A Big Bang-Big Crunch Type-2 Fuzzy Logic System for Explainable Semantic Segmentation of Trees in Satellite Images using HSV Color Space

Hugo Leon-Garza and Hani Hagras
The Computational Intelligence Centre,
School of Computer Science and Electronic Engineering
University of Essex
Colchester, UK
hleong@essex.ac.uk

Anasol Peña-Rios, Anthony Conway and Gilbert Owusu
BT Labs
BT Plc
Adastral Park, Martlesham Heath
Ipswich, UK
anasol.penarios@bt.com

***Abstract*— In recent years, new sensor technologies have increased the accessibility of high-resolution satellite images. The information in these images can help to improve activities like urban planning and growth analysis of cities. Additionally, information extracted from these images can be used for taking decisions related to infrastructure planning, e.g. identifying objects that might interfere with network assets like underground cables. To be able to justify the cost of network planning decisions a high degree of interpretability is required. Convolutional Neural Networks (CNNs) are the state of the art for segmenting these images, but like any black box model they do not offer any explanation for their output. In this paper, we present an approach on how to use a Fuzzy Logic System (FLS) for performing explainable semantic segmentation of trees in satellite images. The FLS uses the HSV (hue, saturation, value) of the pixels as inputs and was optimized by using an evolutionary algorithm called Big Bang Big Crunch. The best configuration for the Interval Type-2 FLS has an Intersection over Union metric measure of 60.6%, which is close to the results obtained from neural network, however the proposed FLS provides interpretable outputs which is highly needed for the real-world operation especially in the telecommunication domain.**

Keywords—Fuzzy Logic System, Neural Networks, explainable AI, semantic segmentation, interpretable models, satellite images.

I. Introduction

Semantic segmentation, or pixel-wise classification, is the task of assigning a semantic label to every pixel in an image, e.g. building or tree [1]. Satellite images contain large amounts of structured and uniform data compared to traditional images, and semantic segmentation can be used to extract that information [2]. Automatic segmentation of remote sensory data facilitates monitoring forest resources [3], agricultural land for economic analysis [4] and urban modelling and growth analysis of specific geographic areas [5]. Utility companies that provide phone, gas and electric services are within the main interested parties for semantic segmentation of satellite images [6]. Their interest for the output of this task is primarily due to the need for information related to buildings, terrain and vegetation from a geographic area, to be able to perform planning and maintenance activities of their infrastructure [6].

In the last few years, the improvements and development in sensor technologies have increased the accessibility to high spatial resolution images [7], which has also enabled the automatic creation of computer-aided designs and models of geographic areas [6]. Additionally, the availability and accessibility of new high performance GPUs have made possible the training of deep neural network architectures and has drawn significant attention towards different architectures for solving the segmentation problem [8]. Currently, deep Convolutional Neural Networks (CNNs) are considered state-of-the-art [8], however, the lack of explainability in how the model makes predictions [8] and the large amount of labelled data required for training the model [9] are the two main drawbacks of this approach. The lack of explainability is the main disadvantage since there is a need for the end users to be assured of a reconstructed model's credibility [6]. The use of an explainable Artificial Intelligence (AI) model can help automate the creation of these models and provide the end user with an explanation of why a geographic area was constructed in a certain way.

In this paper we present a type-2 Fuzzy Logic System (T2 FLS) for performing explainable semantic segmentation of trees in satellite images. The T2 FLS uses the HSV (hue, saturation, value) of the pixels as inputs and it was optimized by using an evolutionary algorithm called Big Bang Big Crunch. The proposed approach provides a better understanding of how the prediction is computed. Also, since it is now a pixel-wise classification, the number of labelled images needed is lower since multiple training patterns can be extracted from each image.

The paper is organized as follows: Section II provides an overview of neural network architectures for semantic segmentation while Section III provides an overview on HSV color spaces. Section IV presents the proposed optimization process for our fuzzy logic system implementation. Section V

978-1-7281-6932-3/20/$31.00 ©2020 IEEE

presents the experiments and results. The conclusions and future work are presented in section VI.

