

Anomaly Detection & Behavior Prediction: Higher-Level Fusion Based on Computational Neuroscientific Principles

Bradley J. Rhodes, Neil A. Bomberger, Majid Zandipour, Lauren H. Stolzar,
Denis Garagic, James R. Dankert and Michael Seibert
*BAE Systems, Advanced Information Technologies, Burlington, MA,
U.S.A.*

1. Introduction

Higher-level fusion aims to enhance situational awareness and assessment (Endsley, 1995). Enhancing the understanding analysts/operators derive from fused information is a key objective. Modern systems are capable of fusing information from multiple sensors, often using inhomogeneous modalities, into a single, coherent kinematic track picture. Although this provides a self-consistent representation of considerable data, having hundreds, or possibly thousands, of moving elements depicted on a display does not make for ease of comprehension (even with the best possible human-computer interface design). Automated assistance for operators that supports ready identification of those elements most worthy of their attention is one approach for effectively leveraging lower-level fusion products.

A straightforward, commonly employed method is to use rule-based motion analysis techniques. Pre-defined activity patterns can be detected and identified to operators. Detectable patterns range from simple trip-wire crossing or zone penetration to more sophisticated multi-element interactions, such as rendezvous. Despite having a degree of utility, rule-based methods do not provide a complete solution.

The complexity of real-world situations arises from the myriad combinations of conditions and contexts that make development of thorough, all-encompassing sets of rules impossible. Furthermore, it is also often the case that the events of interest and/or the conditions and contexts in which they are noteworthy can change at rates for which it is impractical to extend or modify large rule corpora. Also, pre-defined rules cannot assist operators interested in being able to determine whether any unusual activity is occurring in the track picture they are monitoring. Timely identification and assessment of anomalous activity within an area of interest is an increasingly important capability—one that falls under the enhanced situational awareness objective of higher-level fusion.

A precursor of being able to automatically notify operators about the presence of anomalous activity is the capability to detect deviations from normal behavior. To do this, a model of normal behavior is required. It is impractical to consider a rule-based approach for achieving such a task, so an adaptive method is required: that is, a capability to learn what is normal in a scene is required. This normalcy representation can then be used to assess new data in order to determine their degree of normalcy and provide notification when any

Source: Sensor and Data Fusion, Book edited by: Dr. ir. Nada Milisavljević,
ISBN 978-3-902613-52-3, pp. 490, February 2009, I-Tech, Vienna, Austria

activity deviates beyond some level of tolerance from the learned representation of normal behavior. Additionally, learned normalcy models can be used to predict future vessel behavior over timescales beyond the capabilities of standard track fusion/stitching engines—that is, on the order of hours to days (depending on the application domain). Learning of kinematic normalcy models for use in anomalous behavior detection and future behavior prediction are the core elements of the problem addressed in this chapter.

Numerous real-world constraints must be addressed when developing such capabilities if the results are to have any practical, fieldable utility. Key drivers for these constraints are that new track data are continually collected, that activity patterns can change over time, and that operators play no more than a limited role in guiding the evolution of the activity pattern learning system. It is not realistic to train models in batch mode where all data contributing to the learned representation have to be available prior to training onset. Periodic re-training with datasets of ever increasing size is also untenable. A static representation (one that is trained from available data then frozen for use against new data) is a suspect approach for situations where activity patterns are not static. Given the potentially huge amount of data to be processed, it is not reasonable to expect that labels indicating which tracks (or portions thereof) are normal and which are not will be applied to the data. On the other hand, sometimes relevant data are relatively scarce. So, in addition to being able to handle very large amounts of data, it is also important to be able to learn useful representations from limited amounts of data. These factors further define the problem addressed here.

Additional considerations also inform our approach. For instance, it is self-evident from any real-world situation that behavior is often contingent on ambient context. For example, travel patterns of individuals would (it is to be hoped) differ between weekdays and weekends or between daytime and nighttime. These are relatively simple contexts, but even so, they provide an important role in helping produce accurate representations of normalcy. Another example would differentiate between peak hour and non-peak hour periods when considering traffic activity patterns on a highway. During non-peak hours, stopped vehicles (or even those moving slowly) would be unusual, and thus worthy of attention (or even suspicion), whereas during peak-hours slowly moving traffic may be the norm. Some contexts are far more subtle or difficult to determine *a priori*. Consider the case of a relatively permanent change in daily travel of an individual who changes jobs. Importantly, that individual's initial visit(s) to the new job location – during the interview process, for instance – would have registered as deviations from normal workday travel patterns. If the job location were available as context data to the system, then a new model for workday travel could be learned once the individual's status had been updated. In the absence of such context information, the original model would slowly be adapted to the new pattern due to the incremental learning that takes place. Prior to the new pattern becoming mature in the model, this pattern would still be considered deviant.

