Resident Space Object Characterization and Behavior Understanding via Machine Learning and Ontology-based Bayesian Networks

Roberto Furfaro
Department of Systems and Industrial Engineering, University of Arizona
Richard Linares
Department of Aerospace Engineering & Mechanics, University of Minnesota
David Gaylor
Department of Aerospace and Mechanical Engineering, University of Arizona
Moriba Jah
College of Engineering, University of Arizona.
Ramona Walls
CyVerse, University of Arizona

ABSTRACT

In this paper, we present an end-to-end approach that employs machine learning techniques and Ontology-based Bayesian Networks (BN) to characterize the behavior of resident space objects. State-of-the-Art machine learning architectures (e.g. Extreme Learning Machines, Convolutional Deep Networks) are trained on physical models to learn the Resident Space Object (RSO) features in the vectorized energy and momentum states and parameters. The mapping from measurements to vectorized energy and momentum states and parameters enables behavior characterization via clustering in the features space and subsequent RSO classification. Additionally, Space Object Behavioral Ontologies (SOBO) are employed to define and capture the domain knowledge-base (KB) and BNs are constructed from the SOBO in a semi-automatic fashion to execute probabilistic reasoning over conclusions drawn from trained classifiers and/or directly from processed data. Such an approach enables integrating machine learning classifiers and probabilistic reasoning to support higher-level decision making for space domain awareness applications. The innovation here is to use these methods (which have enjoyed great success in other domains) in synergy so that it enables a "from data to discovery" paradigm by facilitating the linkage and fusion of large and disparate sources of information via a Big Data Science and Analytics framework.

1. INTRODUCTION

Over the past few years, Space Domain Awareness (SDA), which is concerned with acquiring and maintaining knowledge of Resident Space Objects (RSOs) orbiting Earth, has become a critical component of any space-based operation. This is mostly due to the hazards to operational satellites caused by the growing number of RSOs including orbital debris. Whereas The U.S. Space Object Catalog currently lists approximately 15,000 trackable objects, the total population is thought to exceed 20,000 objects larger than 10 cm [1]. Importantly, due to emerging capabilities of potential adversaries, threats to operational satellites are also increasing and need to be properly addressed. In order to protect valuable space assets, it is necessary to observe, understand, and predict the behavior of objects in orbit around the Earth. However, characterizing the behavior of RSOs requires a thorough understanding of the functional relationship between sensor measurements and object energy and state parameters. Such functional relationships are generally representative of the physical processes underlying the interaction between the RSO and its environment. Such processes can be modeled using a set of coupled dynamically linked models representing the evolution in time of the RSO energy and state parameters (e.g. angular momentum, propulsive, absorbed, reflected and emitted energy). There is generally no closed-form, explicit representation of the relationship between measurements and RSO energy parameters which represents a major challenge. Moreover, in the context of RSO behavior characterization, even assuming that the energy and state parameters were determined, the following questions must be answered, i.e. 1) given the knowledge of energy and state parameters for a set of RSOs, how can we identify patterns and identify physically-driven emerging behaviors? 2) How can we employ knowledge of energy and state parameters to effectively classify the RSO and provide robust inference on its behavior? Nevertheless, the behavior and classification of RSOs is intimately connected to the problem of assessing and evaluating threat warnings which include predicting intent, opportunity and capability of RSOs. The latter is possible only in presence of a robust understanding of how objects behave in space. More specifically, SDA requires sophisticated knowledge-engineering methodologies to support high-level decision-making (e.g. determine collision threat between RSOs or prevent adverse RSOs to deny and contest access to regions of the domain space). However, building and maintaining consistent knowledge-base systems for such a complex SDA application domain is generally a very difficult problem. In principle, knowledge has to be either elicited from SDA domain experts and captured in the ontological formalism or directly learned from data. In this regard, constructing a Space Object Behavior Ontology (SOBO [2]) represents an ideal solution to build a knowledge model in which the SDA domain knowledge is captured and then subsequently employed to build the knowledge-base. In SOBO, the domain knowledge is acquired via experts and explicitly captured by declaring the classes and instances and relationship between them. However, current logic-based ontology reasoners (e.g. ELK reasoner [3]) generally perform deterministic reasoning which is unsuitable for the SDA domain where uncertainty must be accounted for.

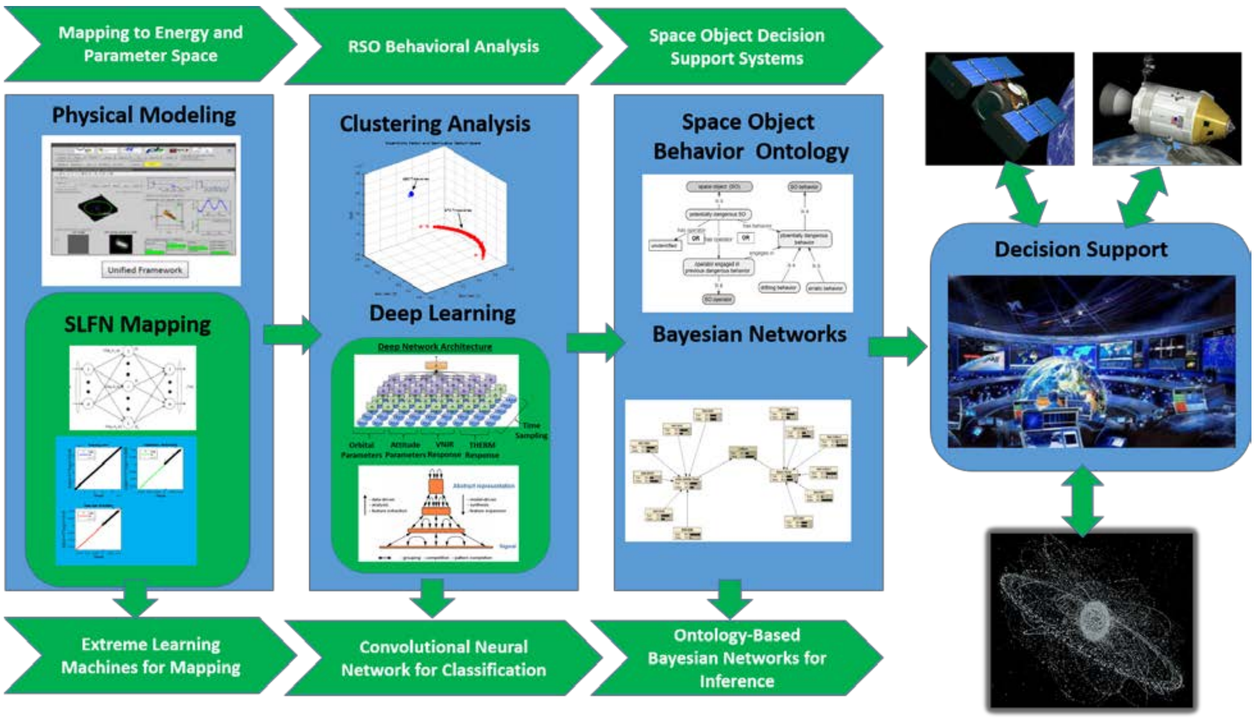


Fig. 1. Machine Learning and Ontology-based Bayesian Networks for Space Object Decision Support System: End-to-End Approach.