II. A Brief Overview of neural network architectures for semantic segmentation

In the last few years, it has become possible to have deep neural networks trained in less time due to the advances in technology. In particular, Convolutional Neural Networks (CNNs) have received significant attention when trying to solve image segmentation problems [10]. CNNs are typically composed of a combination of convolution, pooling, rectified linear units and fully connected layers. They were originally designed for image classification but they have been successfully adapted for semantic segmentation [11]. Despite being considered the state-of-the-art due to their accuracy success [7], CNNs have two major problems: (1) they need a large amounts of data with labels for each pixel to achieve an acceptable error level [9] and (2) they lack of a clear understanding on what kind of features those architectures are learning [8].

A Multilayer Perceptron (MLP) with Back-Propagation Learning is a simple neural network architecture that transforms the segmentation problem to a pixel classification problem by using extracted features from each pixel as inputs [12]. A pixel-wise classification means that the MLP will use each pixel of an image as a training input instead of the complete image, therefore multiple training patterns might be extracted from a single image and the number of labelled images might be reduced. Also, by extracting features and feeding them as inputs, it is possible to at least understand what the network is using to predict.

III. A brief description of HSV color space

The HSV (hue, saturation, value) color space is based on the characteristics and the way the human visual system perceives color [13]. Hue (H) is a single value between 0 and 360 that represents the color, unlike RGB color space where the color is represented by the combination of the three values. Saturation (S) describes how pure the color is, it is measure with a value between 0 and 1, with 0 being white. Value (V) can be seen as the amount of light in the color, it is a value between 0 and 100, where 0 will look black [14]. This color space approach of separating color, intensity and light is preferred because it is considered easier to understand than the combination of color intensities in RGB space.

Combinations of RGB colors are not considered as interpretable as using only a single variable to define color input. E.g. having a green Hue with high saturation and low value is considered easier to interpret than low Red, high Green and low Blue. This close relationship of the HSV color space with how humans perceive color is the main reason to use it as preferred input. In our implementation, RGB images from the dataset were transformed to HSV color space.

IV. The optimization process for the proposed Fuzzy Logic Systems

A. A Brief Overview of Fuzzy Logic Systems

A rule-based FLS is a system that maps crisp inputs to crisp outputs with the help of an inference engine based on rules [15]. The rules are IF-THEN rules, composed of antecedents (IF-part) and consequences (THEN-part), each of them has associated a linguistic label such as low, medium or high. Fuzzy sets are then used to model and describe the uncertainty of the crisp inputs that belong to each of the linguistic labels. The modelling of these imprecise human concepts allows rule-based FLS to mimic the human thinking [16].

Type-1 (T1) and type-2 (T2) fuzzy sets can be used to model the uncertainty in the inputs, each of them model different levels of uncertainty [17]. Type-1 fuzzy sets are described by membership functions that represent the different crisp inputs associated to a linguistic label, however, since the membership functions are certain, they are unable to handle the uncertainty of transferring knowledge from different experts to precise functions and rules. Type-2 fuzzy sets, on the other hand, are useful for handling the uncertainty of determining an exact membership function for an input or output fuzzy set, therefore, they can be used to handle uncertainty related to rules and measurements [15], [18]. This additional handling of uncertainty has proven to make Type-2 FLS better than Type-1 FLS at complex problems [19], [20].

As mentioned in [16], a rule-based FLS is considered an Explainable AI (XAI) model since it covers all the XAI components from [21]. The rule-based FLS generated speaks the same languages as humans [22], this allows any user to easily analyze and interpret the model output [16]. Additionally, the user is capable to augment the model by capturing their expertise in the rules [16].

In later sections of this paper, a Type-1 FLS and a Type-2 FLS are optimized using evolutionary algorithms and presented as an explainable AI model alternative for semantic segmentation of satellite images. An example of the rules of the generated Fuzzy Logic Systems (FLSs) is: IF hue is green, saturation is high, and value is medium THEN the pixel belongs to a tree. These rule based FLSs transform the data to linguistic concepts that humans can easily understand. This level of interpretation is not available in neural network architectures e.g. a neural network will receive crisp numbers for hue, saturation and value and output a class. Further comparison of these models is presented in Section V.

