An Explainable AI Approach to Process Data in Mixed Reality Environments for Field Service Operations

Hugo Leon Garza

A thesis submitted for the degree of Doctor of Philosophy in Computer Science

School of Computer Science and Electronic Engineering

University of Essex

2023

Acknowledgements

I would like to thank the following people that supported me, guided me, and contributed to the success of this PhD.

Professor Hani Hagras, my academic supervisor, I am eternally grateful for his guidance and support throughout my PhD study. He taught me invaluable lessons and pushed me to my limits to help me deliver high-quality work.

I am sincerely grateful to Dr. Anasol Peña Rios, my industrial supervisor, for her invaluable support and mentorship. She kindly shared with me all her knowledge and guided me through the difficulties of a PhD.

I am also grateful to Anthony Conway and Dr. Gilbert Owusu for their support during these last 4 years. Their expertise and advice guided my research and enabled it to be applied to current challenges.

I am also thankful to British Telecom for funding and supporting my PhD study.

Finally, but not least, I am very grateful to Hugo, Adriana, and Alejandra, my family who are always supporting me.

Publications and Patents Arising from this Thesis

Journal Papers

- H. Leon-Garza, H. Hagras, A. Peña-Rios, A. Conway, and G. Owusu, 'A type-2 fuzzy system-based approach for image data fusion to create building information models', *Information Fusion, vol. 88, pp. 115–125, 2022, doi: https://doi.org/10.1016/j.inffus.2022.07.007.*

Conference Papers

- H. Leon-Garza, H. Hagras, A. Peña-Rios, G. Owusu, and A. Conway, "A Fuzzy Logic Based System for Cloud-based Building Information Modelling Rendering Optimization in Augmented Reality", *Proceedings of the 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Glasgow, July 2020.*

- H. Leon-Garza, H. Hagras, A. Peña-Rios, A. Conway, and G. Owusu, "A Big Bang-Big Crunch Type-2 Fuzzy Logic System for Explainable Semantic Segmentation of Trees in Satellite Images using HSV Color Space", *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Glasgow, July 2020.*

- H. Leon-Garza, H. Hagras, A. Peña-Rios, G. Owusu, and A. Conway, "Type-1 Fuzzy Rule-based System using Patch-based Approach for Semantic Segmentation in Floor Plans", *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Luxembourg, July 2021.*

- H. Leon-Garza, H. Hagras, A. Peña-Rios, A. Conway, and G. Owusu, "An Interval Type-2 Fuzzy-based System to Create Building Information Management Models from 2D Floor Plan Images", *Proceedings of the* IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), *Luxembourg, July* 2021.

- H. Leon-Garza, H. Hagras, A. Peña-Rios, O. Bahceci, and A. Conway, "A Hand-Gesture Recognition Based Interpretable Type-2 Fuzzy Rule-based System for Extended Reality", *Proceedings of the 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, October 2022, pp. 2894–2899. doi: 10.1109/SMC53654.2022.9945407.*

Patents

- Pena-Rios, H. Leon-Garza, H. Hagras, A. Conway, and G. Owusu, *"Rendering of spatial environments", GB2591103A.*

- Pena-Rios, A. Conway, G. Owusu, H. Leon-Garza, and H. Hagras, *"Generating three-dimensional data models of two-dimensional floor plans", WO2022194674A1.*

Abstract

Digital Twins is a concept that describes how physical objects can be represented and connected to the virtual world, the main goal of a Digital Twin is to centralise all the available information of an object of interest in a single virtual model. The Digital Twin consist of three main components: the physical object, a virtual representation of that object (typically a 3D model), and a real-time connection between both objects so that any change can be communicated to the other part. The possibility of understanding, visualising, and interacting with physical objects through a virtual environment is, at a very high level, the main benefit of using Digital Twins.

The adoption of this concept has grown a lot in the recent years in industries such as the manufacturing, construction, health, and energy. Utility companies in the telecommunication industry, water services, and gas services are still falling behind in the adoption of these new concepts. The potential benefit for these sectors is huge where some of these benefits are real-time remote monitoring, predictive maintenance, scenario and risk assessment, better collaboration between stakeholders (internal and external), and better documentation.

Existing Mixed Reality, Virtual Reality and Augmented Reality technologies can help with the interaction and visualisation of the virtual twin. The different levels of reality in combination with the digital twins will help with different tasks, for example, Virtual Reality is useful for remote tasks were most of the interaction happens with the virtual twin and Augmented Reality will help bringing the virtual twin and all its information to onsite tasks to help field engineers.

However, there are different challenges when trying to connect all the different components and some of these challenges did slow down the adoption of these technologies by the utility

companies. The research work in this thesis will focus on two main challenges: the cost of creating these digital twins from existing sources of information and the lack of an explainable AI approach that can be used as an enabler for the interaction between human and Digital Twin in the mixed reality environment.

To address the challenge of automating the creation of digital representations at a low cost, two interval type-2 Fuzzy Rule-based Systems are presented as the best explainable AI alternatives to the opaque AI models for processing images and extracting information of the objects of interest. One of them was used to extract information about trees in a satellite image and generate a 3D representation of the geographic area combined with terrain data. This will be used for remote scenario and risk assessment and prediction of the telecommunication equipment getting damaged by natural elements like trees. The proposed approach achieved an 86.90% of accuracy, 3.5% better than the type-1 but 3.0% worse than the opaque Multilayer Perceptron model.

The second interval type-2 Fuzzy Rule-based System is an explainable AI model that incorporates the use of context information in its rule to process 2D floor plan images, identify elements of interest and create a 3D digital representation of the building floors. This will benefit the telecommunication company by automating, at a low cost, the process of creating a more detailed in-building map with the telecommunication assets and improve the collaboration with external stakeholders like contractors for maintenance tasks or construction companies for any works in the building. The proposed method achieved a 97.5% Intersection over Union metric value which was comparable to the 99.3% Intersection over Union of the opaque Convolutional Neural Network model, however our proposed solution is highly interpretable and augmentable by human experts which cannot be achieved via opaque box AI models.

Additionally, another interval type-2 Fuzzy Rule-based System for hand gesture classification is also presented in this thesis. This rule-based system achieved a 96.4% accuracy, and it is an easily adjustable model that can be modified to include more hand gestures, the opaque model alternative, a K-Nearest Neighbour algorithm achieved a 98.9% accuracy, however, this model cannot be easily modified by a human expert and re-training is needed which results in a cost of time. This hand gesture recognition model, alongside another fuzzy rule-based system, will help to address the challenge of the interaction between human and digital twin objects in Mixed Reality environments.

Table of Contents

List of Figures

List of Acronyms

3D – 3-Dimension

AI – Artificial Intelligence

AR – Augmented Reality

BB-BC – Big Bang-Big Crunch

BIM – Building Information Modelling

DT – Digital Twin

ER – Extended Reality

FL – Fuzzy Logic

FLS – Fuzzy Logic System

FRBS – Fuzzy Rule-based System

GA – Genetic Algorithms

HSV – Hue, Saturation, and Value colour space

IoT – Internet of Things

IoU – Intersection over Union

MLP – Multi-Layer Perceptron

MR – Mixed Reality

NDT – Network Digital Twin

NDTp – National Digital Twin programme

UK – United Kingdom

VR – Virtual Reality

XAI – Explainable Artificial Intelligence

Chapter 1. Introduction

1.1 An Introduction to the Concept of Network Digital Twin

The new technological advancements in the recent years have enabled the connectivity of multiple devices and entities across the world. Large quantities of data are generated daily by companies, people, and even machines or systems generate their own historical data. Companies seek to use the data that they are constantly capturing to be better in different aspects, e.g., improve the quality of their services and products, reduce the cost of their processes, guarantee safety of their employees, and prevent failures in their systems.

All the data is stored in large size databases with thousands or millions of rows. In recent years a new concept for modelling and representing the information of objects called "Digital Twins", has been proposed [1], [2]. The digital twin concept proposes the idea of creating a digital representation for an existing object and linking both entities (physical and digital) through a remote connection that enable the exchange of information. Essentially, the idea is to go from having the information of physical objects stored across multiple tables in one or more databases, to have a virtual environment with a replica of the object containing all the information. This is an attractive and game changer concept for organisations with thousands of physical assets that need to be constantly monitored. Digital twins used technologies such as Internet of Things (IoT) to establish a communication between both entities and mixed reality environments for the user to interact with the digital representation. Utility companies (i.e., water service, gas, electric, telecommunication) are part of the organisations with a high interest in developing a digital twin of their assets. The goal for utility companies is to create a Network Digital Twin (NDT), which is a collection of connected digital twins, each of them representing an asset in a big network, replicating with this the actual physical network, e.g., telecommunications company having a virtual world with a 3D digital map of every cable and equipment in the country. The idea of having an engineer do network planning tasks remotely using the virtual world with the digital twins is big cost reduction, considering that before the engineer needed to attend the site to do the survey and analysis.

The UK government has its *National Digital Twin programme* (NDTp) that they use to push utility companies to develop their network digital twins, to then combine all of them in a single national digital twin of all the utility services assets in the county. The UK government believes that the country's society, economy, business and environment will benefit from this digital twin [3], [4].

Utility companies already have large numbers of data about their assets; however, this is not always in a structured format and tends to be a numeric table format. The challenges of a Network Digital Twin are in two areas: 1) creating the digital representations with the available data and 2) interacting with some of the digital twin assets through Mixed Reality (MR) environments.

The cost for utility companies of creating the Network Digital Twin is high if it must be made manually. This thesis presents some alternative solutions for automating part of the creation of digital twins which will reduce the cost. Additionally, solutions presented in later chapters are interpretable models that a human can understand and modify. In this way utility companies can still take advantage of their expert engineers and capture their knowledge in these explainable models.

1.2 Aims of the Thesis

The work in this thesis aims to present novel explainable AI models that can automate the process of creating 3D representations of physical objects in a virtual world to contribute to the idea of the Network Digital Twin. Additionally, this thesis also seeks to create a novel explainable AI approach for guiding the user in this mixed reality environments that contain information from the network digital twin. Deep learning and traditional machine learning models are “black box” models, i.e., they are systems that receive inputs, perform an unknown process with those inputs, and return an output to the user. Their main advantage over other approaches is that these models have a higher performance, and their outputs tend to be more accurate. The disadvantages are that large volumes of data are required to achieve a high-performance level, it is practically impossible for a user to understand the inference process and modify it using their knowledge, and these models require the data to be labelled which has a high cost for acquiring this type of data (cost can be in time, money, or both).

The research in this thesis focuses on evaluating the capabilities of a fuzzy rule-based system (FRBS) as an explainable AI alternative for processing image data. A comparison against a black box model is provided for each of the use cases.

The aims of the research are:

1) To understand if an explainable AI alternative is viable for image processing and if the performance is close to the one of a black box model.
2) To create an explainable system that extracts information from 2D image sources to then generate 3D digital representations automatically for the objects of interest in different type of images.
3) To develop an explainable and augmentable model that understands the hand gesture of a user in a mixed reality environment and if the gesture is expected for the task that is being performed.
4) To guarantee a high degree of interpretability in the developed systems so that the user understands the inference process and can augment it using human expert knowledge.

1.3 Thesis Layout

The thesis is organized in a way that the first part covers the basic concepts needed to understand and implement explainable AI models. Then the application of these explainable AI models to the processing of images is explored and analysed. Finally, the thesis concludes with a description on a proposed explainable AI system for guiding a user in a Mixed Reality environment.

The structure of the thesis chapter by chapter is as follows; Chapter 2 provides a detailed description of fuzzy logic systems, the basic concepts, advantages and how to use fuzzy logic to build a classifier.

Chapter 3 describes the selected Big Bang-Big Crunch optimisation strategy for improving the performance of the fuzzy rule-based systems. It describes what are the three main components to optimise and how each of the components affect the performance and the interpretability of a FRBS. Additionally, it describes how to encode the FRBS into a numeric vector that can be then optimised.

Chapter 4 provides an overview of the concept of digital twins and their applications. This chapter tries to highlight the different challenges for companies that try to integrate the use of digital twins in their processes. These challenges and high-cost tasks of integrating everything can be minimised by the automating the process of creating digital twins, the value and purpose of the work of the next chapters is highlighted in this chapter.

Chapter 5 presents the first application of a FRBS for processing satellite images and automatically generate a 3D object from the output of those images. The type-2 FRBS outperforms the type-1 FRBS, however, the black box model solution has a significantly higher performance value. This chapter discusses the benefits of using an explainable AI model.

Chapter 6 presents a novel interval type-2 FRBS solution that expands on the approach from chapter 5, this new solution incorporates the use of context information which was something missing in the first implementation in chapter 5 of a FRBS for semantic segmentation. The context information is obtained using a similarity measure between patches of images. The advantages of choosing a solution like the FRBS are discussed in this chapter, however, the black box alternative performed better than the FRBS. Although, in this case the difference was considerably smaller than in chapter 5. The models in this chapter were used to automatically create 3D digital representations of a building floor which will help utility companies to connect their outside network with the inner network and elements of the floor.

Chapter 7 starts discussing the need to consume this digital twin information. The digital twin loses a lot of value if it remains as a centralised database of information instead of being used as an object in mixed reality environment. This chapter also presents a detailed description of an overall architecture on how two FRBSs can be applied to combine and process multiple inputs and then return some form of feedback to the user.

Finally, chapter 8 presents the conclusions and future work of the thesis.

Chapter 2. An Overview of Fuzzy Logic Systems

2.1 A brief description on fuzzy logic

Fuzzy Logic (FL) is a form of logic that can handle the uncertain and imprecise nature of human thinking and communication process. The main ideas and concepts were introduced by Lotfi Zadeh in "Fuzzy Sets" [5], where in this paper Zadeh states that most of the classes (or sets of values) that humans use to define and classify objects in the real world are ambiguous, i.e. it is possible to have objects that can be part of multiple classes depending on the context, and even in the same context the boundaries are not clearly defined.

The height of a person can be used as an example of the ambiguity in classes. For example, consider a male person with a height of 180 centimetres (cm), he will be considered of "average height" in the United Kingdom, he will be classified as "tall" in Mexico, and he will be a "short" player in the National Basketball Association (NBA) tournament in the United States. The same person classified differently depending on the context, this makes the human communication process complex and ambiguous, to fully understand what it is meant by "that person is tall" a lot of context information is needed. If context information is available, it is possible to adapt the class to a specific context, e.g., to be considered "tall" in the NBA context you must be above a certain height. However, there is still uncertainty in defining the classes even if context information is available. For example, in the NBA context it can be defined that any player above 200 cm will be considered "tall". Does this mean that a basketball player with a height of 199 cm is not tall? Is one centimetre enough difference to change the class? These classes are arbitrarily defined based on the perspective of a certain group of people and although most people will agree on the average characteristics of a class, there is high uncertainty and ambiguity on the boundaries of a class, which makes it difficult to understand when an object stops belonging to one class.

Elements in traditional logic either belong to a class or they do not (true or false, 0 or 1), which implies that the boundaries are absolute truth, and everything inside the limit is part of the class, with no exception. Additionally, it also implies that all elements of a set have the same membership value, which does not happen. For example, using again the height of a person, if it is assumed that any person with a height between 180 cm and 200 cm is "tall", in reality we know that a person with a height of 180 cm will not be equally "tall" as a person with a height of 195 cm, the latter will be visibly taller, therefore, have a stronger (or higher) belonging to the class "tall".

Fuzzy Logic seeks to handle this ambiguity in classes and instead of assigning a 1 or a 0 (it does or does not belong) to an element in a class, it assigns a degree of membership, which describes how much an element belongs to a given set (or class).

The following sections of this chapter provide an overview about Fuzzy Logic Systems (FLSs), their basic components, how they are created, how they work, and how they use Fuzzy Logic to handle uncertainty in the inputs. Furthermore, there are different types of fuzzy logic that will be described in this chapter alongside an explanation of why rule-based models using fuzzy logic can be considered Explainable AI models and how they will be used in the rest of this thesis.

2.2 Type-1 Fuzzy Logic Systems

A fuzzy logic system (FLS) is a nonlinear mapping of crisp inputs to crip outputs by using a rule base with linguistic labels for the inference process [6]. The process of the FLS has three main stages: 1) map the crisp numeric value to a linguistic label (i.e., a human understandable word), 2) Find the rule (or rules) that have the linguistic labels that were mapped to the crisp input values and get the output linguistic labels, and 3) convert the output linguistic labels to a crisp numeric value that can then be used by other computer-based system that only understand numbers. Fig. 2.1 shows the components of a FLS and how these components are connected. The fuzzifier component is used in the first stage where crisp numeric inputs are mapped to linguistic labels. The inference engine component consists of If-Then rules of linguistic labels, these are used in stage 2 of the FLS process to map input linguistic labels to output linguistic labels. The defuzzifier component is used in stage 3 of the FLS process to map the output linguistic labels to a crisp value, this is important so that the output can be used outside the FLS by other systems. For example, if a FLS is used to control at what distance a vehicle should start breaking the output that goes to the vehicle has to be numeric because the vehicle will not understand "start breaking when you are close", "close" is human label used to define a set of valid distance values for starting to stop the car.

