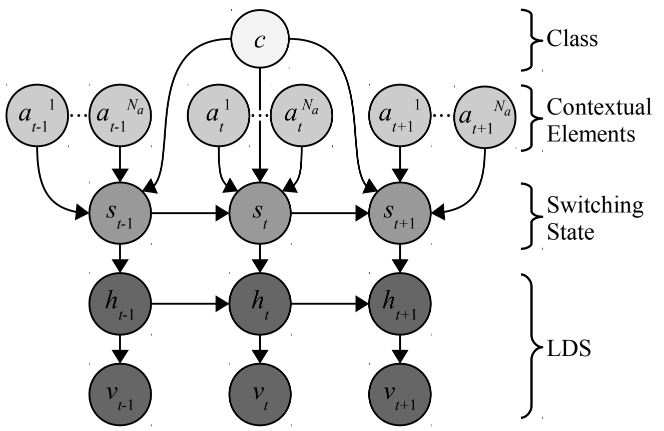


Figure 1: A novel model for context-based behavioural modelling and classification. The visible (observable) node, v_t and hidden node, h_t from a linear dynamical system (LDS). Together, the nodes v_t, h_t and s_t form a switching linear dynamical system (SLDS). The nodes (a_t^n where $n = \{1, ..., N\}$) are observable nodes describing contextual elements. The c node describes the class.

To consider the causal relationships between variables, the DBN links between variables may be examined. The central variable is the switching state variable s_t as it describes behaviour. This variable is influenced by the contextual elements and the class. The behaviour, in turn influences the kinematic dynamics of the system through the LDS.

The inferred class is informed by the parameters that result in the best modelling of the data. The linear dynamic system parameters in the proposed model are conditional on the switching state s_t, the contextual elements $\bar{a}_t$ and the class c. This implies that various linear dynamic system parameters may be defined for different behaviours. The parameters may be defined according to combinations of switching states, contextual elements and classes.

4.1. Variable Definition and Designation

The observed variables include the contextual elements a_t^n, $n = \{1, ..., N\}$ and the visible state variable v_t. In a DBN, a continuous variable may have discrete variable parents. A discrete variable however, should generally not have continuous variable parents (Pavlovic, 1999). The contextual elements are required to be discrete as they are parents of the discrete switching state variable s_t. Continuous contextual elements may be discretised using methods such as partitioning or fuzzy logic such as described in Yang et al. (2010). Increasing the number of variables and variable states increases the complexity of the system. DBNs (and graphical models in general) are susceptible to the curse of dimensionality (Bengio and Bengio, 2000). It is recommended to keep the number of variables or DBN nodes to a minimum. Variables should be combined where possible. For example in this study, variables that include ocean conditions, weather conditions, time of day and the season of year are merged into a single 'sailing conditions' variable.

The visible state variable v_t describes a continuous variable that is a measurement of the LDS. In this study, the sequential locations of a vessel trajectory is provided for this variable. Additional measurements such as velocity and acceleration could also be provided. However, these entities may also be inferred in the hidden variable, h_t.

The LDS parameters could be estimated using algorithms such as the EM algorithm as described in Ghahramani and Hinton (1996). In this study the LDS parameters were not required to be estimated. See Section 6.1 for further discussion.

5. Classification via Inference

Classification in the proposed model is performed by inferring the class variable c. Inferring the class variable given the observed data is equivalent to evaluating

$$p(c|v_{1:T}, \bar{a}_{1:T}). \tag{3}$$

The proposed model includes an SLDS within its DBN structure. Exact inference on an SLDS is NP-hard (Murphy, 2002; Barber, 2012). The number of components increases by a factor of the number of switching states S as time progresses. Similarly, exact inference in the proposed model is intractable. The GSF algorithm has been proposed to perform approximate inference on the SLDS (Barber, 2006, 2012).

A detailed derivation of the GSF algorithm for the proposed model is presented in Appendix A. The purpose of the algorithm is to infer the latent variables given the observed variables in the proposed DBN. The latent variables include s_t, h_t and c. The observed variables include $\bar{a}_{1:t}$ and $v_{1:t}$. The probability distribution of the latent variables given the observed variables is expressed as $p(s_t, h_t, c|v_{1:t}, \bar{a}_{1:t})$. This equation can be factorised as follows (see (A.5) in Appendix A)

$$\begin{aligned} p(s_t,h_t,c|v_{1:t}, \bar{a}_{1:t}) &= \\ &p(h_t|s_t, c, v_{1:t}, \bar{a}_{1:t}) \cdot p(s_t|c, v_{1:t}, \bar{a}_{1:t}) \cdot p(c|v_{1:t}, \bar{a}_{1:t}). \end{aligned} \tag{4}$$

The first factor is a continuous conditional distribution that describes the LDS. The second factor is a discrete conditional distribution that describes the switching state probability. The third factor is a discrete conditional distribution that describes the class probability. Each of these three factors is determined in the GSF algorithm.

The variables h_t and v_t form an LDS. As indicated in (1) and (2), the LDS depends on s_t, c and $\bar{a}_t$. A single Gaussian is not sufficient to represent the conditional distribution representing the LDS. This conditional distribution is given by the first factor in (4). The conditional distribution is approximated by a mixture of I Gaussians as follows

$$\begin{aligned} &p(h_t|s_t, c, v_{1:t}, \bar{a}_{1:t}) \approx \\ &\quad \sum_{i_t=1}^{I} q(h_t|i_t, s_t, c, v_{1:t}, \bar{a}_{1:t}) \cdot q(i_t|s_t, c, v_{1:t}, \bar{a}_{1:t}), \end{aligned} \tag{5}$$

where i_t indicates the i^{th} mixture component at time t. A mixture component is parametrised by mean $f(i_t, s_t, c)$ and covariance $F(i_t, s_t, c)$. In the forward propagation of the LDS, the I mixture of Gaussians is expanded to an $I \times S$ mixture of Gaussians for each class. The mean and covariance parameters associated with the forward propagation of the LDS are determined as (see (A.10) through (A.14) in Appendix A)

$$\Sigma_{hh} = \mathbf{A}(s_t, c, \bar{a}_t)F(i_{t-1}, s_{t-1}, c) \cdot \mathbf{A}^T(s_t, c, \bar{a}_t) + \Sigma_h(s_t, c, \bar{a}_t), \tag{6}$$

$$\Sigma_{vv} = \mathbf{B}(s_t, c, \bar{a}_t)\Sigma_{hh}{}^T(s_t, c, \bar{a}_t) + \Sigma_v(s_t, c, \bar{a}_t), \tag{7}$$

$$\Sigma_{vh} = \Sigma_{hv}^T = \mathbf{B}(s_t, c, \bar{a}_t)\Sigma_{hh}, \tag{8}$$

$$\mu_v = \mathbf{B}(s_t, c, \bar{a}_t)\mathbf{A}(s_t, c, \bar{a}_t)f(i_{t-1}, s_{t-1}, c), \tag{9}$$

$$\mu_h = \mathbf{A}(s_t, c, \bar{a}_t)f(i_{t-1}, s_{t-1}, c). \tag{10}$$

These parameters may be used to determine the a posteriori distribution parameters given by (see (A.16) and (A.17) in Appendix A)

$$\mu_{h|v} = \mu_h + \Sigma_{hv}\Sigma_{vv}^{-1}(v_t - \mu_v), \tag{11}$$

$$\Sigma_{h|v} = \Sigma_{hh} - \Sigma_{hv}\Sigma_{vv}^{-1}\Sigma_{vh}. \tag{12}$$

It may be noted that equations (6) through (12) are closely related to the equations for the Kalman filter.

The mixture weight associated with the i^{th} mixture component is determined as (see (A.20) in Appendix A)

$$\begin{aligned} q(i_{t-1}, s_{t-1}|s_t, c, v_{1:t}, \bar{a}_{1:t}) \propto \\ & q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ & \cdot q(s_t|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ & \cdot q(i_{t-1}|s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ & \cdot p(s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t-1}). \end{aligned} \tag{13}$$

The first factor is a Gaussian with mean μ_v and variance Σ_{vv}. The second factor describes the predefined switching state transition distribution. The third factor contains the weights of the Gaussian mixture determined in the preceding step of the algorithm. The fourth factor is the switching state conditional distribution determined in the preceding step of the algorithm.

Once the parameters of the Gaussian mixture model are determined, the $I \times S$ component Gaussian mixture model is collapsed back to a I component Gaussian mixture model for each class. To collapse the distribution, the mixture components with the lowest weights are merged.

The switching state conditional distribution is given by the second factor on the right hand side of (4). This conditional distribution, denoted by α_t, is determined as follows (see (A.25) in Appendix A)

$$\begin{aligned} \alpha_t = p(s_t|c,v_{1:t}, \bar{a}_{1:t}) \propto \\ \sum_{i_{t-1}, s_{t-1}} & [q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ & \cdot q(s_t|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ & \cdot q(i_{t-1}|s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ & \cdot p(s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t-1})]. \end{aligned} \tag{14}$$

Each of these factors is determined as in (13). The switching state conditional distribution should be normalised after each iteration. The conditional distribution for the class is given by the third factor on the right hand side of (4). This conditional distribution, denoted by β_t, is determined as follows (see (A.28) in Appendix A)

$$\begin{aligned} \beta_t = p(c|v_{1:t},\bar{a}_{1:t}) \propto \\ \sum_{i_{t-1}, s_{t-1}, s_t} & [q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ & \cdot q(s_t|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ & \cdot q(i_{t-1}|s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ & \cdot p(s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ & \cdot p(c|v_{1:t-1}, \bar{a}_{1:t-1})]. \end{aligned} \tag{15}$$

Each of these factors is determined as in (14). The last factor is the a priori for the class and is computed in the previous time step. The class conditional distribution should be normalised after each iteration. The GSF algorithm is presented in Algorithm 1.

Algorithm 1 Gaussian sum filtering (GSF) algorithm for the proposed inference DBN model.

Require: the sample data, the contextual element data, the linear dynamic system parameters and the switching transition probabilities.

At $t = 1$, initialise α_1, β_1 and the Gaussian mixture model.
for $t = 2 \rightarrow T$ **do**
  {Recursion for inference at each time step:}
  for $c = 1 \rightarrow C$ **do**
    for $s_t = 1 \rightarrow S$ **do**
      {Recursion for Gaussian mixture components:}
      for $i_{t-1} = 1 \rightarrow I$ and $s_{t-1} = 1 \rightarrow S$ **do**
        Calculate Σ_{hh}, Σ_{vv}, Σ_{vh}, Σ_{hv}, μ_v and μ_h using (6) through (10).
        Calculate $\mu_{h|v}$ and $\Sigma_{h|v}$ using (11) and (12) respectively.
        Set $q(v_t|i_{t-1}, s_{t-1}, s_t, c, v_{1:t-1}, \bar{a}_{1:t})$ in (13) as a Gaussian distribution parametrised by μ_v and Σ_{vv}.
        Calculate the unnormalised mixture weights given by (13).
      end for
      Collapse the $(I \times S)$ component Gaussian mixture to an (I) component Gaussian mixture.
      Calculate the unnormalised α_t in (14) by marginalizing over i_{t-1} and s_{t-1}.
    end for
    Normalise α_t.
    Calculate the unnormalised β_t in (15) by marginalizing over i_{t-1}, s_{t-1} and s_t.
  end for
  Normalise β_t.
end for

6. Maritime Pirate Detection Application

Large vessels are required by law to carry automatic identification system (AIS) transmitters (Balci and Pegg, 2006).

Small vessels such as fishing vessels are not required to carry AIS transmitters. Evidence exists that indicates that some pirates use AIS data transmitted from vessels for target selection (Kraska and Wilson, 2008). For this reason, large vessels may turn off the AIS transmitters when entering areas where pirates are known to operate. If no AIS data is transmitted from vessels, vessels may not easily be identified. Hence, there are situations where vessel behaviour needs to be identified based on movement and context alone. Sources which may provide movement information are satellite synthetic aperture radar (SAR), optical satellite sensors, coastal radar and on board vessel radar systems. Contextual data may be obtained from sources such as weather systems, geographical maps, pirate attack reports and known shipping routes and corridors.

Real-world data of illegal maritime activities is scarce (Jakob et al., 2011; Kazemi et al., 2013). The proposed model and algorithm are thus applied to generated data. The data is generated from a simulation of pirate, transport and fishing vessels in a maritime environment. The simulated data is produced from a generative statistical model that is discussed in (Dabrowski and de Villiers, 2015). The simulation provides tracked coordinates of maritime vessels over a specified region. Sailing conditions over time are provided. An indicator variable is provided that indicates if a simulated vessel is travelling in a high risk pirate zone or not.