To partially address this type of shortcoming, our learning approach can take advantage of externally-generated feedback about its performance to refine the learned representations. Although they do not ever need supervision to learn normalcy models, our algorithms can certainly exploit human subject matter expertise. Via reinforcement learning, operators can influence the learned models in a number of ways. For instance, regarding the last example above, if an operator determines that the new pattern of workday travel is indeed normal, then that pattern can be selected from a display, labelled as normal, and fed to the learning algorithm, which would then label clusters associated with this new pattern as normal. In effect, this speeds up the learning process on the basis of superior human insight. By the

same token, an operator could select a trajectory that the learned model considers normal and indicate that it is to be considered anomalous, whereupon the system would thenceforth consider any similar trajectory anomalous and produce corresponding notification. Another route to reinforcement learning is available via responses to anomaly detections. An operator can respond to notifications with agreement or disagreement. Disagreement indicates that the responsible behaviour is not anomalous and should be considered normal.

Neurobiological systems such as the human central nervous system are eminently suited to the challenges of such problems, so we draw inspiration for the development of automated high-level fusion support systems from computational neuroscience. Complementary, neurobiologically-inspired learning algorithms reduce massive amounts of data to a rich set of information embodied in models of behavioral patterns represented at a variety of conceptual, spatial, and temporal levels. Our approach, based on neurobiological principles, learns incrementally as new data are available, adapts learned models as underlying activity patterns change, and does not rely on labeled data for learning. Before presenting our approach in more detail, a brief survey of related work follows.

2. Related work

Beyond that from our group, the literature on trajectory-based motion learning and pattern discovery for the type of surveillance outlined in the introduction to this chapter is relatively sparse, largely due to the nature of the application. However, the more limited field of video-based surveillance (surveyed in Hu *et al.*, 2004a and Liao, 2005) has reported advances using a variety of approaches, including Learning Vector Quantization (LVQ) (Johnson & Hogg, 1996), Self-Organising Maps (SOMs) (Owens & Hunter, 2000), hidden Markov Models (HMMs) (Alon *et al.*, 2003), fuzzy neural networks (Hu *et al.*, 2004b), and batch expectation-maximization (EM) (Makris & Ellis, 2005). Most of these techniques attempt to learn high-level motion behavior patterns from sample trajectories using discrete point-based flow vectors as input to a machine learning algorithm. For realistic motion sequences, convergence of these techniques is slow and the learning phase is usually carried out offline due to the high dimensionality of the input data space. In addition, many of these algorithms use supervised and/or batch learning and require statistically sufficient amounts of data for constructing normalcy models of motion pattern behavior upon which to base anomaly detection and prediction. A noteworthy example that uses on-line clustering has been reported by Piciarelli and Foresti (2006). Alas, the dependence of their approach upon data acquisition at fixed time intervals for encoding of temporal information in their representation is a limitation that cannot generally be satisfied in real-world applications.

Our work addresses a wider range of issues relevant to real-world applicability and utility than the approaches noted above. We use incremental, unsupervised learning of non-statistical and statistical representations to deal with variable amounts of data. This produces usable normalcy models early in the learning process while data are still limited, yet refines the specificity of the models as additional data become available.

3. Event-level normalcy learning and anomaly detection

Our approach for detecting anomalous behavior is to assume that normal activity occurs frequently, while activity that is sufficiently different from normal activity is rare and

anomalous. In addition, it must be possible to incorporate explicit knowledge about normal and anomalous activity when it is available. One example is a set of vessel traffic data that has been analyzed by an operator who has verified that it contains no anomalous activity. Such input may also occur after the training data has already been presented; for example, an operator is able to select a vessel track from live data and indicate that its activity is normal. Thus, it is required that normalcy models be (1) learned continuously in response to incoming vessel track data, (2) adaptable to operator input, and (3) capable of recovering from operator mistakes.

To learn context-sensitive models of vessel behavior, we have developed a neural network classifier which incrementally constructs a multidimensional Gaussian (hyper-ellipsoid) model of each category that is relatively insensitive to outliers and learns the normal pattern of behavior independent of the feature dimensions comprising the learning hyper-space. When a new data point falls into a particular category, the network updates its parameters adaptively to the incoming data and provides an accurate measure of normal/anomalous behavior. When a new data point is sufficiently beyond all learned categories, then a new category is formed. During classification, the network reports the distance from the data point to its closest category. If this distance is not within the predefined settable threshold, then that point is reported as a deviation from normalcy. The maximum size of each category hyper-ellipsoid is also a predefined location- and dimension-dependent variable, which controls the representational fineness by constraining the size of each category. The network is capable of learning (by updating the categories and their associated hyper-ellipsoids) and classifying (by comparison to the latest hyper-ellipsoid models) data on-the-fly without any operator intervention. As each model matures, the gradient of certain model parameters reaches an asymptote that can be automatically checked for and utilized to activate models for classification purposes. The speed and performance of this learning algorithm makes it suitable for real-time situations wherein an operator/analyst can interactively facilitate the learning process and/or control over the sensitivity level of system alerting to control false alarms. These reasons also make this technology suitable for event-level learning in maritime domain awareness (MDA) or other tracking applications.