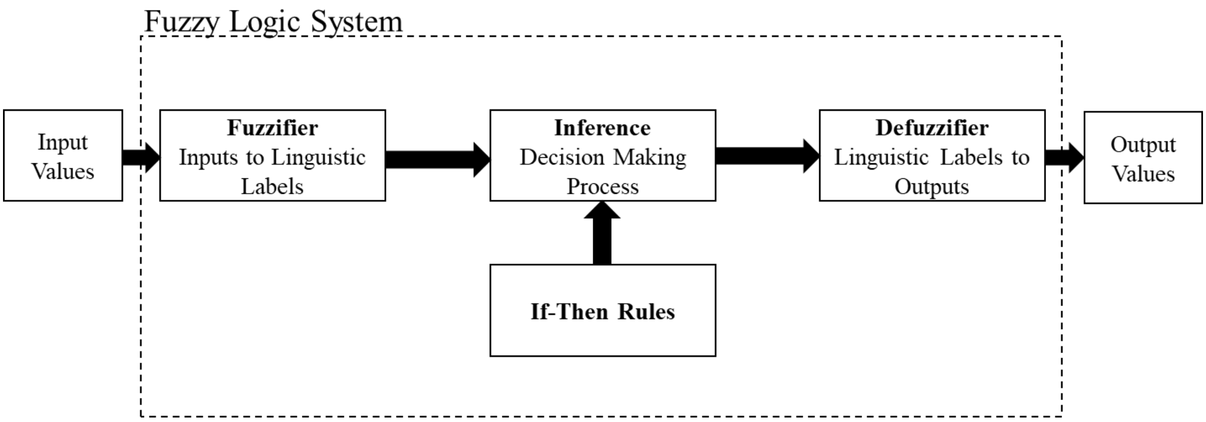


Figure 2.1. The components in a fuzzy logic system, image based on [6].

2.2.1 Type-1 Membership Functions

As mentioned earlier, the first step of the FLS is to convert the crisp inputs to linguistic labels and this is done by the fuzzifier component. Linguistic labels as defined by Zadeh [5], [7] are variables whose values are not numbers but words that represent a set of valid numeric values to describe a class. In other words, a linguistic label is a variable that describes what a range of values mean, e.g., consider the linguistic label "close", distance values from 0 to 5 meters are a "close" distance. The difference in fuzzy logic is that linguistic labels are described by a function that assigns a numeric value from 0 to 1 to each of the input values in the set, this is called degree of membership. The function that assigns the degree of membership is referred to as membership function, and there are different types and shapes of functions.

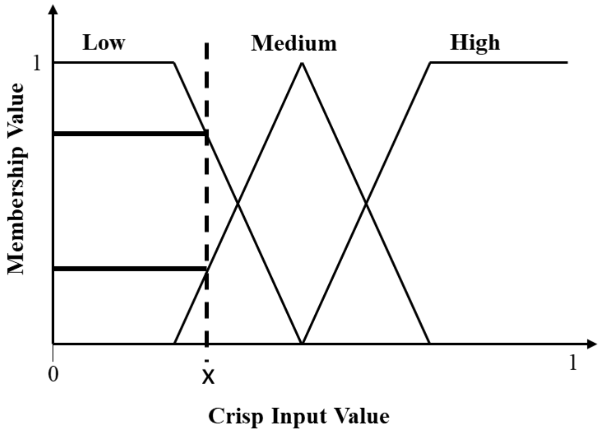


Figure 2.2. Example of a fuzzy set using Type-1 Membership functions. Dashed lines represent an example input value 'x'.

Fig. 2.2 shows and example of a membership function to calculate $\mu(x)$ which is the degree of membership of an input value x. The y axis represent the degree of membership of the input in the x axis, the highest value is always 1. The membership degree for a triangular fuzzy set can be computed as follows:

$$\mu(x) = \frac{x-a}{b-a} \; when \; a \leq x \; \leq b \qquad (2.1)$$

$$\mu(x) = \frac{c-x}{c-b} \; when \; b < x \; \leq c \qquad (2.2)$$

$$Otherwise, \mu(x) = \; 0$$

Similarly, equations 2.3 – 2.5 are used for a trapezoid shape these are as follows:

$$\mu(x) = \frac{x-a}{b-a} \; when \; a \leq x < b \qquad (2.3)$$

$$\mu(x) = \; 1 \; when \; b \leq x \; \leq c \qquad (2.4)$$

$$\mu(x) = \frac{d-x}{d-c} \; when \; c < x \; \leq d \qquad (2.5)$$

$$Otherwise, \mu(x) = 0$$

The singleton fuzzification process, is completed once the membership functions have calculated the membership value of each input. The next step is to use the linguistic labels and their degree of membership for the inference process.

2.2.2 Rules and Inference Process

In this stage, the inference process of a fuzzy logic system decides which rules are relevant to the input data based on the degrees of membership that were computed. This stage is the one that maps the input values to an output value (the output value is the consequence part of the rule) using linguistic variables.

Rules are the logic and the core component of the inference process, they are in a form of If-Then statements, the "If" is the inputs section, and the "then" has the desired output for the specific combination of inputs in the "if" part.

$$R_1 = IF\ x_1\ is\ F_1^l\ and\ \ldots.\ and\ x_l\ is\ F_m^l\ \ THEN\ y_1\ is\ C_1^k \qquad (2.6)$$

Equation 2.6 shows an example of a rule, x_1 is an input value that belongs to fuzzy set F^l, with $l = 1,2, \ldots N$. N is the total number of antecedents or fuzzy sets and m is the number of membership function inside the fuzzy set used to represent the input value x_i.

Once the degree of membership for each value is calculated then the rule needs to calculate the firing strength of the complete rule. To do this the following Equation 2.7 is used (using the min t-norm to represent the AND logical connective):

$$f = \min\ [\mu(x_1), \ldots\ , \mu(x_l)] \qquad (2.7)$$

The firing strength of the rule is calculated by computing the minimum value between the membership values of all the antecedents in the rule. For example, consider a rule with two antecedents A_1 *and* A_2, if $\mu(A_1) = 0.8$ *and* $\mu(A_2) = 0.4$, then the firing strength of the rule will be $f = 0.4$.

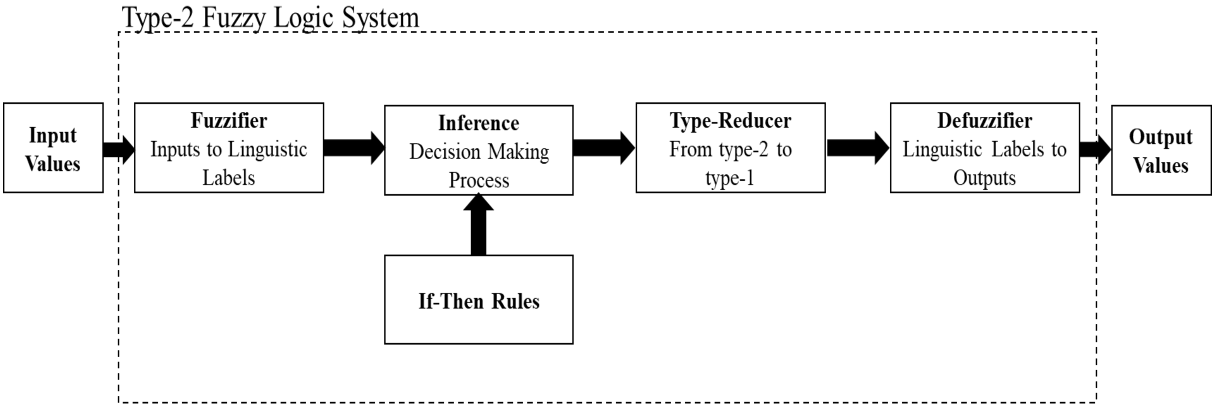


Figure 2.3. Type-2 Fuzzy Logic System. Image based on [8].

2.3 Type-2 Fuzzy Logic Systems

Type-1 fuzzy logic systems have limitations when modelling and minimising the uncertainty in inputs and outputs. Although the type-1 fuzzy membership functions are designed to handle uncertainty by having soft limits and allowing input values to be part of different sets, there is still a degree of uncertainty in the arbitrary definition of the boundaries. Why does input x has a 0.5 membership value for the linguistic label "close" instead of a 0.6 membership value? The answer is that it was an arbitrary decision of human expert, this is the kind of uncertainty that type-1 fuzzy logic systems still have. Expert knowledge may differ between users, which is the type of uncertainty type-1 fuzzy sets cannot handle. A type-2 membership function has a footprint of uncertainty (FOU) to handle type-1 uncertainty [6], [9].

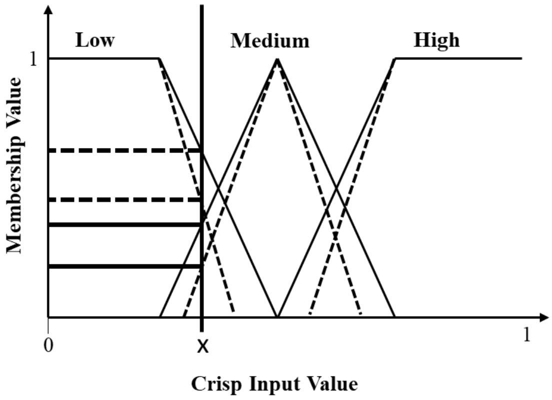


Figure 2.4. Example of a fuzzy set using type-2 Membership Functions. The solid vertical line represents and input value 'x'. The horizontal solid and dashed lines represent the ranges of membership value that describe the input 'x'.

2.3.1 Type-2 Membership Function

Fig. 2.4 shows an example of a fuzzy set using type-2 Membership functions to describe each of the linguistic labels. These types of function assign a range of values as a degree of membership instead of a single crisp value. This is to represent the uncertainty of how much an input belongs to a set. The membership functions shown in Fig. 2.4 are interval type-2 membership function, the name comes from the fact that they use a range or an interval to represent the degree of membership of an input. To calculate the degree of membership of a type-2 membership functions the same Equations from 2.1 to 2.5 are used, however, the difference is that now they are applied to the lower limit of the function and to the upper limit.

$$\bar{\mu}(x) = membership\ value\ of\ x\ for\ the\ upper\ function$$

$$\underline{\mu}(x) = membership\ value\ of\ x\ for\ the\ lower\ function$$

Every type-2 membership function will have two values of degree of membership that represents the lower end of the interval and the upper end of the interval.

2.3.2 Footprint of Uncertainty

The FOU The FOU is created by blurring a type-1 membership function's line to the left and right. This means that some points in the set will have a range of values as a membership degree instead of a crisp value. Interval type2 membership functions use the upper and lower bounds of the range to describe the firing strength of the membership function. The FOU gives interval type-2 fuzzy logic systems additional degrees of freedom beyond those in type-1 fuzzy logic systems.

2.3.3 Type Reduction

In type-2 fuzzy logic systems, there is a type-reduction process that converts the output fuzzy sets from a type-2 to a type-1 fuzzy set, so that it can then be reduced again from a type-1 to a crisp numeric output [6]. This type-reduction step is shown in Fig. 2.3. More details can be found in [8].

2.4 Fuzzy Rule-based System Classifiers

Fuzzy rule-based system (FRBS) classifiers have the same components as a fuzzy logic system except for the defuzzifier, and the reason is that the output of the classifier is a class, i.e., the linguistic label of the consequence part of the triggered rule is used as the output class. If multiple rules are activated then a voting system, using the firing strength of the rule as the vote strength, is used to determine the output class.

2.4.1 Rule Modelling Process

The rule base of a FRBS can be constructed either by 1) human experts or 2) using data from the problem. If data is going to be used for finding the rules, then a rule modelling process is needed. The idea of the rule modelling process is to replace human expert knowledge by extracting from data the consequence for all possible rules. The dataset contains pairs of input data in the format of (x_i, C_j) where x_i is a vector of input values and C_j is the expected class label for that input vector.

For each rule, the firing strength $f(x_i)$ is computed using the membership functions of each antecedent. This value measures the vector x_i belonging to the fuzzy region of that rule. If type-2 fuzzy logic is used then two values define the firing strength $f(x_i)$, the lower $(\underline{f}(x_i)$ and the upper $(\overline{f}(x_i)$ bounds of the interval type-2 membership functions.

The rules that are fired with the input vector x_i will get assigned the consequence label C_j that is paired with x_i.

Two challenges of the automated rule modelling process are:

1) There might be a situation where a possible combination of antecedents does not get assigned a consequence because there is no input data that covers this combination of antecedents. In this case, the rule is not added to the rule base until a human expert or additional training data can provide a consequence for it. If the rule is needed in the prediction phase, then a similarity measure can be used to find a prediction using other rules that already have a consequence assigned, this similarity concept is discussed in section 3.5.1.
2) The second challenge of the rule modelling process is how to handle conflicting rules. Two or more conflict between each other when they share the exact same combination of antecedents, but they were assigned a different consequence.

Using the values of confidence, support, and dominance, it is possible to solve the conflicts in these rules, i.e., decide which of the multiple consequences assigned to the exact same combination of antecedents is the most appropriate.

The confidence value for a class q is used to measure how certain is the rule modelling process that a class q is the consequence for a given set of antecedents. The firing strength of the rule is expected to be higher when fired with input data that has class q as the expected consequence. When working with interval type-2 fuzzy logic systems two confidence values are needed, one for the lower and another one for the upper bounds used to define the fuzzy sets, but the calculation is the same.

Equation 2.8 computes the confidence for class q, the summation of the firing strength of the rule for all the training patterns with an expected class q is divided by the summation of the firing strength of the rule for all training patterns [9]–[11] . For interval type-2 FLS, the upper bound confidence measure calculation uses only the upper bound firing strength, in a similar way, the lower bound confidence uses the lower bound firing strength.

$$confidence\ for\ class\ q, c(A_q \Longrightarrow C_q) = \frac{\sum x_i \in Class\ C_q\ f\ (x_i)}{\sum_{j=1}^{m} f^j(x_i)} \tag{2.8}$$

The support value for a class q is used to measure the coverage of training patterns for a given rule, i.e., if there is enough evidence in the dataset to assign the class q as the consequence. This is to try to avoid any outliers or mislabelled pairs of input data with high confidence to mislead the assignment of the consequence [9]–[11]

Equation 2.9 is used to compute the support for class q. The summation of the firing strength of the training patterns with an expected class q is divided by the total number of training patterns [9]–[11]. In a similar way to the confidence measure, if interval type-2 fuzzy logic is used then two support values are calculated, one for each bound describing the fuzzy set.

$$support\ for\ class\ s(A_q \Longrightarrow C_q) = \frac{\sum x_i \in Class\ C_q\ f(x_i)}{m} \tag{2.9}$$

Finally, the dominance value for class C_q is calculated by multiplying the confidence and support values for C_q. The conflicting rule with the highest dominance value is selected and added to the rule base. The use of dominance-based approach for the rule inference is the low complexity and computationally efficient process; additionally, this approach is known for being capable of handling the uncertainty in an imbalanced dataset, i.e., there is not the same number of examples for each class in the training dataset. When working with images most of the datasets tend to not be perfectly balanced for all classes, which can cause an issue if we are building rules from the data. This dominance-based approach can be easily adapted to handle this imbalance by just modifying the firing strength used in the equations, the following sub-section describes how to calculate the scaled firing strength and use it to find the dominance value.

2.4.2 Scaled Firing Strength

One of the challenges of extracting rules from data is how to handle an imbalanced dataset, i.e., when there are considerably more input pair examples of one class. This kind of imbalance affects the confidence, support, and dominance metrics used in the rule modelling process by favouring the class with a larger number of input pairs.

It is possible to scale the firing strength to handle the imbalance in the dataset. The scaled firing strength of a rule with consequence C_q is calculated dividing the firing strength by the summation of the firing strength of all rules with consequence C_q, this is shown in Equation 2.10.

$$scaled\ firing\ strength, fs = \frac{f}{\sum_{q \in Class\ q} f^q} \tag{2.10}$$

Once the scaled firing strength is calculated then the scaled confidence, scaled support, and scaled dominance, can be calculated just by replacing the firing strength in Equations 2.8 and

2.9 with the scaled firing strength [9]–[11]. Then the $weighted\ scaled\ dominance$ can be calculated by the following Equation 2.11:

$$wd(A_q \Rightarrow C_q) = dominance(A_q \Rightarrow C_q) - dominance_{avg} \qquad (2.11)$$

In Equation 2.11, $dominance_{avg}$ is the average dominance of all the fuzzy rules that share the same set of antecedents A_q but have a difference consequence C_q [10], [11].

If interval type-2 fuzzy logic is used then two values for scaled firing strength are needed, one for the upper bound ($\overline{fs}$) and another one for the lower bound ($\underline{fs}$). Similarly, for the weighted dominance, $\overline{wd}$ is calculated for the upper bound and $\underline{wd}$ is calculated for the lower bound.