In this paper, we present an end-to-end approach capable of integrating machine learning regression methods and classifiers, as well as probabilistic reasoning to support higher-level decision making for space situational awareness applications. First, we develop an approach aiming at understanding and characterizing the functional relationship between sensor measurements and RSOs behavior by using a data-driven, physics-based approach. Here, simulations of coupled physical models and RSO measurements can be employed to train a set of learning machines in a supervised and unsupervised fashion to 1) characterize the functional relationship between sensor measurements and RSO energy and state parameters, 2) Analyze patterns in the energy and state parameters space to identify emerging behavior for RSOs and 3) Classify the RSOs as a function of their corresponding energy and state parameters (as well as directly from measurements). We will employ physical models and a set of state-of-the-art machine learning techniques including Extreme-Learning Machines [4] and Convolutional Neural Networks [5] to train a set of neural networks for both regression (i.e. identify the functional relationship between sensor measurements and RSO energy and state parameters), clustering (i.e. identify patterns in data and models) and classification (i.e. classify RSO using energy and state parameters and/or directly from data). Additionally, we will develop a link between the classification approach and a SOBO knowledge-base via the development of an Ontology-based Bayesian Network (BN) that can support SSA-based decision-making via probabilistic networks. The latter represents the backbone of a Space Object Decision Support System (SODSS) that is capable of inferring intent, opportunity and capability of RSOs against user-defined and/or inferred ontological knowledge and use it to predict threats in a timely and effective fashion. Fig. 1 shows a schematic of the comprehensive approach.

2. RSO ENERGY AND STATE PARAMETERS CHARACTERIZATION VIA MACHINE LEARNING

In this section, we address the first link of the proposed end-to-end approach, i.e. developing physics-based machine learning algorithms for real-time inference of RSO energy and state parameters. More specifically, we propose a model-based, data-driven approach where the unknown functional relationship between sensors measurements and RSO energy parameters and states is learned in a principled way using a set of coupled physical models that simulates the physics of space objects as function of their intrinsic properties. Such models are employed to train a new class of machine learning algorithms called Extreme Learning Machines (ELM, [4], [6]). Such algorithms exhibit an extremely fast learning rate for both regression and classification problems and provide the one of the best generalization capabilities. Importantly, the set of coupled physical models are exercised to generate the training set such that ELM can be trained in a supervised fashion. Supervised training of ELM is extremely fast and computationally efficient. Indeed, it was recently shown that learning the functional relationship between optical images and space object position around small bodies to be less than one (1) second for a training set containing about 2000 data points [7]. ELMs have the ability to learn in batch mode (learn the full training set) or sequential mode (learn in intervals or as data becomes available, [8]). Sequential learning may enable a direct coupling between physics and ELMs as data generated by the model can be employed to directly update the learned functional relationship between sensor measurements and space object energy parameters/states for fast and efficient inference.

Advantages of the proposed approach include 1) Extremely fast training phase, 2) Ability for fast updates as additional data becomes available (sequential learning), 3) Model-based approach enables the generation of training data on-demand, and 4) Fast results when deployed for near real-time (NRT) inference of energy parameters and states from measurements. An example of the proposed approach is illustrated next.

2.1 ELM-based Inference of Orbital Energy and Angular Momentum: A Case Study

Generally, the analytical relationship between ground-based measurements (e.g. angles) and the orbital parameters is unknown. However, physical models can be employed to generate data sets that represent samples of such a relationship. Here, we demonstrate how both orbital parameters (e.g. semi-major axis and eccentricity vector) and/or orbital energy and angular momentum can be directly inferred from angular measurements. For this specific case study, we consider a simplified 2-D orbital motion model where spacecraft can move in the equatorial plane. More specifically we consider a set of planar space objects either in GEO or LEO moving under the influence of the Newtonian field. The equations of motions describing the motion of each objects are:

$$\ddot{x} = -\frac{\mu x}{\|r\|^3} \tag{1}$$

$$\ddot{y} = -\frac{\mu y}{\|r\|^3} \tag{2}$$

Where μ is the Earth's gravitational constant, $\boldsymbol{r} = [x, y]^{\boldsymbol{T}}$ is the position vector assumed to be with respect to the Earth Centered Inertial (ECI) frame. The observation model is given by

$$\theta = atan2(u_x, u_y) \tag{3}$$

Where the vector $\boldsymbol{u} = \boldsymbol{r} - \boldsymbol{r}_{obs} = [u_x, u_y]^T$ represents the position of the RSO relative to the observing station $\boldsymbol{r}_{obs}$ in the ECI frame.

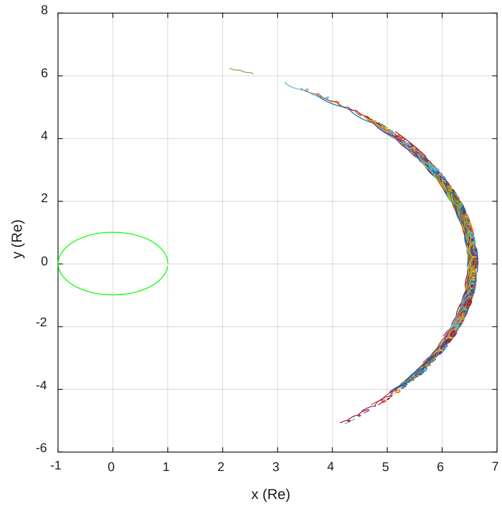


Fig. 2. Trajectories of the 1000 near-GEO objects in relative motion with respect to Earth.

A set of 1000 near-GEO (Fig. 2) and a set of 1000 GTO orbits have been simulated, for a total of 2000 training orbits. For each orbit, a set of 100 measurements have been considered. Each individual orbit is observed for 2.263 days with angular measurements taken approximately every 30 minutes. Both the semi latus rectum SLR and eccentricity vector $\boldsymbol{e} = [e_x, e_y]^T$ as well as orbital energy E and angular momentum h have been recorded to build the overall training set $\{\boldsymbol{X}, \boldsymbol{T}\}$, where $\boldsymbol{X}$ is a 100×2000 matrix and $\boldsymbol{T}$ is a 3×2000 output (2×2000 for energy and angular momentum prediction). Measurements have been corrupted with a Gaussian noise at zero mean and variance denoted by $\sigma_\theta^2 = 0.0085\ deg^2$. Two Single Layer Forward Networks (SLFN) have been designed for $(SLR, \boldsymbol{e})$ and(E, h) prediction. Both SLFNs exhibit 3000 hidden neurons and 100 inputs (i.e. 100 measurements per orbit). According to the ELM theory, the hidden weights are sampled from a random (uniform) distribution, whereas the linear weights are computed by solving a regularized least square problem $min_\beta \left\| H\hat{\beta} - \boldsymbol{T} \right\| + \alpha \left\| \hat{\beta} \right\|$, where the regularization parameter is chosen to be $\alpha = 10^{-5}$. Eighty percent of the 2000 orbits (i.e.1600) have been used for direct training and 20% (i.e. 400) have been used for the validation task. For each of the SLFNs, the training time is less than 1 sec on an Intel Core i7-4900MQ with CPU @2.8GHz and16GB of RAM. Fig. 3 shows the performance of the SLFN for $(SLR, \boldsymbol{e})$ training and validation (near-GEO case). Fig.4 shows the performance of the SLFN for (E, h) training and validation (both GTO and near-GEO). From both figures it is apparent that the SLFNs do very well on the training set but exhibit reduced performance on the validation set, yet within acceptable bounds. These results show that ELMs can learn the mapping between angle measurements and orbital/energy parameters. Performance can be potentially improved by increasing the number of training points via simulation of the physical model. Additional results and discussion can be found in [9].