3.1 Example results

Figure 1 illustrates a two-dimensional projection of the learned representation from vessel track data recorded in the Miami Harbor vicinity during August 2004. Each category is represented by an ellipse, which accounts for 99% (3 standard deviations around the mean) of the data within that category. One aspect of this learned representation is worthy of note here. Panning from west to east (left to right) across the figure the potential locations of vessels become less constrained. In fact, in the east-most section of the region, the learned representation spans the location space. It should also be noted that the great majority of the learned category ellipses in the east-most area are uniformly pale, an indication that the pattern of travel within this area does not follow particular navigation routes. The darker ellipses indicate higher traffic areas. Figure 2 shows the learned 4-dimensional model of same model illustrated in Figure 1. Note that as vessels get closer to the port, they reduce their speed and travel in east-west direction (red-blue ellipsoids) through a narrow channel. The left panel in Figure 3 shows the percentage of track reports as a function of Mahalanobis distance to the center of closest category for a two-dimensional model (based on position: longitude and latitude) of each individual vessel. The thick black curve shows the mean

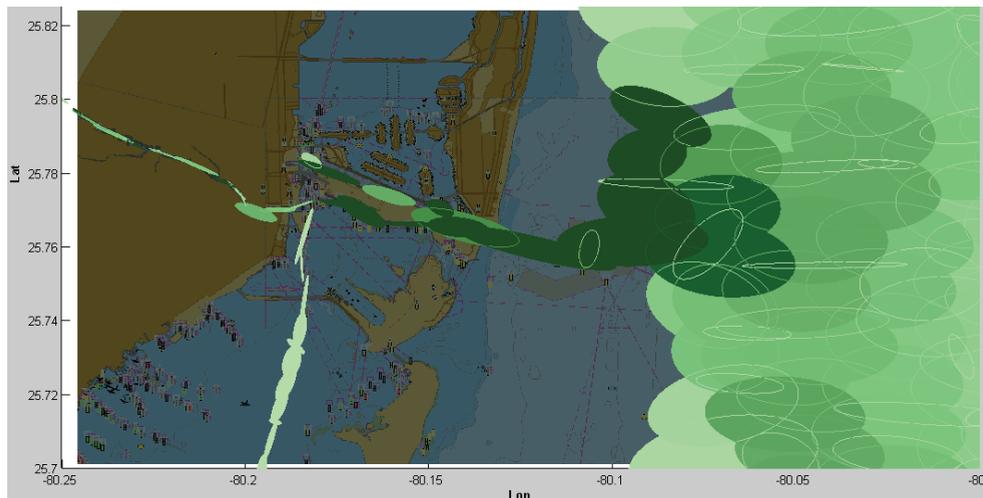


Fig. 1. Two-dimensional depiction of normalcy model learned from six months of real AIS vessel surveillance data from the Miami Harbor area (based on 4 dimensions – latitude, longitude, speed, and course). A map of the relevant region is overlaid with the learned representation of normal event activities as a set of shaded ellipses. Darker shading is proportional to the number of observations in an ellipse.

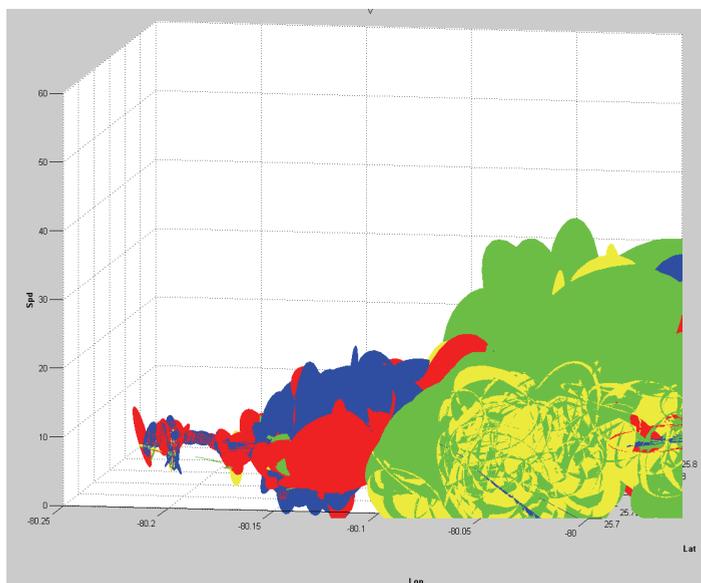


Fig. 2. Four-dimensional depiction of learned model illustrated in Figure 1. Ellipse coloring indicates principal vessel course: red = eastward, blue = westward, green = northward, yellow = southward. Towards the harbor region velocity decreases (as indicated by the lower ellipses).

across all vessels with more than 2000 track reports. The right panel in Figure 3 shows the mean of percent track reports for 2D, $3D_{\text{speed}}$ (position, and speed), $3D_{\text{course}}$ (position and course) and 4D (position, speed, and course) models of normalcy patterns. Note that all four curves approximately follow a Gaussian distribution pattern. As each category accumulates more data, the distribution of data within each category becomes closer to a Gaussian distribution.