2.4.3 Prediction Class Vote

As mentioned before, in FRBS classifiers there is not a defuzzification process because the output is a prediction label or class that tends to be the same as the consequence assigned to the rules.

When passing an input vector through the FRBS at least one of the rules is expected to be fired (if not, a similarity measure needs to be calculated, this is described in section 3.5.1) and the consequences of the fired rules are returned as the predictions.

However, the FRBS classifier should return a single prediction for an input, this can be an issue when multiple rules with different consequences are fired. In these situations, a voting system similar to the one presented in [9]–[11] is used to decide which one of the different consequences is returned as the predicted class. Equation 2.12 shows how to calculate the vote value for each class h.

$$ZClass_h(x^i) = \frac{\sum_{j \in h} f(x^i) \times wd(A_q \Rightarrow C_q)}{\max_{j \in h}(f^j(x^i) \times wd(A_q \Rightarrow C_q))} \qquad (2.12)$$

Equation 2.12 shows that the voting value for a given $class_h$ is calculated by computing the summation of the product of the firing strength and weighted dominance of all rules with $class_h$ as a consequence, and then dividing that value by the maximum of the product of firing strength and weighted dominance of the rules with $class_q$. When using interval type-2 fuzzy logic then two values for $ZClass_h$ are computed, one for the upper bound ($\bar{Z}Class_h$) and another one for the lower bound ($\underline{Z}Class_h$), in both cases using the corresponding upper and lower values of firing strength and weighted dominance. Then both values are merged using the following Equation 2.13:

$$ZClass_h = \frac{\bar{Z}Class_h(x^i)+\underline{Z}Class_h(x^i)}{2} \qquad (2.13)$$

2.5 Using Fuzzy Rule-based Systems as Explainable Artificial Intelligence Models

Artificial Intelligence (AI) algorithms have received a lot of attention in recent years due to their considerable progress in performance, particularly machine learning and deep learning networks [12]–[14]. Despite their good evaluation metrics these AI models are black box models that lack transparency in their decision process, i.e., it is not possible for a human to understand how the model is making a decision [15]. This is starting to become an issue because thanks to the advances in the performance of the models, more black box models are being developed for life-changing decisions, such as in the medical areas [16]. If there is no explanation available on how the models are working and taking the decisions that affect the life of a person, the user's trust and adoption of these systems will be affected [17], [18]. Additionally, there is no way to guarantee a fair judgement of the circumstances, i.e., a model can make decisions that are not fair, diverse, and inclusive, because the data used to train it was biased [19].

Efforts from organizations and governments have been made to push for the ethical development and application of AI systems. The UK government mentioned in their guidelines for AI the need for transparency, understanding the technical development process and the inference process of the final model [20]. Similarly, the Defense Advanced Research Project Agency (DARPA) also stated the importance of having explainable AI (XAI) systems, the idea is to provide humans with explanations to help them in their tasks instead of replacing them, this way AI systems are a tool and not a black box replacing a human [21].

The XAI field has gained attention from researchers in an attempt to replace the current AI systems with alternatives that can be understood by human users [22]. Some of the proposed XAI solutions seek to explain how existing deep learning solutions work. For example, LIME is a technique that presents to the user the set of features that contributed to the model decision and the user can evaluate the model's decision [23]. In a similar way, SHAPLEY technique attributes each of the features an importance value for a specific prediction [24]. Another example is the use of a saliency map of images to understand what features and what areas are the Convolutional Neural Networks (CNNs) giving more importance when analysing input images [25].

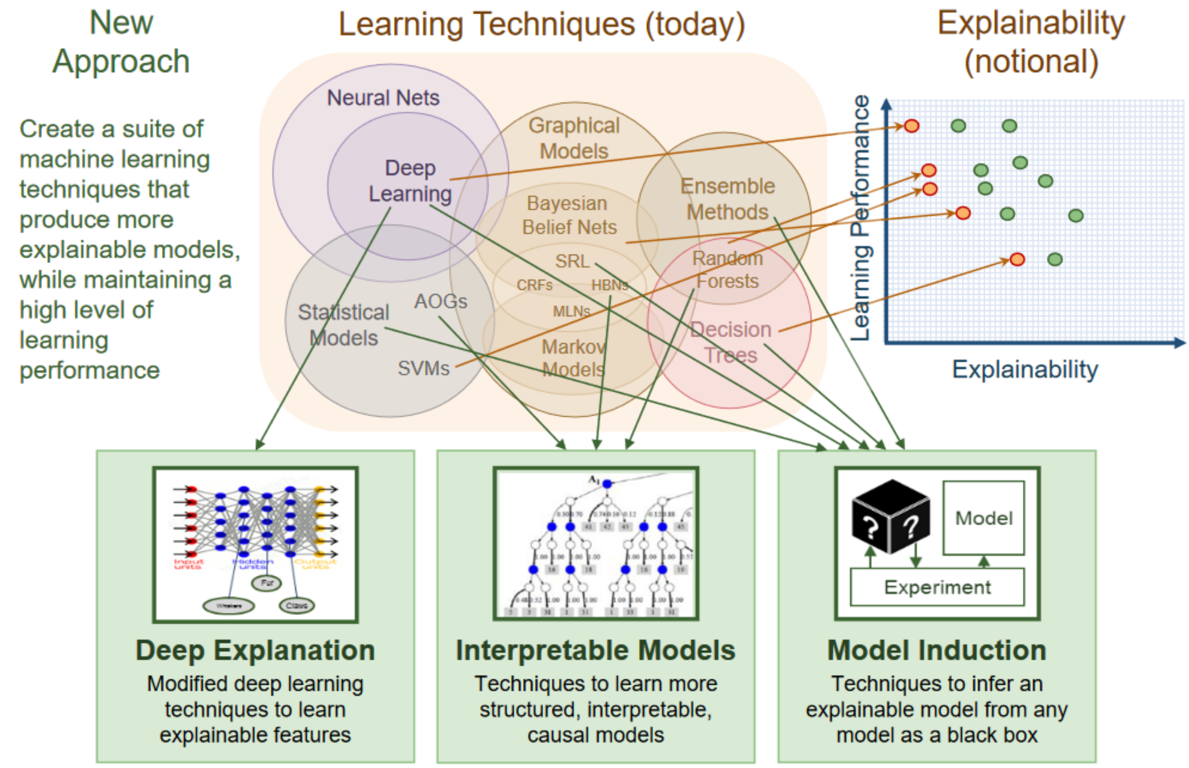


Figure 2.5. A classification of the different explainable AI methods or models according to DARPA report [21].

These are techniques or models used to understand existing black box solutions, instead a XAI model can be used, these models are characterised by using an inference process that can be understood by a human, i.e., there is no need for an additional model or technique to explain the decision process. Transparency is not free, there normally is a trade-off between transparency and performance, complex models tend to achieve better performance scores but are much less transparent [26]. Fig. 2.5. from the DARPA report shows a classification of different existing AI models based on how explainable they are, according to their classification, a Decision Tree is the currently most explainable model [27]. However, this report does not consider the use of fuzzy rule-based systems which are considered to have a high degree of "explainability" thanks to the use of If-Then rules and linguistic labels. Additionally, the use of fuzzy logic allows them to handle uncertainty and achieve good performances [22], [26], [28]. In [29], multiple layers of FRBS are used to create a complex model that can achieve better performance, and it is compared to deep learning solution;

although the use of different layers, the model is still considered to be interpretable, this show how FRBS can be used for complex solutions and still be an explainable AI model.

One of the aims of this thesis is to present a XAI model for processing unstructured and structured data for virtual environments, FRBS is considered as a viable alternative to black box models and it will be evaluated and compared against black box solutions in later chapters.

2.6 Discussion

The human decision process can be modelled using If-Then rules, for a given situation there is an expected behaviour. However, this decision process is imprecise, it is based on estimations perceived by the user's sensors, and this can vary in different situations and differ between different users. This chapter presented an overview of Fuzzy Logic, a concept that can help to handle this kind of uncertainty, the overview includes basic theory and a description of the type-1 and type-2 fuzzy logic systems.

Type-1 FLS can handle the uncertainty associated with the class of an input value, i.e., a crisp value can belong to more than one class, something that is not possible in traditional logic where the input belongs only to one class. However, type-1 FLSs still have uncertainty in their inputs because the definition and boundaries of the fuzzy sets are an arbitrary shape that a user defined. To solve this problem the field of fuzzy logic expanded the ideas of type-1 and created the concept of type-2 fuzzy logic systems. Type-2 FLSs make use of the footprint of uncertainty to handle the uncertainty associated with the definition of the fuzzy sets. Consider the boundaries defined by a type-1 fuzzy set, if those boundaries are blurred to the left and right, the region that is created is the FOU and this represents the possible strength values of membership an input can take.

These concepts of fuzzy logic can be used to create rule-based systems that use either type-1 or type-2 fuzzy sets. As mentioned before the fuzzy logic system maps input values to linguistic labels. This means that the rules used can be understood by a human. Contrary to the inference process of other classifiers or machine learning models where it is not possible for the user to understand how decisions are made. This is what makes fuzzy rule-based systems an explainable AI model that has the following advantages:

- It can be understood by humans, so the user can still be involved in the decision process. The FRBS becomes a tool for helping users, instead of replacing their decision process with a black box machine learning model.
- It can be modified by humans. When the real-world changes and the model becomes old and needs to be re-trained and re-adjusted and labelled training data is not available. Human experts can help to tune the generated models, which can reduce the cost of acquiring and labelling data. Additionally, if there are gaps in the data and some situations are not included then these gaps can be filled by expert knowledge.

The work presented in the next chapters of this thesis proposes the use of FRBS as classifiers for merging information from different sources. The goal is to present an explainable AI alternative solution for situations where it is critical for the user to understand the decision process and where there is no available data for optimising black box alternatives or the cost of acquiring the data is considerably high.

Before describing how the FRBS is used and how it compares to a black box model, the next chapter will describe how to optimise these explainable AI models to be able to achieve the best performance.

Chapter 3. An Overview of Big Bang – Big Crunch Optimisation Strategy

3.1 An Overview on Fuzzy Rule-based Systems Optimisation

The goal of the optimisation (or tuning) stage in the creation process of a Fuzzy Rule-based System (FRBS) is to find the combination of parameters that have the lowest error. There are two main approaches for tuning a FRBS: a manual process using expert human knowledge or an automatic process using labelled data, i.e., pair of data point where we have input data vector and the expected output. It is possible to combine the two approaches in the creation of a FRBS, a first design can be done with either approach, then the other approach can tune the parameters to find a better configuration.

According to Mendel [6], there is no such thing as “the FLS” for all problems, there are multiple architectural decisions that have to be made every time a FLS is built. Some of these decisions are the shape of the membership functions, the use of singleton or non-singleton fuzzifier, t-norm operation to be used by antecedents and consequences, type of defuzzification process.

In this thesis the tuning process of any FRBS will focus on the next 3 components.

1) The shape of the membership functions describing the fuzzy sets and is used to map the crisp input values to linguistic labels.
2) The number of antecedents used in a rule. In case there are too many antecedents per rule, some will be removed and inputs from those antecedents will be ignored. This help to reduce the dimensionality of the problem.
3) The last component to be optimised is the number of rules. A high number of rules will make the inference process more complex, which results in more computational time needed and a less explainable model.

The first component is related to the prediction performance of the FRBS. As described in previous chapter, the shape of the membership function decides what input values are mapped to the linguistic label represented by that function, then the inference process will use the linguistic labels to decide the output. Therefore, changing the set of values that are mapped to a specific linguistic label will change the output, this can result in a better or a worse performance. The objective of the optimisation technique is to find the set of parameters that define the shape of membership functions with the best performance. The number of parameters and the process of changing them is described in the following sections of this chapter.

The second and third components to be optimised are focused on reducing the complexity of the system without affecting its performance. Previously, Fuzzy Logic Systems have been used for different applications such as routing [30]–[32], control [8], and computer vision [33], [34]. In these applications fuzzy logic has been great for translating the expert human knowledge from the real-world to the system. However, there are applications where the FLS will have a lot of parameters and it will be unreasonable for a human to define each of them, additionally, the more complex the system becomes, the less explainable it is and it becomes difficult for the expert human to later on augment (or improve) the inference system of the FRBS [26]. Therefore, there is a need to reduce the complexity of the system without sacrificing performance. In the following sections of this chapter, a detailed description of how rules and antecedents are encoded and optimised is provided.

3.2 Big Bang – Big Crunch Optimisation Strategy

The Big Bang – Big Crunch (BB-BC) optimisation strategy is an algorithm inspired by the “Big Bang” theory in the field of physics, about the creation of the universe. It was presented Erol and Eksin in [35] where it was proved to have a high convergence speed and a low computation time when compared to other Compact Genetic Algorithms (C-Gas). In a different study [36], BB-BC was compared against Genetic Algorithms (Gas) and Particle Swarm Optimisation algorithm (PSO), and it achieved better convergence speed and execution time results. Additionally, in that same study [36], it proved to be less dependent on the random initialisation of the generations, which helps on reducing the number of candidate solutions and iterations needed.

The use of BB-BC optimisation strategy for fuzzy logic was introduced in [37] and since then it has been successfully used for the optimisation of FRBSs in different areas such PID controllers [38], machine vision [9], [39], and workforce optimisation [40]. The algorithm consists of a simple process with easy implementation, and it can be used for optimising the shape of membership functions, the number of antecedents per rule, and the total number of rules. This makes it a viable option for tuning any FRBS using real-world data instead of expert knowledge. In addition to the simple algorithmic implementation, the encoding process of the BB-BC, i.e., the process of generating the numeric vector representation, can also be easily implemented and it is an intuitive representation. This adapts to the focus of the thesis of exploring explainable AI solutions that can be understood by humans. When using GA's optimisation processes, some of the encoding in those algorithms is based on biological chromosomes approaches that cannot be interpreted by humans. A detailed description of the step-by-step process of the BB-BC algorithm and how to use it to tune the different components is presented in the following sections of this chapter.

3.3 A step-by-step overview of BB-BC process

The BB-BC algorithm is divided into two main phases: Big Bang (BB) and Big Crunch (BC). The former phase is where randomness is introduced as the means for modifying and exploring different solutions within the search space [35]. This phase is like any other GA where an initial random population of N members is created, it is important to mention that every member (or candidate solution) of this population is an independent solution and that there will be two constraints for each solution; all solutions should be unique, and all solutions should comply with the limits in the search space.

The big crunch phase is where the convergence to the optimal solution happens. All the candidate solutions created in the big bang phase are evaluated using a fitness function that measures the performance of each solution based on a specific objective. After all candidate solutions are evaluated, they will be ranked and sorted based on their fitness value (the numeric result of the fitness function). Then the centre of mass for this generation can be computed using the following equation:

$$x^c = \frac{\sum_{i=1}^{N} \frac{1}{f^i} x^i}{\sum_{i=1}^{N} \frac{1}{f^i}} \qquad (3.1)[35]$$

In Equation 3.1, x^i is a candidate solution generated and f^i is a fitness value for this candidate solution. The optimal solution is believed to be around the centre of mass point and the idea is to slowly reduce the search space and converge towards that point. It is possible to use the candidate solution with the best fitness value as the centre of mass of the population and make the algorithm converge towards that direction.

The new generation must be created after all candidate solutions are evaluated and the centre of mass is calculated (or selected in case of using the best candidate solution of the iteration), this means the big bang phase is repeated. However, if another random population is created then the algorithm becomes a random search, knowledge of the previous generation is needed, other GAs use a combination of members in the population but BB-BC algorithm uses the centre of mass [35]. Therefore, the new candidate solutions are created by adding or subtracting a random number to the centre of mass. The formula for this is as follows:

$$x^{new} = x^c + \frac{l \times r}{k} \qquad (3.2)\ [35]$$

In Equation 3.2, l is the limit in the search space, e.g., if the search space goes from 0 to 1, then $l = 1$. The parameter r is a random number between [-1,1], thus, the multiplication of $l \times r$ will define a positive or negative change to the centre of mass to create the new point. It is important to note that search space limits still need to be respected and if the new candidate point exceeds the limits, then x^{new} value is changed to be within the limits to an arbitrary point or Equation 3.2 must be repeated until a new candidate point within the limits is generated. The parameter k is the iteration number of the process, the longer the process is running for the smaller the change to the centre of mass will be. This is because the first iterations of the process are meant to be for exploring different solutions across the search space, so there will be large changes. However, as the number of iterations advances, the process is meant to converge towards one of the best solutions found, so the changes are small because the assumption of the process is that the best solution will be in the area nearby the current best solution.

3.4 BB-BC Algorithm for Membership Functions Optimisation

For the optimisation process of the shape of membership functions the key part is to encode the parameters that define the function in a 1-D numeric vector. This numeric vector will be the equivalent to the chromosome in GAs, that are used to represent a candidate solution. The proposed encoded representation used in this research was inspired by the encoding process in [10], [11], [41], some minor changes were made to make it work with the BB-BC algorithm, it was initially presented in [9], [42]–[44].