The proposed model may be considered to be a simplified variation of the model discussed in (Dabrowski and de Villiers, 2015). The simplification is required to reduce the complexity of the model. The complexity is reduced for the purpose of applying the GSF algorithm. The GSF algorithm may be intractable or unreliable when applied to highly complex models.

6.1. State Space Representation

Pirate vessels generally travel at speeds between 20 and 25 knots (ICC-IMB, 2012). Transport vessels are advised to travel at transit speeds of 10, 12, 14, 16 or 18 knots (ICC-IMB, 2012). Vessel speeds provide a potential feature for classification. The state space model is formulated to exploit this by parametrising the state matrix $\mathbf{A}$ with the vessel speed. Consider the following representation of the state vector as described in (1):

$$\begin{bmatrix} x_t \\ y_t \\ \Delta r_t \cos(\theta_t) \\ \Delta r_t \sin(\theta_t) \end{bmatrix} = \mathbf{A} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \Delta r_{t-1} \cos(\theta_{t-1}) \\ \Delta r_{t-1} \sin(\theta_{t-1}) \end{bmatrix} + \eta_t^h. \tag{16}$$

The first two variables in the state vector, x_t and y_t, describe the position of the vessel at time t. The third and fourth variables in the state vector describe the vessel velocity in polar coordinates. The angle of the velocity at time t is described by θ_t. The speed of the vessel at time t is described by Δr_t. A constant velocity model for the vessels is assumed. For constant velocity, Δr_t is a constant value over time. Mathematically, this element can thus be placed in the state transition matrix $\mathbf{A}$. With $\Delta r_t = \Delta r$, the state transition matrix is defined as follows:

Table 1: Relative speeds of vessels for the set of switching states.

	Pirate	Transport	Fishing
Sailing	100%	75%	100%
Drifting/Fishing	25%	-	25%
Anchored	0%	0%	0%

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta r & 0 \\ 0 & 1 & 0 & \Delta r \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{17}$$

This representation deviates from the traditional discrete white noise constant velocity model as described in Blackman and Popoli (1999). In the traditional model, the velocity is inferred and the sampling period is contained in $\mathbf{A}$. The proposed representation in (17) allows for $\mathbf{A}$ to be parametrised by the vessel speed. Furthermore, as indicated in (1), the state transition matrix, $\mathbf{A}$ is conditioned on s_t, c and $\bar{a}_t$. Separate state matrices may be defined for each combination of s_t, c and $\bar{a}_t$. For each of these combinations, $\mathbf{A}$ may be parametrised by a predefined a priori speed (Δr). This provides a means to distinguish vessels based on their speed using $\mathbf{A}$.

In the simulation, the pirate and fishing vessels are set to travel at an average of 25% faster than the transport vessels. When the pirates drift and fishing vessels fish, their drift speed is 25% of their sailing speed. All vessels are stationary in the anchored state. The relative speeds of vessels for the set of switching states are presented in Table 1.

6.2. Contextual Elements

Two contextual elements are provided in the vessel simulations. The first contextual element describes the sailing conditions. The sailing conditions at a particular time are described as $a_t^1 \in$ {poor, adequate, favourable}. These conditions are a result of an amalgamation of various contextual elements. The amalgamation of the contextual elements into a single variable decreases the complexity of the system. The sailing conditions describe ocean conditions, weather conditions, time of day and the season of year (Dabrowski and de Villiers, 2015). Table 2 provides an indication of how the states of a_t^1 may be selected for pirate and fishing vessels.

Evidence indicates that pirates tend to favour particular sailing conditions (Dabrowski and de Villiers, 2015; ICC-IMB, 2012). The switching state and LDS are thus influenced by the sailing conditions. Sailing conditions affect the vessel speed. The simulated vessels speed decrease by 20% in adequate sailing conditions and by 50% in poor sailing conditions. The relative speeds of vessels for the set of sailing conditions are presented in Table 3.

The second contextual element provides an indication of whether the vessel is in a high risk pirate zone or not. Pirate zones have been determined from historical attack data (ICC-IMB, 2012; Dabrowski and de Villiers, 2015).

Table 2: Examples for defining the sailing conditions contextual element for pirate and fishing vessels.

Sailing conditions	Ocean conditions	Weather conditions	Time	Season
poor	high wave height	high wind speed, heavy rain	daytime	monsoon
adequate	mild wave height	mild wind speed, little/no rain	dusk, dawn or dark	non-monsoon
favourable	low wave height	low wind speed, no rain	dusk, dawn or dark	non-monsoon

Table 3: Relative speeds of vessels for the set of sailing conditions.

	Pirate	Transport	Fishing
Favourable	100%	75%	100%
Adequate	80%	60%	80%
Poor	50%	37.5%	50%

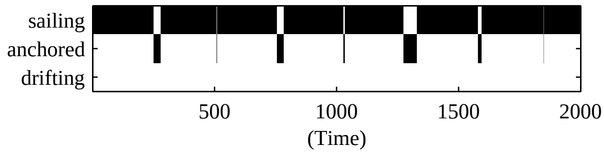
Pirate vessels typically resemble fishing vessels (Heger et al., 2009). In the simulation, fishing behaviour resembles the drifting behaviour of pirates (Dabrowski and de Villiers, 2015). This causes difficulty in distinguishing pirate vessels from fishing vessels. Fishing vessels are known to be targets of pirates (ICC-IMB, 2012). It is assumed that fishing vessels will thus avoid pirate zones. It follows that, vessels exhibiting drifting or fishing behaviour in pirate zones are likely to be pirate vessels. Vessels exhibiting drifting or fishing behaviour outside of pirate zones are likely to be fishing vessels. The result of this assumption is that the probability of a false positive classification of a fishing vessel as a pirate vessel increases. Consequences of a false positive classification are less severe than the false negative classification of a pirate vessel as a fishing vessel. That is, a ship captain may prefer to be warned that a fishing vessel may be a threat rather than no warning being issued when a pirate vessel is misclassified as a fishing vessel.

6.3. Switching States

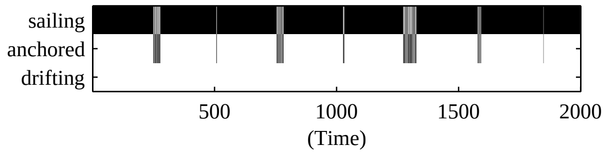
The switching states for the vessels are described as $s_t \in \{\text{sailing, anchored, drifting}\}$. The transition matrices for the vessel types provide a distinction between behaviour. Transport vessels will avoid the drifting state. The transport vessel will alternate between the sailing and the anchored state. All vessels will prefer to avoid sailing in poor sailing conditions. Pirate and fishing vessels are generally smaller than transport vessels. They are thus more susceptible to poor conditions. Pirate and fishing vessels will thus avoid poor conditions more so than transport vessels.

Pirate vessels will transition from anchored to sailing states. From sailing states, the pirate vessel may enter a drifting state if it is in a pirate zone. In preferred sailing conditions and in pirate zones, the pirate vessel is more likely to enter a drifting state.

Fishing vessels will transition from anchored to sailing states. From sailing states, the fishing vessel may enter a fishing state. Fishing behaviour is modelled using the drifting state. From the fishing state, fishing vessels enter a sailing state as to return to shore.

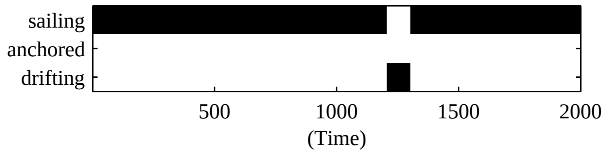


(a) True switching states.

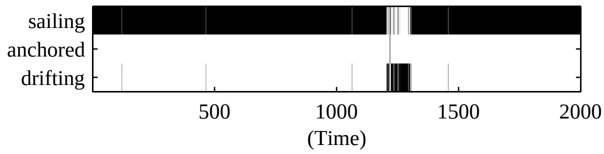


(b) Inferred switching states.

Figure 2: The inferred switching states for a transport vessel.



(a) True switching states.



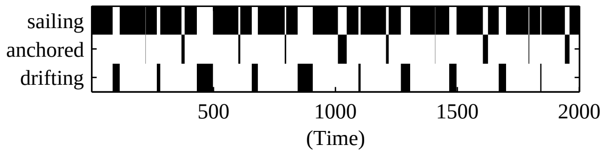
(b) Inferred switching states.

Figure 3: The inferred switching states for a pirate vessel.

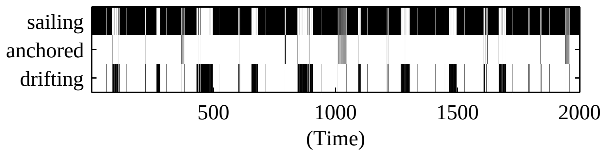
6.4. Inference Results

A simulation with pirate, fishing and transport vessels was evaluated using the proposed model. The results displayed in Figure 2 describe the inferred switching states for a transport vessel. The results illustrate the transport vessel alternating between sailing and anchored states. Noise in the sensor variables results in apparent motion during the anchored states. This results in a degree of uncertainty when in the anchored state. This is apparent for all vessel types.

The results illustrated in Figure 3 describe the inferred switching states for a pirate vessel. The proposed model does not infer the drifting state continuously. This is due to the non-linear random motion of the vessel as it drifts. The drifting and fishing motion is modelled as a random process (Dabrowski and de Villiers, 2015). The non-linear random motion of the vessel is not tracked easily as the proposed DBN models linear motion.



(a) True switching states.



(b) Inferred switching states.

Figure 4: The inferred switching states for a fishing vessel.

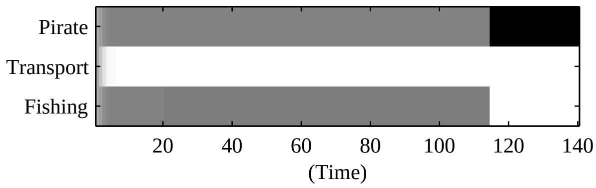


Figure 5: The inferred class for a pirate vessel for the first 140 time steps.

The results illustrated in Figure 4 describe the inferred switching states for a fishing vessel. This example illustrates a fishing vessel alternating between sailing, anchored and drifting (fishing) states.

The results displayed in Figure 5 describe the inferred class for a pirate vessel for the first 140 time steps. In the first time step the model infers equiprobable classes. As more information is provided over time, the model transitions to a more distinctive decision. The model infers that the pirate and fishing vessel classes are equiprobable up to time step $t = 115$. The vessel transitions into a pirate zone at $t = 115$. The model infers the pirate vessel class for the remaining time step samples.

The results displayed in Figure 6 describe the inferred class for a transport vessel for the first 40 time steps. The model initially tends towards inferring the fishing vessel class. As the vessel moves into a pirate zone, the fishing vessel class becomes less probable. In the pirate zone, the model considers the pirate and transport class. The transport class is inferred as the vessel does not enter a drifting state and it travels at a speed that is slower than expected for pirate vessels. The class variable transitions to the correct inference as the vessel sails through the pirate zone. The algorithm continues to infer the transport class for the remaining time step samples.

The results displayed in Figure 7 describe the inferred class for a fishing vessel for the first 90 time steps. Initially, the model is not able to distinguish between a pirate and fishing vessel due to similar dynamics. The model transitions to infer the correct fishing class at time $t = 62$. At this time, the model infers that that the vessel enters a drifting (fishing) state outside of a pirate zone.

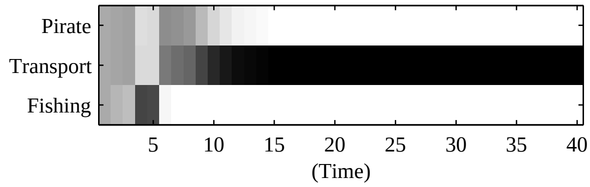


Figure 6: The inferred class for a transport vessel for the first 40 time steps.

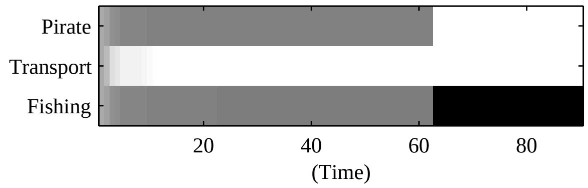


Figure 7: The inferred class for a fishing vessel for the first 90 time steps.

Table 4: Second test: relative speeds of vessels for the set of switching states.

	Pirate	Transport	Fishing
Sailing	100%	90%	100%
Drifting/Fishing	25%	-	25%
Anchored	0%	0%	0%

Table 5: Second test: relative speeds of vessels for the set of sailing conditions.

