

## Research Article

# An Obstacle Recognizing Mechanism for Autonomous Underwater Vehicles Powered by Fuzzy Domain Ontology and Support Vector Machine

Zhen-Shu Mi, Ahmad C. Bukhari, and Yong-Gi Kim

*Department of Computer Science and Engineering Research Institute, Gyeongsang National University, Jinju, Kyungnam 660-701, Republic of Korea*

Correspondence should be addressed to Yong-Gi Kim; [ygkim@gnu.ac.kr](mailto:ygkim@gnu.ac.kr)

Received 12 December 2013; Revised 11 June 2014; Accepted 18 July 2014; Published 5 August 2014

Academic Editor: Tsung-Chih Lin

Copyright © 2014 Zhen-Shu Mi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The autonomous underwater vehicle (AUV) and the problems associated with its safe navigation have been studied for the last two decades. The real-time underwater obstacle recognition procedure still has many complications associated with it and the issue becomes worse with vague sensor data. These problems can be coped with the merger of a robust classification mechanism and a domain knowledge acquisition technique. In this paper, we introduce a hybrid mechanism to recognize underwater obstacles for AUV based on fuzzy domain ontology and support vector machine (SVM). SVM is an efficient algorithm that was developed for recognizing 3D object in recent years and is a new generation learning system based on recent advances in statistical learning theory. The amalgamation of fuzzy domain ontology with SVM boosts the performance of the obstacle recognition module by providing the timely semantic domain information of the surrounding circumstances. Also the reasoning ability of the fuzzy domain ontology can expedite the obstacle avoidance process. In order to evaluate the performance of the system, we developed a prototype simulator based on OpenGL and VC++. We compared the outcomes of our proposed technique with backpropagation algorithm and classic SVM based techniques.

## 1. Introduction and Background

Currently, the underwater obstacle recognition issue has received a lot of attention and, due to its significance, has become one of the hot research topics of machine learning and computer vision. It has been observed that appearance-based object recognition methods can perform better in a variety of ongoing problems. On the other hand, many of these methods either require good whole-object segmentation, which severely limits their performance in the presence of clutter, occlusion, or background changes, or utilize simple conjunctions of low-level features, which cause crosstalk problems as the number of objects is increased [1]. In this research, we investigated an appearance-based obstacle recognition mechanism using fuzzy ontology and SVM that solves many of these problems for AUVs navigation safety.

SVM is a new generation learning system based on recent advances in statistical learning theory. SVMs deliver

state-of-the-art performance in real-world applications such text categorization, hand-written character recognition, and image classification. Their first introduction in the early 1990s led to a recent explosion of applications and the deepening of theoretical analysis has now established support vector machines along with neural networks as one of the standard tools for machine learning and data mining. It is considered as a very effective method of learning and classification [2–4].

Because of their commercial and military potential and the technological challenges in developing them, autonomous underwater vehicles (AUVs) have become a hot topic of research in ocean sciences. There are two main reasons to keep navigation security for AUVs. First, safe AUVs can search, rescue, and facilitate exploration in the scientific field. Second, it can save a lot of money because of the expensive cost of AUVs. Therefore, the AUVs safety problem has become an exciting hotspot research topic. Nevertheless, the multifarious nature of underwater

environments makes the research work a difficult task to handle. Thereby, development of an intelligent 3D obstacle recognition system becomes necessary for AUVs safe navigation. Some notable researches recently have been executed for obstacle detection of AUVs using SVM and other techniques [5–9].

In a normal situation, the AUVs use sonar instead of a good quality camera to recognize obstacles. AUV cannot see even several meters ahead since underwater situation is very dark in the deep ocean or the water is usually not clear in shallow water. However, in some special cases high definition cameras are used to achieve the desired results. For instance, when a vessel needs to be cleaned or repaired for some malfunction, a good quality camera is a very high priority choice to solve such kind of problems. There are a lot of applications for taking some valuable pictures using cameras including underwater resource exploration, bridge survey, and commercial diving.

In this research, we introduce a fast underwater obstacle recognition system for AUV based on fuzzy ontology and SVM [3, 10, 11]. The aim of this paper is to prove that the merger of fuzzy ontology and SVM demonstrates better performance in comparison with BP algorithm and simple SVM technique. The dynamic 3D obstacle simulation designed by OpenGL will provide some test data for getting the better experiment results. Therefore, the SVM will help underwater obstacle recognition, mission planning, and intelligent control of the AUV in a variety of ways [12–14]. At present, ontology has proved itself as an effective technology for relevant information extraction and knowledge sharing. We used the ontology to store and exploit the domain knowledge of the underwater water domain. This unique power of the ontology can be exploited to design an accurate knowledge representation system.

During the past few years, researchers are using ontology widely in different domains for knowledge representation and information extraction. It is another fact that most of the information resides in offline and online resources and is in imprecise format. The ontology only deals with crisp data and cannot find desired results from vague sources of data [15, 16]. To address this issue, researchers have incorporated fuzzy logic theory into classic ontology. Fuzzy theory is widely known among the research community due to its ability to deal with blur information [17, 18]. The rest of the paper is organized as follows. Section 2 describes the concepts and working of SVM; however, Section 3 highlights the importance of fuzzy ontology and its integration with SVM. The fourth part of this paper presents the proposed scheme. Sections 5 and 6 belong to experiments and conclusions, respectively.