B. A Brief Overview of the Big Bang – Big Crunch (BB-BC) Optimization Algorithm

The BB-BC algorithm, presented by Erol and Eksin [23], is an evolutionary optimization technique inspired by the big bang theory about the evolution of the universe in Physics. This approach was selected because of its proven fast convergence when compared to other classical genetic algorithms [23]. The algorithm has two phases: The Big Bang (BB) phase and the Big Crunch (BC) phase. The former is where candidate solutions of a population are randomly generated, the latter is where the convergence to the optimal point happens by compressing all

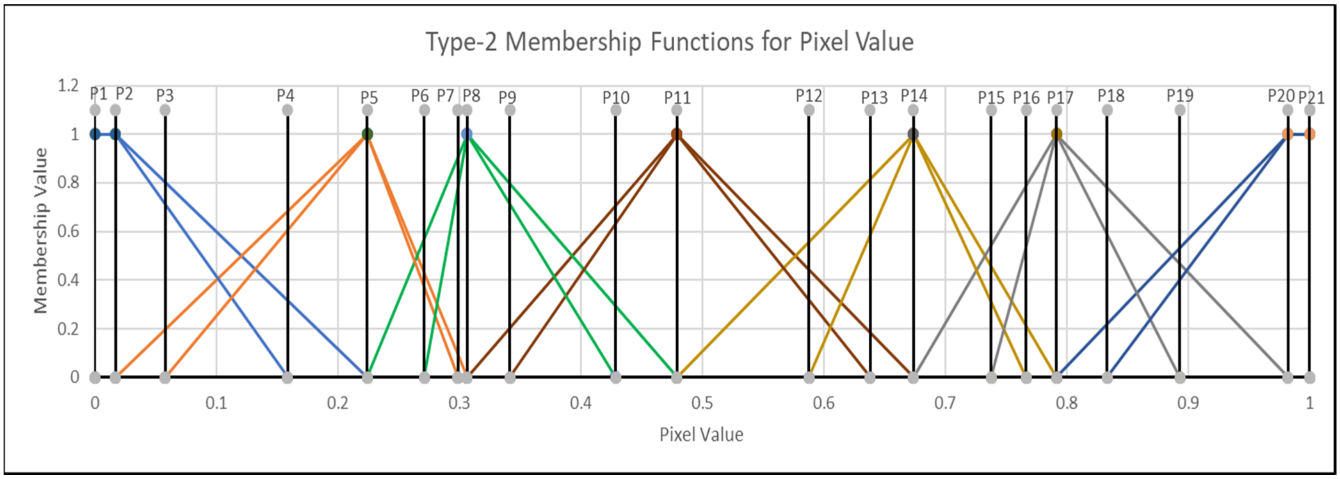

Fig. 1. The fuzzy set for (HSV) '*Value*' channel of a pixel. The points that are optimized by the BB-BC algorithm are shown in vertical lines.

solutions to a representative point via the center of mass. The steps describing the algorithm are as follows:

1) Big Bang phase

a) This phase is similar to other evolutionary search algorithms. An initial population of N candidates is randomly generated within the search space limits.

b) The cost function value of each of the candidates is computed.

2) Big Crunch phase

c) The point to which the algorithm converges is selected. Either the best fit candidate or the center of mass is chosen as the center point. The center of mass is calculated as:

$$x_c = \frac{\sum_{i=1}^{N} \frac{x^i}{f^i}}{\sum_{i=1}^{N} \frac{1}{f^i}} \qquad (1)$$

Where, x_c is the position of the center of mass, x^i is the position of the solution candidate, f^i is the cost function value of the i^{th} candidate and N is the population size.

d) The candidates of the new population will be created. New candidates are calculated around the center of mass in the form of:

$$x^{new} = x_c + \frac{lr}{k} \qquad (2)$$

Where, x_c is the previously computed center of mass, l is limit of the search space, r is a normal random number and k is the iteration or generation number.

e) Return to Step 1.a until stopping criteria have been met.