Before describing the encoding process, it is important to mention some assumptions used that will help define the scope of this research and provide an idea of what kind of FRBS were tested. These assumptions are about the shape of membership functions used in the different fuzzy sets, for the number of rules or number of antecedents per rule. The process later described will focus on and work around these assumptions. However, guidelines and brief descriptions on how to apply this approach to fuzzy sets that do not comply with these assumptions will also be provided. The assumptions are:

1) Each fuzzy set will have at least 2 membership functions. The encoding process described in this section is focused on fuzzy sets with 2 or more membership functions because some points are going to be overlapped and the process describes how to identify which points to use in the vector. Additionally, the presented use cases have multiple linguistic labels for each input variable. However, if the BB-BC optimisation process is going to be applied to a single membership function, the encoding process will be simply using the points described in Fig. 3.1.
2) First and last membership function in the fuzzy set should be a right angle (or shoulder) trapezoid shape. This is to make sure that the degree of membership is 1 at the beginning and at the end of the fuzzy set.
3) All other membership functions between the first and last will have a triangular shape. The advantages of using a triangular or trapezoidal shape function instead of other functions are:
    a. They are computational efficient.
    b. They reach zero value (unlike Gaussian shape functions).
    c. They have proven to have better or similar performance to functions with other shapes [45].

d. And most importantly the focus of this work, is that they are close to human intuition and understanding, which makes them a better option for building an explainable AI system [26].

However, it is important to note that the change in triangular or trapezoidal shape is linear, i.e., the function is a straight line with variations of slope. There might be a case where this characteristic doesn't fit the problem to be solved. A further tuning strategy for the system will be to try different membership functions' shape. Regardless, triangular and trapezoidal shape functions are an adequate starting point for this work because of the previously mentioned advantages.

4) All membership functions in a fuzzy set will have no gaps between their limits, i.e., as soon as a membership function in a fuzzy set starts having a membership value less than 1, the membership value of the function next to it will be greater than 1, and the addition of the membership value of all functions at a given point will always be equal to 1.

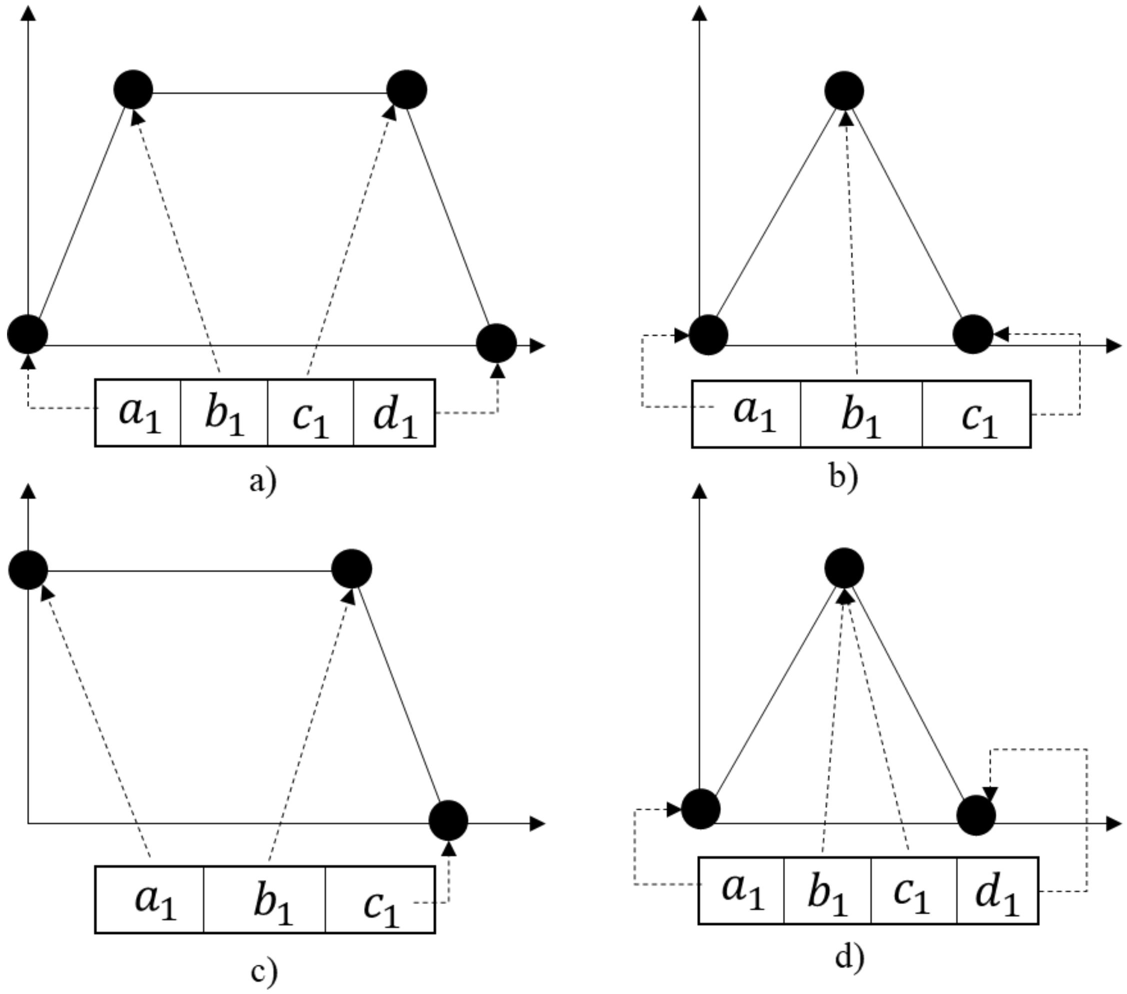


Figure 3.1 trapezoid and triangular type-1 membership function shapes and the points used to encode them represented by a circle. a) trapezoid shape, b) triangular shape, c) right angle (or shoulder) trapezoid shape, d) triangular shape encoded with four points instead of three. For consistency, the encoded points use same variables from equations 2.1 to 2.5.

It is important to first understand how each different membership function shape is flattened to a numeric vector, i.e., how each function is represented in a set of 2 or more numeric values. In Fig. 3.1, the main shapes considered for this research are shown, and a dashed arrow indicates which point in the function will be flattened to the vector representation, essentially, all points where there is a change in the function need to be captured. If the fuzzy set has a combination of triangular and trapezoid membership functions, then a four-point vector is used to represent each function, as shown in Fig. 3.1a and 3.1d. If only triangular shape functions or a combination of triangular and right-angle trapezoid (shown in Fig. 3.1c and this can also be mirrored), then a three-point vector is used as shown in Fig. 3.1b and 3.1c.

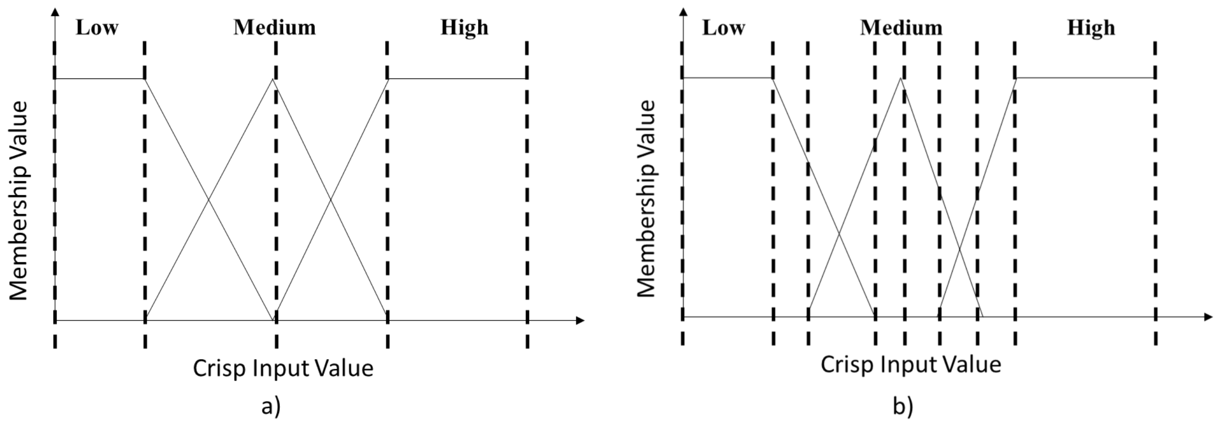


Figure 3.2 Example fuzzy sets with a total of 3 triangular type-1 membership function. a) show how some of the points overlap when there is no gap between membership functions. b) show how to encode a fuzzy set when there might be a gap between the limits of membership functions.

Consider having the fuzzy set shown in Fig. 3.2 with 3 triangular type-1 membership functions. The dashed lines in the sets of those figures show the points that are going to be used in the encoded process. It is possible to see how in Fig. 3.2a there is some overlap between the points of the different functions, this is beneficial because it means there are fewer parameters to be optimised. This overlap happens when there is no gap between limits of membership functions; this is the assumption number 4 previously discussed. In Fig. 3.2b all the points needed to encode the fuzzy set are also shown by dashed lines, more points are needed for this fuzzy set than in Fig. 3.2a, this is because there is no overlap of points, or there is some gap between the limits of the different linguistic labels represented by each membership function, this fuzzy set does not comply with assumption 4. The optimisation process that will be described here can still be used for those kinds of fuzzy sets, however, there will be more parameters to optimise, which may increase the convergence time of the process.

There are two options to create the initial population of candidate solutions:

1) **Use human expert knowledge on where the limits of the membership functions should be.**

In this case, the human expert defines the numeric limits of each linguistic label. The expert should provide 4 points within the possible search space:

- First point is the low confidence lower limit, where the membership value starts to be greater than 0.
- Second point is the high confidence lower limit, where the membership value is equal to 1 for the first time.
- Third point is the high confidence high limit, this is the last point in the function where the memberships values are equal to 1.
- Fourth point is the low confidence high limit, this is the point in the fuzzy set where the function ends and where the membership value becomes 0.

These four points are previously described in Fig. 3.1 and are needed for any triangular or trapezoid membership function shape

2) **When there is no human expert knowledge available, an equally distributed vector of points can be created as the first fuzzy set, or it can be a random set of points.**

   There might be a case where human expert knowledge is not available, but there is data available to create and optimise a FRBS, this is the second option for initialising the candidate solutions. The proposed approach is to calculate the number of parameters needed to represent the fuzzy set and then equally distribute the points across the set.

The following equations can be used to calculate the number of parameters needed for fuzzy sets that comply with the 4 assumptions stated before.

$$\#\, Parameters = P + 2 \qquad (3.3)$$

Equation 3.3 is for calculating the number of parameters needed when the fuzzy set uses type-1 membership functions. P is the number of membership functions in the set.

$$\#\,Parameters = P * 3 \tag{3.4}$$

Equation 3.4 is for calculating the number of parameters needed when the fuzzy set uses interval type-2 membership functions. P is the number of membership functions in the set.

There are two constraints that need to be considered when creating the vector representation of each candidate solution:

1) First and last points of the vector will always be equal to the lower and upper limits of the search space. This guarantees that all valid crisp input values will be mapped to a linguistic label.
2) Every other point in the vector cannot have an equal or lower value than the point before it, and it cannot have a higher value than the point next to it.

Having described the encoding process of the membership functions, here is a summary of the BB-BC optimisation steps applied to membership function tuning.

Step 1. Create an initial population of *N* candidate solutions. Make sure every candidate solution complies with the two previously mentioned constraints. Additionally, if human expert knowledge is available, one of the initial candidate solutions can be designed by expert knowledge and let this solution compete against the others in the optimisation process.

Step 2. Evaluate all candidate solutions. Each candidate solution needs to be converted from a vector representation to a fuzzy set, and a FRBS must be created using these fuzzy sets. Testing data is passed through the FRBS to evaluate the performance (using an appropriate error metric depending on the problem). Regarding the rules of the FRBS, if there is already an existing rule base defined by a human expert this can be used, if there is no rule base available then one must be created using rule modelling techniques to extract the rules from the data using the fuzzy sets defined by the candidate solution.

Step 3. Choose the centre of mass for the generation of candidate solutions. The proposed approach is to use the best candidate solution as the centre of mass. A historical best solution can be saved, and the centre of mass of each generation will be compared against this historical best solution and replaced if it performs better.

Step 4. A new generation of candidate solutions is created using the centre of mass and the Equation 3.2 described in the previous section. For example, if the centre of mass has 5 points, Equation 3.2 will be used 5 times, one time for each point in the vector representation. If the x^{new} does not comply with any of the two constraints than another value must be generated until a valid number is created or an arbitrary value should be assigned, e.g., the value becomes equal to the previous point. Additionally, invalid values can be accepted if they are identified in the evaluation section and a fitness value of zero is automatically assigned to this candidate solution, this way the BB-BC algorithm will get rid of these invalid solutions.

Step 5. Repeat steps 2 – 4 until a valid stopping criterion is met. A stopping criterion can be finding a candidate solution with an acceptable performance level or reaching a maximum number of iterations in the algorithm.

Essentially, the optimisation process described above is trying to find the distance between the dashed lines shown in Figure 2 that will result in a FRBS with the lowest error value. This is the automated evolutionary approach of what a human normally will do by trying to shift the limits of the linguistic labels and then test how the FRBS performs under the new limits.

3.5 BB-BC for Rule Optimisation

The optimisation process for the rules in a FRBS is described in this section. The inference engine of a FRBS consists of if-then rules using linguistic labels as antecedents and consequences. The advantage of this inference engine over the opaque machine learning or prediction processes is that it can be easily interpreted and augmented by humans, which means that an expert human can use their expertise to cover gaps in the data [26]. However, this advantage is only true if the inference engine remains simple enough for a human to understand it. One of the reasons why the FRBS lost popularity was that they tend to suffer from the "curse of dimensionality", i.e., a FRBS solution for a problem with a large number of input features will end up with a large rule base and rules with a large number of antecedents [26]. The issue is that, as the complexity of the rule base of the inference engine increases, the more difficult it becomes for a human to understand the model, which ends up turning the FRBS in a black box model.

Therefore, optimising the number of rules and the number of antecedents per rule is key to maintain the transparency of the FRBS. In this section an optimisation process is presented, the goal of this process is to reduce the complexity of the inference engine of a FRBS while maintaining (or sometimes even increasing) the performance. The process presented here was inspired by the work in [10], [41], [46], [47], where the authors used evolutionary algorithm to find the reduced rule base with the best performance. Their process was adapted to work with the BB-BC algorithm, the main step is to define the encoding process for the rules and the rest of the steps are very similar to the previously described BB-BC process for membership functions optimisation in section 4.2.

3.5.1 Rule Similarity

It is important to first understand the concept of rule similarity, to then understand the complete rule base optimisation process. Rule similarity is a numeric measure of how similar two rules are, and it is completely based on the linguistic labels used by the antecedents of the rules. The rule similarity measure will help us with two things:

1) **To help the BB-BC optimisation algorithm make small and big changes to the rules**. It is necessary to guarantee that a small change of a rule will result in a similar rule and not a completely different rule, if not the BB-BC algorithm will become a random search of the best rules. The rule similarity measure will help us control these changes.
2) **To select a rule when none of the existing rules in the rule base were fired by the input.** There must be at least one rule fired for the FRBS to be capable of giving an output. However, after reducing the number of rules in the optimisation process, there might be the case where an input value does not match with any of the rules that were left, in this case the best thing to do is find the most similar rule and used it predict an output for the given input.

The possible linguistic labels for each antecedent must be encoded using an integer number, this is done by assigning the lowest integer value to the linguistic label that covers the lowest crisp input values, and then sequentially assigning the next to the next linguistic label. For example, a fuzzy set with linguistic labels low, medium, and high, can be encoded as {1=low, 2=medium, 3=high}. The previous encoding process is done for each input variable fuzzy set. Then, the similarity measure between two rules can be calculated using equation 3.5 presented in [10], [46]:

$$Similarity(R_i, R_j) = \prod_{k=1}^{n}\left(1 - \frac{abs(V(R_i) - V(R_j))}{NL_k}\right) \tag{3.5}$$

R_i and R_j are two rules from the rule base, each of them has k antecedents. The value NL_k represents the total number of linguistic labels for the antecedent k, in the case of {low, medium, high} the value of $NL_k = 3$. The highest similarity value is 1, i.e., the rule similarity value of a rule against the same rule is 1.

For example, consider having two fuzzy sets $A_1 and\ A_2$, both antecedents have 3 possible linguistic variables $\{low = 1, medium = 2, high = 3\}$. Now consider the following three rules:

$$R_1 = IF\ A_1\ is\ low\ AND\ A_2\ is\ low\ THEN\ C_1 is\ X$$

$$R_2 = IF\ A_1\ is\ medium\ AND\ A_2\ is\ high\ THEN\ C_1 is\ Y$$

$$R_3 = IF\ A_1\ is\ medium\ AND\ A_2\ is\ low\ THEN\ C_1 is\ X$$

$$Similarity(R_1, R_2) = \left(1 - \frac{abs(1-2)}{3}\right) \times \left(1 - \frac{abs(1-3)}{3}\right) = (1 - .33) \times (1 - .66) = .228$$

$$Similarity(R_1, R_3) = \left(1 - \frac{abs(1-2)}{3}\right) \times \left(1 - \frac{abs(1-1)}{3}\right) = (1 - .33) \times (1) = .67$$

In the previous example the similarity of R_1 with the other rules is calculated and it can be seen that $R_1\ and\ R_3$ are the most similar rules, which is expected since both rules have the same A_2 linguistic value and the two linguistic labels in A_1 are next to each other in the fuzzy set.

