



Computing with Words in Maritime Piracy and Attack Detection Systems

Jelena Tešić¹, Dan Tamir¹(✉), Shai Neumann², Naphtali Rishe³, and Abe Kandel⁴

¹ Texas State University, San Marcos, TX 78666, USA
{jtesic,dan.tamir}@txstate.edu

² Eastern Florida State College, Cocoa, FL 32922, USA

³ Florida International University, Miami, FL 33199, USA

⁴ University of South Florida, Tampa, FL 33620, USA

Abstract. In this paper, we propose to apply recent advances in deep learning to design and train algorithms to localize, identify, and track small maritime objects under varying conditions (e.g., a snowstorm, high glare, night), and in computing-with-words to identify threatening activities where lack of training data precludes the use of deep learning. The recent rise of maritime piracy and attacks on transportation ships has cost the global economy several billion dollars. To counter the threat, researchers have proposed agent-driven modeling to capture the dynamics of the maritime transportation system, and to score the potential of a range of piracy countermeasures. Combining information from onboard sensors and cameras with intelligence from external sources for early piracy threat detection has shown promising results but lacks real-time updates for situational context. Such systems can benefit from early warnings, such as “a boat is approaching the ship and accelerating,” “a boat is circling the ship,” or “two boats are diverging close to the ship.” Existing onboard cameras capture these activities, but there are no automated processing procedures of this type of patterns to inform the early warning system. Visual data feed is used by crew only after they have been alerted of a possible attack. Camera sensors are inexpensive but transforming the incoming video data streams into actionable items still requires expensive human processing.

Keywords: Deep learning · Artificial neural networks · Convolutional neural networks · Fuzzy logic · Computing with Words · Autonomous vehicle · Usability · Human computer interaction

1 Introduction

The rise of maritime piracy and attacks on transportation ships has posed a significant burden on the global economy [1]. Researchers have proposed agent-driven modeling to counter the threats, facilitating the potential of piracy countermeasures but lacking real-time update capability for situational context [2]. Onboard sensor information combined with intelligence from external sources proved valuable for early piracy threat detection [3]. In all scenarios, visual data feed is used by crew only after they have been alerted to

a possible attack: camera sensors are inexpensive but transforming the incoming video data streams into actionable items still requires expensive human processing.

In this paper, we propose to apply recent advances in deep learning to design and train algorithms to localize, identify, and track small maritime objects under varying conditions (e.g., a snowstorm, high glare, night). The crew can benefit from an early automated warning, e.g., “a boat is approaching the ship and accelerating” and decide on the countermeasures based on the developing scenario. Existing onboard cameras capture surrounding activities, but there is no automated processing of threatening patterns to inform early warning systems. Automated warning such as “a boat is circling the ship,” or “two boats are diverging close to the ship” can help direct piracy countermeasures more effectively. The state-of-art deep learning activity detection and recognition systems depend on large corpora of training data, which is infeasible to use in this scenario. We propose a computation-with-words approach, where any activity, e.g., “a boat is circling the ship,” is analyzed using natural language-based human reasoning.

Humans have a remarkable capability to reason, compute, and make rational decisions in the environment of imprecise, uncertain, and incomplete information. In so doing, humans employ modes of reasoning that are approximate rather than exact.

For example, consider a scenario where two friends are each renting a personal wave-runner (small ski-jet boat) and exploring the local lake together while maintaining a “safe distance¹” from each other and other boats on the lake. To this end, a computer-based system might deploy numerous high-resolution sensors for accurately measuring location, speed, velocity, and distance from other marine vehicles. The sensor data and information are generally represented via numerical data. Even this task of maintaining “safe distance” from lake traffic, which is relatively simple for most humans, requires the autonomous system to obtain large amounts of data, maintain a relatively large database/knowledgebase of general knowledge, and perform complex real-time inference. The approach of employing Computing with Numbers (CWN) is well established and is highly effective in many applications. Yet, due to the efficiency and effectiveness of the human mind, there are applications where the human approach, which involves computation using natural language reasoning and “calculations,” that is, Computing with Words (CWW), is more simple, efficient, and effective. This raises an intriguing research question concerning the computational and human-computer interaction benefits of using CWW for tasks involving Artificial Intelligence (AI).