	Pirate	Transport	Fishing
Poor	100%	90%	100%
Adequate	80%	72%	80%
Favourable	50%	45%	50%

6.5. *Results for Varying the Simulation Parameters*

In further investigation, the model was tested with varying speed ratios between pirate, fishing and transport vessels. Pirate and fishing vessels travel with equal speed. In each test their speed was decreased to a value closer to that of transport vessels. In the first test, pirate and fishing vessels travel at speeds 25% faster than transport vessels as described in Table 1. In the second test, pirate and fishing vessels travel at speeds 10% faster than transport vessels. The relative speeds of the vessels for the second test is presented in Table 4 and Table 5. In the third test, all vessels travel at the same speed.

For each test, a set of 1000 vessels were simulated. The 1000 vessel simulations were generated by 50 situation simulations consisting of 20 vessels each. Each situation simulation consisted of 2000 time steps. The proposed model was utilised to classify each vessel. The classification of a vessel is defined as a thresholding function on the expected value of the class variable over the simulation.

The results of the tests are provided in Table 6, Table 7, Table 8 and Table 9. The results presented in the tables describe the recall, precision, F-score and the accuracy of the method respectively. The results are determined using the confusion matrix. The confusion matrix $\mathbf{\Lambda} = [\Lambda(j,k)]$ is defined such that an element, $\Lambda(j,k)$, contains the number of vessels whose true class label is j and are classified to the class with label k

Table 6: Simulated vessel classification recall with varying speed ratios.

Speed Ratio	R_{pirate}	$R_{transport}$	$R_{fishing}$
1.25	100%	99.2%	100%
1.10	100%	94.7%	100%
1.00	70.0%	67.3%	100%

Table 7: Simulated vessel classification precision with varying speed ratios.

Speed Ratio	P_{pirate}	$P_{transport}$	$P_{fishing}$
1.25	98.4%	100%	100%
1.10	90.3%	100%	100%
1.00	47.8%	84%	100%

Table 8: Simulated vessel classification F-score with varying speed ratios.

Speed Ratio	F_{pirate}	$F_{transport}$	$F_{fishing}$
1.25	99.2%	100%	100%
1.10	94.9%	100%	100%
1.00	56.8%	91.3%	100%

Table 9: Overall classification accuracy with varying speed ratios.

Speed Ratio	Accuracy
1.25	99.5%
1.10	96.8%
1.00	71.6%

(Theodoridis and Koutroumbas, 2009). The rows describe the true class. The columns describe the inferred class. Pirate vessels are associated with the label index 1. Transport vessels are associated with the label index 2. Fishing vessels are associated with the label index 3. The confusion matrix for the simulation with the speed ratio of 1.25 is given as follows:

$$\mathbf{\Lambda} = \begin{bmatrix} 314 & 0 & 0 \\ 5 & 595 & 0 \\ 0 & 0 & 86 \end{bmatrix}. \tag{18}$$

In this result, 314 vessels were correctly classified as pirate vessels. 595 vessels were correctly classified as transport vessels. 5 transport vessels were incorrectly classified as pirate vessels. 86 vessels were correctly classified as transport vessels.

The confusion matrix for the simulation with the speed ratio of 1.10 is given as follows:

$$\mathbf{\Lambda} = \begin{bmatrix} 297 & 0 & 0 \\ 32 & 575 & 0 \\ 0 & 0 & 96 \end{bmatrix}. \tag{19}$$

The confusion matrix for the simulation with the speed ratio of 1.00 is given as follows:

$$\mathbf{\Lambda} = \begin{bmatrix} 187 & 80 & 0 \\ 204 & 420 & 0 \\ 0 & 0 & 109 \end{bmatrix}. \tag{20}$$

It may be noted that the ratios between the numbers of vessels in each class may not necessarily reflect reality. The classification results indicate that the classification of fishing vessels is trivial. The classification of pirate and transport vessels is however not as trivial as that for fishing vessels. The ratios are selected to emphasise the results for the classification of pirate and transport vessels.

The recall presented in Table 6 is calculated from the confusion matrix for class j as follows (Theodoridis and Koutroumbas, 2009; Murphy, 2012):

$$R_j = \frac{\Lambda(j, j)}{\sum_{k=1}^{C} \Lambda(j, k)}. \tag{21}$$

Recall may be described as the probability that a vessel is classified with the class label j, given that the true label of the vessel is j. Recall may also be interpreted as a sensitivity measure.

The recall for pirate vessels remains relatively high for all speed ratios. A 100% value for the recall implies that no pirate vessels were incorrectly classified. In hypothesis testing, the incorrect classification of the pirate vessel is known as the false negative hypothesis (Murphy, 2012). This is the most severe of possible hypothesis testing errors. The recall for fishing vessels remains at 100% for all speed ratios. The recall for transport vessels is 99.2%. This is due to 5 transport vessels being misclassified as pirate vessels. In hypothesis testing this error is known as a false positive error.

The precision presented in Table 7 is calculated from the confusion matrix for class j as follows (Theodoridis and Koutroumbas, 2009; Murphy, 2012):

$$P_j = \frac{\Lambda(j, j)}{\sum_{k=1}^{C} \Lambda(k, j)}. \tag{22}$$

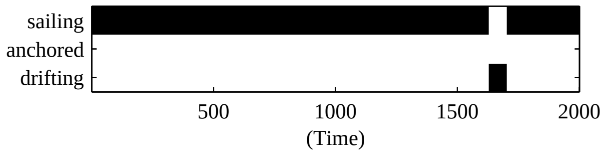
Precision may be described as the probability that the true label of a vessel is j, given that the vessel is classified with the class label j. Precision may also be interpreted as a confidence measure.

The precision values for transport vessels are higher than the respective recall values. Higher values for the transport class precision indicate that fewer vessels were incorrectly classified as transport vessels. The precision values for the pirate vessels are lower than their corresponding recall values. This is due to the incorrect classification of other vessels as pirate vessels. The fishing vessel precision values remain at 100%.

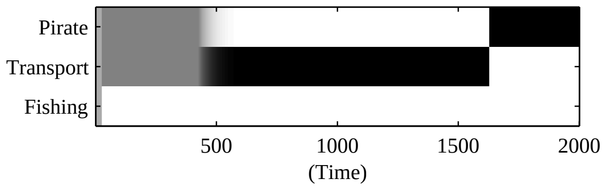
The F-score presented in Table 8 describes the harmonic mean between the recall and precision. The F-score is given as follows (Murphy, 2012):

$$F_j = \frac{2R_jP_j}{R_j + P_j}. \tag{23}$$

The overall accuracy presented in Table 9 is calculated from the confusion matrix as follows (Theodoridis and Koutroumbas,



(a) True switching states.



(b) Inferred class.

Figure 8: Results for a pirate vessel with its speed equal to that of transport and fishing vessels. The vessel is initially classified as a transport vessel. As soon as the vessel starts to drift, the class is changed to a the correct pirate vessel class.

2009):

$$Acc = \frac{\sum_{j=1}^{C} \Lambda(j, j)}{\sum_{j=1}^{C} \sum_{k=1}^{C} \Lambda(j, k)}. \qquad (24)$$

The results presented in Table 8 and Table 9 describe the accuracy of the model. These results indicate that the accuracy of the algorithm increases with increasing pirate and fishing vessel speed. By increasing the pirate and fishing vessel speeds, the discriminatory value of the speed feature increases. The speed feature has no discriminatory value in the case of equal pirate, fishing and transport vessel speeds. The primary remaining discriminating feature is the switching state. The switching state defines that the class of a vessel sailing outside of a pirate zone is ambiguous. The class of a vessel sailing in a pirate zone is more likely to be a transport vessel. The class of a drifting vessel in a pirate zone is more likely to be a pirate vessel. The class of a vessel drifting outside of a pirate zone is likely to be a fishing vessel. The model performs well when this reasoning is considered. An example is provided as follows:

The class and switching states for a pirate vessel are illustrated in Figure 8. For time steps $t = 1$ to $t = 425$ the vessel sails in a non-pirate zone. Initially, the model considers all vessel classes equiprobable. The fishing vessel is expected to enter the sailing state quickly as the fishing zone is near the shore. As the vessel continues to sail, the fishing class becomes less likely. The model thus continues to classify the vessel as either a transport or a pirate vessel. As the vessel enters the pirate zone, the model transitions to the transport vessel class. The transport class is more likely as a transport vessel is more likely to continue sailing through a pirate zone. A pirate vessel is expected to enter a drifting state in a pirate zone. At time step $t = 1628$ the pirate vessel transitions to a drifting state. At this time the model changes the class from the transport class to the pirate class. The classification of behaviour is thus successful.

The time required to classify a vessel depends on the vessel speed and the switching state. In general, if the discriminatory value of the features is higher, the classification time is shorter.

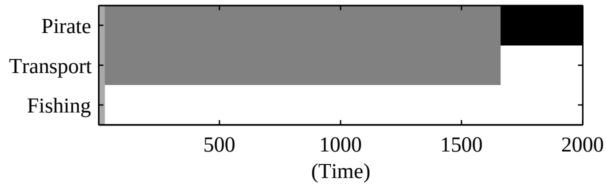


Figure 9: The state transition probabilities between the sailing and anchored states are equal between the transport and pirate vessel classes. With reference to Figure 8, the classifier does not distinguish between the transport and pirate vessel classes until the vessel enters the drifting state.

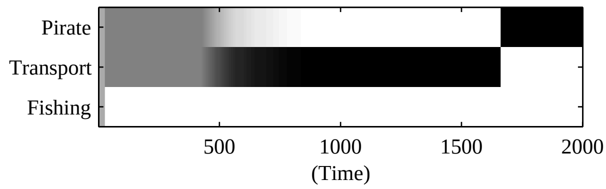
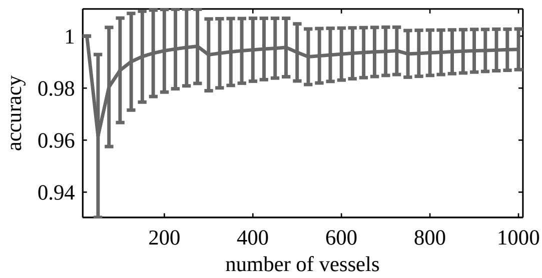


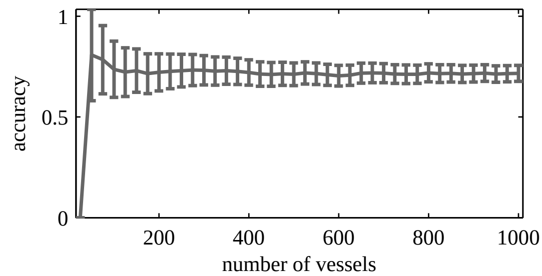
Figure 10: The state transition probabilities between the sailing and anchored states differ by a small 1% between the transport and pirate vessel classes. With reference to Figure 8, the classifier takes longer to classify the vessel as a transport vessel.

That is, if the speeds differ significantly, the model will classify a vessel in a shorter period of time. If the transition matrices for the switching states differ significantly, the model will perform classification in a shorter period of time. The results illustrated in Figure 8 provide an indication of the effect of time on classification. Once the vessel enters the pirate zone, it takes 148 time steps to reach a 99% certainty of the transport class. If the transition probabilities between the sailing and anchored states are set equal between the transport and pirate classes, the classifier continues consider the classes to be equiprobable. This is illustrated in Figure 9. If these transition probabilities only differ by 1%, the discriminatory value is low and more time (414 time steps) is required to reach a 99% certainty of the transport class. This is illustrated in Figure 10. When the vessel enters the drifting state at time step $t = 1628$, the classification of the pirate class is immediate. The transition matrix for a transport vessel specifies that the vessel should never enter a drifting state. Only the pirate vessel is expected to enter the drifting state in a pirate zone.

The error bar plots illustrated in Figure 11 provide an indication of the statistical significance of the sample size. The accuracy per the number of vessels considered is described by the trend lines. The standard deviation of the accuracy per number of vessels is described by the error bars. The trend line converges to a particular value and the standard deviation decreases as the number of vessels increase. The law of large numbers indicates that the average accuracy will tend towards the expected accuracy as the number of vessels increase (Grimmett and Stirzaker, 2001). The converging trend lines indicate that the number of vessels is sufficient to provide a reasonable approximation of the expected accuracy. This result demonstrates some form of statistical significance regarding the number of selected simulated vessels.



(a) Accuracy error bar plot for a speed ratio of 1.25.



(b) Accuracy error bar plot for a speed ratio of 1.00.

Figure 11: Error bar plot for the accuracy results for speed ratios of 1.25 and 1.00 between the pirate/fishing vessels and the transport vessels. The trend lines describe the accuracy of per number of vessels considered. The error bars describe the standard deviation of the accuracy per number of vessels.