C. *The Proposed Optimization Method*

To optimize antecedent membership functions (MFs) using the BB-BC algorithm, the parameters have to be encoded into a form of a population [24], [25]. We have taken a look at successfully optimized FLS using Genetic Algorithms in [26]–[30] and a similar encoding approach as the one in [29] is used in the BB-BC algorithm. The parameters for a single Type-1 (T1) triangle or trapezoid membership function is a set of 4 points that describe the shape of the function. In the case of Interval Type-2 (IT2) MFs, at least two additional points are needed to represent upper and lower membership functions. The number of parameters to be optimized for each input is determined as follows:

$$Number\ of\ parameters\ for\ Type\ 1 = M + 2 \qquad (3)$$

$$Number\ of\ parameters\ for\ Type\ 2 = 3 * M \qquad (4)$$

Where, M is the number of membership functions for the input fuzzy set.

P_1^1	P_2^1	P_3^1	P_4^1	…	$P_{n_1}^1$	…	$P_{n_i-1}^i$	$P_{n_i}^i$

Fig. 2. The encoded points that represent all the membership functions from all the input fuzzy sets of the FLS.

The algorithm uses an encoded configuration in a list format like the one shown in Fig 2, where all the points of the fuzzy sets are stored in ascending order. In Fig. 2, the superscript represents the input number and the subscript is the point number from the input fuzzy set. The fuzzy sets from all inputs should have the following conditions:

1) First membership function is a trapezoid that starts at the lowest value of the set.
2) Last membership function is a trapezoid that ends at the highest value of the set.
3) The rest of the MFs are triangle shape.

Fig. 1, shows the optimized membership functions of the (HSV) 'Value' input fuzzy set, the vertical lines illustrate where the optimized parameters that represent the fuzzy set are located. Each candidate solution created during the Big Bang phase of the algorithm is a set containing parameters from the input fuzzy sets (Fig. 2), where i is the input number and n_i is the maximum number of parameters for the i^{th} input fuzzy set. The optimization process for both, the Type-1 and Type-2 FLS is the following:

1) Randomly initialize a population of N candidate solutions.
2) Each candidate describes a configuration for the membership functions. Use the training data to create a rule base for each candidate configuration.
3) Evaluate the rule base of each candidate.
4) In this paper, the best candidate solution is selected as the center of mass.
5) Start a new population. Create new parameters for each candidate around the center of mass by using (2).
6) Repeat steps 2 – 5 until the number of iterations is reached.

D. Rule base Generation

The rule base is generated by using the available training dataset in a similar process as the one described in [29]. First, we run the raw rule extraction process, which consists of reading all input-output training patterns from the training data and computing the membership value for each of the patterns. Then, IF-THEN rules are generated by combining matched fuzzy sets, i.e. fuzzy sets where the membership values are greater than zero. In case of type-2 fuzzy sets, it is matched by an input when either the upper or lower function is greater than zero.

When doing semantic segmentation of a single class most pixels in the training images will be background, therefore the dataset is unbalanced. To avoid any bias from the FLS towards the class with highest percentage of training patterns, the firing strength of each of the rules is scaled by using the following equations presented in [29]:

$$fs^{jt} = \frac{f^{jt}}{\sum_{j \in Class\ j} f^{j}} \tag{5}$$

Equation (5) states that the scaled firing strength (fs^{jt}) of a given rule t with a consequent Class C_j is given by dividing the non-scaled firing strength (f^{jt}) of the rule t by the summation of the firing strengths of all the rules in the rule base with an output Class C_j. This allows handling the imbalance of the data [29]. For Type-2 fuzzy sets, (5) will be used for the upper and lower firing strengths, resulting in two values: $\overline{fs^{jt}}$ and $\underline{fs^{jt}}$.