## 2. The Support Vector Machine

The foundation of support vector machine (SVM) was developed by Vapnik in 1995 [12]. SVM is a powerful classification methodology that can create specialized functions from a set of labeled training data. These functions can be a classification function or a general regression function. For classification,

SVM operates by finding a hypersurface in the space of possible inputs. This hypersurface will attempt to split the positive examples from the negative examples. The split will be chosen to have the largest distance from the hypersurface to the nearest of the positive and negative examples. Intuitively, this makes the classification correct for testing data that is near, but not identical to, the training data. The goal of SVM is to find out an optimally separable hyperplane to solve the classification task. The optimal hyperplane has maximal distance between the two classes and the hyperplane.

*2.1. Linear Separable Case.* Consider the problem of separating the set of training vectors belonging to two linearly separable classes:

$$(x_i, y_i), \quad x \in R^d, \quad y \in \{+1, -1\}, \quad i = 1, \dots, n, \quad (1)$$

with a hyperplane

$$w \cdot x + b = 0, \quad (2)$$

where the parameters  $w, b$  are constrained by

$$\min_i |w \cdot x_i + b| = 1. \quad (3)$$

The set of vectors is said to be optimally separated by a hyperplane if it is separated without error. The distance between the closest vector and the hyperplane is maximal. Hence, a separating hyperplane in canonical form must satisfy the following constraints:

$$y_i \cdot (w \cdot x_i + b) \geq 1, \quad i = 1, 2, \dots, n. \quad (4)$$

Suppose that the following bound holds:

$$\|w\|^2 \leq c. \quad (5)$$

The VC dimension  $h$  of the set of canonical hyperplanes in  $n$  dimensional space is bounded by

$$h \leq \min([R^2, c], d) + 1, \quad (6)$$

where  $R$  is the radius of a hypersphere enclosing all training vectors. Hence the hyperplane that optimally separates the data is the one that minimizes (4):

$$\phi(w) = \frac{1}{2} (w \cdot w). \quad (7)$$

This is equivalent to minimizing an upper bound on the VC dimension. In practice, such a hyperplane does not exist. So we need to relax the constraints of (4) by introducing slack variables  $\xi \geq 0; i = 1, 2, \dots, n$ :

$$y_i \cdot (w \cdot x_i + b) \geq 1 - \xi, \quad i = 1, 2, \dots, n. \quad (8)$$

In this case, the optimization problem becomes

$$\phi(w, \xi) = \frac{1}{2} (w \cdot w) + C \sum_{i=1}^n \xi_i, \quad (9)$$

with a user defined positive finite constant  $C$ . The solution to optimization problem (9), under the constraints of (8), could be obtained in the saddle point of Lagrangian function:

$$L(w, b, \alpha, \xi, \gamma) = \frac{1}{2} (w \cdot w) + C \sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i |y_i (w \cdot x_i + b) - 1 + \xi_i| - \sum_{i=1}^n \gamma_i \xi_i, \quad (10)$$

where  $\alpha_i \geq 0$ ,  $\gamma \geq 0$ , and  $i = 1, 2, \dots, n$  are the Lagrange multipliers.

The Lagrangian function has to be minimized with respect to  $w$ ,  $b$ , and  $\xi_i$ . Classical Lagrangian duality enables the primal problem, (10), to be transformed to its dual problem, which is easier to solve. The dual problem is given by

$$\max_{\alpha} \left[ \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \right], \quad (11)$$

with constraints

$$\sum_{i=1}^n \alpha_i y_i = 0, \quad (12)$$

$$0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, n. \quad (13)$$

This is a classic quadratic optimization problem. There exists a unique solution. According to the Kuhn-Tucker theorem of optimization theory [8], the optimal solution satisfies

$$\alpha_i [y_i (w \cdot x_i + b) - 1] = 0, \quad i = 1, 2, \dots, n. \quad (14)$$

Equation (14) has nonzero Lagrange multipliers when and only when the points  $x$  satisfy

$$y_i \cdot (w \cdot x_i + b) = 1. \quad (15)$$

These points are termed support vectors (SV). The hyperplane is determined by the SV, which is a small subset of the training vectors. Hence if  $\alpha_i^*$  is the nonzero optimal solution, the classifier function can be expressed as

$$f(x) = \text{sgn} \left\{ \sum_{i=1}^n \alpha_i^* y_i (x_i \cdot x) + b^* \right\}, \quad (16)$$

where  $b^*$  is the solution of (14) for any the nonzero  $\alpha_i^*$ .

**2.2. Nonlinear Case.** When a linear boundary is inappropriate, SVM can map the input vector into a high dimensional feature space. By defining a nonlinear mapping, the SVM constructs an optimal separating hyperplane in this higher dimensional space. Usually nonlinear mapping is defined as following the function:

$$\phi(\cdot) : R^d \longrightarrow R^{dh}. \quad (17)$$

In this case, optimal function (8) becomes (18) with the same constraints:

$$\max_{\alpha} \left[ \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \right], \quad (18)$$

where

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \quad (19)$$

is the kernel function performing the nonlinear mapping into feature space. It may be any of the symmetric functions that satisfy the Mercer conditions [14]. The following functions are commonly used kernel functions.