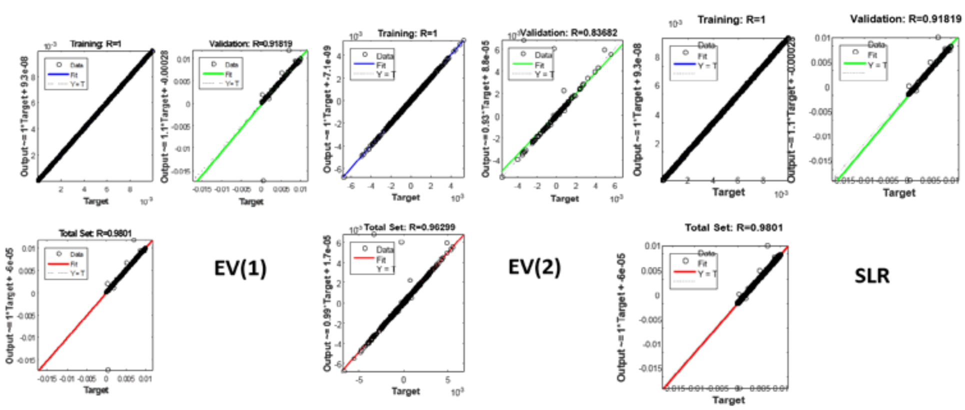


Fig. 3. Training a SLFN using ELM theories for $(SLR, \boldsymbol{e})$ prediction of GTOs (left: $\boldsymbol{e}(1)$, center: $\boldsymbol{e}(2)$, right: SRL). The SLFN learned the training set (blue line) and it is able to accurately generalize on the validation set (green line).

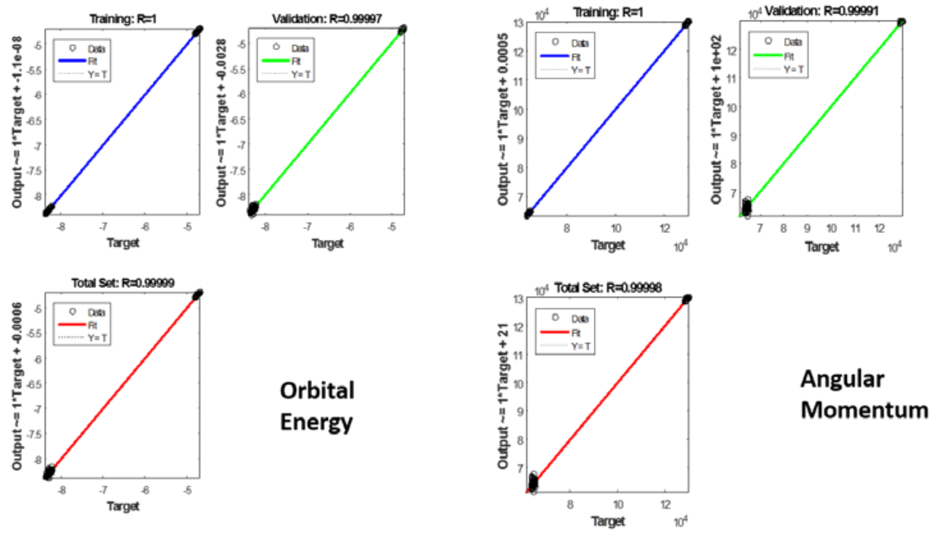


Fig. 4. Training a SLFN using ELM theories for (E, h) prediction of combined near-GEOs and GTOs (left: E, right: h). The SLFN learned the training set (blue line) and it is able to accurately generalize on the validation set (green line).

3. RSO BEHAVIOR ANALYSIS VIA CLUSTERING AND CLASSIFICATION

Model-derived data can help us to understand RSOs behavior in the energy and state parameters space. Indeed, a cluster analysis can be functional to identify patterns and similarities between objects that share similar behavior. Generally, clusters can be observed and analyzed in the energy and parameter space, i.e. once the mapping from measurements to energy and parameter space occurs, determining clusters and sub-clusters can help discover patterns of behavior using physical parameters. Such parameters may help analysts develop their intuition. Indeed, the analysis can be better understood in the physical parameter space. Fig. 5 shows the clustering of the 2000 orbits (i.e. near-GEO and GTOs) in the $(SLR, \boldsymbol{e})$- space (3D). Here, we observe a clear distinction between two major clusters representing the two categories of orbits. Importantly, a similar behavior can be observed in Fig.6 where clustering of the orbits in the (E, h)-space (2D) is visualized for analysis. In both cases, the RSOs clustering clearly separate near-GEO and GTOs. The latter may directly and visually help identify the two classes of behaviors.

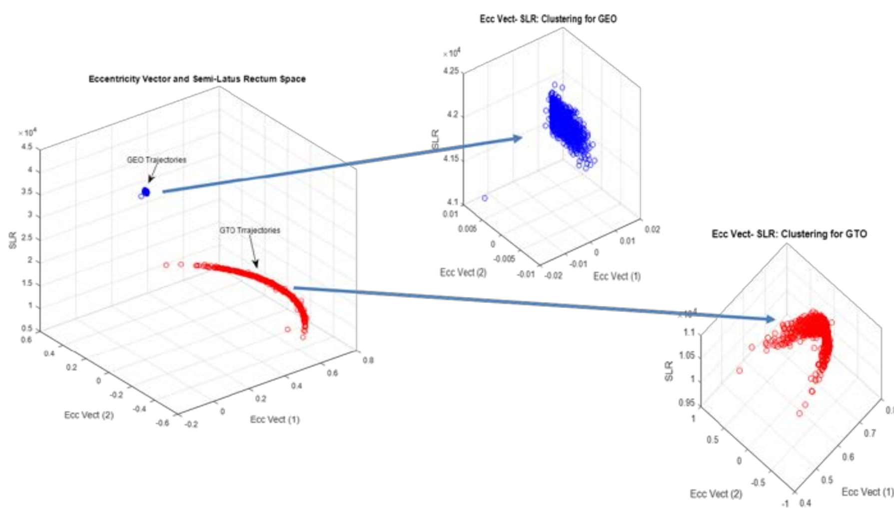


Fig. 5. Clustering of near-GEOs and GTO RSOs in the $(SLR, \boldsymbol{e})$ space. The two clusters are clearly separated and identify the distinctive orbital behavior of the two classes.