The goal of the research reported here is to explore the utility of CWW as a tool for identifying threatening activities where lack of training data precludes the use of deep learning. The main contribution of this paper is that it should serve as an initiator of theoretical and applied research in the field of CWW and its applications in areas such as autonomous vehicles and vehicle activity detection. To the best of our knowledge, there is no published research concentrating on the disruptive concepts of using CWW for efficient computation and Human Computer Interaction (HCI) and Artificial Intelligence (AI) applications.

The rest of the paper is organized as follows. Background concepts are defined in Sect. 2. Section 3 reviews state of the art. Section 4 introduces the CWW-based Maritime

¹ The term “safe distance” has a fuzzy connotation. Nevertheless, a CWN engineer might attempt to provide a crisp and accurate definition for this term.

Activity Detection System (CWWMAD). Section 5 provides a qualitative analysis of the CWWMAD effectiveness and efficiency, as well as its usability. Finally, Sect. 6 concludes and proposes further research.

2 Background

In this section, we provide background concerning Maritime Activity Detection. The CWW theoretical and applied concepts are detailed in a previously published paper [4].

2.1 Maritime Object Localization, Identification, and Tracking Using Deep Learning

Maritime sensor imagery quality varies greatly with the weather conditions, sea surface movements, size, and distance of maritime vessels. This degrades the performance of state-of-art Deep Convolutional Neural Networks (DCNN) for identifying maritime vessels. On the other hand, compared to the consumer domain, the maritime domain lacks the abundance of alternative targets that could be incorrectly identified as maritime objects: this allows relaxing the parameter constraints learned on urban natural scenes in consumer photos, adjusting parameters of the model inference, and achieving robust performance and high average precision (AP) measure for transfer learning scenarios [5].

DCNNs trained on large corpora of labeled consumer images provide robust generalized modeling for starting and initializing a network with transferred features from almost any number of layers and boosts generalization. In our previous work [4], we have relied on this finding and expanded the consumer dataset to the maritime domain for adapting the DCNNs to reflect the target domain. We have utilized domain characteristics to refine the deep learning framework and have shown that our transfer learning strategy produces models that reliably and accurately identify sea-surface objects in overhead imagery data. Furthermore, we have demonstrated successful single-source domain adaptation from consumer and maritime data sources to maritime object recognition [5]. In this paper, we use CenterNet as a baseline [6], as it has emerged as a fast and lean deep network that produces the same quality of recognition results with reduced model size and inference time. CenterNet object detector builds on successful key-point estimation networks, finds object centers, and regresses to their size. The algorithm is simple, fast, accurate, and end-to-end differentiable without expensive non-maximum suppression (NMS) post-processing step. Every object is modeled as a single point, centered at its bounding box, and the approach skips the expensive step of an exhaustive search of the possible object locations.

Frame-to-frame target tracking can greatly improve the accuracy of the system and reduce the number of false positives in the dynamic learning schema. Towards this end, we have adapted the DeepSort algorithm for real-time object tracking. The system optimizes cosine similarity through a simple re-parametrization of the conventional Softmax classification regime [7]. At inference time, the final classification layer facilitates nearest neighbor queries on unseen individuals using the cosine similarity metric [7]. Figure 1 depicts the CenterNet domain adaptation to maritime data. The adaptation

produces high, real time, recall localization inference at a fraction of the model size. As illustrated in the figure, the system, trained on maritime data, can identify and track small objects through longer periods of occlusions.



Fig. 1. CenterNet domain adaptation to maritime data

2.2 The IPATCH Dataset

The IPATCH project data collection focuses on non-military protection measures for merchant shipping against piracy [3]. The goal of the project is to develop an on-board automated surveillance and decision support system providing early detection and classification of piracy threats and supporting the captain and crew in selecting appropriate countermeasures against piracy threats. The data collection was carried out employing the vessel ‘VN Partisan,’ where the ship was traveling at a constant speed, while fishing boats and “pirate” boats re-enacted scenarios. A subset of the IPATCH dataset collected in 2015 was released for public use as the PETS dataset [3]. The recordings, which represents a series of realistic maritime piracy scenarios, presents several challenges of object detection and tracking including the fusion of data from sensors with different modalities and sensor handover, tracking objects passing from one field of view (FoV) to another with minimal overlapping FoVs, event detection, and threat recognition. Piracy attacks on a vessel typically fall into one of five scenarios [3]. The PETS database contains only a small sample of each of the listed scenarios.