In order to adaptively learn not only the model categories, but also the scale at which they are learned over time, we have developed an enhancement to our learning approach that applies the concept of *scale space* to our learning algorithm. This is a familiar concept in the field of computer vision, in which (Gaussian or Laplacian) image pyramids are used to efficiently represent and analyze images at multiple scales of image resolution (Burt & Adelson, 1983). In our multi-scale learning enhancement, multiple models are learned simultaneously as different model layers, with each successive layer having a scale parameter that results in a coarser scale model being learned than the model in the previous layer. This is an efficient learning representation because, while multiple model layers are learned, the coarse-resolution model layers use larger and fewer categories than the fine-resolution model layers (see Figure 4). Although multiple model layers are learned simultaneously, only one of the model layers is “active” at any given time for the purpose of detecting deviations and alerting. As learning proceeds the average category evidence in each layer is monitored, and this value is used as the criteria for switching between model layers.

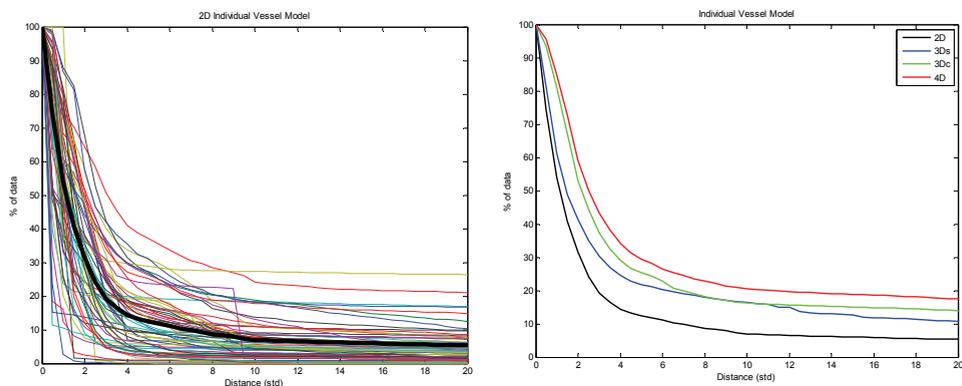


Fig. 3. Proportion of track reports beyond Mahalanobis distance from ellipse centroids as a function of ellipse standard deviations. *Left*: Individual vessel model functions; thick black line is the average function over all vessels. *Right*: Average functions for differing model dimensionality.

4. Inter-event normalcy learning and anomaly detection – behavior prediction

Learning for behavior prediction aims to predict the future position of a vessel given its current behavior (location and velocity). Essentially, this involves learning links between behavioral events. It is important that the prediction learning system operates

autonomously so as to not make demands on already busy operators. Also essential is that learning occurs incrementally in order to allow the system to take advantage of increasing amounts of data without having to take the system offline in order to batch process massive amounts of data. An additional benefit of this incremental approach is that the system will be able to adapt to changing behavior patterns automatically. For these reasons, our learning approach for this task is based on the associative learning algorithm introduced in Rhodes (2007) and extended in Bomberger *et al.* (2006), Rhodes *et al.* (2007a), and Zandipour *et al.* (2008). Weights between grid locations change via presynaptically gated Hebbian learning. The set of weights in which learning takes place is determined by the velocity state of the vessel at the start of each temporal prediction window. Learning is based on the associative learning algorithm, as described in Rhodes *et al.* (2007a):

$$\Delta w_{ijk} = \frac{1}{N_{jk}} \cdot x_{jk} \cdot (x_{ik} - w_{ijk}) \quad (1)$$