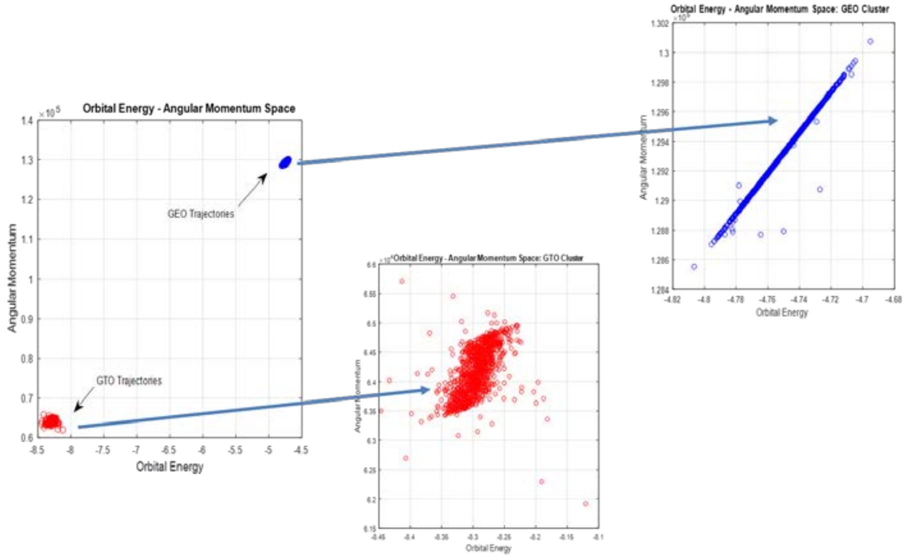


Fig. 6. Clustering of near-GEOs and GTO RSOs in the (E, h) space. Again, the two clusters are clearly separated and identify the distinctive orbital behavior of the two classes.

Note that the simple physics employed to represent the orbital dynamics (i.e. Newtonian forces only) and the distinctive physical difference between GTOs and near-GEOs RSOs contribute to the marked cluster separation. However, things may become more complicated if additional effects are modelled (e.g. aerodynamic drag, solar radiation pressure, gravitational harmonics). In such a case, multiple clusters and sub-clusters may emerge, requiring a more rigorous cluster analysis to identify new or unknown patterns. In this regard, there are many machine learning techniques that can be employed to perform a cluster analysis for pattern discovery (e.g. Fuzzy C-means [10], Self-Organizing Maps [11], Autoencoders [12]). Here, the goal is to employ such techniques to effectively perform clusters and sub-clusters analysis to group classes of objects in the energy and state parameters space. Importantly, such a step is critical to inform the supervised classification phase in a more structured and systematic manner. In the proposed approach, labeling of objects may occur after a thorough clustering analysis reveals how the classes of RSOs are distributed in the parameters space. Such analysis may inform the construction of deep networks that can be trained in a supervised fashion for final classification. Indeed, the next step is to define, train, and test physics-based deep network architectures for RSO classification using multi-temporal, multi-sensors data stream (Fig. 7).

Recent advancements in deep learning [5] have demonstrated ground breaking results across a number of domains. Here the term deep refers to any neural network learning approach with more than one hidden layer. Deep learning approaches mimic the function of the brain by learning nonlinear hierarchical features from data that build in abstraction [13]. In an end-to-end fashion a deep learning approach can be employed to classify RSOs from observational data. One important class of deep networks are the Convolutional Neural Networks (CNN). CNNs have achieved astonishing performance on general image processing tasks such as object classification [14], scene classification [15], and video classification [16]. It is well known that the key enabling factor for the success of CNN architecture is the development of techniques for large scale networks, up to tens of millions of parameters, and massive labeled datasets. Based on such recent success exhibited by the deep architectures, one can employ a physics-based approach to classify RSO with defined behavior informed by the clustering analysis.

3.1 RSO Classification via Convolutional Deep Networks: A Case Study

We present a case study, where CNNs (with max-pooling and dropout) [13] are employed for supervised classification of RSO observational data. Although the deep learning approach used here is trained with simulated data, once trained these models can be applied to real-data classification examples. As opposed to the traditional approaches discussed earlier, this work produces a new way of processing RSO observations where quick determinations of RSO classes are made possible directly from observational data. This case serves as a demonstration of the power of the proposed approach.

The dynamical system investigated here is the rotational dynamics of RSOs. The physical attributes of the RSOs, such as shape and mass distribution, are also included in the classification process. The challenging aspect of the application of CNNs to physical dynamical systems is the generation of labeled training data. As discussed earlier, we use physics-based models to simulate observations by sampling randomly from a distribution of physical attributes and dynamic states. Light curve (intensity flux over time) measurements are used as inputs, and classes of the RSOs are used as outputs for training the CNN approach. The CNN then learns convolutional kernels that look for characteristic features in the data. As opposed to manually specified features, the features are adaptively learned given the training data. The training data set consists of light curve measurement vectors as inputs and class vectors as outputs. The input light curve and output class vector are denoted by $\boldsymbol{x} \in \mathbb{R}^{1\times m}$ and $\boldsymbol{y} \in \mathbb{R}^{1\times n_c}$, respectively, where m and n_c denote the number of light curve measurements and number of classes, respectively. Then a deep neural network with convolutional layers and a fully connected output layer is trained to map from measurement vector, $\boldsymbol{x}$ to classes $\boldsymbol{y}$, using a set of training examples.

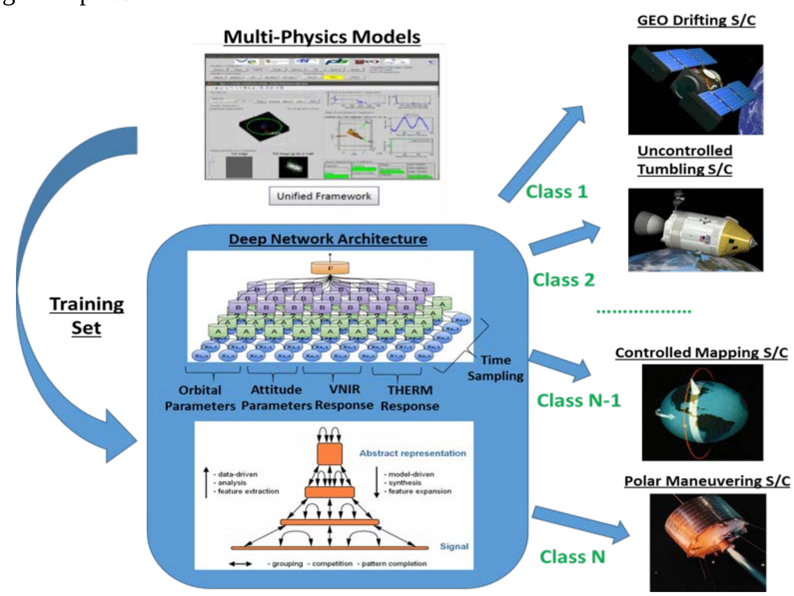


Fig. 7. Physics-based Deep Networks for RSOs classification. The Multi-physics models are employed to generate a training set. Deep Networks can be informed by the unsupervised clustering of RSOs which may help to identify the right number of classes.

Python and Tensorflow are used as the simulation environment for this work. The training of the CNN classification approach is computationally expensive, but it is expected that once trained on a larger dataset, this approach can outperform traditional methods while providing a computationally efficient classification model. The CNN used in the present work uses a four-layer structure, the layers are given by three convolution layers followed by a fully connected layer. The first three layers use a 32, 12, and 6-unit size kernel, respectively. Both max-pooling (1×4 pooling kernel) and dropout are applied after each convolutional layer. The dropout rates used for this work are 0.7 and 0.5 for the convolutional and fully connected layers respectively. For this study we only consider shape classes with one control class, but other classes can be added in the same CNN or with independent CNNs for each class. The classes considered are rocket bodies, controlled payload, uncontrolled payload, and debris.