(1) Polynomial function:

$$K(x_i, x_j) = (x_i \cdot x_j + 1)^q, \quad q > 0. \quad (20)$$

(2) Radial basis function:

$$K(x_i, x_j) = \exp \left\{ -\frac{|x_i - x_j|^2}{\sigma^2} \right\}. \quad (21)$$

(3) Sigmoid function:

$$K(x_i, x_j) = \tanh(\beta \cdot (x_i \cdot x_j) + c). \quad (22)$$

The classifier function can be defined as

$$f(x) = \text{sgn} \left\{ \sum_{i=1}^n \alpha_i^* y_i K(x_i, x) + b^* \right\}, \quad (23)$$

where  $b^*$  is the solution of the following equation for any of the nonzero  $\alpha_i^*$ :

$$\alpha_i \left[ y_i \sum_{j=1}^n \alpha_j y_j K(x_j, x_i) + b - 1 \right] = 0. \quad (24)$$

### 3. The Fuzzy Domain Ontology

Ontology is a branch of metaphysics which focuses on the study of existence. Ontology is an explicit formal specification of a shared conceptualization of any domain which is machine readable and human understandable format [15]. In ontology, we basically focus on concepts (classes) of the domain, their values, and their properties. Ontology arranges the classes in a hierarchical structure, in the form of subclasses and superclasses hierarchies, and also defines which property has constraints on its values. Basically, ontology is developed to share a common understanding of the domain knowledge among people and software, in order to reuse the classes of a domain instead of remodelling them. Ontology is written in a specific language called OWL (Web Ontology Language) [16, 17] which is developed by W3C. OWL is specially designed for ontology modelling. Figure 1 depicts the graphical model fuzzy underwater ontology generated by

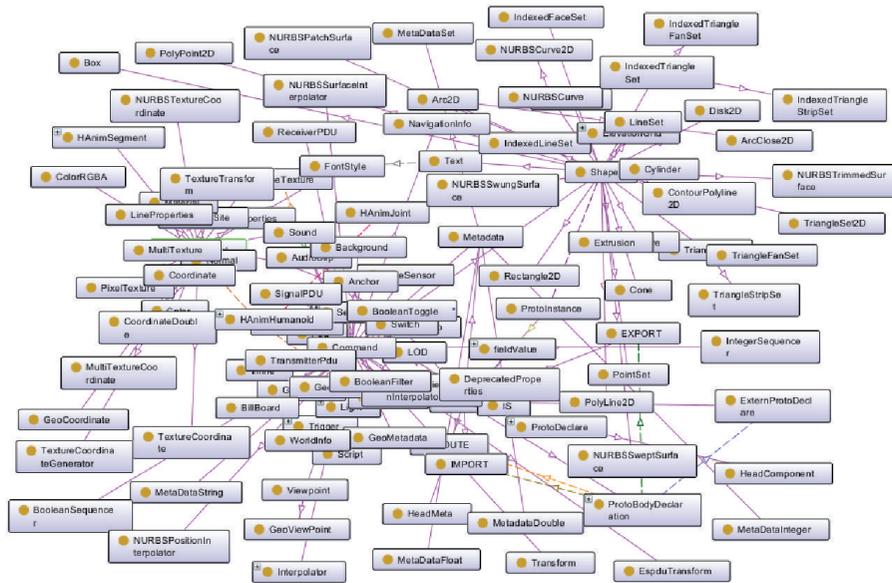


FIGURE 1: Graphical model of fuzzy underwater environment ontology.

a Protégé OnToGraf tool. OWL-2 is the extended version of OWL which holds new features and rationale. In order to increase the expressivity level of OWL, several new constructs were introduced along with their unique modelling features such as extended annotations and complex data representations. Ontology can be expressed in equation format as

$$\text{Ontology} = (\text{Concepts } C, \text{Properties } P, \text{Relationships } R, \text{Values } V, \text{Constraints of property value } V_C). \quad (25)$$

In the above expression notations  $C$ ,  $P$ ,  $R$ ,  $V$ , and  $V_C$  represent concepts, properties of concepts, relationship among concepts, values of the concepts, and constraints on property values, respectively. The Ontology can be classified into metaontology, domain ontology, and application level ontology. Usually, we use domain ontology which is used to represent the information about a specific domain. On the other hand, there is no exactly one way to define ontology [15, 17]. Researchers used different ways to define the ontology according to their needs and demands. The values of concepts and properties in the classical domain ontology are crisp but as mentioned earlier most real-time systems nowadays are based on complex structure which is based on fuzzy set theory.

**3.1. Development of Fuzzy Domain Ontology.** In order to make the paper self-contained, we first define the concept definitions and terminologies before moving towards the formal development of underwater fuzzy ontology. Fuzzy set theory was introduced by Lotfi Zadeh in 1965 [15, 18] to deal with vague and imprecise concepts. In classical set theory, elements either belong to a particular set or do not. There is no concept of partial membership in classical set theory.

However, in fuzzy set theory the association of an element with a particular set lies between 0 and 1, which is called its degree of association or membership degree. Fuzzy set theory adds generalization concept in to classical set theory and makes it diverse enough to represent imprecise boundaries like hot, tall, low speed, and so forth. A fuzzy set “FS” over the universe of discourse “ $Z$ ” can be defined by its membership function  $\mu_{FS}$  which maps element “ $z$ ” to values between  $[0, 1]$

$$\mu_{FS}(z) : z \rightarrow [0, 1]. \quad (26)$$

Here  $z \in Z$  and  $\mu_{FS}(z)$  provide the degree of membership by which  $z$  belongs to  $Z$ .  $z$  is considered as full member of  $Z$  if  $\mu_{FS}(y) = 1$  and is considered partial if  $\mu_{FS}(y)$  is between 0 and 1; say 0.57. The strength of our proposed system is the fuzzy enriched domain ontology which not only is able to present the associated domain knowledge of any obstacle but can also assist obstacle avoidance and intelligent path planning by providing timely information about obstacles.