where N_{jk} is the number of times that node j has been activated in the k^{th} set of weights (which corresponds to the vessel velocity state at the beginning of the prediction interval, indexed by k), w_{ijk} is the connection weight from node j to node i , and x_{jk} and x_{ik} are the activations of grid locations j (location at the start of the period—the source location) and i (location at the end of the period—the target location) respectively. Note that the learning rate is node-dependent, such that it decreases with the amount of activity that has been encountered by a node. For a node j in the k^{th} set of weights, the learning rate first starts at a maximum of 1 and then decreases inversely with N_{jk} . Each node thus begins in a fast-learning mode, and then the weights are slowly tuned as more data is presented. Learning is presynaptically-gated by activation at the source location. If this location is not active, then no connections from this location to other locations will change their weights. If the source location is active, then links with active target locations will increase their weights and links with inactive target locations will decrease their weights. Given the binary activations used in the network, weights are bounded between 0 and 1 and the size of weight changes is governed by the learning rate and the size of the current weight. This data-dependent learning rate causes the learned weights to accurately track the conditional probabilities encountered in the training data. In contrast to neural network approaches that use batch learning to minimize a global error function with a limited set of hidden weights, this associative learning approach is both incremental and local, and each weight can be physically interpreted as part of a probability density function. The incremental and local nature of the learning process causes the model to adapt as new data is received and is less prone to convergence to local extrema since there is no global error function to be optimized. This form of learning has a number of attractive properties for the current application. First, more frequent combinations of source and target locations are rapidly learned, as indicated by larger weights. Second, random/infrequent combinations will cause learning when they occur but will also be unlearned through weight decay when they do not occur. This property also provides noise tolerance. Third, the system is able to automatically track changes in behavior over time. Fourth, the system is also able to maintain multiple sets of models for alternating operating conditions, for example, to capture seasonal differences or other factors. Fifth, the learning is entirely unsupervised, and requires no operator intervention.

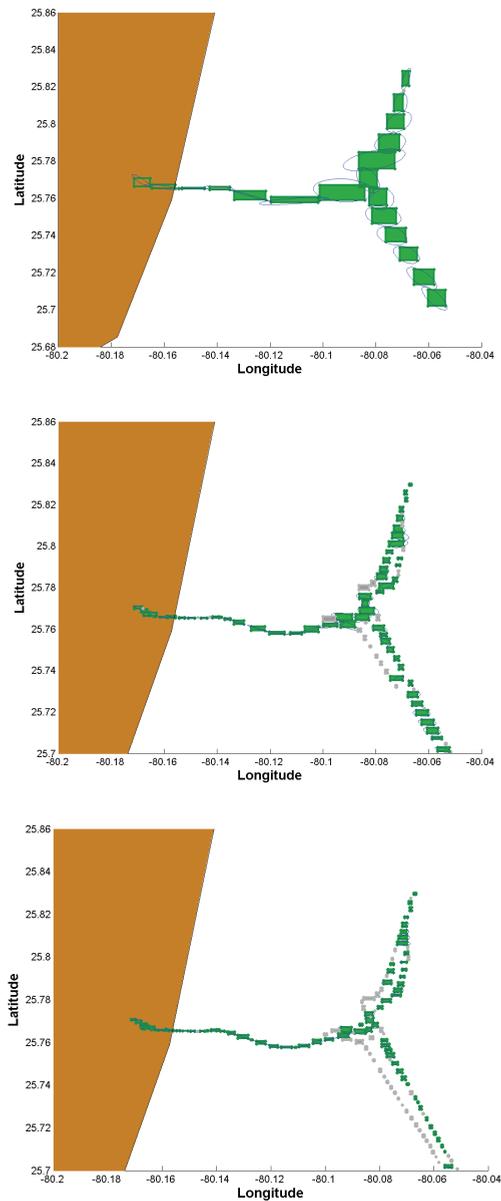


Fig. 4. As more data are received, the normalcy models fill in at the various spatial scales. Those with coarser resolution 'mature' earlier (top), but gradually those with finer resolution develop sufficiently to be used (middle, then bottom). Data-driven utilization of finer resolution models serves to maintain detection sensitivity (while enabling rapid initial use of less precise models).

4.1 Example results

From the same recorded AIS dataset used in Section 3, we utilized vessel location (latitude and longitude) and velocity (course and speed) as the basis for demonstrating our mechanism for predicting future vessel behavior. We placed a square grid over the area of interest surrounding the port of Miami so as to discretize vessel location (see Figure 5). We also defined a discretization of vessel velocity that enables learning to be contextually specific to the behavior of the vessel. Thus for each vessel report, we were able to place the vessel in a grid location having a velocity state. For purposes of exposition, the chosen temporal prediction horizon is 15 minutes.

