In the proceedings of SPIE-09
Visual Analytics for Homeland Defense and Security, Vol 7346

Adaptive Maritime Video Surveillance

Kalyan Moy Gupta[1], David W. Aha[2], Ralph Hartley[2], Philip G. Moore[1]
[1]Knexus Research Corp; 9120 Beachway Lane; Springfield, VA 22153
[2]NCARAI; Naval Research Laboratory; 4555 Overlook Ave, SW; Washington, DC 20375
[1]*firstname.lastname*@knexusresearch.com, [2]*firstname.lastname*@nrl.navy.mil

ABSTRACT

Maritime assets such as ports, harbors, and vessels are vulnerable to a variety of near-shore threats such as small-boat attacks. Currently, such vulnerabilities are addressed predominantly by watchstanders and manual video surveillance, which is manpower intensive. Automatic maritime video surveillance techniques are being introduced to reduce manpower costs, but they have limited functionality and performance. For example, they only detect simple events such as perimeter breaches and cannot predict emerging threats. They also generate too many false alerts and cannot explain their reasoning. To overcome these limitations, we are developing the Maritime Activity Analysis Workbench (MAAW), which will be a mixed-initiative real-time maritime video surveillance tool that uses an integrated supervised machine learning approach to label independent and coordinated maritime activities. It uses the same information to predict anomalous behavior and explain its reasoning; this is an important capability for watchstander training and for collecting performance feedback. In this paper, we describe MAAW's functional architecture, which includes the following pipeline of components: (1) a video acquisition and preprocessing component that detects and tracks vessels in video images, (2) a vessel categorization and activity labeling component that uses standard and relational supervised machine learning methods to label maritime activities, and (3) an ontology-guided vessel and maritime activity annotator to enable subject matter experts (e.g., watchstanders) to provide feedback and supervision to the system. We report our findings from a preliminary system evaluation on river traffic video.

1 INTRODUCTION

US Navy assets are under a constant threat of terrorist attack as evidenced by USS Cole bombing event where 17 sailors lost their lives.[1] Merchant ships are also a convenient target for terrorists attempting to harm a nation's defenses and economy. There is a pressing need for decision support systems that improve maritime domain awareness and reduce such vulnerabilities. Some existing research systems do perform activity pattern learning and anomalous activity prediction pattern learning. However, they focus at a global level on big vessels and open ocean traffic using reliable tracking data from the automatic identification system (AIS). However, no existing video surveillance systems focus on near-shore activities. Those that are available use limited perimeter-based surveillance approaches and do not detect threat intent prior to perimeter breach. We are developing a system, called the Maritime Activity Analysis Workbench (MAAW), to fill this capability gap.

MAAW is designed to be a mixed-initiative real-time maritime video surveillance tool that uses an integrated supervised machine learning approach to label independent and coordinated maritime activities. It shall use the same information to predict anomalous behavior and explain its reasoning. MAAW includes a pipeline of adaptive processors: (1) a video acquisition and preprocessing component that detects and tracks vessels in video images, (2) a behavior analysis component that performs vessel and activity classification using standard and relational supervised machine learning techniques, and (3) a threat analysis component that shall perform mixed-initiative data fusion to assess threat and raise alerts.

We have developed a preliminary version of MAAW's video processing and behavior analysis components and report the following two novel contributions about this version. First, we represent contextual cues in a maritime scene and use them with an emerging technique called *collective case-based inference* to increase the accuracy of maritime object classification.

[1] http://en.wikipedia.org/wiki/USS_Cole_bombing

Second, we investigate the use of tandem classification where the output from an upstream classifier (e.g., object classification) is used to improve the performance of the downstream classification task (e.g., activity classification). We evaluate the effectiveness of these approaches on river traffic video data that we collected using MAAW. We found that exploiting contextual cues with collective case-based inference significantly increased vessel classification accuracy. We also show that tandem classification can significantly increase classification accuracy for low-level maritime activities and that its effectiveness can be dramatically improved by improving the performance of its upstream components (i.e., object classification).

We organize the remainder of this paper as follows. We introduce the topic of maritime domain awareness in the next section. Next, we describe MAAW's functional architecture and its component algorithms in Section 3. We evaluate our methods in Section 4 and we conclude with directions for future research in Section 6.