To achieve perfection in the results and to expedite the process of fuzzy based domain ontology, we employed the service of the domain experts. The domain experts do not have knowledge about SVM and ontology schemes but they can best describe the underwater environment and obstacles. The domain experts monitored all the phases of the ontology development. Furthermore, we followed the ontology development steps which are defined in [19] to develop the underwater domain ontology. The steps can be described as follows.

- (1) Determine the domain and scope of the ontology.
- (2) Consider reusing existing ontologies.
- (3) Enumerate important terms in the ontology.
- (4) Define the classes and the class hierarchy.
- (5) Define the properties of classes.

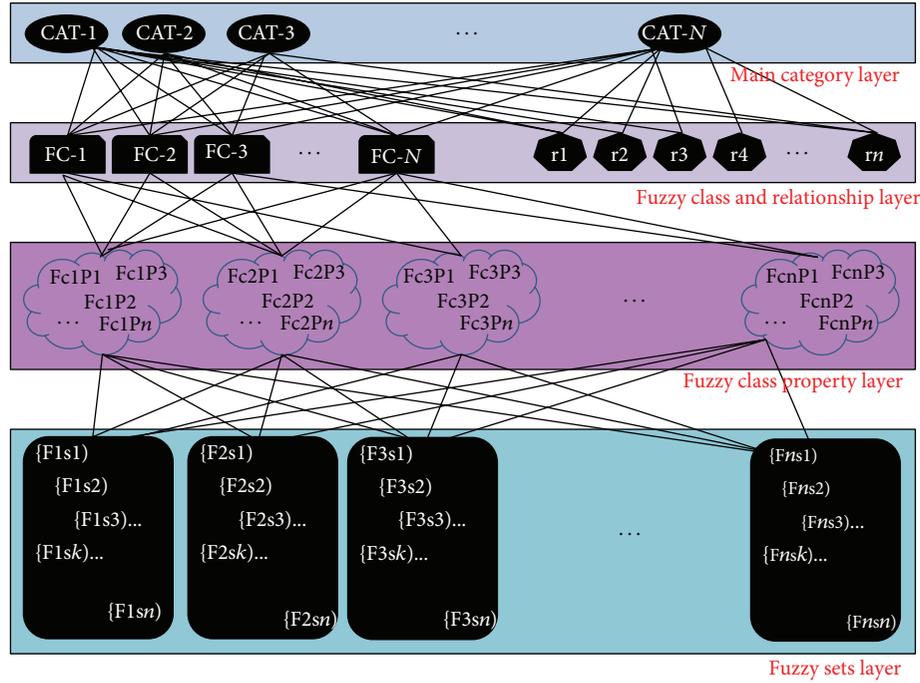


FIGURE 2: Structure of fuzzy underwater domain ontology.

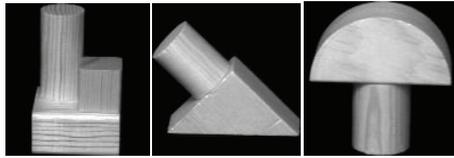


FIGURE 3: Images of 3 objects of the COIL database.

- (6) Define the facets of the slots.
- (7) Create instances.

To author the fuzzy based domain ontology we chose the Protégé 4.1 and OWL-2 due to their technical soundness in information engineering circle. Figure 2 displays the architecture of fuzzy domain ontology.

#### 4. The Proposed 3D Obstacle Recognition System

The SVM in a linear separable case categorizes the same class of data on one side of the optimal hyperplane and the opposite class on the other side. We illustrate an example to explain this problem; see Figure 1. We took a set of data from the COIL image database for virtual experiment purposes. The COIL (Columbia Object Image Library) database consists of 7,200 images of 100 objects. We selected some simple and typical images like simulation obstacles by way of example test data (e.g., Figure 3). The images are color images (24 bits for each of the RGB channels) of 128 × 128 pixels. For each object, the turntable is rotated 5 degrees per image. The tester can design and adjust the rotation degree to get an optimal test image

data. All the selected color image data are according to COIL image standard. Figures 1 and 4 show a selection of the objects in the database and one every three views of a specific object, respectively [20].

4.1. *Processing Based on OpenGL Simulation.* In real research, the underwater environment is hazy and surrounding information is not known. However, it is easy to understand the simulation using OpenGL and we placed some obstacles randomly and created the virtual environment according to the real system. For the experimental purpose, we designed a 3D virtual testbed using OpenGL to get some test data to prove the effectiveness of our proposed algorithm. Each image  $I = (R, G, B)$  was first transformed into a gray-level image through the conversion formula:

$$\text{Gray} = \text{Red} * 30 + \text{Green} * 59 + \text{Blue} * 11, \quad (27)$$

rescaling the obtained gray-level from 0 to 255. The obstacles images spatial resolution was reduced to 32 × 32 by averaging the gray levels over 4 × 4 pixel patches. The purpose of this preprocessing is to identify the selected data for the image containing obstacles and standardize the obstacles images [20, 21].