6.6. *Parameter Sensitivity*

The sensitivity of the switching state parameters is analysed by varying the pirate vessel switching state transition probabilities. The dataset that consists of vessels travelling at equal speeds is utilised. Only the switching state transition probabilities influence the classification outcome this dataset. The results are presented in Figure 12. In Figure 12a, the true switching state is plotted for the pirate vessel. The vessel enters a pirate zone at time $t = 426$. The vessel enters the drifting state at time step $t = 1628$.

In Figure 12b, the switching state transition probabilities for the pirate vessel class are set equal to that for the transport vessel class. In this configuration, the pirate vessel has a zero probability for remaining in a drifting state. As the vessel begins its journey, it is equally classified as a pirate, transport and fishing vessel. The fishing vessel is expected to enter the fishing state quickly. The fishing vessel class becomes less probable as the vessel continues to sail. The vessel is classified as either a pirate or a transport vessel in the remaining time samples. The switching state transition probabilities for a pirate and transport vessel are equal. The result is thus as expected.

In Figure 12c, the state transition probability for remaining in the drifting state for a pirate vessel is set at a non-zero value. All other switching state transition probabilities are set equal to that of the transport vessel classes. The results are similar to Figure 12b. However, as the vessel enters the drifting state, it is immediately classified as a pirate vessel.

In Figure 12d, the probability of remaining in the sailing state for the pirate vessel is increased by 10% when the vessel is inside a pirate zone. When outside of a pirate zone, the state transition probabilities of the pirate and transport vessels are set equal. As the vessel enters and sails through the pirate zone, it is classified as a pirate vessel. This is due to the higher probability of remaining in the *sailing* state for the pirate vessel. For comparison, the pirate vessel is set to have only a 1% higher probability of remaining in the sailing state. The result is illustrated in Figure 12e. In this case, the model seems to take longer to make a decision. That is, it requires more data samples to make a more certain decision. By varying the transition probability the confidence of the model is thus adjusted.

In Figure 12f and Figure 12g the *transport* vessel class is set to have a 10% and 1% higher probability of remaining in the sailing state respectively. As expected, the transport class is inferred as the vessel sails through the pirate zone. However, the pirate class is inferred as the vessel enters the drifting state. The transport vessel has a zero probability for remaining in the drifting state.

6.7. *Model Effectiveness*

The effectiveness of an information fusion system may be described according to quality, robustness and information gain (Blasch et al., 2010). Quality measures the performance of the model. Robustness measures the consistency of the model. Information gain measures the ability of the model to provide improvement.

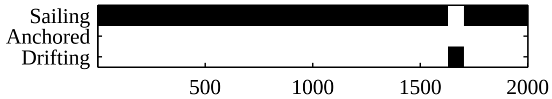
The quality of the model may be described using the precision, recall and accuracy results described in section 6.5. The accuracy of the model in classification provides an indication of the quality of the model. High classification accuracy indicates high quality. The model demonstrated the ability to perform classification with a high level of accuracy.

The robustness of the model can be described by its ability to handle noise, its consistency and its generated errors. Noise has been introduced into the motion models of the simulated vessels. This may cause incorrect inference of the state variable. Examples of noise in the model are discussed in section 6.4. The model demonstrated to have the ability to handle noise in the anchored state. The model demonstrated the ability to cope with a non-linear model for the drifting state 6.4.

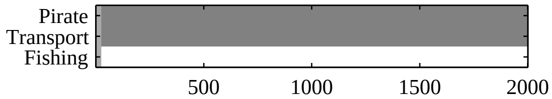
The robustness of the model may be described by the error types generated by the model. The most severe error type is the false negative error. The false negative error occurs when a pirate vessel is incorrectly classified as another vessel class. The adverse consequences of false negative errors are high. The recall values for the pirate vessels provide an indication of the false negative error rate. The recall values are provided in table 6. The high recall values indicate low false negative errors. This indicates a higher level of robustness and quality of the model.

The model used for data generation (Dabrowski and de Villiers, 2015) is not identical to the model proposed in this work. The extent to which the method is able to perform classification, indicates some form of robustness to model mismatch.

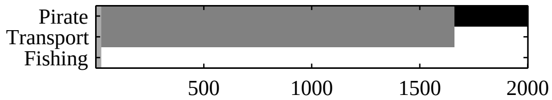
A possible weakness of the model is that it can tend to change the inferred class even if it indicated a high probability of a different class at a previous time. In this circumstance the model is over confident. This property can be managed by adjusting



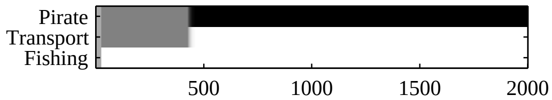
(a) True switching states.



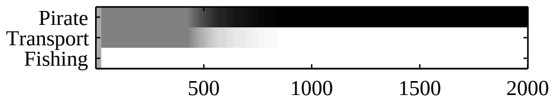
(b) Inferred class: Pirate and transport vessel state transition matrix equal.



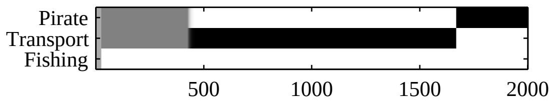
(c) Inferred class: Pirate vessel with non-zero probability for entering drifting state



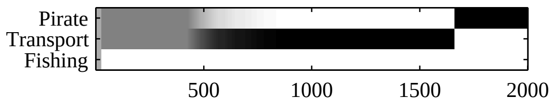
(d) Inferred class: Pirate vessel with 10% higher probability of remaining in the sailing state.



(e) Inferred class: Pirate vessel with 1% higher probability of remaining in the sailing state.



(f) Inferred class: Pirate vessel with 10% lower probability of remaining in the sailing state.



(g) Inferred class: Pirate vessel with 1% lower probability of remaining in the sailing state.

Figure 12: Parameter sensitivity plots.

the transition probabilities for the switching state variable. By reducing the discriminatory value of the transition probability between classes, the classification sensitivity may be decreased as illustrated in Section 6.6.

Information gain may be utilised for the purpose of evaluating an information data source (Blasch et al., 2010; Das, 2008). Information gain may provide a measure of the contribution of a particular data source. The measure may be represented by the Kullback-Liebler divergence or a variant thereof. In the proposed model, the information gain may be utilised to select external factors. External factors that provide higher information gain are preferred. Information gain was not considered in this study as only two contextual elements were considered.

7. Discussion

7.1. Model Strengths and Weaknesses

The proposed model is highly flexible in the sense that it can be applied to many different applications. In general, the model can be applied to any dynamic system that can be represented by a state space model. In this study, the nodes v_t and h_t represent a LDS. This portion of the DBN is not necessarily constrained to this model. It could represent some other system, such as a hidden Markov model or a nonlinear dynamic system. Adjustments to the GSF algorithm may however be required for this adaptation. The contextual elements may represent any desired entity that influences behaviour. This may even include behaviour and class of neighbouring entities. Additional applications may include surveillance of humans, maritime ports, public airfields, military airspace, public roads, financial markets and computer networks. Monitoring of illegal activities such as fraudulent activities, illegal immigration and poaching are also promising applications.

The GSF algorithm that is used for inference in the model is computationally efficient. On a $2.8GHz$ Intel i7 core processor, it takes $1.6ms$ to classify a vessel at each time step. In many applications this speed is sufficient for the model to operate on real time data. The model does not require feature extraction or data preprocessing at each time step. This is a significant contributor to computational efficiency.

A drawback of the proposed model is the relation of complexity to number of variable states. The complexity of the model significantly increases with an increasing number states in the discrete valued variables. This includes increasing the number of contextual elements. By nature, the DBN is subject to the curse of dimensionality. It is recommended that contextual elements be combined where possible as described in Section 4.1.

On initialisation of the model, conditional probability distributions are required to be defined. A large amount of data may be required to define these distributions accurately. Furthermore, the transition probabilities between behavioural states may be difficult to define in some applications. Finding a solution for learning the parameters of the model from data is a promising future research objective.

The nodes v_t and h_t are configured as an LDS in this study. The LDS may not be able to model systems that exhibit nonlinear motion. However, in this study, the drifting and fishing states were modelled as a random walk. Though the LDS was not able to track the vessels well in these states, it was able to distinguish this motion from the sailing and anchored states motion. This provided sufficient means for behaviour classification.

7.2. Comparison With Other Methods

Various machine learning methods such as neural networks are considered to be 'black box' methods. The interactions between variables is not evident. The proposed model is not a 'black box' method. The variables and their interaction is defined through expert knowledge, data and design. These interactions are represented by conditional probability distributions.

To apply general machine learning and classification methods such as neural networks and support vector machines, a windowing technique is generally used (Wu and Lee, 2015; Frank et al., 2001). This involves selecting an appropriate number of lags to be included for classification. This technique reduces the number of time samples that are considered in the classification. In the proposed model, the filtering approach is used to determine $p(c|v_{1:T}, \bar{a}_{1:T})$. The class is thus determined from all time samples.

The Box-Jenkins and related methods are specifically formulated for analysing time series. They are able to consider temporal relationships in variables. These methods are however constrained to modelling stationary processes. Hidden Markov models (HMM) are another example of models that are naturally able to model time series. The HMM is not limited to stationary processes. It is however constrained to modelling systems that follow discrete state changes. These mentioned time series based models are specifically formulated for modelling the time series. They do not provide an immediate means for information fusion and classification. The model proposed in this study is not constrained to stationary processes or discrete observations. Furthermore, it provides the means for information fusion and classification.

The proposed model represents the class in the form of a belief. By nature, it includes uncertainty in the estimation. This is particularly useful when representing information to human operators in surveillance environments. Many machine learning and pattern recognition methods such as support vector machines do not necessarily represent the class with uncertainty. Classification is often a decisive result. In general, methods that are able to represent the classification result with uncertainty include probabilistic and fuzzy logic methods.

Fuzzy time series (FTS) provide a means to model temporal relations in data as well as represent results with uncertainty. Challenges are associated with partitioning the time series data into fuzzy sets. Furthermore, the partitioning operation generalises the data. Information is potentially lost in this process. Many other methods also perform some form of partitioning of the time series; several of which are mentioned Section 2. These include methods for animal behaviour modelling and the windowing approach for machine learning algorithms. The proposed model does not require the partitioning of the data. It receives the continuous data directly as an input into the visible node, v_t. The dynamics of the system are modelled as a continuous entity. Information is not lost through partitioning or clustering of data.

8. Future Research

It is expected that the results described in Section 6.5 may be significantly improved by applying the smoothing operation for inference. The filtering operation considers data sequentially. The smoothing operation considers the entire dataset at any particular time. This will be beneficial in scenarios such as that illustrated in Figure 8. The smoothing operation at time $t = 1$ will take into account the fact that the vessel entered a drifting state in a pirate zone at time $t = 1628$. The pirate class may thus potentially be inferred at this time.

The Gaussian sum smoothing (GSS) algorithm is an algorithm that has been used to perform smoothing in the SLDS (Barber, 2012). The algorithm requires additional approximations to that of the GSF algorithm when applied to the SLDS (Barber, 2012). The GSS algorithm is thus more prone to failure. Further approximations will be required when applying the GSS algorithm to the proposed model due to the additional variables. This will result in a higher level of unreliability of the GSS algorithm. A more appropriate smoothing algorithm needs to be determined in future research. Nonparametric algorithms and approximate algorithms such as the Markov chain Monte Carlo method may be considered.

Learning algorithms may be considered for learning the parameters and distributions of the model. In this study, expert knowledge is use to define the switching state variable. Learning the parameters of the switching state variable from data is not possible as the data is not available. Many applications exist where data is available for learning parameters of variables such as the switching state variable. An emerging approach of particular interest for dealing with such problems is Bayesian nonparametrics (Jordan et al., 2010). Learning and inference in complex dynamical models may be achieved using Bayesian nonparametric methods (Fox, 2009). The approach is to provide a probabilistic model of the dynamic model itself (Teh and Jordan, 2010). The hierarchical Bayesian nonparametric model provides a means to represent infinite-dimensional parameters hierarchically. This allows for representation of models with large degrees of freedom. Furthermore, additional parameters or parameter states may be incorporated into the model dynamically (Fox, 2009).

9. Summary and Conclusion

A novel Bayesian model for context-based behaviour modelling and classification is proposed. Behaviour is modelled using a linear dynamic system that switches between various kinematic behavioural activities. Continuous dynamics are thus used to model various behaviour activities. A means to include contextual elements that may influence a system's behaviour is provided. The GSF algorithm is applied to perform approximate inference to infer the probability of a class given provided time series data. The DBN structure allows for natural application for modelling temporal aspects of time series data.