2 MARITIME DOMAIN AWARENESS

In an act of terrorism at the Yemini port of Aden, the bombing of the USS Cole (DDG 67) completely disabled the ship and claimed the lives of 17 sailors. The scope of such acts is global, as evidenced by a similar attack on the French oil tanker Limburg, also off the coast of Yemen.[2] Attack on military and commercial maritime assets is but one of the many possible ways to harm a nation's defenses and its economy. Others types of harmful acts include trafficking people and other resources across waterways in preparation for future attacks. Maritime domain awareness is the effective understanding of anything associated with the maritime domain that could impact the security, safety, economy, or environment of the United States [1]. It involves securing various maritime assets [2], and continuous intelligence gathering to detect, deter, and prevent terrorist acts. These efforts can occur at many levels. For example, at a global level, one could track merchant vessels and automatically detect non-routine behavior, such as unjustified rendezvous and deviation from manifest, to alert analysts about a potential threat. Some programs and efforts such as the DARPA PANDA program [3] and those in the private sector [4] address the maritime domain awareness problem at this level. In contrast, at a local level, one could monitor maritime traffic at a port or harbor to detect unusual activity to prevent a terrorist plan from its intended execution.

In this paper, we focus on maritime domain awareness at the local level. In particular, we focus on activities of small vessels in littoral areas such as bays, harbors, rivers, and channels. Our focus has its own share of problems and technical challenges that are substantially different from those at the global level. For example, small vessels can exploit a significant vulnerability in security infrastructure and operations because they are hard to detect and track using conventional surveillance methods. That is, they are not easily detected by conventional radar, they do not use AIS, and they are much more maneuverable and agile than large vessels. This vulnerability is compounded by the geographic limitations presented by the waterways in the littoral regions, where large vessels operate in a restricted maneuver mode [5]. A common but limited solution to this problem is to institute a perimeter-based surveillance approach using a combination of regular and thermal video cameras and radar sensors [6]. Perimeter-based surveillance entails detecting mobile objects such as people and vehicles and the breach of a virtual perimeter surrounding the target asset as the suspect objects move toward the target. This approach is limited in the following ways. First and foremost, it can only detect potential malicious intent when the virtual perimeter is breached. That is, it cannot evaluate intent *outside* of the perimeter. Second, a large majority of existing surveillance systems require manual monitoring of the video images from multiple sensors. This is problematic in terms of the resources needed and the potential for missed detection due to human factors such as fatigue and information overload. Recently, some systems have begun to address the second issue by using automatic video analytic approaches [7],[8]. These systems use a combination of image processing and supervised classification approaches to detect and track objects within the area of interest under a variety of conditions with impressive results at roughly 500ft. They generate alarms based on a variety of rules that specify limits on the perimeter and/or on the set of activities. Their approach significantly reduces the manual effort needed for effective video surveillance. However, the task of malicious intent detection outside the threat boundary remains unaddressed.

---

[2] http://en.wikipedia.org/wiki/Limburg_(ship)_bombing

In this paper, we address the problem of malicious intent detection before it becomes a threat. Towards this goal, we are developing an interactive and adaptive video surveillance system that includes fine-grained object categorization, activity analysis, and threat prediction. Research in machine vision is concerned with these reasoning tasks. For example, [9] presents a robust system called AVITRACK for scene understanding from video. Their system includes 24/7 video surveillance with multiple cameras. It also performs motion detection, tracking, and broad categorization of objects by exploiting temporal and spatial relationships in a scene to classify activities. Our approach is similar to theirs at the functional level. However, ours differs markedly in methodology, especially for activity analysis and threat prediction. More specifically, we perform fine-grained categorization of objects and activities over a classification hierarchy. We subcategorize overlapping vessel types instead of merely classifying vastly distinct object types such as *human* and *truck*. Second, we use a case-based reasoning (CBR) approach for supervised learning rather than a probabilistic approach (e.g., see [9]). We also use ontologies to classify objects and behaviors whose categories are hierarchically represented and spatial relations to leverage contextual cues in a scene. Finally, the design of our approach is intended to use iterative feedback between image processing and activity analysis so as to improve MAAW's reasoning and learning capabilities. Section 3 details our approach.

3 AN ADAPTIVE DECISION SUPPORT SYSTEM FOR MARITIME ACTIVITY AND ANALYSIS AND THREAT PREDICTION

Naval assets can be particularly vulnerable when they are moored or berthed in a harbor and when they are underway in a restricted maneuver mode. This is often compounded by limitations on surveillance imposed by local authorities and laws. For example, a vessel may be restricted from using radar when moored at a port. The officers and sailors charged with protecting their vessel must process an enormous amount of information while balancing the competing priorities of defending themselves and preventing engagements with innocent bystanders or friendly forces. By far, the determination of threat intention is the most difficult phase of force protection in a constrained environment. To address this need, we are developing a decision support tool to provide effective and efficient maritime situation awareness for anti-terrorism and force protection (ATFP) missions in the US Navy. MAAW, when completed, shall support and adapt to the decision-making needs of watchstanders and officers on naval vessels entering littoral regions such as harbors, bays, and ports, and in various inland channels and waterways. ATFP operations aboard a ship require continuous monitoring of suspicious activity in *assessment*, *warning*, and *threat* zones (see Figure 1). MAAW shall effectively expand the situation assessment zone based on its information processing and decision support capabilities and will acutely enhance situational awareness. Our goal is to enable detection of hostile intent much earlier than is possible with current methods.