4.2. *Training and Testing the SVM with Semantic Domain Data.* It is a very common problem to select kernel functions in the SVM training phase. In our experiments, the three kernel parameters,  $\sigma$ ,  $c$ , and  $q$ , determine the dynamics of an SVM network. We selected the optimal parameters for optimizing and adjusting the SVM parameters. The radial basis function (RBF) kernel function nonlinearly maps

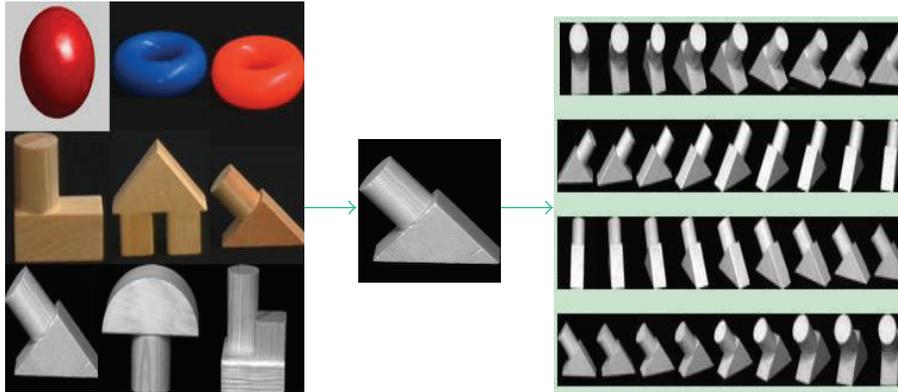


FIGURE 4: 36 images of all the 72 images of one COIL object.

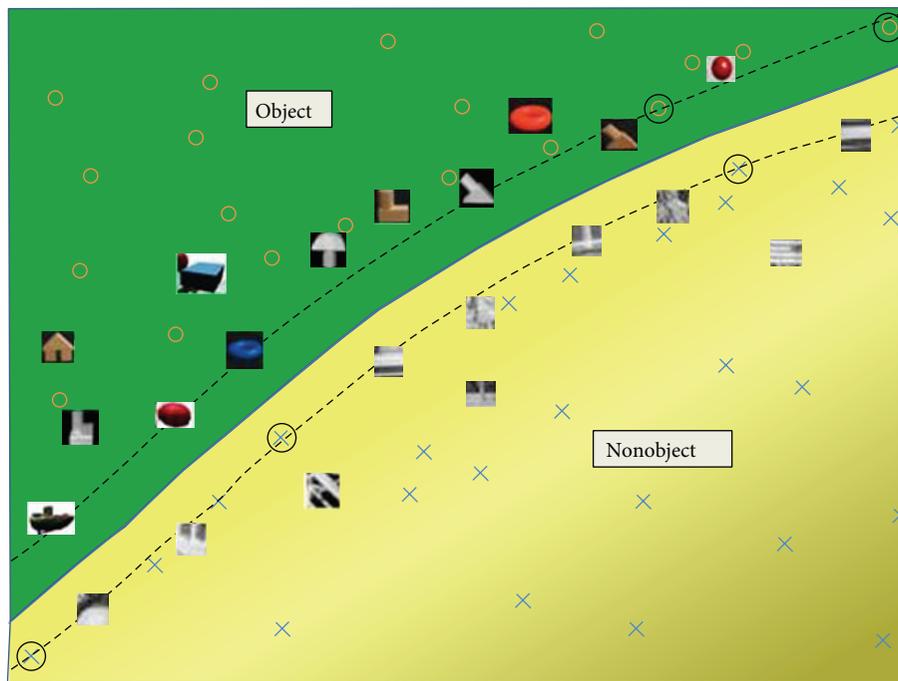


FIGURE 5: SVM feature space for object recognition based on fuzzy domain ontology.

samples into a higher dimensional space. The RBF kernel has less numerical difficulties; its computation is not as complex as the other kernels. Thus the RBF kernel is a better choice for training and testing. We used semantically annotated images to train our SVM classifier. In the SVM recognition system, we adjusted the kernel parameters by training, until the optimal kernel parameters were found. To reduce the computational quantity, we used the principal component analysis method to generate 128-dimensional feature vectors which can represent the characteristics of the image [12].

In one case, 20 objects, total 720 images are selected as samples in the training phase. All the 1440 images for 20 objects are the test data. Given a subset of the 20 objects and the associated training set of 36 images for each object, the test set consists of the remaining 36 images for each object. First, we tested the recognition system on sets of 20 of the 100 COIL objects. The training sets consisted of 36 images

(every 10 degrees for each of the 20 objects) and the test sets consisted of the remaining 36 images. For all the 12 random choices of 20 of the 100 objects, the system got a perfect score. So we decided to select by hand the 20 objects that were more difficult to recognize. To gain a better understanding of how an SVM actually performs recognition, it may be useful to look at the relative weights of the components of  $w$ . A gray-values encoded representation of the absolute value of the components of  $w$  relative to the optimal separating hyperplane [20]. Note that the background is essentially irrelevant, while larger components can be found in the central portion of the image [12, 20]. Figure 5 shows the SVM feature space based on fuzzy domain ontology.

*4.3. Proposed System Overview.* The proposed system consists of five functional phases named as the detection and acquisition phase, obstacle analysis phase, initial recognition