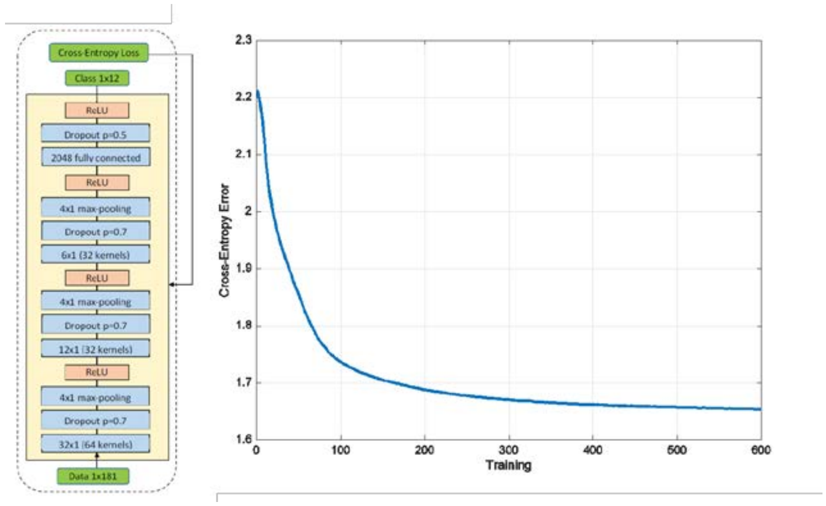
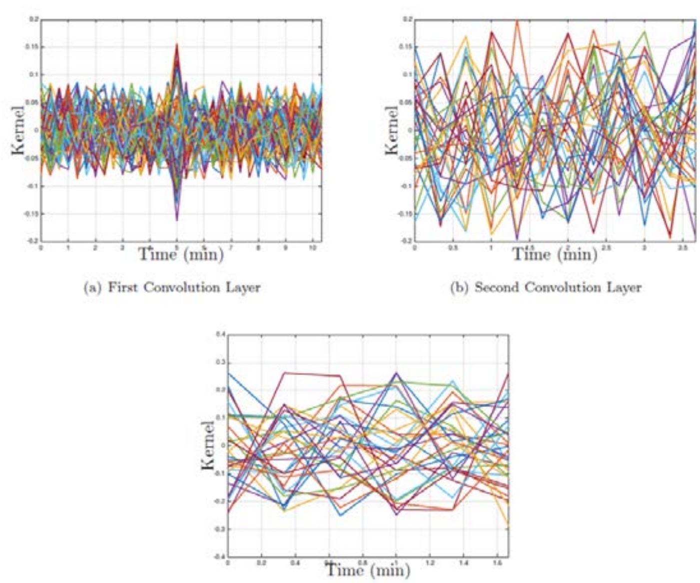


Fig. 8. CNN architecture (left) and Cross-Entropy Error (Right)

Fig. 8 shows the CNN classification kernel features estimated during the training stage. From Fig. 9a,b and c, it can be seen that the CNN approach learns light curve features that are relevant to the classification of the class considered. The first filter has the shape of a derivative type operator with large emphasis on the center location. The CNN learns that for this dataset the best low level operation is this derivative type operator. The higher layers (layer 2 and 3) have complex kernel shapes (as seen by Fig. 9b and Fig. 9c) that don't lend themselves to clear interpretations. Some noise is seen in the learned kernels but this effect can possibly be reduced with a larger training set and longer training durations, this will be looked at for future work. These filters also learn for this particular selection of classes (5 class); it may be possible to learn more general features if more classes are considered and for future work larger number of classes will be considered. The CNN approach reached an overall accuracy of 99.6% correct classification on the set of 5000 samples. Fig.8 (right) shows the cross-entropy loss as a function of gradient descent iteration. From this figure, it can be seen that for the CNN architecture used, the loss value has converged to a steady state value. Additional discussion of the case study can be found in [18].



(a) First Convolution Layer

(b) Second Convolution Layer

(c) Thrid Convolution Layer

Fig. 9. CNN Classification Results

4. RSO BEHAVIORAL INFERENCE: SPACE-OBJECT ONTOLOGY-BASED BAYESIAN NETWORKS

Although machine learning techniques are demonstrated to be powerful in classifying and potentially identifying RSOs behaviors, decision support systems for SDA must enclose structured knowledge that 1) can be incorporated by humans and 2) capable of processing both hard and soft information as well as potential results coming from machine learning classifiers. Ontologies are computing methodologies that can be employed to represent knowledge in a specified domain. The knowledge captured in by a specific ontology can be employed to reason over data in a coherent fashion. Here, we are interested in showing how Space-Object Behavioral Ontologies (SOBO [2]) can be directly translated in a BN to provide direct inference over the SDA domain for reasoning under uncertainty. More specifically, we address the problem of how to design and implement Bayesian Networks using a sharable SOBO to support the deployment of a Space Objects Decision Support System (SODSS). The overall goal demonstrate a process to semi-automatically construct BNs from ontologies and demonstrate that a SOBO-based Bayesian Network (SOBO-BN) is potentially effective in providing decision support in a variety of SDA applications.

4.1 Space Object Behavior Ontology-Based Bayesian Networks

Probabilistic networks, here referred as Bayesian Networks (BN) are graphical structures devised to represent probabilistic relationships between a large numbers of variables as well as for doing probabilistic inference between them [19]. According to the Bayesian interpretation of probability, such probability of an event X represents the degree of personal belief the event itself. In such a framework, data are employed to update, i.e. strengthen or weaken, the belief or assumption encoded in the probabilistic networks [20]. Bayesian networks can form the backbone of a decision support system for SDA [21]. Constructing BN suitable for SDA domain applications requires the following articulated steps: 1) identification of variables that are relevant to the SDA application problem (also known as nodes), 2) Identification of the relationships between the variables (also known as links) and 3) definition/creation of the CPT, which are generally employed to express how the potential states of the parent nodes affect the posterior probability of the node under consideration. Constructing a BN for SODSS may require specialized methodologies. One approach is to the automatically construct the network (concepts and links) via data elicitation [22], [23]. However, such approaches suffer from the bias problem because such BNs are generally constructed from a limited amount of available data which may be insufficient for practical application [24]. Another approach would be to elicit human expert knowledge to define the concepts involved in the problem domain and their conditional dependence. In this regard, ontologies are a potential solution to support the construction of BNs [25]. SOBO ontologies define the terms comprising the vocabulary of the SDA domain and include properties and relationships to extend such vocabulary [2]. SOBO classes, properties and individuals can be employed to capture and represent the SDA domain knowledge. As BNs employ nodes and links to represent knowledge in a probabilistic fashion, one can use the semantics of the ontologies classes, individuals and properties to generate a BNs that performs probabilistic reasoning on the SSA domain. Here, we show how to integrate the SOBO ontology with a knowledge-base for a semi-automatic construction of BNs. Importantly, the field of ontology-based construction of BNs is relatively new. Many possible approaches have been recently proposed [26], [27], [28], [29], including pursuing developing a specialized Bayesian ontology language (PR-OWL [30]) to directly model the uncertainties in the ontology. We apply a methodology that can directly translate the SOBO into a BN [31]. The latter does not rely on learning BNs from the data but leverages on the effort required to construct a well-defined SOBO from domain experts. Although, BNs have been proposed and implemented as decision support for SSA (e.g. non-cooperative GEO satellite monitoring [21]), the construction of an ontology–based BN capable of fusing hard and soft data for SSA decision support has never been proposed and implemented. We leverage our expertise in devising decision support system for space exploration [32], [33], [34], [35] and available methods for ontology-based BN [31] to construct state-of-the-art SOBO-BN that automatically reasons on hard and soft data in a probabilistic fashion.