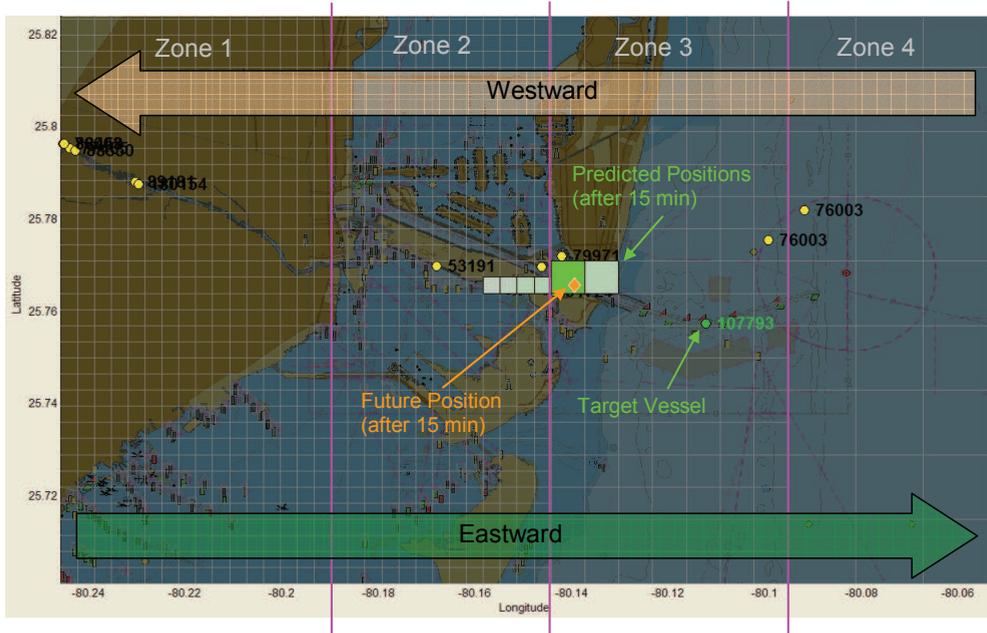


Fig. 5. Snapshot from Miami Harbor surrounds depicting system operation. The location multi-scale grid is superimposed over an ENC map of the area. Current vessel location is indicated on the map by circular markers and identification numbers. One vessel (ID 107793) has been selected for prediction display (as indicated by the larger, brighter marker). The actual future position of this vessel at the end of a 15 minute prediction horizon is indicated by the diamond. Model predictions of future location are indicated by highlighted grid locations. The strength of the weight underlying each prediction is indicated by the highlight intensity (pale=small weights; dark=large weights). Since the actual future location falls within a predicted grid location, this example represents a hit. The map is overlaid with zones that we have imposed for analysis of prediction results. Grids in zone 4 are four times larger than grids in zone 3, and 16 times larger than grids in zones 1 and 2.

To determine performance, we compared the set of grid locations (and corresponding weights) predicted by the model based on the current location and velocity state of each vessel to each vessel’s corresponding actual location 15 minutes into the future.

Each location prediction consists of a set of grid locations (the target states) and the corresponding model weights from the grid location determined by the known location and velocity of the vessel (the source state). The set of weights from the source state to the target states forms a probability density function, where the weight to each target state represents the conditional probability that it will occur in 15 minutes given that the source state has occurred.

Recall, *precision*, *accuracy*, and *coverage* statistics were calculated periodically. *Coverage* provides a measure of how well the learning has progressed in terms of being able to make predictions for all events presented to the model. *Recall* and *precision* are standard information retrieval metrics for assessing model performance. *Recall* is equivalent to P_D (probability of correct detection) and is an absolute measure of prediction accuracy. *Precision* is related to P_{FA} (probability of false alarms), which decreases as *recall* increases. *Accuracy*—as defined here—is a relative measure of prediction accuracy in that it measures the probability of correct prediction made. In contrast, *recall* factors in all events irrespective of whether a prediction was made or not. Rhodes et al. (2007a) showed that *recall* is the most relevant metric for evaluation of prediction performance. In order to generate a prediction at a requested recall level, a subset of the predicted grid locations is selected by adding predicted locations (in order from highest to lowest weight) until the sum of the weights exceeds the requested recall level.

Due to fast learning (to a weight of 1) at a node when it is first activated, *coverage* less than 1 indicates that some of the vessel states for which predictions are to be made have never been previously encountered. *Accuracy* differs from *recall* only to the extent that vessel states for which predictions are to be made have not been encountered before. The important measure is whether the predicted grid locations contain the actual future vessel location with the same probability as the recall level that is requested. That is, does the *actual recall* match the *requested recall threshold* (T_R)? Ideally, *actual recall* should always match *requested recall*. Therefore, the plot of *actual recall* vs. *requested recall threshold* should ideally produce a straight line with slope of 1 (and 0 intercept).

The *recall* vs. *requested recall threshold* (T_R) is plotted in Figure 6 for all zones and speed states, along with the *coverage*, *accuracy*, and *precision*. The solid black line (slope = 1) illustrates the *recall* performance of an ideal predictor for reference, for which the *actual recall* matches the *requested recall* level. As described earlier, *coverage* is less than 1 when vessel states for which predictions are to be made have not been encountered before, and thus is constant as T_R increases. *Accuracy* differs from *recall* only when *coverage* is less than 1. *Precision* decreases with increased T_R , having a shallow slope. The most important quantity from Figure 6 is how well *recall* matches the *requested recall* level. If the match is good, the predictions are accurate with respect to the uncertainty in the underlying data distribution, so lower precision can be tolerated.