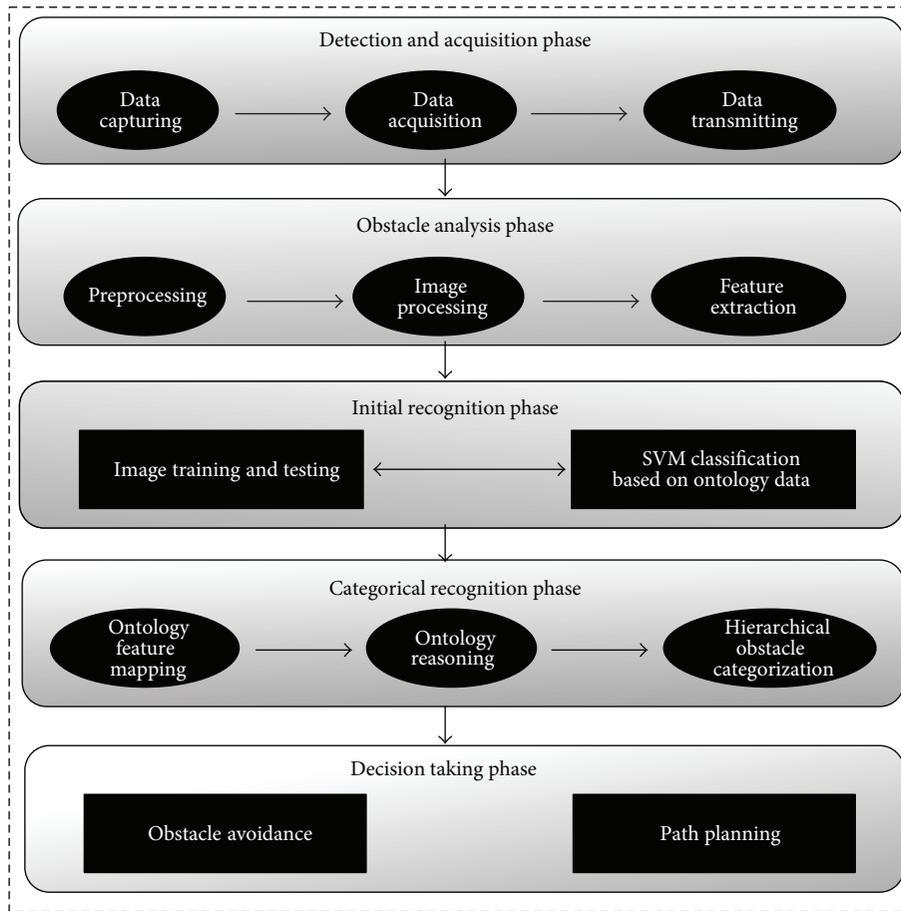


FIGURE 6: The SVM and fuzzy ontology supported obstacle recognition system architecture.

phase, categorical recognition phase, and decision making phase. While maneuvering, the AUV continuously takes images of its surroundings and sends them back to the coastal station for analysis. The sophisticated analysis phase applies the predefined algorithms against the image data to get results. Subsequently, these analysis results are forwarded to the decision making module which is specially designed to make intelligent decisions and sends the instructions back to the AUV for safe navigation. In the subheadings, we elaborate the system in detail.

**4.3.1. Detection and Acquisition Phase.** This phase can be further divided into three phases which are data capturing, data acquisition, and data transmitting. To accomplish the underwater mission, an AUV has to acquire optimal path information from the onboard system or from the surface control station. To get the advice for safe maneuvering, the AUV repeatedly captures the images with an onboard high definition camera. Usually, the AUV has internal memory which stores the current situation and the transmission function of the detection and acquisition phase assists in sending the image data back to the surface station for the analysis.

**4.3.2. Obstacle Analysis Phase.** Similar to the previous phase, this phase can also be divided into a series of subactivities. In the preprocessing stage, the system acquires the image data and performs the initial screening. The initially processed images are automatically sent to the image processing phase which is the core module. The job of this module is to deeply analyze the images after sharpening and noise removing. Further, from an image, it can extract the features such as color, width, height, diameter, shape, texture, and orientation towards the AUV.

**4.3.3. Initial Recognition Phase.** This phase is considered the most important phase of the entire system. In fact, the efficiency of the entire system is dependent upon this phase. This phase performs two main tasks. It calls the SVM classifier and starts the training and testing activities. Before the testing, we have to train the classifier with sample data. In our technique, we introduced a fuzzy ontology based trainer, which trains the SVM classical algorithm with semantically annotated image data. This allows the classifier to be aware of the color, texture, geometry, and orientation of the image. As the SVM is a binary classifier at initial phase, it segregates the obstacle from nonobstacle data as shown in Figure 6.

**4.3.4. Categorical Recognition Phase.** The categorical recognition phase is based on the fuzzy ontology scheme. The main purpose of this phase is to categorize the detected obstacles into classes and subclasses. We introduced several fuzzy classes and subclasses in our ontology as the following: underwater animals, rock, icebergs, long obstacle, short obstacle, dangerous obstacle, very dangerous obstacle, soft obstacle, and hard obstacle. Each class has its own properties and values. The image processing phase extracts the obstacle features and categorical recognition phase compares their properties with semantically defined properties and segregates the obstacles according to their matching classes. This categorical recognition is essential to expedite the process of decision making. For instance, if the fuzzy ontological classifier finds an obstacle as soft or nondangerous, it simply advises the AUV not to change the route. We use a *Pellet* reasoner in Protégé to reason against the supplied data [15, 16].

**4.3.5. Decision Making Phase.** Last but not least, all the analysis results are forwarded to the decision making phase. The underlying algorithm performs the computations and finds an appropriate decision. Lastly, the commands are transmitted to the AUV. Obstacle avoidance and path planning do not fall in the scope of our paper. Figure 6 describes the proposed system architecture.

## 5. Experiments and Results

To evaluate the efficiency of our proposed system, we developed a virtual testbed which holds all the components of the real-world environment. In Figure 7, we considered the small black ball as a starting point and the right red ball as an end point. In our algorithm, we added underwater survey as a task for the AUV. When we play the simulation, the AUV starts grabbing the surroundings information and takes the high definition images with the help of the mounted camera. After capturing the images from the virtual environment, the AUV sends those to the base station for further processing [21].