Novelty is found in the unified approach to perform information fusion, behavioural modelling and classification simultaneously. The way in which contextual information is fused into the model for behaviour modelling is unique. The method of classifying entities according to their behaviour using the Bayesian approach to determining the probability of a class given trajectory and contextual information is distinct. Finally, a GSF algorithm is derived particularly for the proposed model.

The proposed model is demonstrated as a new approach for identifying maritime pirates. A simulated maritime piracy situation dataset is utilised. The model is applied to classify vessels as pirate vessels, fishing vessels or transport vessels. The data presented to the model includes tracked vessel data, sailing conditions and region data. The sailing conditions describe a set of various conditions such as ocean conditions, weather conditions, time of day and the season of year. The variables and their states were carefully selected to minimise complexity related to the curse of dimensionality in DBNs. The results that are presented indicate that the method performs well in classification. Various measures are considered in evaluating the method. These measures are used to argue a high level effectiveness of the method.

Many existing methods in literature do not naturally consider temporal relationships in data. Sequential data is often partitioned or clustered into groups for processing resulting in potential information loss. Methods that do consider temporal relationships and do not cluster data are often application or task specific. The DBN in the proposed model provides a natural means to model temporal relationships. Even if there are no differences in activity kinematics between vessel classes, the model is able to classify vessels purely on the transitions between different activities. A second key insight is that, by varying the switching state transition probabilities, the confidence of the model is altered. If the transition probabilities significantly differ between classes, the model will be more confident in its classification. That is, it will be more decisive in a shorter period of time.

The proposed model simultaneously performs context-based behavioural modelling, information fusion and classification in a computationally efficient manner. The model is flexible and may be used in a wide range of applications and fields. These may include signal processing, econometrics, computer vision and robotics.

Appendix A. Derivation of the GSF Algorithm for the Proposed DBN

The Gaussian sum filter (GSF) algorithm is applied to the proposed DBN model. The GSF algorithm is a recursive algorithm. The derivation of the algorithm for the proposed model involves a number of steps that are repeated. These steps are described as follows:

1. Marginalise over the previous hidden variable(s).
2. Divide up the evidence into evidence at time t and evidence at time $1:t-1$.
3. Apply Bayes rule to determine the posterior of the evidence at time t.
4. Apply the chain rule of probability to form several factors.
5. Apply the Markov property or D-separation to simplify the factors.

According to the above procedure, the joint distribution of the hidden variables is marginalised over previous hidden variables. The evidence is split into the current observation at time t and previous observations at times $1:t$ as follows

$$p(s_t, h_t, c|v_{1:t}, \bar{a}_{1:t}) = \sum_{s_{t-1}} \int_{h_{t-1}} p(s_t, h_t, s_{t-1}, h_{t-1}, c|v_t, v_{1:t-1}, \bar{a}_{1:t}). \tag{A.1}$$

Using Bayes' rule, (A.1) may be expressed according to the evidence at time t as

$$\begin{aligned}p(s_t, h_t, c|v_{1:t}, \bar{a}_{1:t}) = &\\ \zeta \sum_{s_{t-1}} \int_{h_{t-1}} &[p(v_t|s_t, h_t, s_{t-1}, h_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t})\\ &\cdot p(s_t, h_t, s_{t-1}, h_{t-1}, c|v_{1:t-1}, \bar{a}_{1:t})],\end{aligned} \tag{A.2}$$

where ζ is a normalising constant. The factors in (A.2) may be simplified using D-separation. The v_t variable in the first term is only conditioned on h_t. The remaining terms in the first factor are connected serially through h_t to v_t. If h_t is observed, the remaining terms are independent of v_t. Finally, the second term in (A.2) may be expanded using the chain rule of probability as follows

$$\begin{aligned}p(s_t, h_t, c|v_{1:t}, \bar{a}_{1:t}) = &\\ \zeta \sum_{s_{t-1}} \int_{h_{t-1}} &[p(v_t|h_t)\\ &\cdot p(h_t|s_t, s_{t-1}, c, h_{t-1}, v_{1:t-1}, \bar{a}_{1:t})\\ &\cdot p(s_t|s_{t-1}, c, h_{t-1}, v_{1:t-1}, \bar{a}_{1:t})\\ &\cdot p(s_{t-1}, h_{t-1}, c|v_{1:t-1}, \bar{a}_{1:t})].\end{aligned} \tag{A.3}$$

D-separation may be applied to the last three factors in (A.3) by considering Figure 1as follows

Second factor $p(h_t|s_t, s_{t-1}, c, h_{t-1}, v_{1:t-1}, \bar{a}_{1:t})$**:**
The s_t and h_{t-1} nodes are directed into the h_t node. The variable s_{t-1} forms a serial connection with h_t through s_t. The variable h_t is independent of s_{t-1} given s_t. Similarly, h_t is independent of h_{t-2} given h_{t-1}. The variables $v_{1:t-1}$ are connected to h_t through diverging connections from $h_{1:t-1}$. These variables are however considered to be blocked by the instantiation of h_{t-1}. The variable h_t is conditioned only on s_t and h_{t-1}.

Third factor $p(s_t|s_{t-1}, c, h_{t-1}, v_{1:t-1}, \bar{a}_{1:t})$**:**
The s_{t-1}, $\bar{a}_t$ and c nodes are directed into the s_t node. The h_{t-1} node is connected to the s_t node through a diverging connection from s_{t-1}. The variable s_t is independent of h_{t-1} given s_{t-1}. The variable v_{t-1} is connected to s_t through a diverging connection through s_{t-1}. The variable s_t is independent of v_{t-1} given s_{t-1}. In general, the variables $v_{1:t-1}$ are blocked by s_{t-1}. The nodes $\bar{a}_{1:t-1}$ are blocked by the node s_{t-1}. The variable s_t is conditioned only on c, $\bar{a}_t$ and s_{t-1}.

Fourth factor $p(s_{t-1}, h_{t-1}, c|v_{1:t-1}, \bar{a}_{1:t})$**:**
The node s_{t-1} and h_{t-1} form converging connections to node $\bar{a}_t$ through nodes s_t and h_t respectively. The variables s_{t-1} and h_{t-1} are only independent of $\bar{a}_t$ if the converging node or any of its children are instantiated. The children of s_t and h_t include $s_{t+1:T}$, $h_{t+1:T}$ and $v_{t:T}$. None of these variables are instantiated. The variables s_{t-1} and h_{t-1} are independent of the variables $\bar{a}_t$.

By D-separation, (A.3) may be simplified to

$$p(s_t, h_t, c|v_{1:t}, \bar{a}_{1:t}) = \zeta \sum_{s_{t-1}} \int_{h_{t-1}} [p(v_t|h_t) \cdot p(h_t|s_t, h_{t-1}) \cdot p(s_t|s_{t-1}, c, \bar{a}_t) \cdot p(s_{t-1}, h_{t-1}, c|v_{1:t-1}, \bar{a}_{1:t-1})]. \quad (A.4)$$

The first factor is the emission probability. The second and third factors describe transition probabilities. The fourth factor is the joint distribution at time $t-1$. The current joint distribution is thus a function of the previously determined joint distribution. This forms the desired recursion. The GSF algorithm performs inference by propagating $p(s_t, h_t, c|v_{1:t}, \bar{a}_{1:t})$ forward using exact dynamics. This factor can be separated into continuous and discrete parts as follows

$$p(s_t, h_t, c|v_{1:t}, \bar{a}_{1:t}) = p(h_t|s_t, c, v_{1:t}, \bar{a}_{1:t}) \cdot p(s_t|c, v_{1:t}, \bar{a}_{1:t}) \cdot p(c|v_{1:t}, \bar{a}_{1:t}). \quad (A.5)$$

The first factor is a continuous distribution. The second and third factors are discrete distributions and are denoted by α_t and β_t respectively. The second factor infers the switching state. The third factor infers the class and is equivalent to (3) when $t = T$. The continuous distribution will first be considered, followed by the discrete distributions.

Appendix A.1. Continuous Distribution

The continuous factor in (A.5) is approximated by a mixture of I Gaussians as follows

$$p(h_t|s_t, c, v_{1:t}, \bar{a}_{1:t}) \approx \sum_{i_t=1}^{I} q(h_t|i_t, s_t, c, v_{1:t}, \bar{a}_{1:t}) \cdot q(i_t|s_t, c, v_{1:t}, \bar{a}_{1:t}). \quad (A.6)$$

The variable i_t is the indicator variable that references the i^{th} Gaussian mixture component. In the proposed DBN, the indicator variable is associated with the variable h_t. Approximate distributions are denoted by q. The i^{th} Gaussian mixture component is described by the first factor in (A.6). This mixture component is parametrised by vectors containing the mean $f(i_t, s_t, c)$ and the covariance $F(i_t, s_t, c)$. The weight of the i^{th} Gaussian mixture component is described by the second factor in (A.6).

At each time step, the I mixture of Gaussians is expanded to an $I \times S$ mixture of Gaussians for each class $c = 1, ..., C$. This distribution is given as

$$q(h_t|s_t, c, v_{1:t}, \bar{a}_{1:t}) = \sum_{i_{t-1}, s_{t-1}} q(h_t|i_{t-1}, s_{t-1}, s_t, c, v_{1:t}, \bar{a}_{1:t}) \cdot q(i_{t-1}, s_{t-1}|s_t, c, v_{1:t}, \bar{a}_{1:t}). \quad (A.7)$$

The first factor describes a Gaussian mixture component. The second factor describes the weight of the Gaussian mixture component.

Appendix A.1.1. Gaussian Mixture Components Evaluation

The linear dynamic system is comprised of the h_t and v_t nodes. To evaluate the Gaussian, consider the following joint distribution that describes the linear dynamic system

$$q(h_t, v_t|i_{t-1}, s_{t-1}, s_t, c, v_{1:t-1}, \bar{a}_{1:t}) = \int_{h_{t-1}} [q(h_t, v_t|h_{t-1}, i_{t-1}, s_{t-1}, s_t, c, v_{1:t-1}, \bar{a}_{1:t}) \cdot q(h_{t-1}|i_{t-1}, s_{t-1}, s_t, c, v_{1:t-1}, \bar{a}_{1:t})]. \quad (A.8)$$

By D-separation, h_t and v_t are independent of s_{t-1} given h_{t-1} and s_t. The variables h_t and v_t are independent of the indicator variable i_{t-1} at time $t-1$. In the second term, h_{t-1} is independent s_t and a_t given that h_t and any of its descendants are not observed. Equation (A.8) is thus simplified as

$$q(h_t, v_t|i_{t-1}, s_{t-1}, s_t, c, v_{1:t-1}, \bar{a}_{1:t}) = \int_{h_{t-1}} [q(h_t, v_t|h_{t-1}, s_t, c, v_{1:t-1}, \bar{a}_{1:t}) \cdot q(h_{t-1}|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t-1})]. \quad (A.9)$$

The second term corresponds to the Gaussian component in (A.6) at time $t-1$. This mixture component is parametrised by mean $f(i_{t-1}, s_{t-1}, c)$ and covariance $F(i_{t-1}, s_{t-1}, c)$. This Gaussian component may be propagated forward using the linear dynamics of the system. Lemma 1 in Appendix B may be utilised to evaluate (A.9) using this forward propagation (Barber, 2012). Lemma 1 implies that (A.9) is a Gaussian with the following mean and covariance elements

$$\Sigma_{hh} = \mathbf{A}(s_t, c, \bar{a}_t) F(i_{t-1}, s_{t-1}, c) \cdot \mathbf{A}^T(s_t, c, \bar{a}_t) + \Sigma_h(s_t, c, \bar{a}_t), \quad (A.10)$$

$$\Sigma_{vv} = \mathbf{B}(s_t, c, \bar{a}_t) \Sigma_{hh}{}^T(s_t, c, \bar{a}_t) + \Sigma_v(s_t, c, \bar{a}_t), \quad (A.11)$$

$$\Sigma_{vh} = \Sigma_{hv}^T = \mathbf{B}(s_t, c, \bar{a}_t) \Sigma_{hh}, \quad (A.12)$$

$$\mu_v = \mathbf{B}(s_t, c, \bar{a}_t) \mathbf{A}(s_t, c, \bar{a}_t) f(i_{t-1}, s_{t-1}, c), \quad (A.13)$$

$$\mu_h = \mathbf{A}(s_t, c, \bar{a}_t) f(i_{t-1}, s_{t-1}, c). \quad (A.14)$$

To determine $q(h_t|i_{t-1}, s_{t-1}, s_t, c, v_{1:t-1}, \bar{a}_{1:t-1})$ in (A.7), (A.9) is conditioned on v_t to provide

$$q(h_t|i_{t-1}, s_{t-1}, s_t, c, v_{1:t}, \bar{a}_{1:t-1}) = \mathcal{N}(h_t|\mu_{h|v}, \Sigma_{h|v}). \quad (A.15)$$

Lemma 2 in Appendix B is utilised to determine the values of $\mu_{h|v}$ and $\Sigma_{h|v}$ (Barber, 2012). Lemma 2 indicates that the values of $\mu_{h|v}$ and $\Sigma_{h|v}$ are defined according to (A.10), (A.11), (A.12), (A.13) and (A.14) as follows

$$\mu_{h|v} = \mu_h + \Sigma_{hv} \Sigma_{vv}^{-1} (v_t - \mu_v), \quad (A.16)$$

$$\Sigma_{h|v} = \Sigma_{hh} - \Sigma_{hv} \Sigma_{vv}^{-1} \Sigma_{vh}. \quad (A.17)$$

Equations (A.10) through (A.14), (A.16) and (A.17) are closely related to the equations found in the Kalman filter.