4.2 Constructing SOBO-BN for SODSS: Satellite Collision Threat Evaluation

The semi-automatic construction of a prototype SOBO-BN for SODSS, will fundamentally require three steps, i.e. 1) determine the portion of the SOBO architecture relevant to the SSA decision problem 2) extend the SOBO to include values and weights classes (required to construct CPTs); and 3) Construct the BN graphical structure and CPTs. To illustrate the proposed methodology, we describe a simplified working example where the SOBO-BN is constructed to provide a probabilistic assessment of the collision threats between active satellites and debris.

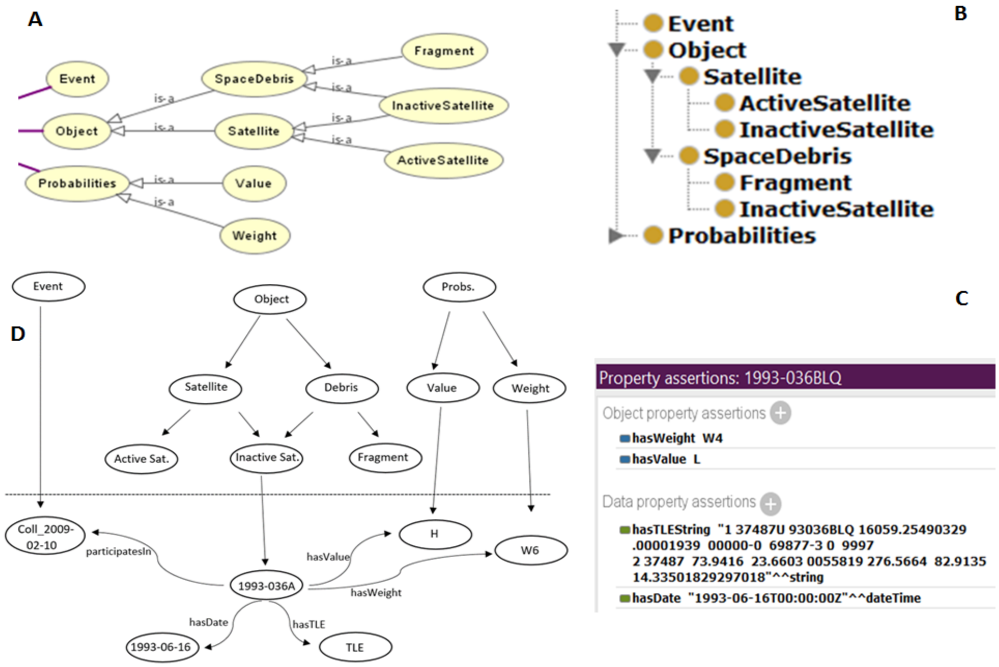


Fig. 10. A) &B) simplified SOBO for collision threats as captured in protégé management system; C) Example of properties assertions for satellite 1993-036BLQ as captured in the ontology; D) Schematic relationships between the ontology classes/individuals and the probability subclasses (Values and Weights) for CPTs constructions.

Fig. 10 shows a sample ontology that defines a simplified "satellite collision threat" problem as captured in the ontology management system (Protégé 4.1). Here, the superclass "object" has two classes: "satellite" and "debris" which are connected via the relationship "is a". Additionally, subclasses such as "active satellite", "inactive satellite" and "fragment" share the same relationship with the correspondent superclass. Individuals are defined as part of subclasses and specific properties captured in the ontology. Given the simplified SOBO, the semi-autonomous generation of SOBO-BN require extending the ontology to include classes that can be employed to compute the CPTs. More specifically, an additional class called "Probabilities" was added as well as two additional sub-classes that provide values and weights to the instances (See Fig. 10D). The "Values" subclass consists of specific chances of an instance itself being a threat based on some criteria. The purpose of the weights class is to calculate the threat probabilities of the instances' super-classes (or the parent classes). The latter specifies which instance for that particular class is more or less of a threat. Generally, the weights are chosen based on the amount of instances the classes have, and based on the risk assessment completed from the classes' properties (which were decided on the same way as the "values" class).

Conditional Probability Table for Active Satellite Threat

1993-036A	1993-036BLQ	1993-036BLR	1993-036BLV	2002-005J	2014-089C	True	False
True	True	True	True	True	True	100	0
True	True	True	True	True	False	82	18
True	True	True	True	False	True	84.5	15.5
True	True	True	True	False	False	66.5	33.5
True	True	True	False	True	True	84.5	15.5
True	True	True	False	True	False	66.5	33.5
True	True	True	False	False	True	69	31
True	True	True	False	False	False	51	49
True	True	False	True	True	True	84.5	15.5
True	True	False	True	True	False	66.5	33.5
True	True	False	True	False	True	69	31
True	True	False	True	False	False	51	49
True	True	False	False	True	True	69	31
True	True	False	False	True	False	51	49
True	True	False	False	False	True	53.5	46.5
True	True	False	False	False	False	25.5	74.5
True	False	True	True	True	True	84.5	15.5
True	False	True	True	True	False	66.5	33.5
True	False	True	True	False	True	69	31
True	False	True	True	False	False	51	49
True	False	True	False	True	True	69	31
True	False	True	False	True	False	51	49
True	False	True	False	False	True	53.5	46.5
True	False	True	False	False	False	35.5	64.5
True	False	False	True	True	True	69	31

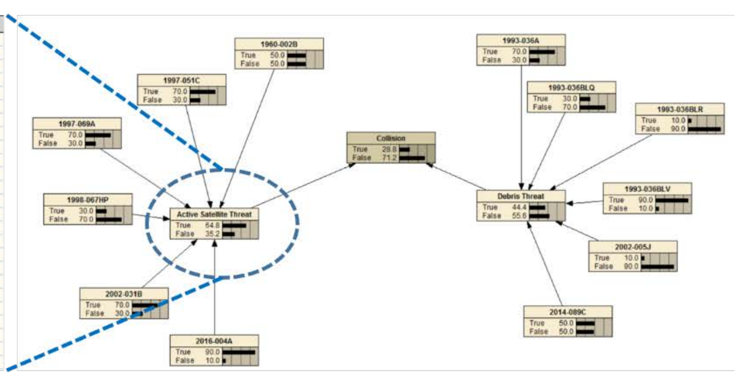


Fig. 11: Sample SOBO-BN derived from the simplified SOBO. CPT for the "Active Satellite Threat" node is reported. The CPT has been computed according to the methodology described in the text.