The base station computer gets the random images as test data each second. The simulation generates different kinds of obstacles around AUVs and facilitates the evaluator to get the 3D obstacle images data directly from the experiment. One of the experiments is tested by a MATLAB tool based on the OpenGL test images [22]. In our experiment, the OpenGL test images contain 720 images of 20 objects. For each object, we can generate 36 images from OpenGL simulation, and the simulation is powerful enough to produce a new image of the same obstacle for every 10 degrees of rotation. We used the same training and testing data sets for SVM and BP algorithms; therefore, we can get a clear analysis based on both algorithms.

We performed a series of experiments using different training sets to evaluate the performance of the whole system [22, 23]. For the BP algorithm experiment, we selected the optimal parameters in the experiments while optimizing the neural network architectures. To train the artificial neural networks, we used the back propagation algorithm. We used

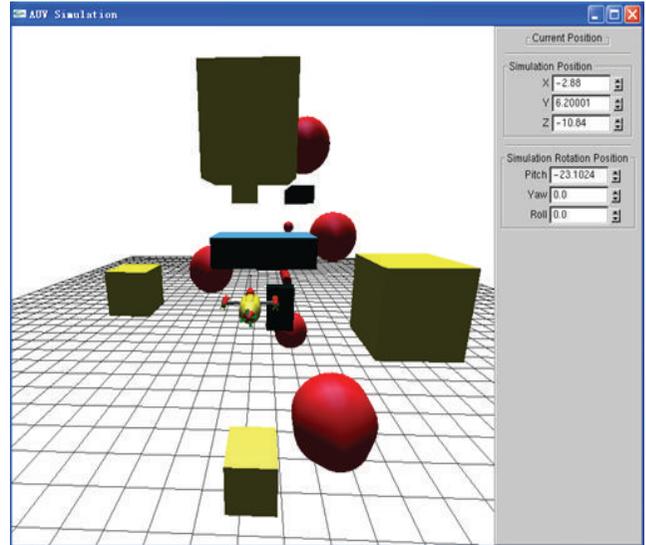


FIGURE 7: The virtual testbed to evaluate the proposed system.

TABLE 1: SVM optimal parameters.

Arg.	C	Training error	Training time
1	100	0.0106	48.09 s
1	200	0.0127	41.45 s
2	400	0.0171	39.52 s
2	500	0.0062	63.15 s

TABLE 2: Adjusting BP network.

Neurons in hidden layer	Training error	Training time
6	0.0148	631.17 s
7	0.0136	683.52 s
8	0.0127	495.76 s
9	0.0112	451.23 s

the tensing function in the hidden layer and logsig function in the output layer. It was trained to find out the optimal neurons, which can generate the best output results [4, 24].

Tables 1 and 2 have the optimal SVM parameters and adjusted BP network, respectively. Table 3 shows the performance results of classic SVM, BP, and fuzzy ontology based SVM schemes. Figure 8 depicts the average performance of all three schemes in graphical format. We can easily understand from the graph that high accuracy rate can be achieved by applying the proposed hybrid approach. By applying the proposed scheme, not only can we reduce training and testing time but we can also improve the efficiency of the system overall.

## 6. Conclusion

In this research, we introduced a hybrid mechanism based on fuzzy domain ontology and support vector machine to expedite the process of underwater obstacle recognition. Several

TABLE 3: The performance result of SVMs, BP algorithm, and the proposed scheme.

Obstacles classifier (object sample)	BP training time (s)	SVM training time (s)	Proposed system training time	BP testing time (s)	SVM testing time (s)	Proposed system testing time	BP accuracy	SVM accuracy	Proposed system accuracy
5–24	72.86	15.29	10.43	0.28	0.19	0.09	86.17%	90.87%	97.91%
10–24	153.76	19.93	9.47	3.91	2.28	0.14	85.12%	86.73%	98.17%
20–24	217.91	67.35	33.52	12.83	6.12	1.26	84.94%	89.90%	95.77%
5–36	141.26	47.82	26.51	2.93	1.06	0.07	87.95%	91.78%	98.31%
10–36	248.26	78.11	50.09	11.96	7.18	2.33	86.21%	92.99%	97.47%

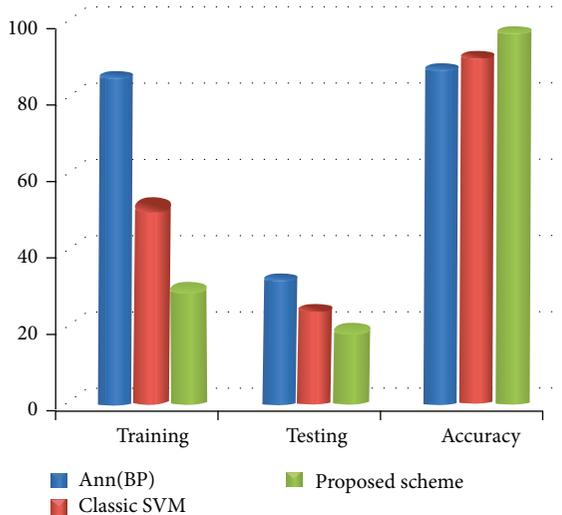


FIGURE 8: An average performance comparison graph among BP algorithm, classical SVM, and proposed scheme.

researchers have addressed this issue and many of them proposed solutions to solve this problem. However, the efficiency of their proposed algorithms decreases with the increase in data vagueness collected from sensors. We coupled fuzzy ontology with SVM which holds the underwater domain data in machine readable and human understandable format which further assists the SVM classifier in fast recognition of the obstacle. Moreover, the reasoning power of the fuzzy ontology provides accurate and timely information to the obstacle avoidance and path planning modules and makes the underwater navigation safe. To analyze the performance of our proposed system, we developed a prototype system and performed a number of experiments by using the training set from the COIL database. To monitor the overall performance of the system, we compared our proposed scheme with BP and classic SVM schemes. Our proposed scheme showed more accurate results in a shorter span of time. As a future work, we will train our algorithm with diverse training sets, so it can be able to work with any kind of underwater data.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgment

This work was supported by the Korea National Research Foundation (NRF) Grant funded by the Korean Government (no. 2012R1A1A2038601).