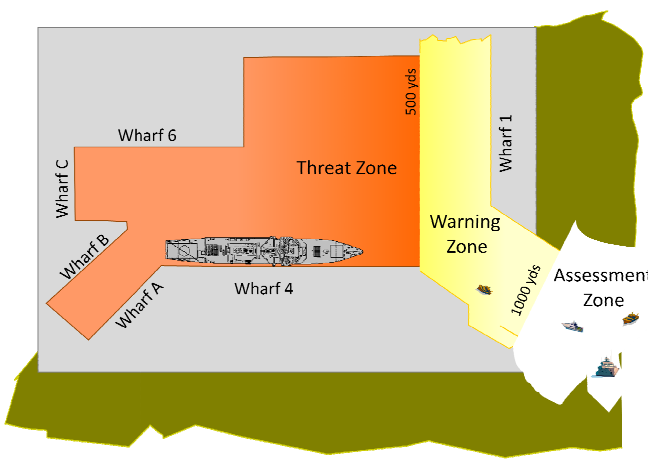

Figure 1: Threat zones about a Navy ship of interest to AT/FP operations

MAAW focuses on collecting and analyzing maritime surveillance video and fusing it with information from other sources to provide qualified threat assessments and issue alerts to the shipboard security personnel. It includes a series of adaptive processors, ranging from video acquisition to threat analysis, designed to interact with its user to issue alerts, provide threat assessments, and receive performance feedback with corrections (see Figure 2). Currently, we have implemented preliminary versions for the following components: Video Acquisition, Video Processor, Behavior Interpreter, and the Track Viewer and Annotator. We detail these components in the following subsections, and report our findings from evaluating the performance of the Behavior Interpreter in Section 4.

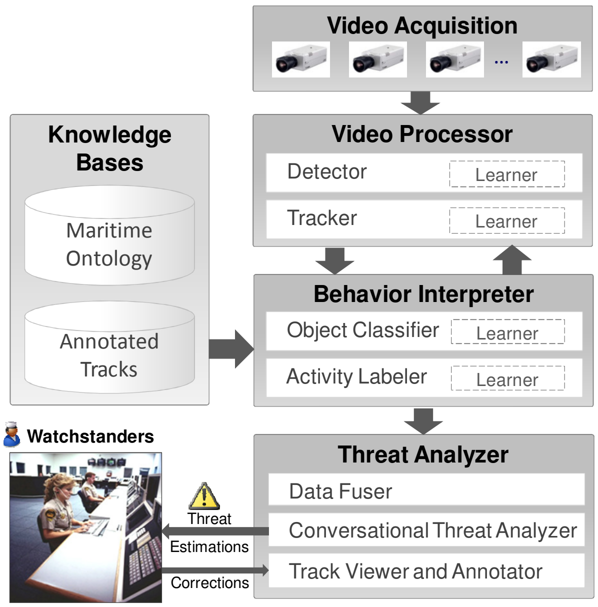

Figure 2: MAAW's Functional Architecture

3.1 Video Acquisition

MAAW's Video Acquisition component operates with fixed video cameras at a variety of resolutions. For example, using the system, we recorded maritime traffic overlooking the Potomac River in Washington, DC, and the maritime traffic in Boston's Inner Harbor from a publicly accessible web camera.[3] The Potomac River images were recorded in a 1024x1024 pixel 12 bit format at 1 second intervals. The Boston Harbor video was recorded at a lower resolution (320x240 pixels) and were collected at over 1 second intervals. Video Acquisition performs several low-level image processing operations such as image compression and cropping. For example, it cropped the images to exclude most of the sky. The region of each image sequence representing the water surface is labeled by hand. The images from the web camera contained substantial compression artifacts.

3.2 Image Processing

The main task of the Image Processor is to detect and track moving vessels, which are performed by *Detector* and the *Tracker* subcomponents respectively.

Object Detection

The Detector identifies a moving object by a process of change detection. It does this by identifying those regions in individual frames that differ significantly from the background. It constructs a Gaussian Background Model of intensity for each pixel in the scene based on its recent history (e.g., [10]). It calculates the mean intensity $\bar{I}$ and its standard deviation σ^2 of each pixel over a weighted time window. The value of the pixel in the image from time t_s contributes to the background model used at the current time t with the weight:

---

[3] http://www.seatowboston.com/harborcam.html

$$W_{t-t_s} = \mathrm{e}^{-(t-t_s)/T_{long}} - \mathrm{e}^{-(t-t_s)/T_{short}}$$

The first term allows the model to adjust to changes in conditions and the second prevents the object from being included as part of the background model. For each pixel, it computes a significance score $s^2 = (I - \bar{I})^2/\sigma^2$ and compares it with a threshold. The pixels above the threshold are filtered by a shrink and expand process and the connected components are extracted as a detected image object. Components that do not touch the water surface, or those that are too small to be boats, are discarded.