5. Discussion

The neuro-cognitively inspired learning algorithms and representational paradigms described here have been remarkably successful in a variety of application domains. We have previously reported their use in a prototype program for port and littoral zone surveillance and automated scene understanding (Rhodes et al. 2006, 2007b). We also have

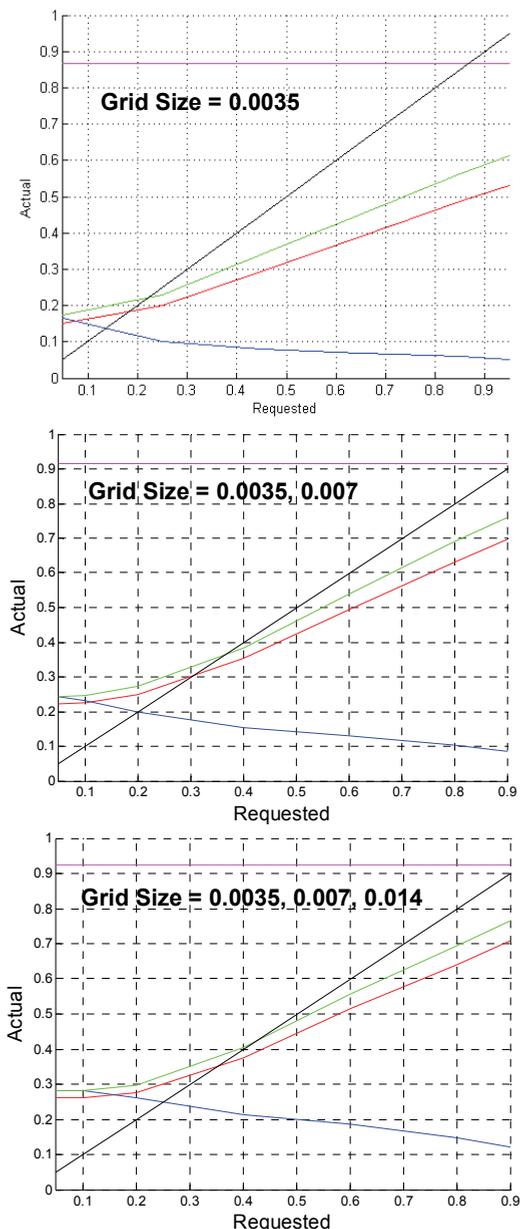


Fig. 6. Prediction results compiled across all zones. The top panel is based on a uniform grid; the middle and bottom panels are based on 2-scale and 3-scale grids respectively. The multi-scale grids had significantly better results on the metrics. *Recall* (red), *coverage* (magenta), *accuracy* (green) and *precision* (blue) are plotted vs. requested recall. The solid black line (slope=1) illustrates the recall performance of an ideal predictor for reference.

unreported success in other maritime domain awareness applications as well as land-based applications. The latter have been based on data from platforms such as surveillance towers and UAVs.

Learning-based track data analysis and exploitation as a surveillance and monitoring capability is an emerging new capability that becomes increasingly important as constraints on personnel clash with increasing needs for vigilant watchkeeping. These capabilities contribute to higher-level fusion situational awareness and assessment objectives. They also provide essential elements for automated scene understanding to shift operator focus from sensor monitoring and activity detection to assessment and response.

While having performed well in a variety of prototype level situations, our current effort represents first-generation technology. It is not yet mature enough for operational use. Each new application area produces new insight into the strengths and weaknesses of the algorithms and how they should be embedded into an overall system. Studying performance characteristics under a variety of circumstances enables the robustness and generality of the algorithmic components to be identified and enhanced. This also permits incorporation of situation specific functionality as needed to meet specific operational requirements. It is also often the case that insights gained from a new domain yield solutions that are beneficial across numerous domains.

6. Future research

Although the approaches described here have met with considerable success in a variety of domain applications, much remains to be done to produce a truly effective capability. For example, we have begun to move beyond the kinematic trajectory domain to address abnormality detection problems in other fields. Once we have multi-domain normalcy learning capabilities, it will be important to fuse across those domains in order to enhance anomaly detection. Consider, for example, a potential situation where a given activity pattern is considered normal in each of two domains judged independently, but determined to be deviant when the domains are jointly judged.

Other lines of pursuit include enhancing the flexibility of the contextually-sensitive aspect of our learning approach and refining the reinforcement learning approach used to incorporate operator feedback. In the former case, our current approach treats contexts in a discrete manner, proscribing capabilities such as mixing contexts to determine normalcy of current activity patterns or interpolating between contexts to account for previously unseen combinations of contextual conditions. As for reinforcement learning, enhancing the model refinement utility offered to operators is the key objective. Model fidelity and integrity need to be maintained while enabling user-specific insights and expertise to be incorporated via simple, intuitive interactions with the system. Moreover, potentially divergent interests of different users have to be accommodated in any tool in order for it to be useful in situations where multiple operators will be interacting with it.