### References

- [1] T. Sergios and K. Konstantinos, *Pattern Recognition*, Elsevier Science Publishing, 2003.
- [2] C. J. C. Burge, "A tutorial on support vector for pattern recognition," *Neural Network*, pp. 1076–1121, 2001.
- [3] C. Colin, "Algorithmic approaches to training support vector machines: a survey," in *Proceedings of the 8th European Symposium on Artificial Neural Networks*, pp. 27–54, 2000.
- [4] E. Zheng, P. Li, and Z. Song, "Performance analysis and comparison of neural networks and support vector machines classifier," in *Proceedings of the 5th World Congress on Intelligent Control and Automation*, pp. 52–58, June 2004.
- [5] I. Quidu, A. Hetet, Y. Dupas, and S. Lefevre, "AUV (Redermor) obstacle detection and avoidance experimental evaluation," in *Proceedings of OCEANS—Europe*, 2007.
- [6] S. Zhao, T. Lu, and A. Anvar, "Automatic object detection for AUV navigation using imaging sonar within confined environments," in *Proceedings of the 4th IEEE Conference on Industrial Electronics and Applications (ICIEA '09)*, pp. 3648–3653, May 2009.
- [7] Q. Li, J. Zhao, and Y. Zhao, "Detection of ventricular fibrillation by support vector machine algorithm," in *Proceedings of the International Asia Conference on Informatics in Control, Automation, and Robotics (CAR '09)*, pp. 287–290, Bangkok, Thailand, February 2009.
- [8] M. S. Bewley, B. Douillard, N. Nourani-Vatani, A. Friedman, O. Pizarro, and S. B. Williams, "Automated species detection: an experimental approach to help detection from sea-floor AUV images," in *Proceedings of the Australasian Conference on Robotics and Automation (ACRA '12)*, December 2012.
- [9] S. Karabchevsky, B. Braginsky, and H. Guterman, "AUV real-time acoustic vertical plane obstacle detection and avoidance," in *Proceedings of Autonomous Underwater Vehicles (AUV '12)*, IEEE/OES, 2012.
- [10] V. Blanz, B. Scholkopf, H. Bulthoff, C. Burges, V. Vapnik, and V. Vetter, "Comparison of view-based object recognition algorithms using realistic 3D models," in *Proceedings of International Conference on Artificial Neural Networks*, vol. 112 of *Lecture Notes in Computer Science*, Springer, pp. 251–256, Berlin, Germany, 1996.

- [11] H. Murase and S. K. Nayar, "Visual learning and recognition of 3-d objects from appearance," *International Journal of Computer Vision*, vol. 14, no. 1, pp. 5–24, 1995.
- [12] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.
- [13] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, Mass, USA, 1995.
- [14] R. Courant and D. Hilbert, *Methods of mathematical physics*, vol. 1, Wiley-Interscience, New York, NY, USA, 1953.
- [15] A. C. Bukhari and Y.-G. Kim, "Integration of a secure type-2 fuzzy ontology with a multi-agent platform: a proposal to automate the personalized flight ticket booking domain," *Information Sciences*, vol. 198, pp. 24–47, 2012.
- [16] A. C. Bukhari and Y. Kim, "Ontology-assisted automatic precise information extractor for visually impaired inhabitants," *Artificial Intelligence Review*, vol. 38, no. 1, pp. 9–24, 2012.
- [17] A. C. Bukhari and Y. Kim, "Exploiting the heavyweight ontology with multi-agent system using vocal command system: a case study on e-mall," *International Journal of Advancements in Computing Technology*, vol. 3, no. 6, pp. 233–241, 2011.
- [18] C. Ahmad and Y. Kim, "Incorporation of fuzzy theory with heavyweight ontology and its application on vague information retrieval for decision making," *International Journal of Fuzzy Logic and Intelligent System*, vol. 11, no. 3, pp. 171–177, 2011.
- [19] A. C. Bukhari, A. A. Fareedi, and Y. G. Kim, "HO2IEV: heavyweight ontology based web information extraction technique for visionless users," in *Proceedings of the 7th International Conference on Networked Computing and Advanced Information Management (NCM '11)*, pp. 90–95, June 2011.
- [20] M. Pontil and A. Verri, "Support vector machines for 3D object recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 6, pp. 637–646, 1998.
- [21] A. Arif, L. Young-il, J. Hee, and Y. Kim, "Unsupervised real-time obstacle avoidance technique based on a hybrid fuzzy method for AUVs," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 8, pp. 82–87, 2008.
- [22] N. Yoshikawa and Y. Ii, "Three-dimensional object recognition using multiplex complex amplitude information with support function," in *Proceedings of the 1st International Conference on Innovative Computing, Information and Control (ICICIC '06)*, vol. 1, pp. 314–317, September 2006.
- [23] J. Licheng, *Application and Realization of Neural Networks*, Xian Electron and Science University, 1993.
- [24] C. H. Chen, "A comparison of neural network models for pattern recognition," in *Proceedings of the 10th International Conference on Pattern Recognition*, pp. 45–46, June 1990.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