2.3 Activity Recognition

Recent two-stream DCNNs have made significant progress in recognizing human actions in videos. Despite their success, methods extending the basic two-stream DCNN have not systematically explored possible network architectures to utilize spatiotemporal dynamics within video sequences further. Furthermore, activity recognition in maritime videos lacks sufficient annotations. The annotated activities are categorized into seven groups:

- **Speeding:** Sudden acceleration of the mobile object.
- **Loitering:** The detected object stands still or moves slowly in the same area.
- **Group formation:** A mobile object comes close to another and holds an interaction.

- **Group Separation:** A mobile object departs from a group.
- **Moving Around (Circling):** A boat is moving and has appeared in two or more sides of the ship.
- **Sudden Direction Change:** sudden change of the trajectory.

2.4 Computing with Words, Human-Computer Interaction, and Usability

The theoretical and to some degree the applied background of the present paper is described in detail in [4] “Computing with Words—a Framework for Human-Computer Interaction,” authored by a subset of the present authors with a contribution by Lotfi Zadeh. A further in-depth review of the background can be found reference [4] citations.

3 Related Work

The DCNN framework is designed so that the individual neurons respond to overlapping regions in the visual field [8]. DCNNs for object recognition in images consist of multiple layers of small neuron collections, inspecting small portions of the input image, called receptive fields. The results of these collections are then tiled so that they overlap to obtain a better representation of the original image; this is repeated for every layer. One major advantage of DCNNs is the use of shared weights in convolutional layers, which means that the same filter (weights bank) is used for each pixel in the layer; this reduces the required memory size and improves performance. The performance of the DCNNs in the ImageNet Large Scale recognition challenge has approached the capabilities of human recognition [9]. When it comes to noisy imagery, however, the processes humans use to identify a specific target are largely unknown. Nevertheless, recent advancements in DCNN research have changed the expectations from an image and video understanding system, significantly raising the bar. DCNNs, however, continue to exhibit shortcomings, which has spurred great activity in the research community but with limited its effectiveness in real-life situations. Due to a large number of network parameters that have to be trained on, every DCNN system requires a significant number of labeled training samples to perform well. Pascal VOC, COCO, and ImageNet [9], and benchmarks motivated a breakthroughs in the field as training samples were collected via a well-executed and expensive crowd sourcing endeavor to label millions of object instances in imagery created by consumers using their hand-held devices. To achieve similar performance in other domains, one has to consider the replication of these process at comparable scale, and this is prohibitive when it comes to the periscope imagery domain, where crowd sourcing effort or labeling uniformity to achieve comparable benchmark at such a large scale are not available. Recent advances in DCNN development for object recognition have demonstrated that one can apply high-capacity DCNNs to bottom-up region identification in order to localize and segment objects, and when labeled training data is scarce, supervised pre-training for an auxiliary task, followed by domain-specific fine-tuning, yields a significant performance boost [8, 10–12].

Transfer learning focuses on storing knowledge gained while solving one problem and applying it to a different but related problem and has gained traction in domain adaption problem in computer vision. In the domain adaption problem, we focus on

utilizing multiple existing source data to build a model that performs well on different but related dataset. For tasks where sufficient number of training samples is not available, a DCNN trained on a large dataset for a different task is tuned to the current task by making necessary modifications to the network and retraining it with the available data [13, 14]. Lately, multiple groups proposed a one-shot learning approach for deep learning setup, and demonstrated that it is consistent with ‘normal’ methods for training deep networks on large data [13]. Domain adaptation of DCNN systems has been used to produce segmentation maps and to improve category identification when applied to satellite imagery and remote sensing [15]. It should be noted that generating synthetic data can serve as an intermediate stage for DCNN training.

Numerous publications address the topic of CWW from the theoretical point of view as well as relevant applications [16–20]. Several of these papers and patents, e.g., [16, 17] allude to the possibility of using CWW for Maritime navigation, activities, and Piracy Alert systems. Nevertheless, a thorough search for literature that is using the approach presented in this paper did not yield any relevant publications.

Several papers, cited in [4] address the topic of HCI in CWW-based system and conclude that the affinity between the way that the CWW-based system operates and natural language oriented HCI significantly improve these systems’ usability. These observations are in line with our expectations. Nevertheless, except for reference [4], we could not identify papers that specifically address the HCI of CWW-based applications such as Maritime Activity Detection, and Piracy Alert systems.