For each detection, a base point must be determined to enable positional tracking. An important criterion for choosing the base point is that when the detected image is a vessel, its base point should be near the bottom of the detected object where it contacts the water. The base point is determined as the centroid of the detected image in the horizontal direction and the lowest vertical position of the image at which the width of the region is half its maximum.

Tracking

A *track* is a collection of individual detections representing a moving object. The Tracker takes as input the image object detections from the Detector and clusters them into segments. Segments are then pieced together into a track. Each segment is made up of a set of image detections that have roughly the same velocity. The base point locations of the detected images are converted into world coordinates by projecting them onto the horizontal surface of the water as follows:

$$X_w = x_i e/(y_i - h) \qquad Y_w = fe/(y_i - h)$$

where (X_w, Y_w) is the world position, (x_i, y_i) is the base point in the image (rotated to correct for camera roll if any), f is the focal length of the camera (in pixels), e is the elevation of the camera above the water (in meters), and h is the vertical position of the horizon in the image. Linear functions of time are fit to the base point locations using least squares estimation. The errors in the detected positions are in image space, and are related to changes in position in the world by a non-linear function. To take this into account, the error of each point is weighted by a matrix obtained by linearizing the detected position. The detected images are assigned to segments by minimizing the total cost of all the segments by simulated annealing, a heuristic approach to optimization. We include the total position error of the segments, the number of segments, the number of detections not assigned to any segment, the number of "gaps" in each segment (images in which the segment should have had a detection but didn't), and the variance in computed height of each segment in the cost function for simulated annealing.

Segments are combined into complete tracks by stringing together segments that match at their start and end points. The assignment of detections to segments within each track are further tuned using the constraint that segments in a track must be disjoint in time; that is, the last detection of one segment must precede the first detection in the next.

A major source of tracking error is that the base point of each detection does not always represent the same point on a vessel. The location of the base point is subject to noise from several sources. For example, the vessel may only be partially detected, occluded, or combined with part of its wake. In future work, image matching will be used to improve the alignment of the detections.

3.3 Behavior Interpretation

The Behavior Interpreter's function is to take as input the track information extracted by the Video Processor and classify the objects in the track and its activity. A track is represented as a series of segments or *events*, each referring to a maritime object and its attributes. The *Object Classifier* and the *Activity Labeler* are the two components within the Behavior Interpreter that perform object and activity classification, respectively.

At an abstract level, our classifiers are functions that take a vector of attributes as input and predict a label for the object represented by the input vector. For example, the Object Classifier in MAAW takes a vector containing features of an object such as its position, speed, and image signature and predicts the label of that object. Such a classification function can be manually developed (e.g., one that uses hand-crafted decision rules). However, a more robust approach is to induce a classifier from the observed data, also called the *training data*, which includes the actual classification labels for the object. This approach, called *supervised learning*, has obvious advantages over a manually developed classifier. For example, if the conditions of the domain change, then a new classifier can be induced by adding new labeled data and re-running the learning algorithm. More specifically, changing the AT/FP location (e.g., a different port or harbor) could completely change the set of objects and activities of interest. New training data representing this change could be gathered and the Object Classifier could be retrained to address this change in the decision environment. Furthermore, when a classifier is used for supporting operations, the users can continue to provide feedback and correct mislabeled objects. This feedback can then be used to further increase a classifier's accuracy. Numerous supervised learning methods for inducing classifiers exist. Commonly used approaches include support vector machines (SVM), the Naïve Bayes classifier, and case-based (e.g., k-nearest neighbor) classifiers. In MAAW, we currently use case-based classifiers for all classification tasks. We briefly overview their application in MAAW classification tasks later in this section. In our future work, we will explore additional methods.

Although supervised learning approaches are more convenient to use than a manual development process, they require *labeled* observation data, which itself must be acquired manually. This can be expensive depending on the nature of the classification task, the domain, and the desired classification accuracy. Case-based methods have a potential advantage in this regard; they are simple to implement and explain, and can perform comparably to more complex classifiers (e.g., SVMs) with relatively few examples. Given that our project is in the initial phases and that we have a relatively small set of labeled data, we chose case-based methods for MAAW's classification tasks.