Appendix A.1.2. Gaussian Mixture Weights Evaluation

The observations $v_{1:t}$ in the Gaussian mixture weights in (A.7) may be separated as follows

$$q(i_{t-1}, s_{t-1}|s_t, c, v_{1:t}, \bar{a}_{1:t}) = q(i_{t-1}, s_{t-1}|s_t, c, v_t, v_{1:t-1}, \bar{a}_{1:t}). \quad (A.18)$$

Where v_t, s_t and c are particular realisations of their respective random variables. Using Bayes' rule, (A.18) may be written as

$$\begin{aligned} q(i_{t-1}, s_{t-1}|s_t, c, v_{1:t}, \bar{a}_{1:t}) = \\ &\zeta q(v_t, s_t|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &q(i_{t-1}, s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t}), \end{aligned} \tag{A.19}$$

where ζ is a normalization constant. By the chain rule of probability, (A.19) may be expanded as follows

$$\begin{aligned} q(i_{t-1}, s_{t-1}|s_t, c, v_{1:t}, \bar{a}_{1:t}) \propto \\ &q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(s_t|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(i_{t-1}|s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ &\cdot p(s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t-1}). \end{aligned} \tag{A.20}$$

By D-separation, the third and fourth factors are conditionally independent of the external factors given at time t. The variables s_{t-1} and i_{t-1} are connected to $\bar{a}_t$ through converging connections through s_t and i_t respectively. The variables s_t, i_t and their children are not instantiated.

The first factor in (A.20) is a Gaussian with mean μ_v and variance Σ_{vv}. The second factor is the transition distribution between states. By D-separation, s_t in the second factor is conditionally independent of $v_{1:t-1}$ given s_{t-1}. Furthermore, s_t in the second factor is conditionally independent of i_{1-t}. The second factor is thus given by

$$p(s_t|s_{t-1}, c, \bar{a}_t). \tag{A.21}$$

The third factor in (A.20) contains the weights of the Gaussian mixture at time $t-1$. The fourth factor in (A.20) is the distribution of the switching state at time $t-1$ given the class, data and contextual elements. This fourth factor is denoted by α_{t-1} and is computed in the preceding step of the GSF algorithm.

Appendix A.1.3. Gaussian Mixture Components Collapsing

For each class, the mixture of $I \times S$ Gaussian mixtures described in (A.7) is collapsed to an I Gaussian mixture described by (A.6). A method to collapse the Gaussian mixture is to retain the $I-1$ mixture components with the highest weights. The remaining mixture components are merged to a single Gaussian with the following parameters (Barber, 2006)

$$\mu = \sum_i p_i \mu_i, \quad \Sigma = \sum_i p_i \left(\Sigma_i + \mu_i \mu_i^T\right) - \mu\mu^T. \tag{A.22}$$

The values p_i, μ_i and Σ_i are the weight, mean and variance of the i^{th} Gaussian component that is merged.

Appendix A.2. Discrete Components

The first discrete component defined in (A.5) may be expressed as

$$\begin{aligned} \alpha_t =& p(s_t|c, v_{1:t}, \bar{a}_{1:t}) = \\ &\sum_{i_{t-1}, s_{t-1}} q(i_{t-1}, s_{t-1}, s_t|c, v_t, v_{1:t-1}, \bar{a}_{1:t}). \end{aligned} \tag{A.23}$$

Where v_t and c are particular realisations of their respective random variables. Using Bayes' rule, the term within the summation in (A.23) may be written as

$$\begin{aligned} q(s_t, i_{t-1}, s_{t-1}|c, v_t, v_{1:t-1}, \bar{a}_{1:t}) = \\ &\zeta q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(s_t, i_{t-1}, s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t}), \end{aligned} \tag{A.24}$$

where ζ is a normalization constant. This result may be expanded using the chain rule of probability to provide the following result for α_t:

$$\begin{aligned} \alpha_t = p(s_t|c, v_{1:t}, \bar{a}_{1:t}) \propto \\ &\sum_{i_{t-1}, s_{t-1}} [q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(s_t|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(i_{t-1}|s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ &\cdot p(s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t-1})]. \end{aligned} \tag{A.25}$$

Each of these factors are determined in the calculation of the mixture weights in (A.20). As in (A.20), the third and fourth factors are D-separated from $\bar{a}_t$.

The second discrete component defined in (A.5) may be expressed as

$$\begin{aligned} \beta_t = p(c|v_{1:t}, \bar{a}_{1:t}) = \\ &\sum_{i_{t-1}, s_{t-1}, s_t} q(i_{t-1}, s_{t-1}, s_t, c|v_t, v_{1:t-1}, \bar{a}_{1:t}). \end{aligned} \tag{A.26}$$

Where v_t and c are particular realisations of their respective random variables. Using Bayes' rule, the term within the summation in (A.26) may be written as

$$\begin{aligned} q(i_{t-1}, s_{t-1}, s_t, c|v_t, v_{1:t-1}, \bar{a}_{1:t}) = \\ &\zeta q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(s_t, i_{t-1}, s_{t-1}, c|v_{1:t-1}, \bar{a}_{1:t}), \end{aligned} \tag{A.27}$$

where ζ is a normalization constant. This result may be expanded using the chain rule of probability to provide the following result for β_t

$$\begin{aligned} \beta_t = p(c|v_{1:t}, \bar{a}_{1:t}) \propto \\ &\sum_{i_{t-1}, s_{t-1}, s_t} [q(v_t|s_t, i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(s_t|i_{t-1}, s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t}) \\ &\cdot q(i_{t-1}|s_{t-1}, c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ &\cdot p(s_{t-1}|c, v_{1:t-1}, \bar{a}_{1:t-1}) \\ &\cdot p(c|v_{1:t-1}, \bar{a}_{1:t-1})]. \end{aligned} \tag{A.28}$$

The first four factors in (A.28) are determined in the calculation of the mixture weights in (A.20). As in (A.20), the third and fourth factors are D-separated from the external factors given at time t. In the fifth factor, c and $\bar{a}_t$ are D-separated given that the converging node s_t between c and $\bar{a}_t$ and the children of s_t are not instantiated.

Equations (A.20), (A.25) and (A.28) are determined up to a normalisation constant. After each iteration, α_t and β_t are thus normalised to form probability density functions. The algorithm for the GSF procedure on the proposed model is provided in Algorithm 1.

Appendix B. Lemmas

Lemma 1. *Let y be related to x through $y = Mx + \eta$, where $x \perp\!\!\!\perp \eta$, $\eta \sim \mathcal{N}(\mu, \Sigma)$, and $x \sim \mathcal{N}(\mu_x, \Sigma_x)$. Then the marginal $p(y) = \int_x p(y|x)p(x)$ is a Gaussian given by (Barber, 2012)*

$$p(y) = \mathcal{N}(y|M\mu_x + \mu, M\Sigma_x M^T + \Sigma).$$

Lemma 2. *Consider a Gaussian distribution parametrised by μ and Σ, defined jointly over two vectors x and y of potentially differing dimensions,*

$$z = \begin{pmatrix} x \\ y \end{pmatrix},$$

with corresponding mean and partitioned covariance

$$\mu = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \Sigma_{xx}\Sigma_{xy} \\ \Sigma_{yx}\Sigma_{yy} \end{pmatrix}.$$

Where $\Sigma_{yx} = \Sigma_{xy}^T$. The marginal distribution is given by (Barber, 2012)

$$p(x) = \mathcal{N}(x|\mu_x, \Sigma_{xx})$$

and conditional

$$p(x|y) = \mathcal{N}(x|\mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y), \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}).$$

Acknowledgement

This work was funded by the South African National Research Foundation (NRF) and the the Advanced Sensors and Electronics Defence (ASED) Centre of KACST through the Council for Scientific and Industrial Research (CSIR).

Al-Hmouz, R., Pedrycz, W., Balamash, A., 2015. Description and prediction of time series: A general framework of granular computing. Expert Systems with Applications 42 (10), 4830 – 4839.
URL http://www.sciencedirect.com/science/article/pii/S0957417415000834

Alspach, D., Sorenson, H., 1972. Nonlinear bayesian estimation using gaussian sum approximations. Automatic Control, IEEE Transactions on 17 (4), 439–448.

Alves, J., Herman, J., Rowe, N. C., June 2004. Robust recognition of ship types from an infrared silhouette. Tech. rep., Monterey, California. Naval Postgraduate School, command and Control Research and Technology Symposium, San Diego, CA, USA.

Anagnostopoulos, C., Sekkas, O., Hadjiefthymiades, S., 2007. Context fusion: Dealing with sensor reliability. In: Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on. pp. 1–6.

Arnesen, T. N., Olsen, R. B., 2004. Literature review on vessel detection. Tech. rep., Forsvarets forskningsinstitutt, fFI/RAPPORT-2004/02619.

Askari, S., Montazerin, N., 2015. A high-order multi-variable fuzzy time series forecasting algorithm based on fuzzy clustering. Expert Systems with Applications 42 (4), 2121 – 2135.
URL http://www.sciencedirect.com/science/article/pii/S0957417414005855

Balci, M., Pegg, R., 2006. Towards global maritime domain awareness - recent developments and challenges. In: Information Fusion, 2006 9th International Conference on. pp. 1–5.

Baldini, G., Shaw, D., Dimc, F., 2010. A communication monitoring system to support maritime security. In: ELMAR, 2010 PROCEEDINGS. pp. 243–246.

Bar-Shalom, Y., Li, X., 1993. Estimation and Tracking: Principles, Techniques, and Software. The Artech House radar library. Artech House, Incorporated.

Barber, D., 2006. Expectation correction for smoothed inference in switching linear dynamical systems. The Journal of Machine Learning Research 7, 2515–2540.

Barber, D., 2012. Bayesian Reasoning and Machine Learning. Cambridge University Press.

Bengio, S., Bengio, Y., May 2000. Taking on the curse of dimensionality in joint distributions using neural networks. Neural Networks, IEEE Transactions on 11 (3), 550–557.

Bennett, C., Stewart, R. A., Beal, C. D., 2013. Ann-based residential water end-use demand forecasting model. Expert Systems with Applications 40 (4), 1014 – 1023.
URL http://www.sciencedirect.com/science/article/pii/S0957417412009736

Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D., 2010. A survey of context modelling and reasoning techniques. Pervasive and Mobile Computing 6 (2), 161 – 180, context Modelling, Reasoning and Management.

Blackman, S., Popoli, R., 1999. Design and Analysis of Modern Tracking Systems. Artech House Radar Library. Artech House.

Blasch, E., Valin, P., Bosse, E., 2010. Measures of effectiveness for high-level fusion. In: Information Fusion (FUSION), 2010 13th Conference on. pp. 1–8.

Bon, N., Hajduch, G., Khenchaf, A., Garello, R., Quellec, J.-M., September 2008. Recent developments in detection, imaging and classification for airborne maritime surveillance. Signal Processing, IET 2 (3), 192–203.

Bossé, E., Shahbazian, E., Rogova, G., 2013. Prediction and Recognition of Piracy Efforts Using Collaborative Human-Centric Information Systems. NATO Science for Peace and Security Sub-Series E: Human and Societal Dynamics. IOS Press, Incorporated.

Bouejla, A., Chaze, X., Guarnieri, F., Napoli, A., 2014. A bayesian network to manage risks of maritime piracy against offshore oil fields. Safety Science 68 (0), 222 – 230.
URL http://www.sciencedirect.com/science/article/pii/S0925753514000952

Boyen, X., Koller, D., 1998. Tractable inference for complex stochastic processes. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence. UAI'98. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 33–42.