4 CWW Procedure

4.1 Background

Assume that a boat that is monitoring maritime activity is the “friendly boat,” referred to as Boat Φ , and the boats that are monitored are the “adversary boats,” referred to as the Ψ boats. In the present paper, we study the problem of identifying whether a single Ψ -boat is “circling” Boat Φ . A byproduct of our procedure is identifying whether any Ψ is “too close” to Φ .

The DCNN is fed by onboard video cameras placed according to the schematics of Fig. 2. These cameras provide incomplete and overlapping coverage of the area around Φ . Hence, at certain times, the location of an adversary boat in the image obtained by a specific camera is known, and at other times it is unknown.

“Four AXIS P1427-E Network cameras were added to the ship; three of them at the side and one at stern. The camera technical characteristics are the following: 5-megapixel resolution; 35°–109° FoV – Autofocus; Frame rate 30 fps; Digital PTZ” [3].

We are fixing a computer graphics frame (affine coordinate system and a point of reference - origin), referred to as the Cameras View Frame (CVF), where Boat Φ is assumed to be in its origin. The CVF is time dependent as is generated by applying affine transformations on the location of the adversary boat as obtained from onboard cameras using DCNN. Next, we mark the subregions of the CVF, as depicted in Fig. 3. The regions {A, B, C, D} denote proximity of the adversary boat to Boat Φ . The virtual region ‘INV’ denotes that none of the cameras identifies an adversary boat at the given CVF. To distinguish between individual frames of each camera and the view frameset

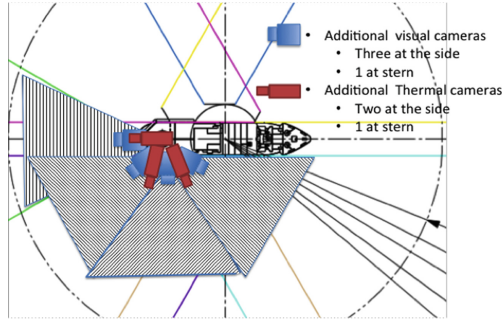


Fig. 2. Schematic representation of added cameras in the VN Partisan [3]

by the monitoring system, we refer to the camera view frames as the CVFs, while the video frames of the individual cameras are referred to as Image Frames (IFs). We assume that the cameras mounted on the boat, along with the DCNN, can provide the following information, as illustrated in the Fig. 3, at a rate of 30 CVFs per second.

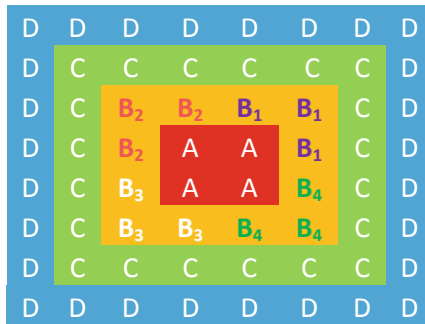


Fig. 3. Cameras view frame subregions

We further assume that at every time unit (of 1/30 of a second or longer) the system provides the following information: (1) either $(x, y, 0)$ – meaning that an adversary boat is invisible in the current frame, or (2) $(x, y, 1)$ meaning that an adversary boat is visible and is located in the affine point $(x, y, 1)^T$. In this system, $(x, y, 0)$ means a vector to the direction of (x, y) , which represents the direction at which the adversary boat was visible in the last frame. In this case, $(0,0,0)$ is the 0 vector – meaning that there is no knowledge about an adversary boat. Given the above, we can construct the following list of directions (Table 1):

Table 1. List of directions

NW	N	NE
W	ST	E
SW	S	SE

NW	N	NE
W	INV	E
SW	S	SE

For example, the direction NW means that the adversary boat is “currently” moving in the Northwest direction in the CVF, ST means that the boat is loitering, and INV means that the boat is invisible at the current CVF. Note that the smallest time unit for ‘currently’ is 1/30 of a second (the frame rate of the cameras). The largest ‘currently’ unit is arbitrary and is determined by monitoring considerations.

A “circling” activity in the context of the present paper is defined to be:

Circling: An adversary boat is moving and has appeared at two or more sides of Boat Φ . It should be noted that this activity can be detected by the dynamics of the boat location change. For example: consecutive movement through regions B_1 , B_2 , and B_3 constitutes a circling event.