To classify a new *problem case* (e.g., a maritime event extracted by the Video Processor composed of attributes such as speed, location, and object signature), a case-based method reuses the classifications of previously classified *cases* that are the most *similar* to the new case. This requires a database of solved cases called a *case base*. For example, MAAW's Object Classifier relies on a case base of maritime events that include the object labels generated from annotated tracks. We describe this process of annotating tracks later in this section. To assess the similarity of two cases (i.e., a problem case and a previously classified case), the classifier uses a similarity metric. For example, the Euclidean distance metric can be used to assess the similarity of the positions of two maritime objects. The cases that are the most similar to the unclassified object are called its *nearest neighbors*. The classifier retrieves the *k* nearest neighbors from the case base and uses a voting method to predict the class label of the problem case. Training the classifier for a task typically implies estimating the parameters of its similarity metric. We describe our case-based approaches to object and activity classification in Section 3.4.

3.4 Maritime Object Classification

We categorize maritime objects using a hierarchy of object categories that are encoded in a Maritime Ontology encoded using OWL.[4] For example, our hierarchy includes "Touring and Sightseeing Vessels", "Patrol Boats", and "Trash Skimmers" as category labels. We developed this hierarchy in consultation with a subject matter expert and the navigation rules handbook [5]. In the supervision phase of our application, we use categories from this ontology to label/annotate the tracks that have been detected by the Video Processor (see Figure 5).

[4] http://www.w3.org/TR/owl-features/

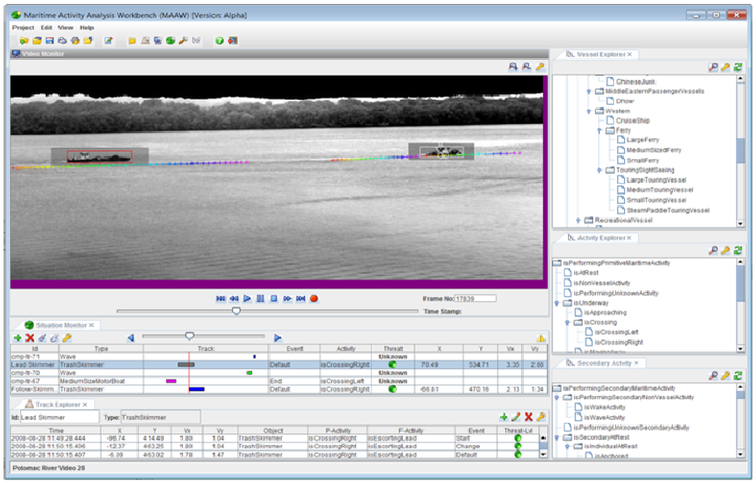

Figure 5: MAAW's user interface for annotating extracted tracks

Each event in an annotated track is then transformed into a case. A case for the object classification task is represented by the following attributes:

1. *Object position*: This represents the position of a maritime object in a two-dimensional coordinate system. It is a tuple $<p^x, p^y>$ comprising two continuous real values. The same are represented by the Tracker as x_i,y_i in World coordinates
2. *Object velocity*: This represents the velocity vector (i.e., speed and direction) of a maritime object. Like object position, the velocity vector is represented in two dimensions using a tuple $<v^x, v^y>$.
3. *Object image moments*: The Video Processor extracts images of objects from a scene including its shape, which it converts into a characteristic shape signature. Shape signatures or moments are a commonly used technique for analysis and comparison of 2D shapes. They capture information such as orientation, size, and shape boundary [11]. We generate fourth-order moments, which is a tuple comprising 15 real continuous values $<m^0 \ldots m^{14}>$.

The maritime object classification task can be challenging because tracks extracted by the Video Processor can be noisy depending on a variety of application conditions such as the weather, time of day, the size and the number of objects, and occlusion. For example, a single object in the scene could result in multiple spurious tracks with inaccurate attribute value estimates. We explore one way to address this problem: whether taking application and scene context into account can improve classification accuracy, even when faced with noisy track data. We include the context of a maritime scene by employing the following group of *relational* attributes:

4. *Closest track object*: These three attributes encode the spatial relationship of a reference object (i.e., the object that the case represents) in a maritime scene to a related maritime object that is the closest to it. The distance between a reference object and a related object is computed based on their positions in the two-dimensional real world coordinates. The attributes comprise a tuple of three values <*roc, rod, rob*>:
   a. <u>Related object category</u> (*roc*): This is a categorical label of the related object selected from our Maritime Ontology.
   b. <u>Related object distance</u> (*rod*): This is the distance of the related object from the reference object represented by a continuous real value greater than or equal to 0. (We define our distance function below.)
   c. <u>Related object bearing</u> (*rob*): This is the angle between the velocity vector of the reference object and the position vector of a related object.