Since all of the required information to create a Bayesian Network was included in the extended SOBO, it is subsequently possible to auto-generate a Bayesian network by using a specialized plugin in the Netica software (NorSys Inc.). In both cases, the ontology serves as the knowledge model employed to store all of the needed classes, properties, and relationships. More specifically, to create the SOBO-BN, the nodes were first created for the Bayesian Network based on the classes from the ontology (Fig. 11). In this application, the classes range from specific instances, to the threat level of the instances' superclasses ("Active Satellites" and "Debris"), and to the final child "Collision." The proposed set-up calculates the collision probability of the objects in space, and allows the user to see if the satellites or debris classes have an impact probability. Besides the classes, properties were also imported from the extended SOBO. If the properties were not involved in calculating the collision risks, they were excluded from the quantitative aspect of the Bayesian network and included in the description of its respective class. Importantly, the properties "Value" and "Weight" were used as inputs to the Bayesian Network. "Values" was used as an observation input for each instance, while "Weight" was used for the children nodes' CPTs which were computed according to the following formula [31]:

$$P(N|X_1, \cdots, X_n) = \left(\frac{s_{X_1}}{h_{X_1}} * w_{X_1}\right) + \cdots + \left(\frac{s_{X_n}}{h_{X_n}} * w_{X_n}\right) * C\left(S_{X_1}, \cdots, S_{X_n}\right) \quad (4)$$

Where w_{X_1} is the weight of the parent node X_1 affecting the conditional dependence of children node N. Importantly, h_{X_1} describes the highest possible numerical state of the parent node X_1.
Once all of the CPTs were calculated, the result was a Bayesian network that calculated the risk for a collision. Based on the instances, values, and weights given the Bayesian network, it computed a collision risk of 28.8% (see Fig. 11), while showing the "Debris Threat" as having a risk of 44.4% due to their high speed and small size. Their impact on active satellites is reduced though, because the active satellites are provided with shields and maneuverability, which are reflected in their values and weights.

4.3 Inter-Agency Space Debris Coordination Committee (IADC) SOBO-BN: Preliminary Results

Recently, the Inter-Agency Space Debris Coordination Committee (IADC) has issued space debris mitigation guidelines [36]. The latter include a key recommendation that before mission's end, any spacecraft should move far enough from GEO so as not to be an operational hazard to other objects in active missions. Generally, it can be extremely difficult to determine if an operator or the spacecraft itself is in compliance with this guideline, as it requires prediction of future actions based upon many data types. The University of Arizona's Space Object Behavioral Sciences (SOBS) is committed to study the objects in space with the goal of assessing the current and future behavior and evaluate the impact on safety of the any object in space. To further this objective, SOBS researchers and scientists are developing an ontology-based system to support both integration, use and sharing of data for SDA applications, including integration and development of decision support systems. To demonstrate how to develop and implement a SOBO system (i.e. an application ontology), the IADC guideline for GEO end-of-life disposal was selected as use case [2]. More specifically, to show a SOBO proof of concept, the team has focused on post-mission disposal of RSO in GEO.

The methodology described on the collision threat example can be directly applied to semi-automatically construct both graphical structure and CPT of a SOBO-BN that can effectively infer if an observed object has met (or predicting it will meet) the IADC guideline. Fig.12 shows how a portion of the prototyped ontology can be directly transformed in BN. In this case, input nodes (gray nodes, e.g. orbit/attitude/historical datum) capture information from hard data (e.g. probability of the object to be controlled/uncontrolled) and soft data (e.g. probability of the system to be in an anomalous state due to web-based information posted by an operator). Such input information is propagated by the network to probabilistically infer if the object is within the IADC GEO Graveyard compliance. Challenges will include the definition of the values and weight classes that are required to construct the CPTs. Importantly, such parameters can be directly tuned as additional knowledge is acquired and captured in the ontology.

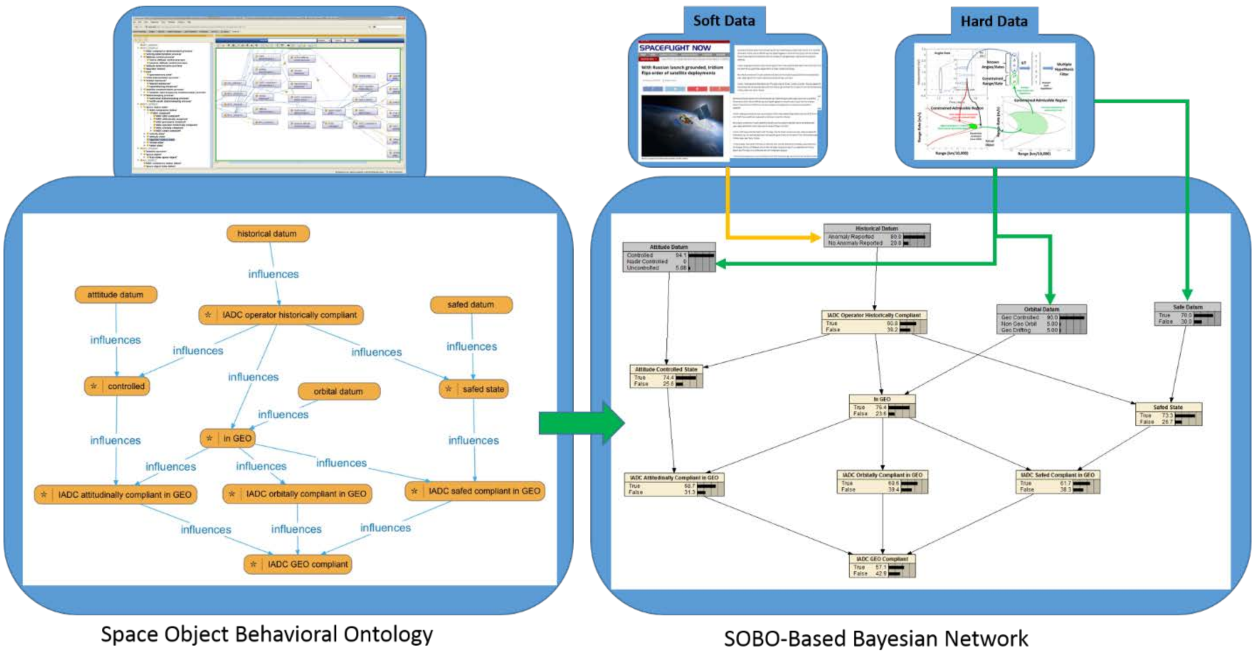


Fig.12. Preliminary SOBO and SOBO-based BN for the IADC guideline use case [2]. The SOBO (left) is translated in a graphical structure BN is a semi-automatic fashion that preserves the semantic of the ontology. The SOBO-BN is fed with pre-processed hard and soft data to the input nodes (e.g. attitude datum, orbital datum, historical datum) in a probabilistic form. The SOBO-BN probabilistically infer if the satellite is IADC GEO compliant.