The situation assessment unit, referred to as CWWMAD (MAD stands for Maritime Activity Detection), is capable of providing the crew of Boat Φ with several indications of alert levels, which can be considered as flags, where “Alarm” is the “Red Flag.” The alert levels are denoted as {invisible, watch, inspect, suspect, and Alarm}. The alert levels are determined by the current location, history of directions, and history of activities of the adversary boat. Additionally, the direction is utilized to visualize the adversary boat’s trajectory and to produce synthetic training data.

4.2 CWWMAD Inference Procedure

The CWWMAD procedure for identifying whether an adversary boat is too close Boat Φ , or is circling Φ , can be implemented using the DCNN readings denoted in Fig. 3. It is described using the following notation:

- 1) **No adversary**- If the current and the previous n readings of the adversary boats’ locations are ‘INV’, then it is assumed that no adversary boat is in proximity to Boat Φ .
- 2) **Watch** - If the current reading of an adversary boat’s location is ‘D,’ it is assumed that the adversary boat is in the ‘D’ region and might become a threat.
- 3) **Inspect** - If the current reading of an adversary boat’s location is ‘C,’ it is assumed that the adversary boat is in the ‘C’ region and requires an elevated level of monitoring.

- 4) **Suspect** - If the current reading of and an adversary boat's location is 'B₁', 'B₂', 'B₃', or 'B₄', it is assumed that the adversary boat is in one of the 'B' regions and might become a reportable threat if it crosses into the 'A' region completes a circling operation, as defined above.
- 5) **Alarm** - If the current reading of an adversary boat's location is 'A' or if the adversary boat has completed a circling activity (as defined above) then the crew must receive an alarm and undertake a response.

5 Analysis

5.1 System Verification

A simplification of the problem of detecting circling and several other maritime activities via a CWW enables system implementation using a relatively simple rule-based system. We have constructed a rule-based system RB-CWWMAD for the detection of "circling" and of "getting too close to our boat".

Due to the simplicity of the CWW problem statement, we have been able to manually verify the soundness and completeness of CWWMAD. This is significant since currently there is not enough realistic video data that can be used for automatic verification.

5.2 Usability

The following is quoted from [4]. Internal citation numbers are removed to fit the present paper.

"Tamir et al. have developed an effort-based theory and practice of measuring usability [-]. Under this theory, learnability, operability, and understandability are assumed to be inversely proportional to the effort required from the user in accomplishing an interactive task. A simple and useful measure of effort can be the time on the task. Said theory can be used to determine usability requirements, evaluate the usability of systems (including comparative evaluation of "system A vs. System B"), verify their compliance with usability requirements and standards, pinpoint usability issues, and improve usability of system versions. It is quite obvious, and supported by research literature, that an interface that uses a natural language or a formal language that are "close" to natural languages reduces the operator effort and can improve system usability [-].

A natural language interface can accompany a CWN-based as well as a CWW-based system. A CWW-based system, however, provides the advantage that the system itself operates and is being controlled in a way that is closer to human reasoning, decision making, and operation. This increases the coherence between the system and its user interface and, thus, it simplifies the system design and human-controlled operation of the system."

5.3 Additional Activities

The CWWMAD system for circling detection can be expanded to include additional activities, such as “divergence of boats” and “acceleration towards our boat.” These activities would be detected via simple extensions of the current CWW rule-based inference engine. Another activity of interest, which might be more challenging, is detecting that an adversary boat is following Boat Φ . This system would require a more complex inference engine.

6 Conclusion and Further Research

We have demonstrated that the integration of a CWW-based inference engine with a DCNN is a viable tool for detecting and providing alerts concerning several maritime activities, including boat circling. The inference can be implemented with a relatively simple rule-based system. Additionally, we have discussed the utility of a CWW-based human-computer interface (HCI) to enable ergonomic interaction with the CWW system. Analysis of the system’s performance and its usability shows that the approach has a potential for the facilitation of mitigation of maritime piracy, with an ergonomic and efficient system.

In the future, we plan to extend the activity detection capability of the system and include other maritime activities such as “following,” “diverging,” and “accelerating towards our boat.” We also plan to examine the utility of more complex inference engines that include game-theory considerations and space search-based inferencing.