We compute similarity across these four attributes to assess the overall similarity of a new case to a stored case. We compute the overall similarity by a weighted aggregation of attribute similarities, where each attribute similarity is computed in a domain-specific manner (for details, please see [12]). Using relational attributes in a case representation can be problematic because some of the values (e.g., "roc") are initially unknown. For this purpose, we use a *collective inference* process with our case-based classifier [13]. Briefly, our collective case-based classifier first uses a conventional case-based classifier with the non-relational attributes to predict the test objects' labels. It then iteratively performs collective inference by (1) estimating the values of the relational attributes and (2) using them, along with the non-relational attributes, to re-predict the test objects' labels. In our implementation, the predictions converge quickly, and we simply use a pre-determined number of iterations (10) for this procedure. This algorithm is called the *Iterative Classification Algorithm* [14].

Activity Classification

We classify maritime activities at two levels using two separate classifiers:

1. *Primary*: This level takes the perspective of the asset under protection and identifies the basic maneuvers that a vessel can perform independently. For example, "Crossing-left", "Crossing-right", and "Approaching" are some of the activity labels at this level. These labels are organized in a hierarchy of primitive activity types and are encoded in the Maritime Ontology.
2. *Secondary*. This takes an asset independent view and considers activities at a functional level of the activity. For example, "Cruising", "Sightseeing and Touring", and "Bunkering" are some of the labels denoting activities at this level. Like the primitive activity labels, these activities are also hierarchically organized and encoded in the Maritime Ontology.

The *Primary Activity Classifier* uses the following attributes for case representation. Like the object classifier, it uses the position and the velocity of an object as attributes. In addition, we use the "predicted object category" output by the Object Classifier as an additional attribute. Therefore, the Object Classifier and the Primary Activity Classifier must operate in *tandem* to classify primary activities. In other words, with the predicted object category as an attribute, the activity classifier must rely on the Object Classifier to provide its value and subsequently perform the activity classification.

In addition to the attributes used by the Primary Activity Classifier, the Secondary Activity Classifier uses the "predicted primary activity" as an attribute. Like the Primary Activity Classifier, it must operate in tandem with Object and Primary Activity Classifiers to predict secondary activities. We compute the similarity of predicted objects and predicted primary activities using taxonomic distance [15], the details of which we omit here due to lack of space.

The Behavior Interpreter hands off the automatically labeled tracks to the *Threat Analyzer*, which shall fuse the labeled tracks with harbor database information to assess threats and issue alerts. End users will be able to accept or reject MAAW's decisions and provide corrective feedback, which MAAW will use to update the track database.

3.5 Threat Analysis

The goal of the Threat Analyzer is to take the classification predictions from the Behavior Interpreter as its input and combine the data with additional information sources to further assess threat. For example, the Behavior Interpreter could classify a particular vessel and/or its activity into an unknown category. This would cause the Threat Analyzer to raise an alarm for the watchstander. However, the content of data bases such as a Harbor Masters Message Database, (e.g., a message about an onboard systems failure) could be used to reclassify the object or activity as non-threatening. MAAW will include a *conversational CBR* [16] component for gathering and fusing data from electronic databases and/or human operators to offer a final threat classification. Conversational CBR systems progressively gather information as needed from end-users and systems to improve their precision in case retrieval.

We have implemented a basic version of all the MAAW components except the Threat Analyzer. Next, we report our findings from evaluating these components.

4 EMPIRICAL STUDIES

4.1 Objectives

Our goals for evaluating MAAW's Behavior Interpreter address the following questions:

1. Does the use of relational attributes and collective inference improve object classification performance?
2. Does tandem primary activity classification outperform a non-tandem version?
3. Does tandem secondary activity classification outperform a non-tandem version?

4.2 Method

Data: We selected two days of video of maritime activities from the Potomac River in Washington, DC. We applied the Video Processor to this data to detect tracks of moving maritime objects and their attributes (e.g., position and velocities at different points in time). Using MAAW, we then labeled all the events in a track with appropriate object categories, primary activity labels, and secondary activity labels (see Figure 5).

Our database included 1578 cases in 23 object categories from our Maritime Ontology, with proportions ranging from 46.64% to 0.13%. The top three most populous labels were *wave* (46.64%), *small-touring-vessel* (9.76%), and *wake* (7.41%). Half the object categories (e.g., *steam-paddle-touring-vessel*) were relatively rare and occurred less than 2% of the time in our data set.

The primary activity was labeled using 6 categories from the primary activity ontology. A large majority of activities pertained to non-vessel phenomena such as waves and wakes (51.6%). The remainder were distributed between "crossing left" (24.72%), "crossing right" (21.93%), "approaching" (0.13%)" and "unknown activity" (1.57%). Likewise, the secondary activities were labeled using 14 category labels from the secondary activity ontology. The top five most populous labels in the set were "wave activity" (37.36%), "touring and sightseeing" (19.71%), "cruising" (19.52%), "non-vessel activity" (8.62%), and "wake activity" (5.64%).