3.5.2 Optimisation of the Rule Base

The total number of rules and the total antecedents per rule can be optimised at the same time, this means that there are two inputs for the rule base optimisation process, an integer TR which is the maximum number of rules allowed in the rule base, and another integer TA which is the maximum number of antecedents allowed per rule. To do so:

Step 1. Create the search space, this is a list of all possible rules for the rule base. When creating this list, it is necessary to include an additional linguistic label for each antecedent, this linguistic label will be "empty", which means that any rule that has an "empty" linguistic label in any of its antecedents, those antecedents should be removed from the rule. This way it is possible to create rules, that comply with the TA parameter. All rules that have number of antecedents greater than TA will be removed from the list.

Step 2. Sort the list so that all similar rules are close to each other. The similarity measure will help us to do this sorting. A dummy empty rule will be created, and it will be used as the base rule, i.e., the similarity between this rule and other rules will be calculated, each rule will have a similarity value assigned and they will be ranked based on this similarity. For example, consider having two fuzzy sets A_1 and A_2, both with the following 3 linguistic labels $\{low, medium, high\}$. In step 1, "empty" linguistic label is added, which results in the sets being $\{empty = 1, low = 2, medium = 3, high = 4\}$. Then, the following rule is created:

$$R_1 = IF\ A_1\ is\ empty\ AND\ A_2\ is\ empty$$

A similarity value between R_1 and all other rules is calculated by following the process described in the previous section. Then the search space of possible rules is sorted based on this measure from highest to lowest.

Step 3. Create the candidate solutions for the BB-BC population. Each candidate solution represents a possible set of rules for the rule base of the FRBS. The candidates are encoded in a 1-D vector of size TR that contains integers from 1 to N, where N is the total number of possible rules in the search space, each integer in the candidate solution vector refers to the one index in the list of possible rules. For example, a candidate solution might have the integer 1, this means that the first rule in the list of possible rules is included as part of this potential set of rules.

Step 4. Evaluate each of the candidate solutions. For each candidate solution, create a rule base using the list of all possible rules, only select the rules with an index value that is included in the candidate solution. Use this rule base in a FRBS and evaluate the performance of the FRBS by using an appropriate error metric for the problem. Only create one rule inside the rule base for repeated index values.

Step 5. Choose the candidate solution with the best performance as the centre of mass.

Step 6. Create the next generation of candidate solutions by using Equation 3.2 (presented earlier in this chapter) and the centre of mass chosen in Step 5. The l parameter of Equation 3.2 will be equal to the total number of possible rules in the search space and the x^c parameter will be the integer that is being modified from the centre of mass. The idea is that the algorithm explores different combinations of rules by making big changes and as the iterations of the algorithm progress the changes will become smaller which means that a rule will be changed for similar rules because the search space is sorted by similarity. If the same rule index is selected multiple times, then there are some options:

1) Keep adding 1 to the value until a non-duplicate rule index is found.
2) Use Equation 3.2 to calculate another x^{new}.
3) Leave the duplicate inside the candidate solution, duplicates should be ignored in step 4, there should not be duplicated rules when evaluating the rule base.

Step 7. Repeat steps 4 – 6 until a stopping criterion is met.

The process is very similar to the process for tuning membership functions, the main two differences are: a) how the rules are encoded and b) how the search space of rules is represented.

This concludes the chapter on how to optimise some of the main components of a FRBS to increase its performance and avoid the “curse of dimensionality”.

3.6 Discussion

This chapter presented the Big Bang-Big Crunch optimisation strategy as the selected approach for finding the set of parameters that creates FRBS with the lowest error metric or highest performance value. The BB-BC is an easy to implement algorithm that consists of two main phases: big bang and big crunch. The goal of the former phase is to search the universe of possible solutions and the goal of the latter phase is to slowly converge toward the optimal solution. In the first phase new candidate solutions are created and in the second phase solutions are evaluated to be able to select the one with the best performance.

This optimisation process is used for the following 3 components of a FRBS:

1. Shape of the membership functions in the different fuzzy sets.
2. The number of antecedents in a rule.
3. The total number of rules in the rule base used by the inference engine.

The goal of optimising the first component is to reduce the error of the FRBS. On the other hand, the main goal of optimising the second and third components is to prevent an increase of the complexity of the FRBS. A larger number of rules or a larger number of antecedents per rule increases the complexity of the inference engine which results in a loss of interpretability of the system.

It is necessary to encode the FRBS parameters in a one-dimension numeric vector to be able to apply this optimisation process. The encoding process of the FRBS for each component optimisation was described in this chapter.

The described process in this chapter will be used in the following chapters to find the best configuration of parameters for the different FRBSs. The next chapters present the concepts related to digital twins to be able to understand what kind of information is processed by the optimised FRBS.

Chapter 4. An Overview of Digital Twins and their Applications

4.1 The Concept of Digital Twins

Digital Twins (DT) tackle the challenge of operating, maintaining, or improving processes and systems remotely based on the concept that high-fidelity virtual models can mirror real-world physical assets using real-time data. Currently, state-of-the-art use cases for DTs have been explored in manufacturing, automotive, aerospace, construction, and the built environment.

It is difficult to define the concept of digital twins (DTs), everyone in the industry seems to have a different understanding or a different implementation of what they believe a digital twin is, in recent years it has become more of a buzzword than an actual concept. I want to highlight the idea that a DT is a concept for modelling physical objects in a digital environment, it is not a specific tool, computer system, or algorithm, that can be bought by a company as a solution for the challenges in their processes. The DT concept was first introduced by Grieves in the early 2000s as the idea of gathering from all available sources the data of a single object and centralise it in a digital representation (in recent years, a 3D model has been used) of the physical object [1]. The concept continued to evolve over the years, and although it remains a broad concept, the following three core components have been defined as a requirement of every digital twin [1], [48]:

1) Physical asset or entity. This is the object of interest that is going to be linked to a digital representation. It can be any physical object that the users want to monitor, e.g., a computer server, a building, a car, or even simple bench in the park.
2) Digital representation or entity. This is the “twin” of the physical object, and it belongs only in the digital environment, hence the name “Digital Twin”. This entity normally

has a 3D model associated with it, which is an exact replica of the latest state of the physical object. However, the important aspect is to centralise all the information that fully describes the physical object [2], the lack of a 3D model for visualisation purposes does not stop the possibility of having a DT. This digital entity acts as a centralised digital database of all the information of the object, instead of having the information spread across different tables in one or more traditional databases. It is a different way of modelling real-world object.

3) Bi-directional connection between both entities. Both “twins” (physical and digital) must be connected and capable of exchanging information. A digital twin is useless if it does not have the latest state of the physical asset, updates from the physical to the digital world should be possible. In a similar way, any change in the configuration of the DT must be transmitted to the physical object which needs to reflect the desired new configuration. This is where current technologies such as Internet of Things (IoT) sensors and fast-internet connections (such as 5G) help establish a connection between and make the DT concept possible [48]. This is the most relevant component that defines if both entities can be considered digital twins or not.

The concept of digital twins was created with the idea of modelling in the virtual world a physical object and using the digital twin to reduce the cost of tasks that require the physical object. It is now possible to simulate forces on the object over time, test changes on any property, or simply monitor the object remotely, all of this by using the digital twin [2]. Before these were high cost (both in time and money) tasks that required an interaction between human and object, e.g., building a replica of the physical object to test if the current design can resist cold temperatures.

4.1.1 Maturity Levels of Digital Twins

The basic definition of a digital twin is not specific, if the 3 components exist then it can be considered a digital twin. However, it is not the same a digital twin with an automated bi-directional connection and a digital twin that requires a manual process in its connection. The definition has continued to evolve and in [49] a 3-level classification of digital twin was proposed. This classification considers the different maturity level and it consists of the following new concepts:

- Digital Model. This consists of a digital replica of a physical object, the digital representation is linked to the physical object, however, the communication process between elements is not automated. A manual process tends to be a high-cost process which means that in terms of costs it does not offer all the benefits of a digital twin.
- Digital Shadow. This is when there is an automated connection between the physical and the digital object that goes from the physical world to the virtual world. It is mostly a connection to keep the digital object up to date, however, the digital object does not pass any information to the physical object or is not an automated process.
- Digital Twin. This is the last level of maturity, it is achieved when both directions of the connection are automated, this considerably reduces the need for human intervention which reduces the cost of maintenance.

The digital twin maturity level is the ideal scenario, but it does not mean that a digital model or digital shadow is useless, actually, if a correct manual process is in place, then the functionality of the first two levels can be almost the same as a digital twin, however, the cost will be greater. Additionally, each level builds on top of the previous which means that in a later stage a digital model can become a complete digital twin. The main use case of a digital model or digital shadow is to provide context information to an existing digital twin or for monitoring purposes, where the information of the latest state of the physical object is the only thing needed.

4.1.2 Twinning Rate of Digital Twins

Another important concept of digital twins is the twinning rate. Twining is the act of synchronising the virtual and the physical objects, therefore, twinning rate is a value that describes how often this synchronisation process is triggered [50].

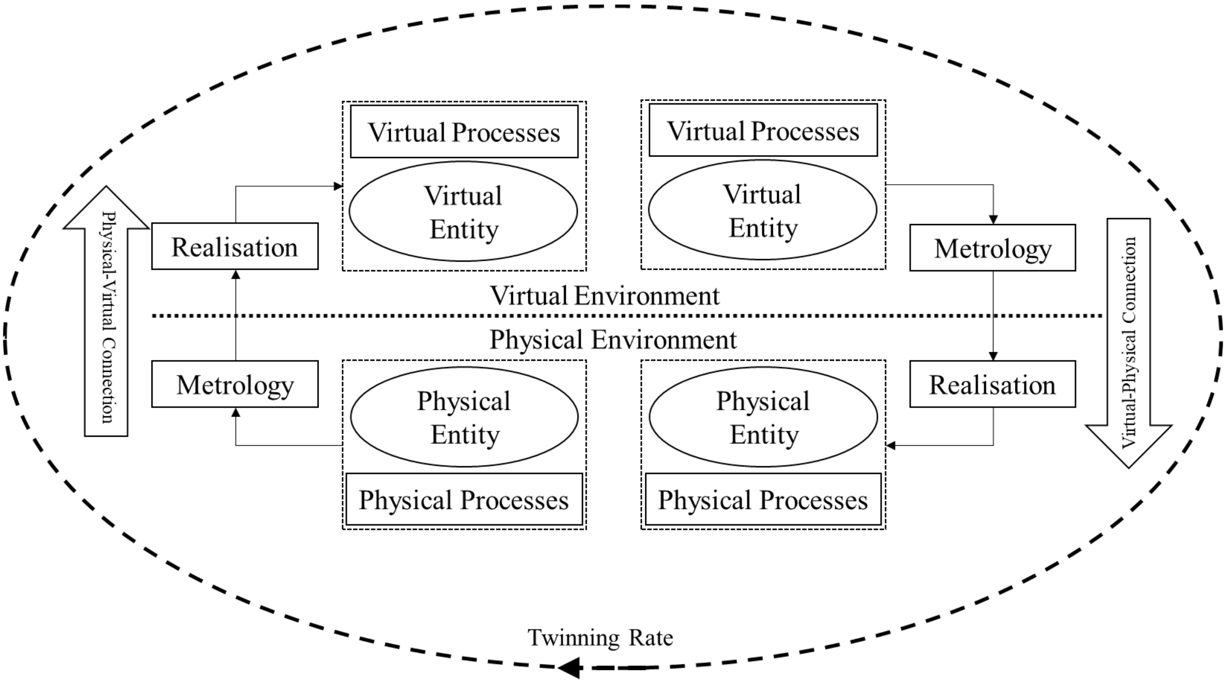


Figure 4.1. The synchronisation process between the digital and the physical objects, proposed by [50].

Fig. 4.8 shows the synchronisation process between the virtual and the physical environments. Between the two entities, there are two stages: metrology and realisation. Metrology refers to the capturing of data and is normally done in an automated way through sensors. Realisation refers to processing the recently captured data and updating the twin (either digital or physical).

4.1.3 Mixed Reality Environments for Digital Twins

By definition a digital twin is a 3D replica of a physical object that lives in the virtual world. There is a natural need to be able to visualise and interact with this digital replica, therefore, technologies that connect the physical and the virtual world are needed.

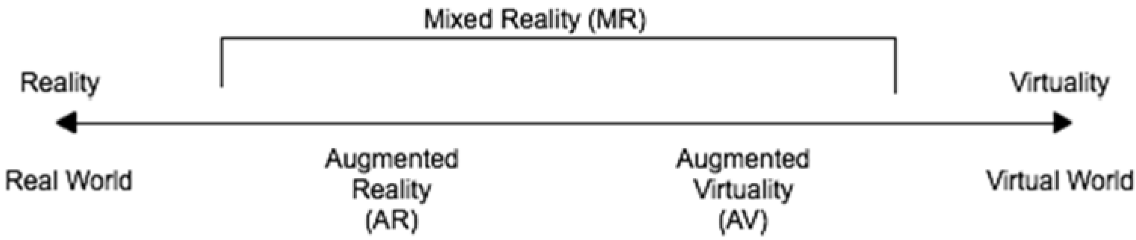


Figure 4.2. Reality-Virtuality Continuum proposed by [51].

Fig. 4.9 shows the reality-virtuality continuum proposed by [51], in there he defines the concepts of mixed reality (MR), augmented reality (AR) and virtual reality (VR). MR is an overall concept that covers everything in between the real and the virtual world, i.e., if in the user's view there is a combination of both worlds, it is considered a mixed reality application [51]–[54]. In this broad spectrum an important concept called augmented reality can be found, AR overlays information from the virtual world to the user’s view and it is the closest point in the continuum to reality, and the interaction with the objects tends to be limited [52], [53]. AR applications make use of screen displays to show additional information and also of tracking techniques to understand the user’s environment and to save the location of the virtual information displayed [55]. As shown by the continuum in Fig. 4.9, VR is one of the ends of it and it is characterised by completely removing the real-world from the user’s view. This is normally achieved by headsets that completely immerse the users in a new view. MR technologies are the perfect partners for DTs, they merge both realities, therefore, they can merge both twin objects in a single view for the user. Additionally, VR technologies can display only the digital twin and allow to visualise a replica of the physical object remotely. MR technologies are currently being used for training purposes [56]–[59]. However, there is also research looking at field engineers accessing virtual information on-site through MR technologies [60]–[62].

4.1.4 Handling 3D Objects Rendering in Mixed Reality Environments

One of the challenges of using MR technologies during outdoors tasks is that 3D models can be considerably large, and the hardware devices will not be capable of loading them, especially the MR or AR devices that are not connected to an external processor and have limited memory such as the Microsoft HoloLens shown in Fig.4.10.

Figure 4.3. Microsoft HoloLens.

To address this situation a type-1 FRBS was proposed to handle the loading of 3D objects into the user's view [63]. The FRBS uses the following 4 input variables:

- Average frame per second (FPS) rate, provided information on the performance at runtime.
- Distance of the next object to the user's location.
- The memory usage of the device.
- Download speed available for the device.

The membership functions and the rules were created using expert knowledge and a heuristic approach.

The value of using a FRBS is that it understands when to reduce the number of requests and when to increase them based on the computing power of the device. When compared to a static loading process using non-fuzzy heuristic rules the performance of the application was 2.33 times worse.

As the computational power of the hardware increases this solution will no longer be needed but for now it helps to keep the application running at a better FPS rate than when having a static load of objects.

4.2 National Digital Twin

For the built environment, many companies and governments are looking at digital transformation programmes to create large-scale digital twin ecosystems towards delivering on the grand challenges facing society, including sustainable development, efficient urbanisation, population growth, and escalating infrastructure costs [64]. For example, the UK's *National Digital Twin programme* (NDTp) aims to enable better use, operation, maintenance, planning and delivery of national and local assets, systems and services to benefit the country's society, economy, business, and environment [3], [4]. Similarly, Singapore's National Research Foundation (NRF) launched *Virtual Singapore*, an authoritative 3D digital twin platform to enable tools and applications for test-bedding, planning, decision-making and research to solve emerging and complex challenges for Singapore [65].