We plan to obtain and annotate additional realistic video data, which would enable automatic testing of CWW, as well as training a deep-learning artificial neural network (ANN) and comparing the performance of the CWW-based inference engines to the performance of ANN-based approaches. We also plan to construct a simulation system to generate synthetic data for training deep-learning ANNs. With the addition of computer graphics capabilities to the simulation, stakeholders will be able to assess the utility and usability of maritime activity inference engines.

References

1. Marchione, E., Johnson, S.D.: Spatial, temporal and spatio-temporal patterns of maritime piracy. *J. Res. Crime Delinquency* **50**(4), 504–524 (2013)
2. Vaněk, O., Jakob, M., Hrstka, O., Pěchouček, M.: Agent-based model of maritime traffic in piracy-affected waters. *Transp. Res. Part C: Emerg. Technol.* **36**, 157–176 (2013)
3. Patino, L., Cane, T., Vallee, A., Ferryman, J.: PETS 2016: dataset and challenge. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1–8 (2016)
4. Tamir, D., Newman, S., Rische, N., Kandel, A., Zadeh, L.: Computing with words a framework for human computer interaction. In: *The International Conference on Human Computer Interaction HCI-2019*, pp. 356–372 (2019)
5. Russakovsky, O., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vision* **115**(3), 211–252 (2015). <https://doi.org/10.1007/s11263-015-0816-y>

6. Warren, N., Garrard, B., Staudt, E., Tesic, J.: Transfer learning of deep neural networks for visual collaborative maritime asset identification. In: 2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC), Philadelphia, PA, pp. 246–255 (2018)
7. Wojke, N., Bewley, A.: Deep cosine metric learning for person re-identification. In: IEEE Winter Conference on Applications of Computer Vision (WACV) (2018)
8. Pailla, D.R., Kollerathu, V., Chennamsetty, S.S.: Object detection on aerial imagery using CenterNet. [arXiv:1908.08244](https://arxiv.org/abs/1908.08244), August 2019
9. Hou, X., Wang, Y., Chau, L.: Vehicle tracking using deep SORT with low confidence track filtering. In: 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Taipei, Taiwan, pp. 1–6 (2019)
10. Feichtenhofer, C., Fan, H., Malik, J., He, K.: SlowFast networks for video recognition. In: The IEEE International Conference on Computer Vision (ICCV), pp. 6202–6211 (2019)
11. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN, International Conference on Computer Vision (ICCV), 2017, p. 5274 (2017)
12. Zhou, X., Wang, D., Krähenbühl, P.: Objects as Points, [arXiv](https://arxiv.org/abs/1904.00029), April 2019
13. Hara, K., Vemulapalli, R., Chellappa, R.: Designing Deep Convolutional Neural Networks for Continuous Object Orientation Estimation, [arXiv](https://arxiv.org/abs/1708.05122) (2017)
14. Çevikalp, H., Dordinejad, G.G., Elmas, M.: Feature extraction with convolutional neural networks for aerial image retrieval. In: 25th IEEE Signal Processing and Communications Applications Conference (SIU) (2017)
15. Maggiori, E., Tarabalka, Y., Charpiat, G., Alliez, P.: Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Trans. Geosci. Remote Sensing* **55**(2), 645–657 (2017)
16. Zadeh et al.: Method and System For Feature Detection, United States, Patent Office, US9,916,538B2 (2018)
17. Liu, J., Martínez, L., Wang, H., Rodríguez, R.M., Novozhilov, V.: Computing with words in risk assessment. *Int. J. Comput. Intell. Syst.* **3**(4), 396–419 (2010)
18. Zhou, C., Yang, Y., Jia, X.: Incorporating perception-based information in reinforcement learning using computing with words. In: Mira, J., Prieto, A. (eds.) IWANN 2001. LNCS, vol. 2085, pp. 476–483. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45723-2_57
19. Wójcik, A., Hałas, P., Pietrzykowski, Z.: Modeling communication processes in maritime transport using computing with words. *Arch. Transp. Syst. Telematics* **9**(4), 47–51 (2016)
20. Le Pors, T., Devoegele, T., Chauvin, C.: Multi agent system integrating naturalistic decision roles: application to maritime traffic. In: IADIS International Conference Intelligent Systems and Agents, pp. 196–210 (2009)