Algorithms: To answer the questions we raised earlier, we implemented the 8 algorithms using the Knexus Classification Workbench (KCLAW), a Java library for classification tasks (See Table 1).

Table 1: Summary of classification algorithms evaluated in our experiments

Task	Algorithm	Description
Object Classification	OC-R	Case-based collective classifier that uses relational attributes representing the contextual cues from a maritime scene
	OC-NR	Case-based classifier that performs a context free classification by using only non-relational attributes
Primary Activity Classification	PAT	Tandem case-based classifier that uses labels predicted by the object classifier
	PAT-P	Tandem case-based classifier that assumes perfect (P) (i.e., 100% accurate) object classification predictions as input
	PA	Non-tandem case-based classifier that ignores the predicted object category attribute
Secondary Activity Classification	SAT	Tandem case-based classifier that includes inputs from the Object Classifier in its first stage and from the Primary Activity Classifier in its second stage
	SAT-P	Tandem case-based classifier that assumes perfect (P) (i.e., 100% accurate) object classification predictions and perfect primary activity classification
	SA	Simple non-tandem classifier that ignores the predicted object and the predicted primary activity attributes

Test Procedure: We used a leave-one-out cross validation (LOOCV) test procedure with some modifications. Conventional LOOCV procedures use one case from the database for testing and the remainder for training, cycling through the entire case base and averaging the results of individual tests. We cannot use this here because collective inference operates on a *graph* of related cases, and we chose to eliminate any relations between the training and test cases. Therefore, we grouped cases that refer to co-occurring tracks and events within the same track; each such grouping yields a single *fold* (i.e., each fold's cases have no relations with cases in other folds). Next, we treated each fold as a test set and the union of cases from the remaining folds as the corresponding training set (i.e., the case base). This yields 1578 cases over 177 folds; this includes 77 *relational* folds containing 1315 cases that have relational attribute values. The average number of cases per fold across the entire data set is 8.92. The average number of cases in relational folds was marginally greater (10.79). All the algorithms were applied to each test set (i.e., fold) and their classification accuracy was recorded. We analyzed the results using one-tailed paired t-tests.

4.3 Results

The results of our evaluation are summarized in Table 2. First, we compared the performance of a collective case-based classifier (OC-R) with a non-relational classifier (OC-NR) on the object classification task. We found that collective classification outperforms the non-relational classifier (56.22 % vs. 53.36%, p=0.0001). This answers our first question: the use of relational attributes and collective inference significantly increases object classification accuracy.

Next, we compared the three algorithms for primary activity classification to assess the effectiveness of tandem classification. The tandem version (PAT) of the classifier outperforms the non-tandem version (PA) (82.69% vs. 81.29, p=0.050). This provides support for our second hypothesis: Tandem primary activity classification significantly increases activity classification accuracy versus a non-tandem method.

To examine whether there is room for performance improvement, we reviewed the performance of PAT-P, an idealized version of the tandem classifier that assumes perfect object classification. Its performance is significantly higher (89.08%) compared to the non-idealized version (82.69%). This shows that the effectiveness of tandem classification can be dramatically improved by improving the performance of the Object Classifier.

Finally, we compare the three algorithms for secondary activity classification. The tandem version (SAT) attains a lower accuracy than the non-tandem version (63.19% vs. 62.83%, p=0.353), although this difference is statistically insignificant. Like the Primary Activity Classifier, we examined the potential for performance improvement by assessing the performance of an idealized version of the tandem classifier that assumes perfect object and primary activity classification. This is substantially higher (80.98% compared to 63.19%) than when using the (possibly incorrect) predicted values from the upstream classifiers. Thus, the performance of secondary classification can be substantially improved by increasing the accuracy of object and primary activity classification.

Table 2. Average classification accuracies of the eight algorithms

Object Classifier	**OC-R**	**OC-NR**	
	56.22	53.36	
Primary Activity Classifier	**PAT**	**PAT-P**	**PA**
	82.69	89.08	81.29
Secondary Activity Classifier	**SAT**	**SAT-P**	**SA**
	63.19	80.98	62.83

5 CONCLUSION

The existing surveillance infrastructure for maritime asset and force protection is vulnerable due to the lack of adequate decision support capabilities. In this paper, we reported on the development and capabilities of a system to reduce this capability gap. Our system called MAAW uses a pipeline of processors that include a Video Processor, Behavior Interpreter, and a Threat Analyzer. Together, these components shall provide a mixed-initiative threat assessment ability with the goal of improving the ability to detect malicious intent far beyond the immediate threat zone. Although, the current version of MAAW is preliminary and partially implemented, we reported on two technical contributions. First, we applied an approach to classification over relational data called *collective case-based classification* to the task of maritime object classification. We successfully exploited the elements of a maritime scene to significantly increase maritime object classification accuracy. Second, we used a novel problem representation for maritime activity classification that requires a sequence of classifiers (i.e., tandem classification). We showed that using a suitable problem representation with the tandem classification approach can significantly increase accuracy, thereby illustrating the utility of our tandem classification approach.