Bueger, C., Stockbruegger, J., Werthes, S., 2011. Pirates, fishermen and peacebuilding: Options for counter-piracy strategy in somalia. Contemporary Security Policy 32 (2), 356–381.

Carvalho, R., Haberlin, R., Costa, P., Laskey, K., Chang, K., 2011. Modeling a probabilistic ontology for maritime domain awareness. In: Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on. pp. 1–8.

Castaldo, F., Palmieri, F., Bastani, V., Marcenaro, L., Regazzoni, C., July 2014. Abnormal vessel behavior detection in port areas based on dynamic bayesian networks. In: Information Fusion (FUSION), 2014 17th International Conference on. pp. 1–7.

Chatfield, C., 1996. The Analysis of Time Series: An Introduction, 5th Edition. Chapman & Hall. CRC Press.

Chen, C.-H., Khoo, L. P., Chong, Y. T., Yin, X. F., 2014. Knowledge discovery using genetic algorithm for maritime situational awareness. Expert Systems with Applications 41 (6), 2742 – 2753.
URL http://www.sciencedirect.com/science/article/pii/S0957417413007963

Corbane, C., Marre, F., Petit, M., 2008. Using spot-5 hrg data in panchromatic mode for operational detection of small ships in tropical area. Sensors 8 (5), 2959–2973.

Costa, P. C. G., Laskey, K. B., Chang, K.-C., Sun, W., Park, C. Y., , Matsumoto, S., August 2012. High-level information fusion with bayesian semantics. In: UAI 9th Bayesian Modeling Applications Workshop. Catalina Island, CA, pp. –, held at the Conference of Uncertainty in Artificial Intelligence (BMAW UAI 2012).

Dabrowski, J. J., de Villiers, J. P., 2015. Maritime piracy situation modelling with dynamic bayesian networks. Information Fusion 23 (0), 116 – 130.

URL http://www.sciencedirect.com/science/article/pii/S1566253514000840

Das, S., 2008. High-Level Data Fusion. Artech House electronic warfare library. Artech House, Incorporated.

Dean, T., Kanazawa, K., February 1989. A model for reasoning about persistence and causation. Computational Intelligence 5 (2), 142–150.

Dey, A. K., Jan. 2001. Understanding and using context. Personal Ubiquitous Comput. 5 (1), 4–7.
URL http://dx.doi.org/10.1007/s007790170019

Dias, J. G., Ramos, S. B., 2014. Dynamic clustering of energy markets: An extended hidden markov approach. Expert Systems with Applications 41 (17), 7722 – 7729.
URL http://www.sciencedirect.com/science/article/pii/S0957417414003108

Doucet, A., de Freitas, N., Murphy, K., Russell, S., 2000. Rao-blackwellised particle filtering for dynamic bayesian networks. In: Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence. UAI'00. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 176–183.

Du, W., Leung, S. Y. S., Kwong, C. K., 2014. Time series forecasting by neural networks: A knee point-based multiobjective evolutionary algorithm approach. Expert Systems with Applications 41 (18), 8049 – 8061.
URL http://www.sciencedirect.com/science/article/pii/S0957417414003820

Durbin, J., Koopman, S., 2001. Time Series Analysis by State Space Methods. Oxford Statistical Science Series. Clarendon Press.

Dutta, R., Smith, D., Rawnsley, R., Bishop-Hurley, G., Hills, J., Timms, G., Henry, D., 2015. Dynamic cattle behavioural classification using supervised ensemble classifiers. Computers and Electronics in Agriculture 111 (0), 18 – 28.
URL http://www.sciencedirect.com/science/article/pii/S0168169914003123

Esher, L., Hall, S., Regnier, E., Sanchez, P., Hansen, J., Singham, D., 2010. Simulating pirate behavior to exploit environmental information. In: Simulation Conference (WSC), Proceedings of the 2010 Winter. pp. 1330–1335.

Fooladvandi, F., Brax, C., Gustavsson, P., Fredin, M., 2009. Signature-based activity detection based on bayesian networks acquired from expert knowledge. In: Information Fusion, 2009. FUSION '09. 12th International Conference on. pp. 436–443.

Fox, E. B., September 2009. Bayesian nonparametric learning of complex dynamical phenomena. Ph.D. thesis, Massachusetts Institute of Technology.

Frank, R., Davey, N., Hunt, S., 2001. Time series prediction and neural networks. Journal of Intelligent and Robotic Systems 31 (1-3), 91–103.
URL http://dx.doi.org/10.1023/A%3A1012074215150

Franklin, G., Powell, J., Emami-Naeini, A., 2002. Feedback Control of Dynamic Systems, 4th Edition. Addison-Wesley series in electrical and computer engineering. Control engineering. Prentice Hall.

Garcia, J., Gomez-Romero, J., Patricio, M., Molina, J., Rogova, G., 2011. On the representation and exploitation of context knowledge in a harbor surveillance scenario. In: Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on. pp. 1–8.

George, J., Crassidis, J., Singh, T., 2009. Threat assessment using context-based tracking in a maritime environment. In: Information Fusion, 2009. FUSION '09. 12th International Conference on. pp. 187–194.

Ghahramani, Z., Hinton, G. E., 1996. Parameter estimation for linear dynamical systems. Tech. rep., Technical Report CRG-TR-96-2, University of Totronto, Dept. of Computer Science.

Ghahramani, Z., Hinton, G. E., 2000. Variational learning for switching state-space models. Neural Computation 12 (4), 831–864.

Gibbins, D., Gray, D., Dempsey, D., 1999. Classifying ships using low resolution maritime radar. In: Signal Processing and Its Applications, 1999. ISSPA '99. Proceedings of the Fifth International Symposium on. Vol. 1. pp. 325–328 vol.1.

Gómez-Romero, J., Serrano, M. A., Patricio, M. A., García, J., Molina, J. M., 2012. Context-based scene recognition from visual data in smart homes: an information fusion approach. Personal and Ubiquitous Computing 16 (7), 835–857.

Gonzalez, L., Bishop-Hurley, G., Handcock, R., Crossman, C., 2015. Behavioral classification of data from collars containing motion sensors in grazing cattle. Computers and Electronics in Agriculture 110 (0), 91 – 102.
URL http://www.sciencedirect.com/science/article/pii/S0168169914002798

Gooijer, J. G. D., Hyndman, R. J., 2006. 25 years of time series forecasting. International Journal of Forecasting 22 (3), 443 – 473, twenty five years of forecasting.
URL http://www.sciencedirect.com/science/article/pii/S0169207006000021

Górecki, T., Łuczak, M., 2015. Multivariate time series classification with parametric derivative dynamic time warping. Expert Systems with Applications 42 (5), 2305 – 2312.
URL http://www.sciencedirect.com/science/article/pii/S0957417414006927

Gowsikhaa, D., Abirami, S., Baskaran, R., 2014. Automated human behavior analysis from surveillance videos: a survey. Artificial Intelligence Review 42 (4), 747–765.
URL http://dx.doi.org/10.1007/s10462-012-9341-3

Greidanus, H., 2005. Assessing the operationality of ship detection from space. Tech. rep., European Commission Joint Research Centre (JRC), eURISY Symposium New space services for maritime users: The impact of satellite technology on maritime legislation.

Greidanus, H., 2008. Satellite imaging for maritime surveillance of the european seas. In: Barale, V., Gade, M. (Eds.), Remote Sensing of the European Seas. Springer Netherlands, pp. 343–358.
URL http://dx.doi.org/10.1007/978-1-4020-6772-3_26

Grimmett, G. R., Stirzaker, D. R., 2001. Probability and Random Processes, 3rd Edition. Oxford University Press, Great Clarendon Street, Oxford OX2 6DP.

Hanninen, M., Banda, O. A. V., Kujala, P., 2014. Bayesian network model of maritime safety management. Expert Systems with Applications 41 (17), 7837 – 7846.
URL http://www.sciencedirect.com/science/article/pii/S0957417414003704

Hanninen, M., Kujala, P., 2014. Bayesian network modeling of port state control inspection findings and ship accident involvement. Expert Systems with Applications 41 (4, Part 2), 1632 – 1646.
URL http://www.sciencedirect.com/science/article/pii/S0957417413006817

Hassan, M. R., Nath, B., Kirley, M., 2007. A fusion model of hmm, {ANN} and {GA} for stock market forecasting. Expert Systems with Applications 33 (1), 171 – 180.
URL http://www.sciencedirect.com/science/article/pii/S0957417406001291

Hegde, R., Kurniawan, J., Rao, B., 2009. On the design and prototype implementation of a multimodal situation aware system. Multimedia, IEEE Transactions on 11 (4), 645–657.

Heger, M., Oberg, J., Dumiak, M., Moore, S., Patel-Predd, P., 2009. Technology vs. pirates. Spectrum, IEEE 46 (2), 9–10.

Hernàndez, J., Cabido, R., Montemayor, A. S., Pantrigo, J., 2014. Human activity recognition based on kinematic features. Expert Systems 31 (4), 345–353.
URL http://dx.doi.org/10.1111/exsy.12013

Hu, W., Tan, T., Wang, L., Maybank, S., Aug 2004. A survey on visual surveillance of object motion and behaviors. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 34 (3), 334–352.

ICC-IMB, January 2012. Icc-imb piracy and armed robbery against ships report - annual report 2011. Annual report, ICC International Maritime Bureau.

Jakob, M., Vaněk, O., Pěchouček, M., 2011. Using agents to improve international maritime transport security. Intelligent Systems, IEEE 26 (1), 90–96.

Joo, T. W., Kim, S. B., 2015. Time series forecasting based on wavelet filtering. Expert Systems with Applications 42 (8), 3868 – 3874.
URL http://www.sciencedirect.com/science/article/pii/S095741741500041X

Jordan, M., Sudderth, E., Wainwright, M., Willsky, A., Nov 2010. Major advances and emerging developments of graphical models [from the guest editors]. Signal Processing Magazine, IEEE 27 (6), 17–138.

Kalman, R. E., 1960. A new approach to linear filtering and prediction problems. Journal of Basic Engineering 82 (1), 35–45.

Kazemi, S., Abghari, S., Lavesson, N., Johnson, H., Ryman, P., 2013. Open data for anomaly detection in maritime surveillance. Expert Systems with Applications 40 (14), 5719 – 5729.
URL http://www.sciencedirect.com/science/article/pii/S0957417413002765

Kennedy, L. S., Chang, S.-F., 2007. A reranking approach for context-based concept fusion in video indexing and retrieval. In: Proceedings of the 6th ACM international conference on Image and video retrieval. CIVR '07. ACM, New York, NY, USA, pp. 333–340.

Kim, C.-J., 1994. Dynamic linear models with markov-switching. Journal of Econometrics 60 (1-2), 1 – 22.

Kocadağlı, O., Aşıkgil, B., 2014. Nonlinear time series forecasting with bayesian neural networks. Expert Systems with Applications 41 (15), 6596 – 6610.
URL http://www.sciencedirect.com/science/article/pii/S0957417414002589

Kodagoda, S., Sehestedt, S., 2014. Simultaneous people tracking and motion pattern learning. Expert Systems with Applications 41 (16), 7272 – 7280.
URL http://www.sciencedirect.com/science/article/pii/S0957417414002991

Koller, D., Friedman, N., 2009. Probabilistic Graphical Models: Principles and Techniques. Adaptive Computation and Machine Learning. MIT Press.

Kourentzes, N., Barrow, D. K., Crone, S. F., 2014. Neural network ensemble operators for time series forecasting. Expert Systems with Applications 41 (9), 4235 – 4244.
URL http://www.sciencedirect.com/science/article/pii/S0957417413009834

Kraska, J., Wilson, B., 2008. Fighting pirates: The pen and the sword. World Policy Journal 25 (4), 41–52.

Kruger, M., Ziegler, L., Heller, K., 2012. A generic bayesian network for identification and assessment of objects in maritime surveillance. In: Information Fusion (FUSION), 2012 15th International Conference on. pp. 2309–2316.

Lai, R. K., Fan, C.-Y., Huang, W.-H., Chang, P.-C., 2009. Evolving and clustering fuzzy decision tree for financial time series data forecasting. Expert Systems with Applications 36 (2, Part 2), 3761 – 3773.
URL http://www.sciencedirect.com/science/article/pii/S0957417408001474

Lam, P. R. B. E. Y. (Ed.), 2013. Autonomous ship classification using synthetic and real color images. Vol. 8661.
URL http://dx.doi.org/10.1117/12.2005749

Li, H., Wang, X., Dec 2008. Automatic recognition of ship types from infrared images using support vector machines. In: Computer Science and Software Engineering, 2008 International Conference on. Vol. 6. pp. 483–486.