Many of the generated rules in the extraction process will be conflicting rules, i.e. they have the same antecedents but a different consequent. To resolve conflicts, confidence and support values for each rule will be used. The confidence that class C_q is the output class for the set of antecedents $\widetilde{A_q}$ is computed using the following equation:

$$c\left(\widetilde{A_q} \Rightarrow C_q\right) = \frac{\sum_{x_s \in Class\ C_q} fs^{jt}(x_s)}{\sum_{j=1}^{m} fs^{jt}(x_s)} \tag{6}$$

In Equation (6) the summation of the scaled firing strength of the conflicting rules where the expected output class for the training pattern x_s is C_q is divided by the summation of the firing strength of all conflicting rules, this is considered a validity measure for $\widetilde{A_q} \Rightarrow C_q$ [29].

The measure of support of a rule is viewed as the coverage of training patterns for $\widetilde{A_q} \Rightarrow C_q$. This can be computed as following:

$$s\left(\widetilde{A_q} \Rightarrow C_q\right) = \frac{\sum_{x_s \in Class\ C_q} fs^{jt}(x_s)}{m} \tag{7}$$

Where m, is the total number of available training patterns. When computing confidence and support for a Type-2 FLS, (6) and (7) should be applied to the upper and lower scaled firing strength, resulting in upper and lower values for each measure.

Using confidence and support measures we compute dominance of each rule as follows:

$$d\left(\widetilde{A_q} \Rightarrow C_q\right) = c\left(\widetilde{A_q} \Rightarrow C_q\right) * s\left(\widetilde{A_q} \Rightarrow C_q\right) \tag{8}$$

The rule with highest dominance value from all conflicting rules is the only one kept in the rule base. For Type-2 FLS, an upper and lower value of dominance is computed using upper and lower confidence and support measures, the rule with the highest average between upper and lower dominance is the rule selected.

V. EXPERIMENTS AND RESULTS

In this section we present the results of using a Multi-Layer Perceptron (MLP) Neural Network, T1FLS and T2FLS for semantic segmentation. The experiments were performed using images form the Inria Aerial Image Labeling Dataset [11]. Although CNNs are considered state-of-the-art [8], we are not including them in this comparison due to their inputs being different from the ones of T1 and T2 FLS, resulting in an unfair comparison for the models as the inputs may affect the performance. CNNs focus on learning filters used to extract features at different levels of the image through their convolution layers. The extracted features tend to be uninterpretable by humans, that is why they lack of interpretability [8]. On the other hand, an MLP network can use the same inputs as the T1 and T2 FLS, this allow us to manually extract a set of human interpretable features from the images and pass them to the different AI models.

The images in the dataset are RGB images and have a high resolution of 30 cm. The dataset was manually labelled for the class tree, the only object of interest for these experiments, hence we are evaluating the performance for binary classification. Finally, the selected images were transformed to HSV color space and processed to be stored as CSV (comma-separated values) files. Each row in the file contained the HSV values and the class for a given pixel in the images. The training dataset contains 20,000 training patterns, later divided into 80% training and 20% validation. The test dataset contains another 20,000 HSV pixel values. The limitations of only using pixel level information are accepted since the focus is on proposing an explainable AI model and comparing its performance against a black box.

The standard metric for comparing the performance of models when segmenting a given class is Intersection over Union (IoU), as it is a metric that considers the unbalance between background pixels and desired class pixels [31]. This measure value can be computed as follows [2]:

$$IoU = \frac{TP}{TP+FP+FN} \quad (9)$$

Where TP is the number of pixels correctly predicted as the given class, FP are background pixels predicted to be of the given class and FN are pixels from the given class predicted to be background. Accuracy, Dice and Recall metrics will be included in the results for completeness but the main comparison is by using IoU metric due to the unbalanced dataset and the use of this metric in the state of the art [31].