7. Acknowledgements

The material presented here is based upon work supported by the AFOSR under Contract No. FA9550-06-C-0018.

8. References

- Alon, J., Sclaroff, S., Kollios, G., & Pavlovic, V. (2003). Discovering clusters in motion time-series data, *Proceedings of IEEE Computer Vision & Pattern Recognition*, Madison, WI, USA, June, 2003.
- Bomberger, N.A., Rhodes, B.J., Seibert, M., & Waxman, A.M. (2006). Associative learning of vessel motion patterns for maritime situation awareness, *Proceedings of the 9th International Conference on Information Fusion*, Florence, Italy, July, 2006.
- Burt, P.J. & Adelson, E.H. (1983). The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, COM-31, 532-540.
- Endsley, M. (1995). Towards a theory of situation awareness in dynamic systems. *Human Factors*, 37(1), 32-64.
- Hu, W., Tan, T., Wang, L., & Maybank, S. (2004a). A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, 34(3), 334-352.
- Hu, W., Xie, D., Tan, T., & Maybank, S. (2004b). Learning activity patterns using fuzzy self-organizing neural networks. *IEEE Transactions on Systems, Man & Cybernetics – Part B: Cybernetics*, 34(3), 1618-1626.
- Johnson, N. & Hogg, D. (1996). Learning the distribution of object trajectories for event recognition. *Image Vis. Computation*, 14(8), 609-615.
- Liao, T.W. (2005). Clustering of time series data – a survey. *Pattern Recognition*, 38(11), 1857-1874.
- Makris, D. & Ellis, T. (2005). Learning semantic scene models from observing activity in visual surveillance. *IEEE Transactions on Systems Man & Cybernetics – Part B: Cybernetics*, 35(3), 397-408.
- Owens, J. & Hunter, A. (2000). Application of the self-organising map to trajectory classification. *Proceedings of IEEE International Workshop Visual Surveillance*, 77-83.
- Piciarelli, C. & Foresti, G.L. (2006). On-line trajectory clustering for anomalous events detection. *Pattern Recognition Letters*, 27, 1835-1842.
- Rhodes, B.J. (2007). Taxonomic knowledge structure discovery from imagery-based data using the Neural Associative Incremental Learning (NAIL) algorithm. *Information Fusion*, 8, 295-315.
- Rhodes, B.J., Bomberger, N.A., & Zandipour, M. (2007a). Probabilistic associative learning of vessel motion patterns at multiple spatial scales for maritime situation awareness, *Proceedings of the 10th International Conference on Information Fusion*, Quebec City, Canada, July, 2007.
- Rhodes, B.J., Bomberger, N.A., Klamer, D., L'Italien, A.C., Stauffer, C., Creamer, W., Santos, B.C., & Seibert, M. (2007b). SeeCoast: Bringing automated scene understanding to US Coast Guard port surveillance. *Journal of Underwater Acoustics*.
- Rhodes, B.J., Bomberger, N.A., Seibert, M., & Waxman, A.M. (2006). SeeCoast: Automated port scene understanding facilitated by normalcy learning. In *Proceedings of IEEE MILCOM 2006 Military Communications Conference*, Washington, DC, USA, October, 2006.

Zandipour, M., Rhodes, B.J., & Bomberger, N.A. (2008). Probabilistic prediction of vessel motion at multiple spatial scales for maritime situation awareness, *Proceedings of the 11th International Conference on Information Fusion*, Cologne, Germany, June-July, 2008.



Sensor and Data Fusion

Edited by Nada Milisavljevic

ISBN 978-3-902613-52-3

Hard cover, 436 pages

Publisher I-Tech Education and Publishing

Published online 01, February, 2009

Published in print edition February, 2009

Data fusion is a research area that is growing rapidly due to the fact that it provides means for combining pieces of information coming from different sources/sensors, resulting in ameliorated overall system performance (improved decision making, increased detection capabilities, diminished number of false alarms, improved reliability in various situations at hand) with respect to separate sensors/sources. Different data fusion methods have been developed in order to optimize the overall system output in a variety of applications for which data fusion might be useful: security (humanitarian, military), medical diagnosis, environmental monitoring, remote sensing, robotics, etc.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Bradley J. Rhodes, Neil A. Bomberger, Majid Zandipour, Lauren H. Stolzar, Denis Garagic, James R. Dankert and Michael Seibert (2009). Anomaly Detection & Behavior Prediction: Higher-Level Fusion Based on Computational Neuroscientific Principles, Sensor and Data Fusion, Nada Milisavljevic (Ed.), ISBN: 978-3-902613-52-3, InTech, Available from:

http://www.intechopen.com/books/sensor_and_data_fusion/anomaly_detection__amp__behavior_prediction__higher-level_fusion_based_on_computational_neuroscienti

INTECH

open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2009 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.