Digital twins present a significant opportunity to model an organisation's assets, services, and operations. Utility companies are crucial stakeholders in these objectives due to their significant investment and ownership of infrastructure assets. They own many assets that expand over extensive areas, from towns to cities and even different countries. These companies have multiple information about their assets, from location to their latest status. However, a big challenge is that data comes from many sources and is presented in many formats, adding to the interoperability challenge. For example, a great deal of the information is 2D, from digitalised floorplans (CAD) to paper schematics. A survey from [66] showed that 89% of the companies use GIS databases to store the location of their assets, but only 19% use 3D enabled frameworks (such as Building Information Modelling or BIM) to map the physical assets in buildings. From the telecommunication industry perspective, 2D network schematics contain assets' location and cables routes from geolocated points A to B. However, the information can become complex when managing assets in extended areas. Furthermore, traditional 2D CAD plans and schematics show assets' location, lacking context information or historical data. Whilst decisions might be made via 2D data, a 3D digital twin provides visual confirmation and spatial context, making it easier for end-users to understand what, where, and why the work is required.

A DT of the telecoms network and its assets can be fundamental for understanding the network, assisting in monitoring and optimising the operation and maintenance of physical assets, systems, and processes during onsite and remote tasks [66], enabling better-informed decision making and leading to improved outcomes in the physical world. This could translate into operational cost reduction and better service for customers. Despite these benefits, the implementation of digital twins for service and operations is not widespread. The challenges include:

a) **Model context-aware DTs using different data sources with different data formats.** Multiple data sources are needed for creating accurate DT. Each data source should constantly feed information into the DT to keep it updated. Data that can be used include geolocated data, BIM models for information about manufactured structures, satellite images to identify elements of interest in an area, and existing network assets' databases. This thesis proposes extracting elements and information of interest from each data stream to create standard 3D objects that encapsulate relevant information, which virtual or augmented applications can use.

b) **Create DTs capable of interacting with each other.** The ability of a digital twin to communicate with other digital twins will provide the model with context and allow it to react to changes around it. This is the first significant step towards creating autonomous digital twins [67].

c) **Enable better interaction between DTs and their end-users.** Immersive technologies, such as Augmented Reality (AR) and Virtual Reality (VR), can visually present rich, volumetric data, which, combined with context-aware models, can allow users to access specific data when they need it. These solutions relieve cognitive overload, promote knowledge generation, and provide decision-making support. Creating relevant solutions requires a collaborative process of building context through the user and the system, keeping humans in the loop, and providing a more natural interaction between the user and system [68]. VR and AR interfaces could enable interactive data visualisations to support users with domain-specific tasks**.**

Existing research on digital twins for service and operations focuses on using one source of information (e.g., BIM) to visualise assets during onsite tasks[61], [69], or to monitor the construction process and assets' deployment [60]. However, there is a lack of research on the creation of context-aware digital twin assets in a geographic area using multiple data sources. This thesis proposes a processing pipeline to create 3D digital twins for elements of interest in a geographic area, merging data from different sources and visualising their latest state. The principal benefits of this approach are twofold:

a) **Unify models.** Since the digital twin can consume data from different sources (e.g., terrain data, BIM data, satellite images, and network assets' location), it would create a unified 3D digital twin of an area with all the available information. Each data stream can feed the DT with the latest information to keep the digital twin updated. Context information around the network is added to the 3D twin by merging multiple data streams. This allows the users to visualise and run simulations on how the surroundings will interact or affect the assets.

b) **Onsite assets visualisation**. Field engineers in maintenance tasks use a variety of data sources, from CAD plans and databases to traditional paper-based documentation. This represents a challenge for end-users because of how the information is conceptually presented in the documents (i.e., without context) and because these documents are rarely updated since their creation. Our proposal includes a 3D spatial visualisation presented to the field engineer through augmented reality headsets which can better support surveyors and field engineers, reducing maintenance tasks time.

Chapters 5 and 6 present our work related to the creation of a Network Digital Twin and how this is not only a virtual representation of the network assets but also needs to include additional context information and interoperability with other digital twins to become a fully functional NDT.

4.3 Discussion

This chapter introduced the concept of digital twins, which describes the idea of modelling and centralising the information of a physical object in a 3D digital model located in a virtual environment. Every digital twin consists of at least the following components:

- The physical object being modelled.
- Virtual representation of the physical object, normally using a 3D object of the actual object since visualisation is a big part of the DTs.
- A connection between the digital and the physical object. This connection is used to keep both sided updated and to influence the state of their twin.

This definition does not consider the scenarios where there is a 3D replica of the physical object without a bi-directional connection. The concept has evolved over the years and new definitions were proposed to describe the different levels of maturity of a digital twin.

- Digital Model. A virtual representation of a physical object, however, there is no connection between objects, or the connection is a manual connection.
- Digital Shadow. A virtual representation of a physical object. There is a connection between both objects, but it only goes from the physical to the virtual world.
- Digital Twin. The highest level of maturity is achieved when there is a real-time bi-directional connection between both objects.

The difference between levels is how much automated the exchange of information is, the capabilities of a digital model can be the same as a digital twin, if a manual process for exchanging information exists. The issue is the cost of using just digital models because manual processes tend to have a high cost, hence why digital twins are preferred.

Since each level builds on top of the previous a digital twin can start as a digital model and as it matures it will become a complete digital twin. This concept is important for the rest of the thesis because most of the data processing work that will be presented is to generate digital models or digital shadows, the idea is to get to achieve digital twin maturity level, but this will not be possible in all scenarios, and it does not mean that the output is useless. Having a digital model of an object is enough in cases where the state of the physical object does not change often and a semi-automated or manual process for exchanging information is enough. Additionally, the main purpose of these type of models is to serve as a context information in a virtual environment that contains multiple digital twins. Furthermore, an automated process for creating digital models from 2D data will reduce the manual cost of creating them.

A utility company can use its information and sensors in their assets to create a digital twin of their network. Additionally, this can then be improved by including digital models, digital shadows, and digital twins of physical objects surrounding the network assets, this is the idea of a network digital twin. Furthermore, combining the network digital twin of all utility companies will create a virtual environment with all the information of the physical infrastructure of a country and it will enable better collaboration between stakeholders. This is the idea of the National Digital Twin concept also introduced in this chapter.

The work presented in the following chapters of this thesis seeks to automate the process of creating the digital models and digital shadows of physical objects that interact with the network assets and include them in a virtual environment where the network digital twin will be. The next chapter starts by identifying and extracting the vegetation in satellite images to create a digital model that will be connected to the network digital twin.

Chapter 5. Processing Satellite Images to Create Digital Twin

5.1 Context Information for a Telecommunications Network Digital Twin

A network digital twin (NDT) for a utility company is a concept inspired by the idea of a National Digital Twin [3]. The idea is to create a digital representation of all the different assets that are part of the network, e.g., for British Telecom (UK's main telecommunications company) the assets modelled will be cables and distribution points (such as green cabinet outside houses or an exchange, i.e., a site with the server racks). The network digital twin will be stored in the cloud and objects will be loaded to a virtual environment when a user wants to access it.

This will shift the paradigm used by utility companies to store their data, now every asset will be represented and have all the information centralised in the object. This will be a complex task that will require abandoning the legacy database systems with thousands of tables and moving to a virtual map of the network.

For a digital twin to reach its full potential it is necessary to have communication with other DTs. It is not enough to create a 3D model representation of a physical asset, placing those models in a virtual environment and feeding them with all the available information. That will be enough for a detailed visualisation of the network and to run simulations of how changes within the network will affect only the objects inside the network. The DT needs to be capable of communicating with nearby DTs to understand how internal changes affect the external environment and vice versa. This is when contextual information takes an important role in maximising the value that a DT can bring to a company.

There are two types of contextual information for a digital twin:

1) Communication between DTs located nearby to each other. For example, exchange of information between the network digital twin from the water service company and the telecommunications company will improve the simulations each of the DTs does.

2) Environment objects that are not necessary already a DT. These objects have an impact on the NDT but are modelled as a property in the DT objects. Additionally, it can even be modelled as some digital model or digital shadow (DT maturity levels) that will be part of the virtual environment of the NDT. Still, it doesn't reach the maturity level of a digital twin.

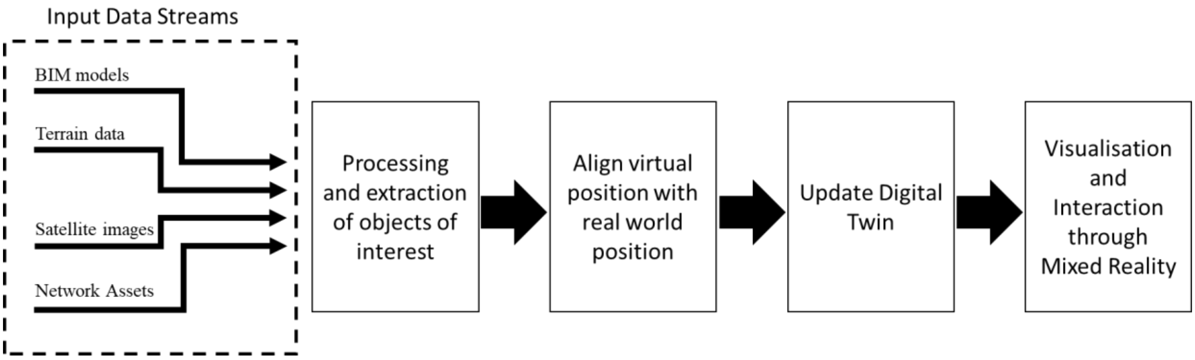


Figure 5.1. The type of information used to build the network digital twin and a high-level view of the steps needed to visualise it in the virtual environment.

Fig. 5.1. shows the proposed processing pipeline to create a digital twin from a geographic area using different data sources. The different stages of the pipelines are described in the following subsections.

5.1.1 Input Data Streams

The first stage of the proposed approach is to read the available data for the network assets. One of the goals of creating this Digital Twin is to visualise the interaction of the network assets with surrounding elements. Multiple data sources are used to provide context. There are two main types of data:

1) Structured network assets data. This refers to any information about a specific asset available in existing company systems (e.g., databases) and can include assets' type, ID, geographic location and historical data.

2) Context information. This refers to information in assets surroundings. In this thesis, three sources of information are considered when defining scope of the data processing work to be done.
    a. Terrain data. This is a simple context information which will be needed by the digital twin to do some more advanced simulations. However, this data is included in the object as a property, e.g., terrain elevation at the location of the asset.
    b. BIM models. Building information is essential for connecting the outside network with the in-building network, understanding where the network assets are in the building and helping field engineers with their tasks. This is further discussed in the next chapter.
    c. Satellite images. Satellite images are a good and cheap source of information about the objects in a geographic region. Processing this type of 2D data is the focus of this chapter. The idea is to extract the information about vegetation in the area and create a 3D object of it in the virtual environment with the network digital twin.

5.1.2 Extracting Objects of Interest

During this stage, the information received from the data streams is processed to extract only the elements of interest. Each data stream is processed differently. This allows us to transform and unify the data into the 3D DT.

Satellite images require the most complex processing because computer vision techniques are required to find the objects of interest. For our use case, we identified all the trees in specific areas, adding context to our DT from these images. This will help users identify if there are trees in the way of an underground network route and if further analysis is required to determine if the overgrowth of the trees might damage the closest assets. To do so, we used semantic segmentation to identify pixels that belong to trees. After that, “tree pixels” are merged in a single cluster of pixels and finally transform the trees’ position in the image to a 3D spatial position.

5.1.3 Pairing Objects of Interest to its Virtual Representation

The next stage creates a 3D representation of every information element extracted from the data streams. These 3D objects contain metadata such as their geolocation, position within the virtual representation, properties, and any available historical information.

5.1.4 Updating the 3D Digital Twin

The 3D objects created in the previous step are stored and further updated or presented in the 3D visualisation. For example, in our use case, every tree identified in the satellite images is saved as a 3D object, terrain data is saved as tiles, and assets and elements in BIM models are saved independently from the others.

5.1.5 Visualising the 3D Digital Twin

The information produced in previous steps can be presented to end-users via VR-enabled devices for remote visualisation or AR devices for onsite visualisations. One of the current challenges for these devices is to load significant amounts of data in real-time due to their limited processing capabilities [70]. The proposed system explores loading the elements spatially closest to the user first and changing the number of elements loaded per frame based on the device capacity [63].

5.2 Extracting Vegetation Information from Satellite Images

5.2.1 Building a 3D Map of Trees in a Geographic Area

In utility companies the remote network planning engineers are responsible for two main tasks: 1) deciding the best route for new network assets and 2) scheduling maintenance of the current network assets. For these two tasks one of the obstacles or hazards is the vegetation (such as trees) in the area where the network assets are located. For example, consider the following two use cases:

1) Planning the deployment of new network assets requires a field engineer to evaluate the area where the cable is going to pass through. Currently, this is done manually by field engineers that are sent to survey and evaluate the area. This is a high-cost task for the company, both in field engineer time and money. The reason for doing this is that before starting or scheduling any works in the area the company needs to guarantee that it is possible to have a cable or cabinet there without damaging any other elements in the area. Trees are some of the objects that delay the deployment of new assets because they are on the way to the planned route and a long process for removing trees must be started or a new route must be found.
2) Other scenario is of existing overhead cables, i.e., the cables are routed in the air by using poles. There are situations when trees grow so much that they become a hazard to the cables between poles and cables might get damaged. There is a maintenance task that needs to be scheduled to trim the tree. This kind of hazard problem can also happen with underground cables and the roots of the trees; however, this is not common because during the planning phase cables are not routed close to any trees.

The idea is to identify all the trees nearby network assets and create a digital representation that can be included in the NDT as context information of external objects of interest. Having both type of objects sharing a virtual space will allow for advanced simulations and smart planning of the deployment of network assets.

The challenge of creating this 3D digital twin of trees in a geographic area is to find data at a low cost. Using people to survey the area is not an option since it has a high cost in time and money, and to keep the model updated it will require revisits to the area, which again is a high-cost task. One of the cheapest sources of information is 2D satellite images, they can also be obtained easily through automated drones, therefore, the cost of keeping the data updated is low when compared to sending a field engineer to the area.

5.2.2 Semantic Segmentation of Satellite Images

Analysing the satellite images using computer vision techniques can help to automate the extraction of information about trees in a geographic area. This information can then be used to automatically create a digital representation of the tree in a virtual environment. Previous research in this area includes the monitoring of forest resources through images [71], monitoring and analysis of agricultural land [72], and urban modelling and growth analysis of geographic areas [73]. The common approach in previous research is to use semantic segmentation techniques to find objects of interest. Semantic segmentation, pixel-wise classification is the task of assigning a label to every pixel in an image, e.g., in the case of finding trees in satellite images, every pixel can be classified as “tree” or “other” [74]. This approach is the one that provides the most detailed in the output when compared to other approaches, such as *Image Classification* and *Object Detection*. Image classification analyses features in the image and assigns a label to the complete image and object detection searches for objects of interest in the image and returns the location of those objects in the form of a bounding box. The level of detail in the segmented image provided by the semantic segmentation algorithms helps to create more accurate 3D representations of the trees.

In recent years, the popularity of deep learning approaches has grown considerably due to the advances in computing power that make it more accessible for anyone to create and optimise this kind of neural networks. In the particular area of computer vision, Convolutional Neural Networks (CNNs), have become the state-of-the-art for solving any image processing task [75]. These networks were originally designed for image classification tasks but they have been successfully adapted for semantic segmentation [76]. Some of the related research includes the use of CNNs based deep learning models to find poles in street view images and analyse their current state [77], which can be useful for monitoring network assets. Additionally, there is existing work on using deep learning approaches for analysing satellite images to find vegetation that needs trimming or maintenance because they are becoming a hazard [78]. The current research work lacks the part of creating of updating 3D digital representation of those objects found in the images, the current research is limited only to providing some insights for decision making based on some detections in satellite images. Furthermore, CNNs and other machine learning models used in image processing have two major problems:

1) They require large amounts of data with labels for each pixel to achieve an acceptable performance [79]. This will result in a monetary cost if the company is buying the data or in a monetary and time cost if the company decides to capture its own data [80].
2) They lack transparency in their inference process, it is not possible for a human user to understand the decision process of the prediction, therefore, it is not possible to augment models with human expertise [81]. The models are limited to doing what they “learn” from the data, if the data provided during the training process is not high quality, then there is no way to complement this with human expertise. The only way to improve the deep learning solutions is to acquire more data and train them again or change the architecture parameters and train them again, both are high-cost options.

In this chapter an interval type-2 fuzzy rule-based system is presented as the best explainable AI alternative for doing pixel classification. The FRBS is compared against a Multilayer Perceptron (MLP) with Back-Propagation Learning that uses pixel-level features as an input vector to classify each pixel [82]. The reason to use the MLP instead of a CNN is to fairly compare the result of both the black box model and the explainable model using the same input vector, the CNN would not have used the same input vector due to the nature of the architecture of using convolution operations to extract features.

5.3 An Explainable AI Fuzzy Rule-based System for processing Satellite Images

5.3.1 Feature Selection and Data

The selected features used to classify each pixel is the HSV (hue, saturation, value) colour space which is based on the characteristics and the way human perceive colour [83]. Hue (H) component is a value between 0 and 360 that represents the colour, unlike RGB that uses a combination of three values to represent colour, using a single value for colour is more human friendly. Saturation (S) component describes the purity of the colour and Value (V) component represents the amount of light in the colour, both of these are measured with a value between 0 and 1 [84].