TABLE I. TEST SET RESULTS

	Metric Results			
	IoU	*Recall*	*Accuracy*	*Dice*
IT1 FLS	49.30%	56.40%	83.40%	66.10%
IT2 FLS	60.60%	70.20%	86.90%	75.50%
MLP	69.00%	77.30%	89.90%	81.70%

The MLP network architecture used is similar to the one presented in [32]. It uses the 3 HSV channels of the images as inputs, a hidden layer with 20 neurons and a single output neuron. We trained it with the previously described dataset, using TensorFlow with a stochastic gradient descent optimizer that had a learning rate of 0.3 and 0.8 for momentum. The best IoU measure obtained for tree class on the test set was 69.0% (Table 1).

Fig. 1. Example of image used for testing the models.

The membership functions that define each fuzzy set were optimized using the BB-BC algorithm described in Section IV. Different configurations of encoded parameters for membership functions selected using a heuristic approach, were optimized by running the BB-BC algorithm for 250 generations. During each generation, a population of 30 sets of encoded parameters was used, each of them created based on the current best solution. The same training dataset used by the neural network, was used with each of the encoded parameters set to generate a rule base and evaluate the performance of those parameters. The configuration with the best performance was the following:

- 16 Membership Functions for Hue.
- 18 Membership Functions for Saturation.
- 7 Membership Functions for Value.

TABLE II. MODEL COMPARISON

	Metric Results		
	IoU	*Degree of Explainability*	*Level of Understanding*
IT1 FLS	49.3%	High	Low
IT2 FLS	60.6%	High	Low
MLP	69.0%	Low	High

The same number of membership functions was used in both, the T1 and T2 FLS. The difference is the addition of the footprint of uncertainty for the Interval Type-2 FLS membership functions.

The Type-1 Fuzzy Logic System (T1FLS) has an IoU measure of 49.3%, while the Interval Type-2 Fuzzy Logic System (IT2FLS) has an IoU of 60.6%. The IT2FLS, which gave the best result, was still outperformed by the MLP network by 8.4%. However, the IT2FLS has the advantage over the MLP network of being a white box model, which means that a human can easily understand how the system predicted a given class. In addition to the high degree of explainability, the IT2FLS uses a rule base that can be modified, allowing human experts to augment and complement the model with their knowledge by adding or modifying rules. For example, end users can see rules like 'IF Hue is green AND Saturation is high AND Value is medium THEN label is tree pixel' and they will be capable of changing the output label of the rule to be 'background pixel' or augment the model capabilities by adding a new label like 'building'. This modification of the model does not require understanding of the training process since it is not being trained again, unlike the MLP.

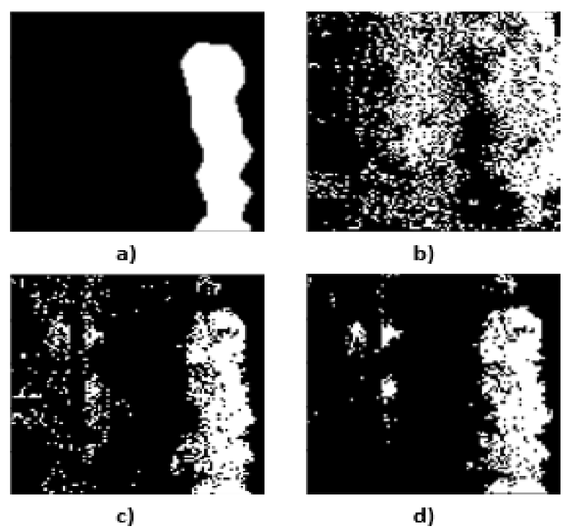

Fig. 2. Segmentation outputs for the image in Fig. 3. a) ground truth output, b) T1 FLS output, c) T2 FLS output and d) MLP network output