5. CONCLUSIONS

We have detailed an end-to-end approach that combines machine learning techniques and Ontology-based Bayesian Networks for the development and implementation of a decision support systems to help characterize, predict and understand the behavior of resident space objects. The many components and techniques needed to implement such an approach and their advantages are highlighted. For example, state-of-the-art machine learning architectures (e.g. Extreme Learning Machines, Convolutional Deep Networks) that can be trained on physically-based models, are demonstrated to be effective in learning the RSO features in the vectorized energy and momentum states and parameters. The mapping from measurements (e.g. angles) to vectorized energy and momentum states and parameters enables behavior characterization via clustering in the features space and subsequent RSO classification. CNNs are showcased as a premiere technique that can directly learn to classify objects directly from measurements. Class and sub-class selection of RSOs (i.e. training set generation for classification) is naturally informed by a clustering analysis in the desired physical feature space (e.g. orbital energy and momentum). Importantly, such a data-driven approach must be integrated with methodologies that can explicitly represent space object domain knowledge. Indeed, SOBO ontologies are employed to define and capture the domain knowledge-base and BNs are constructed from the SOBO in a semi-automatic fashion to execute probabilistic reasoning over conclusions drawn from trained classifiers and/or directly from processed data. Such an approach enables integrating machine learning classifiers and probabilistic reasoning to support higher-level decision making for space domain awareness applications. Importantly, constructing a space object reusable and sharable ontology is a key ingredient for the successful development and deployment of a SODSS. Our team has been working on building proof-of-concept SOBO ontologies (i.e. use cases) and integrating them with BNs for reasoning under uncertainty. Future work includes integrating SOBO and BNs with hard and soft-data and demonstrate effectiveness in decision support under a variety of use cases for RSOs behavior characterization and prediction.

6. REFERENCES

1. Payne TP, Morris RF: The Space Surveillance Network (SSN) and Orbital Debris. Adv Astronaut Sci 2010, 137.
2. Walls, R., Gaylor, D., Reddy, V., Jah, M., Assessing the UN IADC Space Debris Mitigation Guidelines: A case for ontology-based data management, AMOS Paper 2016 (this conference)
3. Kazakov, Y., Krötzsch, M., & Simančík, F. (2014). The incredible ELK.Journal of Automated Reasoning, 53(1), 1-61.
4. Huang, G. B., Zhou, H., Ding, X., & Zhang, R. (2012). Extreme learning machine for regression and multiclass classification. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 42(2), 513-529.
5. LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." Nature, 521.7553 (2015): 436-444.
6. Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: theory and applications. Neurocomputing, 70(1), 489-501.
7. Furfaro, R., Law, A., (2015). Relative Optical Navigation Around Small Bodies via Extreme Learning Machines: Preliminary Results, Proceedings of the 2015 AAS/AIAA Astrodynamics Specialist Conference, Vail Co, Aug 9-13, 2015.
8. Liang, N. Y., Huang, G. B., Saratchandran, P., & Sundararajan, N. (2006). A fast and accurate online sequential learning algorithm for feedforward networks. Neural Networks, IEEE Transactions on, 17(6), 1411-1423
9. Furfaro, R., Linares, R., Gaylor, D., Jah, M., Mapping Sensors Measurements to the Resident Space Objects Behavior Energy and State Parameters Space via Extreme Learning Machines, to be presented at the 67th International Astronautical Conference, Guadalajara, Mexico, Sept 25-30, 2016.
10. Linares, Richard, and Roberto Furfaro. "Space Object classification using deep Convolutional Neural Networks." In 2016 19th International Conference on Information Fusion (FUSION), pp. 1140-1146. IEEE, 2016.
11. Bezdek, J. C. (2013). Pattern recognition with fuzzy objective function algorithms. Springer Science & Business Media.
12. Kohonen, T. (1998). The self-organizing map. Neurocomputing, 21(1), 1-6.
13. Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103). ACM.
14. Lee, H., Pham, P., Largman, Y., & Ng, A. Y. (2009). Unsupervised feature learning for audio classification using convolutional deep belief networks. InAdvances in neural information processing systems (pp. 1096-1104).
15. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
16. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep features for scene recognition using places database. In Advances in neural information processing systems (pp. 487-495).
17. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (pp. 1725-1732).
18. Linares, Richard, and Roberto Furfaro. "Space Object classification using deep Convolutional Neural Networks." In 2016 19th International Conference on Information Fusion (FUSION), pp. 1140-1146. IEEE, 2016.
19. Neapolitan, R. E. (2012). Probabilistic reasoning in expert systems: theory and algorithms. CreateSpace Independent Publishing Platform.
20. Heckerman, D. (2008). A tutorial on learning with Bayesian networks. In Innovations in Bayesian networks (pp. 33-82). Springer Berlin Heidelberg.
21. Abbot, R. I., & Wallace, T. P. (2007). Decision support in space situational awareness. Lincoln Laboratory Journal, 16(2), 297.
22. Lam, W., & Bacchus, F. (1994, July). Using new data to refine a Bayesian network. In Proceedings of the Tenth international conference on Uncertainty in artificial intelligence (pp. 383-390). Morgan Kaufmann Publishers Inc..
23. Friedman, N., & Koller, D. (2003). Being Bayesian about network structure. A Bayesian approach to structure discovery in Bayesian networks. Machine learning, 50(1-2), 95-125.
24. Druzdzel, M. J., & Díez, F. J. (2003). Combining knowledge from different sources in causal probabilistic models. Journal of Machine Learning Research, 4(Jul), 295-316.

25. Neches, R., Fikes, R. E., Finin, T., Gruber, T., Patil, R., Senator, T., & Swartout, W. R. (1991). Enabling technology for knowledge sharing. AI magazine, 12(3), 36.
26. Helsper, E. M., & Van Der Gaag, L. C. (2002, July). Building Bayesian networks through ontologies. In ECAI (Vol. 2002, p. 15th).
27. Ding, Z., & Peng, Y. (2004, January). A probabilistic extension to ontology language OWL. In System Sciences, 2004. Proceedings of the 37th Annual Hawaii international conference on (pp. 10-pp). IEEE.
28. Devitt, A., Danev, B., & Matusikova, K. (2006). Constructing Bayesian networks automatically using ontologies.
29. Zheng, H. T., Kang, B. Y., & Kim, H. G. (2008). An ontology-based bayesian network approach for representing uncertainty in clinical practice guidelines. In Uncertainty Reasoning for the Semantic Web I (pp. 161-173). Springer Berlin Heidelberg.
30. Da Costa, P. C. G., Laskey, K. B., & Laskey, K. J. (2008). PR-OWL: A Bayesian ontology language for the semantic web. In Uncertainty Reasoning for the Semantic Web I (pp. 88-107). Springer Berlin Heidelberg.
31. Fenz, S. (2012). An ontology-based approach for constructing Bayesian networks. Data & Knowledge Engineering, 73, 73-88.
32. Furfaro, R., Dohm, J. M., Fink, W., Kargel, J., Schulze-Makuch, D., Fairén, A. G., ... & Tarbell, M. A. (2008a). The search for life beyond Earth through fuzzy expert systems. Planetary and Space Science, 56(3), 448-472.
33. Furfaro, R., Lunine, J. I., Kargel, J. S., & Fink, W. (2008b). Intelligent systems for the autonomous exploration of Titan and Enceladus. In SPIE Defense and Security Symposium (pp. 69600A-69600A). International Society for Optics and Photonics.
34. Furfaro, R., Kargel, J. S., Lunine, J. I., Fink, W., & Bishop, M. P. (2010). Identification of cryovolcanism on Titan using fuzzy cognitive maps.Planetary and Space Science, 58(5), 761-779.
35. Furfaro, R., Fink, W., & Kargel, J. S. (2012). Autonomous real-time landing site selection for venus and titan using evolutionary fuzzy cognitive maps. Applied Soft Computing, 12(12), 3825-3839.
36. IADC guidelines, 2002. Available from http://www.iadc-online.org/index.cgi?item=docs_pub