Figure 5.2. Image used for testing, similar to what can be found in the training set.

The images used for the experiments are RGB satellite images with a resolution of 30 cm per pixel [44]. The dataset was manually labelled for the class tree, then images were flattened to a 4-column table, the first column contains Hue value, second column contains Saturation value, third column contains the Value component, and the final column contains the expected label for that combination of HSV values. Fig. 5.2 shows a similar image to the ones used to generate the dataset; the image shown here is used in the evaluation process. After processing the available images, a csv file with 20,000 data rows was created, 80% was used for training and 20% for validation. This same dataset was used for both, the training of the MLP and for creating the explainable FRBS.

5.3.2 Creating the Fuzzy Rule-based System

The rule base of the FRBS is created using the rule modelling techniques described in the overview of fuzzy logic of chapter 2.

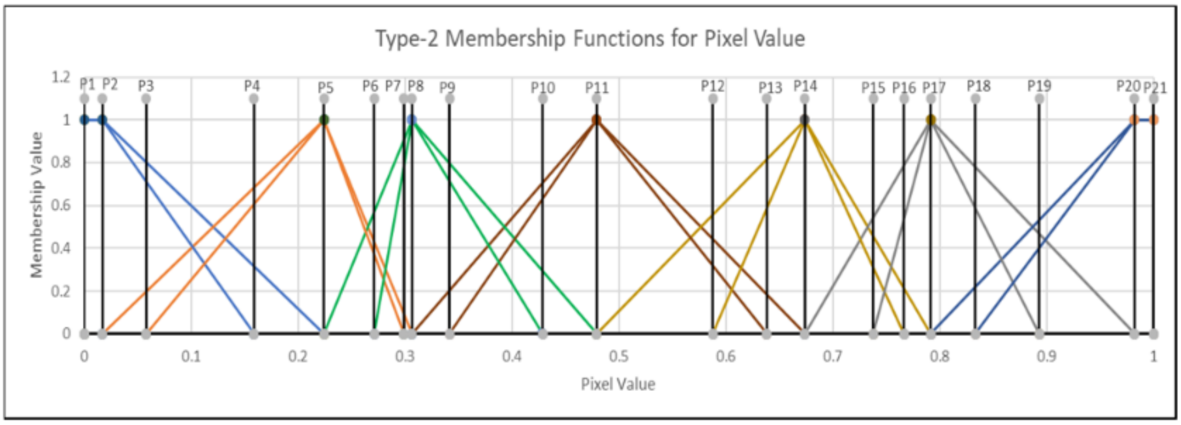


Figure 5.3. Optimised fuzzy set for 'Value' component of HSV features. The vertical lines show the points to be optimised by the BB-BC algorithm.

Fig. 5.3, shows the optimised fuzzy set for the ‘Value’ component of the HSV features and the vertical solid lines show the points that need to be used by the BB-BC optimisation process described in chapter 3, to find the configuration of points with the lowest possible error. The

BB-BC algorithm was executed for a maximum of 250 generations, each with 30 candidate solutions. A type-1 FRBS was also created and compared to the interval type-2 FRBS. The same fuzzy sets were used for both types of FRBS, but the footprint of uncertainty was removed for the type-1 FRBS.

5.3.3 Results of the Experiments

The three different models (type-1 FRBS, interval type-2 FRBS and MLP) were evaluated against images that are like the ones used for the training dataset. Fig. 5.4 shows a visual segmented output from the different models. The MLP model has the best performance, the interval type-2 FRBS is not far away from the MLP, and the type-1 FRBS has the worst performance, and it is not even a useful result from this model.

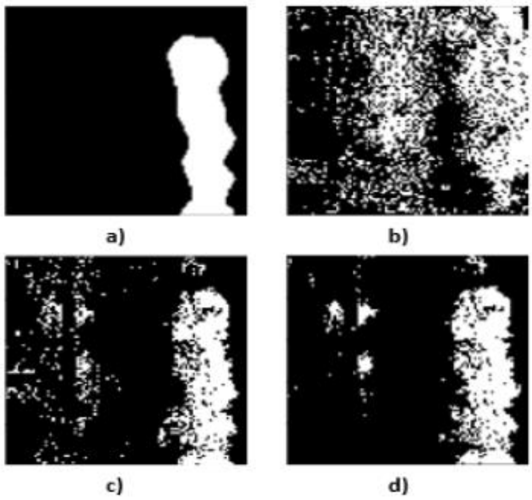


Figure 5.4. Semantic segmentation output from the image in Figure 2. a) expected result for a perfect model, b) result from type-1 FRBS, c) result from type-2 FRBS and d) result for MLP model.

The performance metric values for these experiments presented in [44] can be seen in Table 1. The Intersection over Union (IoU) metric is used for evaluation purposes since it is a standard in semantic segmentation problems and it can deal with the unbalanced results of a segmented output image normally having more background than actual predictions [85]. The type-1 FRBS has an IoU measure of 49.3%, while the Interval Type-2 Fuzzy Rule-based System (IT2FRBS) has an IoU of 60.6%. The IT2FRBS, which gave the best result, was still outperformed by the MLP network by 8.4%. However, the IT2FRBS has the advantage over the MLP network of being a transparent box model, i.e., a human can easily understand how the system predicts a given class. In addition to the high degree of interpretability, the IT2FRBS 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. This helps with the data problem, if not enough high-quality data is available the human expert can complement what the FRBS learned from the data. For example, end users 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 an understanding of the training process since it is not being trained again, unlike the MLP [44].

A post-processing stage is needed after segmenting the satellite images to convert the 2D segmented mask result to a 3D tree in a virtual world. The post-processing stage is a short and fully automated process using OpenCV python library for image processing, the stage consists of the following four steps:

1) Noise removal of the segmented mask. Use a combination of median, dilation, and erosion filters to remove isolated white pixels in the segmented output. There is no perfect model for segmenting, they will all have some errors, and it is important to try to identify them and remove them in an automatic way.
2) Clustering pixels together, finding the contours of the clusters and extracting the location in the image of these contours.
3) Using the geographic location of the top left corner of the image or the centre of the image it is possible to calculate the estimated real-world location of the different

objects. The process also uses the resolution of the image to estimate the actual real-world size of the tree crown, i.e., the upper part of the tree.

4) Create a 3D object that reflects the estimated size of the tree crown and place it in part of the virtual world that represents the geographic location of the image. The heigh of the tree is unknown in this type of images, it is not possible to extract it, however, if the image contains LiDAR data, in step 3 this can be extracted for the location of the 3 and estimate the height.

Fig. 5.5 shows the result how the 3D tree objects look in the virtual world after the segmentation and post-processing stages.

One limitation of this approach and features is the use of only pixel level information, this limits the model to only colour-based characteristics of the objects, in some cases the distinctive factor is in the shape of the object, and this is not considered in this FRBS or MLP. In the next chapter, this work is expanded to combine context information and pixel-level information to be able to classify pixels of elements that share the same colour characteristics.

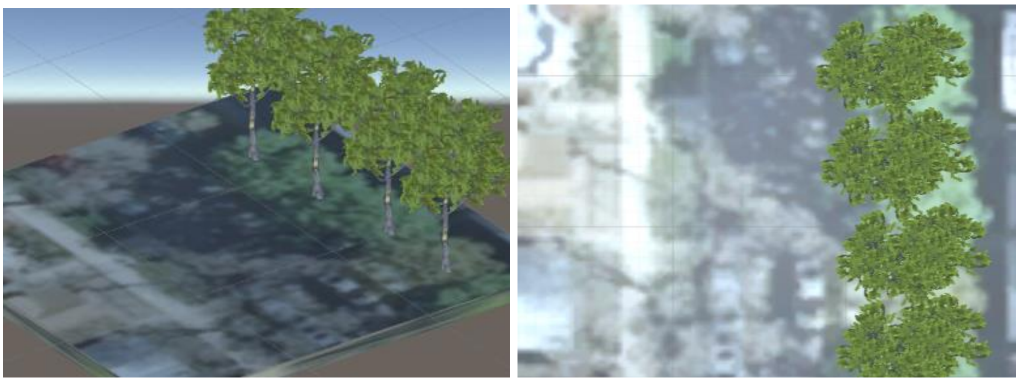


Figure 5.5. View of the automatically generated 3D tree objects in the virtual environment. Left side is view from one side and right side is a top view.

5.4 Discussion

As mentioned earlier in this chapter, trees are one of the main obstacles and hazards to the utility companies network assets, especially for the pipes and cables routed across the city. Therefore, it is important to include the location of the vegetation in a geographic area as part of the network digital twin and be capable of visualising the proximity of trees to network assets and identifying potential issues.

One of the main challenges for creating these 3D digital models of the trees in a geographic area is the cost of sending a person to survey the area of interest. Even for utility companies with a large workforce it is a high-cost task to create the initial map and to update it. Satellite images are a cheap source of data that contains the necessary information on the location of vegetation in a geographic area, it can be used to replace the need of sending a person. Semantic segmentation is the process of labelling each pixel in an image and it is used in this chapter as the solution for automatically extracting the information of the area covered by a tree.

This chapter proposed the use of an interval type-2 FRBS as an explainable AI approach for semantic segmentation of satellite images to extract the location of trees in a geographic area. Additionally, a comparison between this proposed approach, a type-1 FRBS (which is also an XAI approach), and a neural network model (black box alternative) was also presented as part of the results.

The results in this chapter can be summarised as follows:

- The type-2 FRBS outperforms its type-1 FRBS counterpart by a significant amount
- The black box model alternative still has a higher performance than the explainable AI model. However, by adding the noise removal step into the post-processing of the result, the explainable AI model achieves a similar result and still has the advantage of being understood and possibly modified by a human.

Once the image has been segmented using either the proposed XAI method or the black box alternative, the pixels with similar characteristics are clustered together and by using image processing techniques, it is possible to find the contours and the location of these clusters.

This information will allow us to create a 3D model of a tree that covers the same area as the extracted tree from the image. Therefore, a digital model is created, and it is possible to connect it to the network digital twin as context information. The use of contextual information helps improve digital twins by understanding what the surroundings of the physical objects are. This contributes to the idea of a national digital twin, where digital models, digital shadows, and digital twins interact all together to provide an accurate replica of the infrastructure and natural objects in a geographic area.

The idea of creating these digital models of trees in a geographic area is to add context to the "outside" network, i.e., assets that are not inside the premise of a customer. The next chapter presents how to create a digital model for the building elements that surround the network assets in a customer premise. The presented methods in this chapter use pixel-level information to decide on the label, however, if the pixels of the object of interest share characteristics with other objects pixel-level information will not be enough to distinguish between elements. Next chapter expands on the use of a FRBS as an explainable AI approach for semantic segmentation and presents a process on how to include information on the surroundings of the pixels in the inference process of the FRBS.

Chapter 6. From 2D Floor Plan Images to 3D Virtual Representations of Buildings

6.1 An Overview of Building Information Modelling and its Challenges for Utility Companies

Building Information Modelling (BIM) is the standardised process for multiple stakeholders to work on a building's lifecycle, from the early design phase to the post-construction maintenance phase [86]. The output of this process is a 3D data-rich digital model, called BIM model, that has relationships, properties, historical records of changes and geometric information for every element in the building [87]. It is important to highlight that the information is available per element and the relationships are what allows us to visualise all the elements as a single building, this is the main difference with other 3D models where the geometry shape information and visualisation is the most important factor.

The BIM process was born in the construction industry to help manage the project from start to end. The reality is that most of the actual work needed in a construction project is already figured out; however, the projects still have a lot of complexity that comes from the interaction and integration of a lot of different people from different organizations [88]. The exchange of large numbers of documents with partial information related to just one part of the project makes the interaction between stakeholders confusing and difficult, which tends to cause conflicts, a need for further clarifications, frequent calls and a lack of trust [88]. Therefore, BIM is proposed as the standard process with a centralised 3D model that contains all the relationships, properties, and geometry of all elements, and that every stakeholder is capable of viewing at any point in time. There are many benefits in the construction industry, not only for visualisation and management, but also for using the relationships and properties of building elements in tasks such as cost estimation, forecasting, scheduling, and smart decision making. These additional tasks are extra dimensions on top of the data-rich 3D model, “time” related tasks such as scheduling and forecasting the end of the project are part of the 4th Dimension (4-D) and “cost” related tasks are considered part of the 5th Dimension (5-D), and additional type of simulations of tasks can be explored in new dimensions [89]. The characteristics of these new dimensions is that it takes advantage of all the information from different stakeholders that is centralised in this BIM model, therefore, the systems can find conflicts that will take time to find if the information of stakeholders was in separate silos [90]. For example, it is possible to identify if there is a clash in the time of the construction of two elements, i.e., it is not possible to build the actual structural elements (such as walls, doors, windows, columns) in the planned time because the work of one will interfere with the other, therefore, a change in schedule must happen. Additionally, the scheduling suggestions are not limited to only location and interference between the building work, the decision can also be influenced by other factors such as information on estimated construction time, date of arrival of materials, dependency on other elements.

Although the BIM process was born in the construction industry and it was meant to solve most of the problems during the construction phase of the project, the BIM models are still beneficial for utility companies. For starters, utility companies are meant to be one of the stakeholders during the construction phase, ideally, they will participate and contribute to creating the BIM model. The difference comes in the later stages of the building lifecycle, once the construction project is completed all the construction related stakeholders forget about the building, but maintenance and utility companies still need to interact with it for any future maintenance tasks. Some of the benefits related to on-site task guidance, updated historical records, and network planning [9] are:

1) Have an updated and accurate record of the assets inside the building. Additionally, if the BIM is used for collaboration between stakeholders, the record will also be an

updated status of the elements belonging to other organizations, this helps to have an accurate visualisation of the building without being there.

2) Provide on-site guidance for field engineers in their maintenance or repair tasks. Using mixed reality technologies, it is possible to extract information from the BIM model and display it in the field engineer's view to help him. For example, some of the assets (such as cables, pipes, and telecom switches) might be hidden behind walls, and the field engineers need to consult the schematics of the network in the building which are normally in a 2D paper-like PDF format, visualising the information through a mixed reality headset will facilitate the task.
3) Another great benefit is the use of BIM for planning and understanding how different elements interact with each other. With a BIM model there is a clear picture of what is inside a building, and it is possible to connect the assets inside to the network outside, creating a fully connected network in a digital environment. Remote planning tasks will benefit the most of it, for example, it will be possible to understand what assets and how many customers will be affected by a disruption or change in the network since everything is not connected.

6.2 Connecting Buildings to the Network Digital Twin

As mentioned in chapter 4, a Digital Twin consists of three main components: a) the physical asset, b) the virtual asset and c) the bi-directional connection for exchanging data between the two assets. A BIM model is already a virtual representation of a physical asset, which is the actual building with all its elements. The missing component is the bi-directional connection between assets, the physical asset will pass information to the virtual asset and update it, while the virtual asset processes the information and might trigger an action in the physical asset. A BIM model just by itself has no connection to the actual building, considering the levels of digital twin proposed by Kritzinger et al. [49], the BIM model will be considered just a digital model. There are four different levels of BIM Maturity (from 0 to 3) according to the UK BIM Maturity Model [91], level 3 BIM (highest level) is a fully interoperable model capable of working together and exchanging information with other entities, mainly the physical building, this level of BIM can be considered a digital twin [50]. The interoperability is key for the BIM model to become a digital twin because most of the updates will be gathered from different organizations. The different utility companies can provide information about how much resources (such as energy, internet, gas, and water) the building is consuming and if advanced smart meters are used then this can be known at any point in the day. Essentially, the utility companies act as a sensor to monitor the current state of the building, the same way any other sensor will monitor other assets. The digital twin can then trigger events like maintenance tasks or emergency repairs if it detects anomalies in the information of the physical asset, this will result in a change in the physical asset, this way the bi-directional communication between both assets is achieved.

The components seem to be available, and it is only a matter of connecting them for them so that the BIM models start cooperating with the network digital twin. However, from the perspective of a utility company there are two main challenges:

1) Creating BIM models is a complex task, it requires 3D modelling for the geometric shape and the combination of multiple sources of information. The cost of creating them is high, which might be the main reason why the adoption of BIM outside the construction industry is low [92].
2) BIM is a process developed in recent years, which means that most of the existing buildings were not designed using BIM. Utility companies still need to work and interact with existing buildings that are connected to their network. This means that utility companies need to work with two different sources of information (existing

buildings and new buildings with BIM). The integration of information is key for progressing and improving the processes, it is not only important to have the data [90], but being able to standardised all information from the building to a single format and integrated with the network data is essential for utility companies to start thinking of using BIM.

Both challenges are connected and are mainly related to the information on existing buildings and how to integrate them into the network digital twin that utility companies seek to create. New buildings will have their BIM model created by the construction companies, and then reused and maintained by stakeholders of later stages in the building lifecycle.