A comparison of the models where metrics like degree of explainability and required level of knowledge to modify the model are included in Table 2. The assigned values for degree of explainability and level of understanding needed to modify the model were given based on the analysis of interpretability in [16]. For this type of evaluation metrics, FL approaches are better than a black box model like an MLP network. The IT2FLS is a better explainable AI solution than the Type-1 since it performs 11.3% better, visual results are shown in Fig. 4 and it is considered to have the same degree of explainability and augmentation by end user [16], as shown in Table II. The Footprint of Uncertainty (FOU) in the IT2 FLS gave the system an IoU measure 11.3% higher than the T1 FLS. In Fig. 4, each of the pixels in these images is labeled as part of tree or background by each model. In Fig. 4d, the prediction is closer to Fig. 4a, however, Fig. 4c provides the user with an explanation for each assigned label via the T2FLS. The end user can understand when the model is working fine and when it is not, increasing with this the trust and credibility needed in real-world operation [6]. On the other hand, Fig. 4d shows a better performance from the MLP but there is no explanation for each assigned label and no ability to add any rules or expert knowledge to the model to improve its performance. In the MLP, the user will not understand when the model should be trusted.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, three models for performing semantic segmentation of trees in high resolution satellite images were presented: two approaches using fuzzy logic systems (Type-1 and Interval Type-2) and one using neural networks. The Interval Type-2 Fuzzy Logic System (IT2FLS) outperformed the Type-1 FLS (T1FLS) by an Intersection over Union (IoU) measure of 11.3%. The Multi-Layer Perceptron (MLP) network has the best performance of all three models with an IoU of 69.0%, which is 8.4% higher than the IT2FLS. However, the IT2FLS has a higher degree of explainability and augmentation of model by the end user. Utility companies required a high level of trust in the generated models [6] and in this paper we presented an alternative solution using Type-2 FLS that has a high level of explainability and a higher degree of trust compared to a black box model [16].

One of the main limitations from the presented work is the use of pixel level information, which limits the models to specific characteristics of certain objects instead of using features that allow to generalize. As part of our future work we expect to include more complex features that provide context information of the pixel and do not limit the model to the color characteristics of a certain area. In addition, we plan to explore the use of the output in the automatic creation of virtual environments for assisting humans in visualization and planning tasks.

REFERENCES

[1] R. Guo et al., “Pixel-wise classification method for high resolution remote sensing imagery using deep neural networks,” ISPRS International Journal of Geo-Information, vol. 7, no. 3, Mar. 2018.

[2] I. Demir et al., “DeepGlobe 2018: A Challenge to Parse the Earth through Satellite Images,” 2018.

[3] A. N. Skurikhin, N. G. McDowell, and R. S. Middleton, “Unsupervised individual tree crown detection in high-resolution satellite imagery,” Journal of Applied Remote Sensing, vol. 10, no. 1, p. 010501, 2016.

[4] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, “Semantic Segmentation of Small Objects and Modeling of Uncertainty in Urban Remote Sensing Images Using Deep Convolutional Neural Networks.”

[5] M. Volpi and V. Ferrari, “Semantic segmentation of urban scenes by learning local class interactions.”

[6] J. Hu, S. You, and U. Neumann, “Approaches to Large-Scale Urban,” no. December, pp. 62–69, 2003.

[7] K. Nogueira, M. Dalla Mura, J. Chanussot, W. Robson Schwartz, and J. A. Dos Santos, “Learning to Semantically Segment High-Resolution Remote Sensing Images.”

[8] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 818–833, 2014.

[9] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, “What’s the Point: Semantic Segmentation with Point Supervision,” Jun. 2015.

[10] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, “High-Resolution Aerial Image Labeling with Convolutional Neural Networks,” IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 12, pp. 7092–7103, 2017.

[11] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, “Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark,” International Geoscience and Remote Sensing Symposium (IGARSS), pp. 3226–3229, 2017.

[12] C. Amza, “A Review on Neural Network-Based Image Segmentation Techniques.”

[13] C.-Y. Wen and C.-M. Chou, “Color Image Models and its Applications to Document Examination,” 2002.

[14] D. J. Bora, A. Kumar Gupta, and F. A. Khan, “Comparing the Performance of L*A*B* and HSV Color Spaces with Respect to Color Image Segmentation,” Internation Journal of Emerging Technology and Advanced Engineering, vol. 5, no. 2, 2015.

[15] J. M. Mendel, Uncertain Rule-Based Fuzzy Logic Systems Introduction and New Directions, 1st editio. Prentice-Hall, Inc., 2001.