Like any preliminary research development effort, ours has many limitations and shall require much future work. First, we reported results using video from one location. We will consider additional locations in our future evaluations. Second, we will complete the implementation of the Threat Analyzer components and include feedback from the Behavior Interpreter to the Image Processor to investigate the potential improvement for detection and tracking. Third, we will conduct an empirical study of detection and tracking performance, which could have a significant bearing on the Behavior Interpreter. Fourth, we will consider several algorithmic improvements to the basic case-based classifier such as similarity metric weight learning and representation discovery. Finally, we will investigate the effectiveness of alternative classification methods such as support vector machines in our architecture.

ACKNOWLEDGEMENTS

We thank the Office of Naval Research and the Naval Research Laboratory for funding this research. We also thank LT Rex Trudell for providing a concept of operations for deploying MAAW.

REFERENCES

[1] MDA. “National Plan to achieve maritime domain awareness”. Washington, DC: Department of Homeland security. Retrieved on 4 March 2009 from [http://www.dhs.gov/xlibrary/assets/HSPD_MDAPlan.pdf] (2005)

[2] ISPF. “International ship and port facility security code”. Retrieved on 4 Mach 2009 from [http://en.wikipedia.org/wiki/International_Ship_and_Port_Facility_Security_Code] (2004)

[3] Moore, K. “Predictive analysis of naval deployment activities (PANDA)”. Retrieved from [http://www.darpa.mil/ipto/programs/panda/docs/PANDA_Overview.pdf] (2005)

[4] Rhodes, B.J., Bomberger, N.A., Seibert, M., & Waxman, A.M. “Maritime situation monitoring and awareness using learning mechanisms”. *Proceedings of Situation Management: Papers from the Military Communications Conf.* Atlantic City, NJ: IEEE Press (2005).

[5] NavRules. “Navigation rules, International-Inland”. Washington, DC: U.S. Department of Transportation, United States Coast Guard. ISBN 0-16-050057-5 (1999).

[6] PSR. “Perimeter surveillance radar”. Retrieved on 5 March 2009 from [http://en.wikipedia.org/wiki/Perimeter_Surveillance_Radar] (2009).

[7] Lipton, A.J., Heartwell, C.H., Haering. N., & Madden, D. “Critical asset protection, perimeter monitoring, and threat detection using automated video surveillance”. In *Proceedings of the Thirty-Sixth Annual International Carnahan Conference on Security Technology*. [www.objectvideo.com/objects/pdf/products/vew/OV_WP_IVS.pdf] (2002).

[8] OmniAlert. “The Omni Alert Perimeter Monitoring System”. Retrieved on 5 March 2009 from [http://www.remotereality.com/omnialert360-productsmenu-121/perimeter-monitoring-productsmenu-120](2009).

[9] Fusier, F., Valentin, V., Brémond, F., Thonnat, M., Borg, M., Thride, D., & Ferryman, J. “Video understanding for complex activity recognition”. *Machine Vision and Applications*, **18**,167-188 (2007).

[10] Huwer, S., & Niemann, H. “Adaptive change detection for real-time surveillance applications”. *Proceedings of the IEEE Workshop on Visual Surveillance* (pp. 37-43). Dublin, Ireland: IEEE Press (2000).

[11] Leu, J.-G. “Computing a shape’s moments from its boundary”. *Pattern Recognition,* **24**(10), 949-957 (1991).

[12] Gupta K.M., Aha, D.W., & Moore P.G. "Case-based collective inference for maritime object classification". Manuscript submitted for review (2009).

[13] McDowell, L.K., Gupta, K.M., & Aha, D.W. "Case-based collective classification". *Proceedings of the Twentieth International FLAIRS Conference*. Key West, FL: AAAI (2007).

[14] McDowell, L., Gupta, K. M., and Aha, D.W. "Cautious inference in collective classification". In *Proceedings of the Twenty-Second Conference on Artificial Intelligence* (pp. 596-601). Vancouver (BC), Canada: AAAI Press (2007).

[15] Gupta, K.M., Aha, D.W., & Sandhu, N. "Exploiting taxonomic and causal relations in conversational case retrieval". *Proceedings of the Sixth European Conference on Case-Based Reasoning* (pp. 133-147). Aberdeen, Scotland: Springer (2002).

[16] Aha, D.W., McSherry, D., & Yang, Q. "Advances in conversational case-based reasoning". *Knowledge Engineering Review*, **20**(3), 247-254 (2005).