Liwang, H., Ringsberg, J. W., Norsell, M., 2013. Quantitative risk analysis: Ship security analysis for effective risk control options. Safety Science 58 (0), 98 – 112.

Lu, W., Pedrycz, W., Liu, X., Yang, J., Li, P., 2014. The modeling of time series based on fuzzy information granules. Expert Systems with Applications 41 (8), 3799 – 3808.
URL http://www.sciencedirect.com/science/article/pii/S095741741300969X

Luo, G., Yang, S., Tian, G., Yuan, C., Hu, W., Maybank, S., Dec 2014. Learning human actions by combining global dynamics and local appearance. Pattern Analysis and Machine Intelligence, IEEE Transactions on 36 (12), 2466–2482.

Luo, Y., Wu, T.-D., Hwang, J.-N., 2003. Object-based analysis and interpretation of human motion in sports video sequences by dynamic bayesian networks. Computer Vision and Image Understanding 92 (2-3), 196 – 216, special Issue on Video Retrieval and Summarization.

Marchione, E., Johnson, S. D., 2013. Spatial, temporal and spatio-temporal patterns of maritime piracy. Journal of Research in Crime and Delinquency 50 (4), 504–524.

Marchione, E., Johnson, S. D., Wilson, A., 2014. Modelling maritime piracy: A spatial approach. Journal of Artificial Societies and Social Simulation 17 (2), 9.
URL http://jasss.soc.surrey.ac.uk/17/2/9.html

Marsh, C. D., September 2009. Counter piracy: A repeated game with asymmetric information. Master's thesis, Naval Postgraduate School.

Mazzarella, F., Vespe, M., Damalas, D., Osio, G., July 2014. Discovering vessel activities at sea using ais data: Mapping of fishing footprints. In: Information Fusion (FUSION), 2014 17th International Conference on. pp. 1–7.

Mesot, B., Barber, D., 2007. Switching linear dynamical systems for noise robust speech recognition. Audio, Speech, and Language Processing, IEEE Transactions on 15 (6), 1850–1858.

Middleton, R., October 2008. Piracy in somalia: Threatening global trade, feeding local wars. Chatham House, briefing Paper.

Minka, T. P., 1999. From hidden markov models to linear dynamical systems. Tech. rep., Tech. Rep. 531, Vision and Modeling Group of Media Lab, MIT.

Minka, T. P., 2001. A family of algorithms for approximate bayesian inference. Ph.D. thesis, Massachusetts Institute of Technology.

Murphy, K. P., July 2002. Dynamic bayesian networks: Representation, inference and learning. Ph.D. thesis, University of California, Berkeley.

Murphy, K. P., 2012. Machine Learning: A Probabilistic Perspective. Adaptive computation and machine learning series. MIT Press.

Nguyen, K. A., Stewart, R. A., Zhang, H., 2014. An autonomous and intelligent expert system for residential water end-use classification. Expert Systems with Applications 41 (2), 342 – 356.
URL http://www.sciencedirect.com/science/article/pii/S0957417413005368

nón, S. O., na, J. L. M., Gonzalez, A. J., 2014. A dynamic-bayesian network framework for modeling and evaluating learning from observation. Expert Systems with Applications 41 (11), 5212 – 5226.
URL http://www.sciencedirect.com/science/article/pii/S0957417414001262

Oh, S. M., Ranganathan, A., Rehg, J. M., Dellaert, F., 2005a. A variational inference method for switching linear dynamic systems. Tech. Rep. GVU Technical Report;GIT-GVU-05-16, Georgia Institute of Technology.

Oh, S. M., Rehg, J. M., Balch, T., Dellaert, F., 2005b. Data-driven mcmc for learning and inference in switching linear dynamic systems. In: PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE. Vol. 20. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, p. 944.

Pavlovic, V., Rehg, J., 2000. Impact of dynamic model learning on classification of human motion. In: Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. Vol. 1. pp. 788–795 vol.1.

Pavlovic, V., Rehg, J., Cham, T.-J., Murphy, K., 1999. A dynamic bayesian network approach to figure tracking using learned dynamic models. In: Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on. Vol. 1. pp. 94–101 vol.1.

Pavlovic, V. I., 1999. Dynamic bayesian networks for information fusion with applications to human-computer interfaces. Ph.D. thesis, University of Illinois at Urbana-Champaign.

Pearl, J., 1988. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Representation and Reasoning Series. Morgan Kaufman.

Petrovskaya, A., Thrun, S., 2009. Model based vehicle detection and tracking for autonomous urban driving. Autonomous Robots 26 (2-3), 123–139.

Rather, A. M., Agarwal, A., Sastry, V., 2015. Recurrent neural network and a hybrid model for prediction of stock returns. Expert Systems with Applications 42 (6), 3234 – 3241.
URL http://www.sciencedirect.com/science/article/pii/S0957417414007684

Robinel, A., Puzenat, D., 2014. Alcohol consumption detection through behavioural analysis using intelligent systems. Expert Systems with Applications 41 (5), 2574 – 2581.
URL http://www.sciencedirect.com/science/article/pii/S0957417413008105

Rogova, G., Garcia, J., 2013. Contextual Knowledge and Information Fusion for Maritime Piracy Surveillance. Vol. 109 of NATO Science for Peace and Security Series - E: Human and Societal Dynamics. IOS Press.

Saâdaoui, F., Rabbouch, H., 2014. A wavelet-based multiscale vector-ann model to predict comovement of econophysical systems. Expert Systems with Applications 41 (13), 6017 – 6028.
URL http://www.sciencedirect.com/science/article/pii/S0957417414001675

Sanderson, J. G., Teal, M. K., Ellis, T., 1999. Characterisation of a complex maritime scene using fourier space analysis to identify small craft. In: Image Processing and Its Applications, 1999. Seventh International Conference on (Conf. Publ. No. 465). Vol. 2. pp. 803–807 vol.2.

Sekkas, O., Anagnostopoulos, C. B., Hadjiefthymiades, S., 2007. Context fusion through imprecise reasoning. In: Pervasive Services, IEEE International Conference on. pp. 88–91.

Sekkas, O., Hadjiefthymiades, S., Zervas, E., 2006. Enhancing location estimation through data fusion. In: Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on. pp. 1–5.

Sevillano, J. C., Rios Insua, D., Rios, J., 2012. Adversarial risk analysis: The somali pirates case. Decision Analysis 9 (2), 86–95.

Shumway, R. H., Stoffer, D. S., 2010. Time series analysis and its applications:

with R examples. Springer Science & Business Media.
Smith, D., Dutta, R., Hellicar, A., Bishop-Hurley, G., Rawnsley, R., Henry, D., Hills, J., Timms, G., 2015. Bag of class posteriors, a new multivariate time series classifier applied to animal behaviour identification. Expert Systems with Applications 42 (7), 3774 – 3784.
URL http://www.sciencedirect.com/science/article/pii/S0957417414007192
Steinberg, A., Rogova, G., 2008. Situation and context in data fusion and natural language understanding. In: Information Fusion, 2008 11th International Conference on. pp. 1–8.
Stone, L., Barlow, C., Corwin, T., 1999. Bayesian multiple target tracking. Artech House radar library. Artech House.
Strang, T., Linnhoff-Popien, C., 2004. A context modeling survey. In: In: Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing, Nottingham/England. pp. –.
Szpak, Z. L., Tapamo, J. R., 2011. Maritime surveillance: Tracking ships inside a dynamic background using a fast level-set. Expert Systems with Applications 38 (6), 6669 – 6680.
URL http://www.sciencedirect.com/science/article/pii/S0957417410013060
Teh, Y. W., Jordan, M. I., 2010. Hierarchical bayesian nonparametric models with applications. In: Bayesian Nonparametrics. Cambridge University Press, pp. 158–207.
Tetreault, B., Sept 2005. Use of the automatic identification system (ais) for maritime domain awareness (mda). In: OCEANS, 2005. Proceedings of MTS/IEEE. pp. 1590–1594 Vol. 2.
Teutsch, M., Kruger, W., 2010. Classification of small boats in infrared images for maritime surveillance. In: Waterside Security Conference (WSS), 2010 International. pp. 1–7.
Theodoridis, S., Koutroumbas, K., 2009. Pattern Recognition, 4th Edition. Elsevier.
Thrun, S., Burgard, W., Fox, D., 2005. Probabilistic Robotics. Intelligent robotics and autonomous agents series. Mit Press.
Turaga, P., Chellappa, R., Subrahmanian, V. S., Udrea, O., Nov 2008. Machine recognition of human activities: A survey. Circuits and Systems for Video Technology, IEEE Transactions on 18 (11), 1473–1488.
Umstatter, C., Waterhouse, A., Holland, J., 2008. An automated sensor-based method of simple behavioural classification of sheep in extensive systems. Computers and Electronics in Agriculture 64 (1), 19 – 26, smart Sensors in precision livestock farming.
URL http://www.sciencedirect.com/science/article/pii/S0168169908001464
Van Der Merwe, R., 2004. Sigma-point kalman filters for probabilistic inference in dynamic state-space models. Ph.D. thesis, University of Stellenbosch.
Vaněk, O., Bošanský, B., Jakob, M., Pěchouček, M., 2010. Transiting areas patrolled by a mobile adversary. In: Computational Intelligence and Games (CIG), 2010 IEEE Symposium on. pp. 9–16.
Vaněk, O., Jakob, M., Lisý, V., Bošanský, B., Pěchouček, M., 2011. Iterative game-theoretic route selection for hostile area transit and patrolling. In: The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 3. AAMAS '11. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, pp. 1273–1274.
Varol, A. E., Gunal, M. M., 2013. Simulation modeling of maritime piracy using discrete event and agent-based approaches. In: SIMULTECH 2013 - 3rd International Conference on Simulation and Modeling Methodologies, Technologies and Applications. pp. 438–445.
Verner, J. F., Nielsen, T. D., 2007. Bayesian Networks and Decision Graphs. Information Science and Statistics. Springer.
Walia, G. S., Kapoor, R., 2014. Intelligent video target tracking using an evolutionary particle filter based upon improved cuckoo search. Expert Systems with Applications 41 (14), 6315 – 6326.
URL http://www.sciencedirect.com/science/article/pii/S0957417414001419
Wang, L., Liu, X., Pedrycz, W., Shao, Y., 2014. Determination of temporal information granules to improve forecasting in fuzzy time series. Expert Systems with Applications 41 (6), 3134 – 3142.
URL http://www.sciencedirect.com/science/article/pii/S0957417413008634
Wang, L., Zeng, Y., Chen, T., 2015. Back propagation neural network with adaptive differential evolution algorithm for time series forecasting. Expert Systems with Applications 42 (2), 855 – 863.
URL http://www.sciencedirect.com/science/article/pii/S0957417414004941
Wang, X., Ji, Q., 2014. Context augmented dynamic bayesian networks for event recognition. Pattern Recognition Letters 43 (0), 62 – 70, {ICPR2012} Awarded Papers.
URL http://www.sciencedirect.com/science/article/pii/S0167865513002882
Wiggers, P., Mertens, B., Rothkrantz, L., 2011. Dynamic bayesian networks for situational awareness in the presence of noisy data. In: Proceedings of the 12th International Conference on Computer Systems and Technologies. CompSysTech '11. ACM, New York, NY, USA, pp. 411–416.
Wu, S.-F., Lee, S.-J., 2015. Employing local modeling in machine learning based methods for time-series prediction. Expert Systems with Applications 42 (1), 341 – 354.
URL http://www.sciencedirect.com/science/article/pii/S0957417414004394
Xu, J., Chang, H.-I., Chien, C., Kaiser, W., Pottie, G., May 2014. Context-driven, prescription-based personal activity classification: Methodology, architecture, and end-to-end implementation. Biomedical and Health Informatics, IEEE Journal of 18 (3), 1015–1025.
Yang, G., Lin, Y., Bhattacharya, P., 2010. A driver fatigue recognition model based on information fusion and dynamic bayesian network. Information Sciences 180 (10), 1942 – 1954, special Issue on Intelligent Distributed Information Systems.
Yu, T. H.-K., Huarng, K.-H., 2010. A neural network-based fuzzy time series model to improve forecasting. Expert Systems with Applications 37 (4), 3366 – 3372.
URL http://www.sciencedirect.com/science/article/pii/S0957417409008793
Zhang, Y., Ji, Q., 2006. Active and dynamic information fusion for multisensor systems with dynamic bayesian networks. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 36 (2), 467–472.
Zia-ur Rahman, Stephen E. Reichenbach, M. A. N. (Ed.), 2008. Visual surveillance in maritime port facilities. Vol. 6978.
URL http://dx.doi.org/10.1117/12.777645