[16] H. Hagras, “Toward Human-Understandable, Explainable AI,” Computer, vol. 51, pp. 28–36, 2018.

[17] J. Andreu-Perez, F. Cao, H. Hagras, and G. Z. Yang, “A Self-Adaptive Online Brain-Machine Interface of a Humanoid Robot Through a General Type-2 Fuzzy Inference System,” IEEE Transactions on Fuzzy Systems, vol. 26, no. 1, pp. 101–116, 2018.

[18] C. Lynch, H. Hagras, and V. Callaghan, “Embedded interval type-2 neuro-fuzzy speed controller for marine diesel engines,” in Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, 2006, pp. 1340–1347.

[19] H. Hagras and C. Wagner, “Introduction to Interval Type-2 Fuzzy Logic Controllers - Towards Better Uncertainty Handling in Real World Applications,” The IEEE Systems, an and Cybernetics eNewsletter, no. 27, 2009.

[20] A. Sakalli, T. Kumbasar, E. Yesil, and H. Hagras, “Analysis of the performances of type-1, self-tuning type-1 and interval type-2 fuzzy PID controllers on the Magnetic Levitation system,” in Proceedings of the 2014 IEEE International Conference on Fuzzy Systems, 2014.

[21] C. Wierzynski, “The Challenges and Opportunities of Explainable AI,” 2018. [Online]. Available: https://www.intel.ai/the-challenges-and-opportunities-of-explainable-ai/.

[22] A. Steventon and S. Wright, “Programming iSpaces: A Tale of Two Paradigms,” in Intelligent Spaces: The Application of Pervasive ICT, 2005, pp. 389–421.

[23] O. K. Erol and I. Eksin, “A new optimization method: Big Bang-Big Crunch,” Advances in Engineering Software, vol. 37, no. 2, pp. 106–111, 2006.

[24] B. Yao, H. Hagras, D. Alghazzawi, and M. J. Alhaddad, “A big bang-big crunch optimization for a type-2 fuzzy logic based human behaviour recognition system in intelligent environments,” Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013, pp. 2880–2886, 2013.

[25] T. Kumbasar and H. Hagras, “Big Bang-Big Crunch optimization based interval type-2 fuzzy PID cascade controller design strategy,” Journal of Information Sciences, vol. 282, no. October, pp. 277–295, 2014.

[26] M. Antonelli, D. Bernardo, H. Hagras, and F. Marcelloni, “Multiobjective Evolutionary Optimization of Type-2 Fuzzy Rule-Based Systems for

Financial Data Classification," IEEE Transactions on Fuzzy Systems, vol. 25, no. 2, pp. 249–264, 2017.

[27] A. Starkey, H. Hagras, S. Shakya, and G. Owusu, "A multi-objective genetic type-2 fuzzy logic based system for mobile field workforce area optimization," Journal of Information Sciences, vol. 329, pp. 390–411, 2016.

[28] H. Hagras, M. Colley, V. Callaghan, and M. Carr-west, "Online Learning and Adaptation of Autonomous Mobile Robots for Sustainable Agriculture," Autonomous Robots, vol. 13, pp. 37–52, July 2002.

[29] D. Bernardo, H. Hagras, and E. Tsang, "A Genetic type-2 fuzzy logic based system for financial applications modelling and prediction," IEEE International Conference on Fuzzy Systems, 2013.

[30] H. Hagras, V. Callaghan, M. Colley, and M. Carr-West, "A Fuzzy-genetic based embedded-agent approach to learning and control in agricultural autonomous vehicles," in Proceedings - IEEE International Conference on Robotics and Automation, 1999, no. 2, pp. 1005–1010.

[31] S. Ohleyer, "Building segmentation on satellite images." 2018.

[32] R. S. Sarkate, K. P. B, and T. S. B, "BPNN Approach in Pixel Classification based Precision Segmentation for Agriculture Images," in IJCA Proceedings on National Conference on Recent Advances in Information Technology, 2014, pp. 25–27.