An automatic data processing pipeline is proposed to address the challenges mentioned above and help utility companies use BIM in combination with the existing information of their network assets. The idea of the pipeline is to use 2D images of the floor plan of the building to generate a 3D digital representation of the building. The 3D digital representation of the building will be in a standardised open-source format for BIM. This output BIM model will not automatically be a digital twin of the building, it will be a digital model (a low-level digital twin). However, this is the first step towards a digital twin, and it is the step with the highest cost [92], once the structure and 3D model of the building is created then it is possible to start connecting everything.

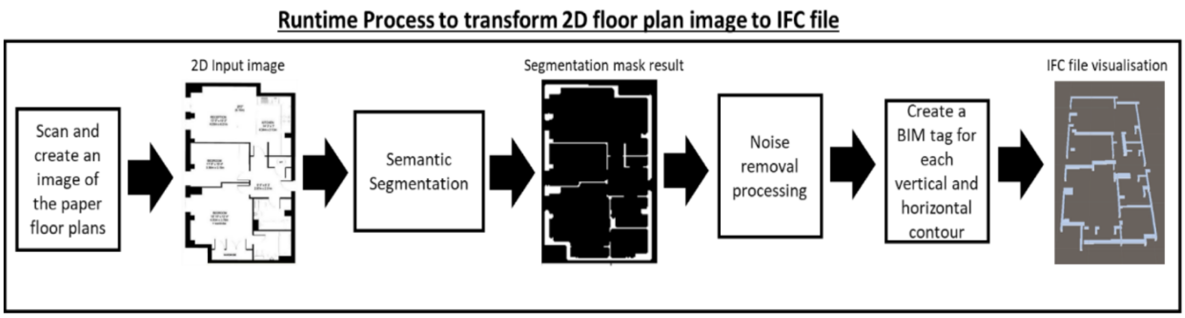


Figure 6.1. Proposed processing pipeline for converting 2D floor plan images to standardised BIM models.

The data processing pipeline is shown in Fig. 6.1, this was initially presented in [9], [42]. The goal is to transform 2D floor plan images into standardised BIM models. BIM models are very detailed 3D models and not all the information that you will normally find in a BIM model can be extracted from an image. However, it is believed that at least the basic structure of the building floor can be extracted and create a Digital Model of the building. This process is divided into the following four main stages:

1) Digitalisation of Existing Documentation
    a. The first stage in the processing pipeline and the goal is to prepare the input data, and make sure the format of the data is adequate for the next steps. The idea of the complete process is to use 2D data of existing buildings to generate the 3D structure of it. However, the proposed process does not work with any 2D data, it requires 2D images as input format and some of the available information will not be in this format, e.g., architectural paper drawings detailing the walls and dimensions, or PDF format files with the floor drawings. Hence, the need for this pre-processing step to standardise the input data to be in any of the common digital image formats (such as PNG or JPG). It is important to mention that this is not a fully automated step, in some cases it will require manual work from a human, e.g., if the available information of a building is only architectural paper floor plans, then a human is required to scan these paper floor plans.
2) Semantic Segmentation of the Digital 2D Images
    a. This is the main stage of the complete pipeline, the other 3 stages are there to help with data preparation, data cleaning, and standard format tasks; however, identifying elements of interest is the main goal of this step. The structure of a building floor is composed of all the wall elements in the

image, the walls define the boundaries of the floor and the space distribution inside it. Every wall element in a 2D image is a collection of one or more pixels; depending on the resolution of the image, and the size of the wall how many pixels belong to the same wall. This means that a semantic segmentation process is required to correctly identify which pixels are part of a "wall element" and which ones are either "image background" or "other element". In previous the chapter an explainable semantic segmentation approach using a FRBS was presented where a similar approach is used here. However, using only pixel value information, as previously presented, represents a challenge with floor plan images since different elements tend to share colour features. Therefore, an improved FRBS for semantic segmentation is presented in this chapter, this new approach seeks to use context information to differentiate between elements of interest and other elements that share colour (or pixel-level) features.

3) Noise Removal Process

   a. The process from stage 2 is not a 100% accurate process and incorrect pixel classifications are expected to happen. However, there are some post-processing image techniques to remove some of the noise generated in the predictions. The "salt look" is one of the most expected effects when working with this pixel classification tasks, this happens when some isolated pixels are incorrectly labelled as being part of one of the objects of interest, this isolated wrong label will look in the output as a white dot surrounded of black background, hence the name of "salt look". All these pixels are removed by using median, erosion, and dilation filters on top of the output image. However, this step works only for high resolution images where there are no “wall elements” represented by a single or a very small set of pixels, if that is the case, this automated noise removal step will think those isolated pixels are noise and change them to be part of the background.

4) Converting Segmented Output Image to BIM file

   a. This last stage is about pixel grouping and tagging stage. During this stage, the goal is to group and separate the different clusters of wall element pixels in the segmented image. As mentioned earlier, BIM models have the distinctive characteristic of having each of the different elements separated, therefore, it is important to create a 3D geometric shape for each wall instead of single 3D shape for the complete floor. The segmented image is divided into segments of straight lines using image processing Sobel filters, with the assumption that all “wall elements” are represented in the floor plan by straight lines, and there will be no circular or curve “walls”. Once all the straight lines are separated, the pixels belonging to those lines are clustered, and the contour of those clusters is found, all of this with the help of the python image processing library called OpenCV. Finally, using the extracted information of those clusters of pixels an Industry Foundation Classes (IFC) tag is created for each “wall element”. IFC is a global open standard for data exchange, and it is used to describe and share construction and facilities management information, it is also considered to be the open-source format for BIM models [92].

6.3 Extracting the Wall Elements from 2D Floor Plan Images

In a floor plan image, it is possible to find non-structural building elements (such as kitchen elements, furniture, and background) and there are structural building elements (such as doors, windows, and walls). Every pixel in the image belongs to an element, or the background. The objective of the process is to find all the “wall elements” so that a 3D BIM model of the building floor structure can be constructed. Therefore, the approach for finding the elements is to do a semantic segmentation, where essentially, as described in chapter 5, each pixel is classified and assigned a label. As mentioned in section 6.2, using the same approach as in chapter 5 will not be possible because of the similarity in pixel colour between different elements, and due to the lack of context information it is not possible to distinguish between those pixels. Fig. 6.2 shows an example of the input image and the expected output, this is the kind of output the process is trying to achieve; every white pixel represents a part of a wall element. In this section, the approach to how to use patches of images for contextual information is described.

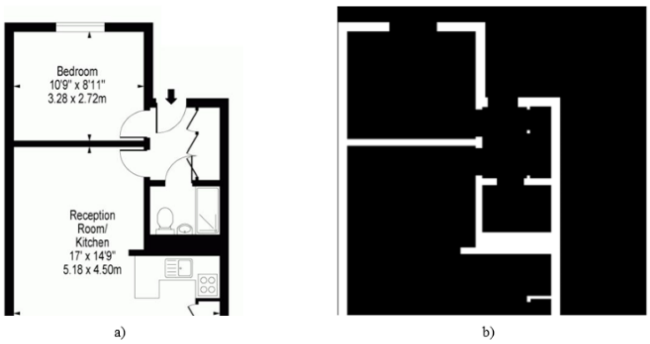


Figure 6.2. These are: a) an example input image that is going to be converted to 3D BIM model and b) the expected output for the segmentation stage.

6.3.1 Visual Words Dictionary

The concept of visual words dictionary (or Bag-of-Visual-Words) was inspired by the Bag-of-Words concept in the Natural Language research field, and it was adapted to work with images instead of text. The idea is that you have a representation of a small image patch (normally in a numeric vector format) that represent an element in the images. These representations of image patches are pre-computed and saved to calculate the similarity of input patches with well-known patches. Similarity is computed using Euclidean distance between vector representation of patches, a low distance value means high similarity to the known patch.

This idea of using image patches for semantic segmentation was first introduced by [93], [94], they combined the information from patches with a Markov Random Field model. The approach was later extended by [95], [96], where the authors trained a Support Vector Machine (SVM) model to segment floor plans using only patch-based input features, the input pixel to be labelled gets assigned a class probability based on its distance to different patches. This is an interesting idea because instead of using pixel-level characteristics, they use only context information (represented by the patches) to classify a single pixel. The floor segmentation process to create 3D BIM models was presented in [9], and it was based on the initial research of building a type-1 FRBS using patches as contextual information [43] and the continuation of that research using an interval type-2 FRBS [42].

The necessary first stage is to pre-compute a list of patches (visual words dictionary) that is going to be used as the knowledge base of our system. The number of visual words is a hyperparameter of the proposed approach and it needs to be modified according to the task, in [9] numbers between 50 and 300 visual words. Create the visual words dictionary using the training dataset of images in the following two steps described by [9]:

1) Divide the training image into patches. The image is divided into sections by using a grid. An overlapping grid is used to avoid the location dependency of the object in the image [96]. After extracting the patches, they are transformed into a numeric vector representation following a row-wise approach and then Principal Component Analysis (PCA) feature selection is applied to it. Two main parameters can be changed in this step: the overlapping value in the grid, i.e., how many pixels overlap between patches), and the size of the image patches extracted. [9]
2) Cluster the extracted patches using the k-means algorithm. The centroid of the computed clusters is the numeric vector used as a visual word. There is a visual word for each cluster; therefore, the number of visual words determines the value of k in the

clustering algorithm. As mentioned before, this k is a hyperparameter of the model, and it can be optimised. It is essential to consider that the higher the value of k, the smaller the clusters and the more specific to a training set the model becomes, it can be seen as overfitting of the model. On the other hand, the smaller the value of k the larger the clusters and therefore the most likely to have different patches clusters together, this will be the equivalent of underfitting. Additionally, when increasing the k value, the number of possible rules will also increase. [9]

6.3.2 Rule Extraction and FRBS Optimisation

The goal of this approach is to find a way of combining individual pixel-level information and context information of the surrounding of the pixel of interest, and by doing so, be able to distinguish between pixels with similar individual characteristics (such as colour).

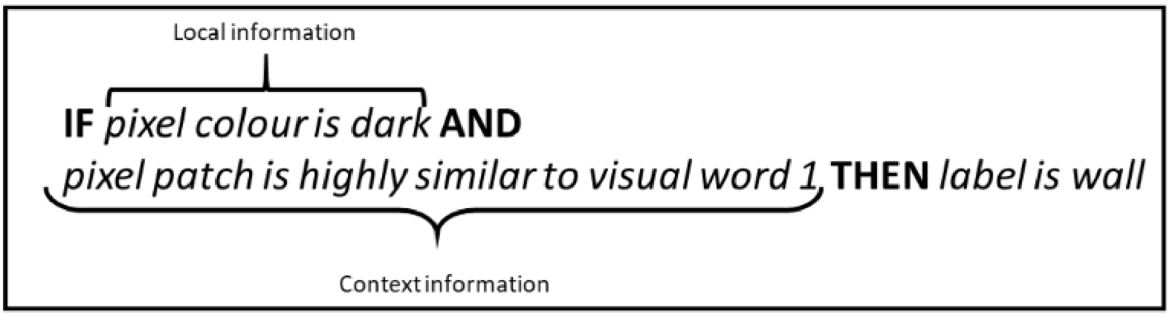


Figure 6.3. An example of a rule used in the FRBS model. The two types of information (pixel-level and contextual) are highlighted in the image.

The intention is to have rules as the one shown in Fig. 6.3, two antecedents, one for each type of information: pixel-level features and context features. The pixel-level features (or local information) are the characteristics of the pixels, in this case, the colour is used, similarly to what was done in chapter 5. On the other hand, context features are the similarity of the patch where the pixel of interest is located, it provides information about the surrounding of the pixel and helps to differentiate from pixels with similar colour that belong to a different element.

The use of this context information is the main improvement from the FRBS used in chapter 5 to the ones used in here.

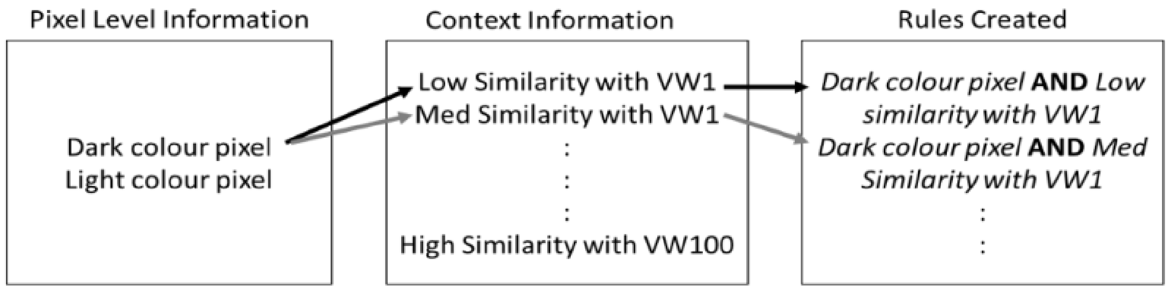
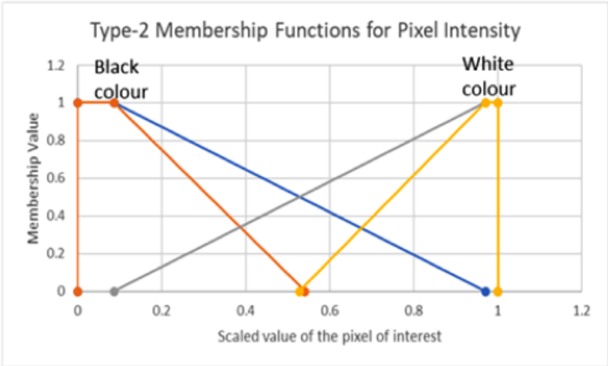


Figure 6.4. Combining the different antecedents to create the initial set of rules.

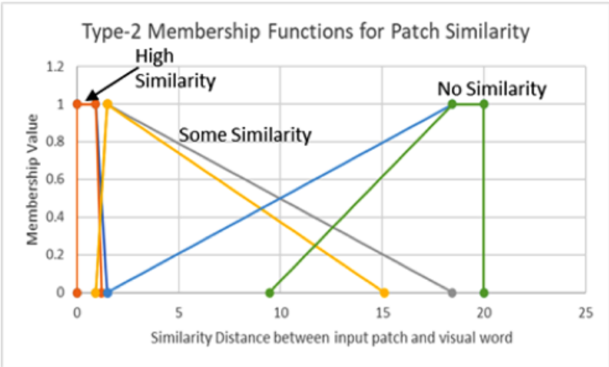
Fig. 6.4, shows how to generate all the possible rules for the system, it is important the constraints used in this process. The first constraint is that the rules will only have two antecedents, the second constraint is that the first antecedent is the pixel-level information, and the second antecedent is the context information. For “Pixel Level Information”, since the images are grayscale images, only two linguistic values were defined, one for dark colour pixel and one for light colour pixel. The number of linguistic labels in context information is defined by the number of visual words in our approach. In the work presented in [9], different values were tested but the best performance metric was achieved with 100 visual words. The similarity with each visual word is described with 3 linguistic values (low similarity, some similarity and high similarity). This means that for context information there will be 300 possible antecedent values. In a FRBS model with 100 visual words, the initial total number of rules will be 600, the similarity to each visual word is described with three linguistic labels (similar, somehow similar, not similar), and the pixel intensity value is described by two linguistic labels (dark and white). When extracting the consequence of the data, we make use of all the rules. In a later stage this rule base can be optimised using the BB-BC process described in chapter 3 of this thesis.

The previous is the process for creating all possible rules, but the consequence label still needs to be assigned to each of those rules. For the rule modelling phase, the process described in [9] is followed. Each of the M rows will be a training pattern called $t(m), where\ m = 1,2, \dots, M$ that consist of a vector $x(m)$ and a class $c(m)$. The vector $x(m)$ has all the needed information related to the pixel, and this includes the pixel intensity value and the similarity of the image patch centred at the pixel to the patches in the visual words dictionary. For example, if the model uses a dictionary of 100 visual words, the vector $x(m)$ will contain the numeric value of the pixel intensity and 100 numeric values. Additionally, $r(m)$ will also have the linguistic value $c(m)$, which is the expected class for the pixel information $x(m)$. Rules without a consequence will be removed from the rule base and conflicts between rules will be solved by using the rule modelling process described in chapter 2.

It is important to mention that this is a computational expensive process, because the fuzzy sets are being optimised at the same time. This means that for every candidate solution in the BB-BC algorithm process described in chapter 3, there is a need to extract the rules when using those fuzzy sets. This happens because the rule modelling process is dependent on the fuzzy sets to calculate firing strength of the training data. However, at the beginning the optimal fuzzy sets have not been found. The rule modelling steps will be part of the BB-BC algorithm when an encoded candidate solution is converted to an actual FRBS. The candidate solution is evaluated, and the best solution will use the optimal configuration of membership functions decided by the BB-BC and the rule base extracted by the rule modelling process using those membership functions. Fig. 6.5 shows the optimal fuzzy sets after the BB-BC optimisation experiments done by [42].



a)

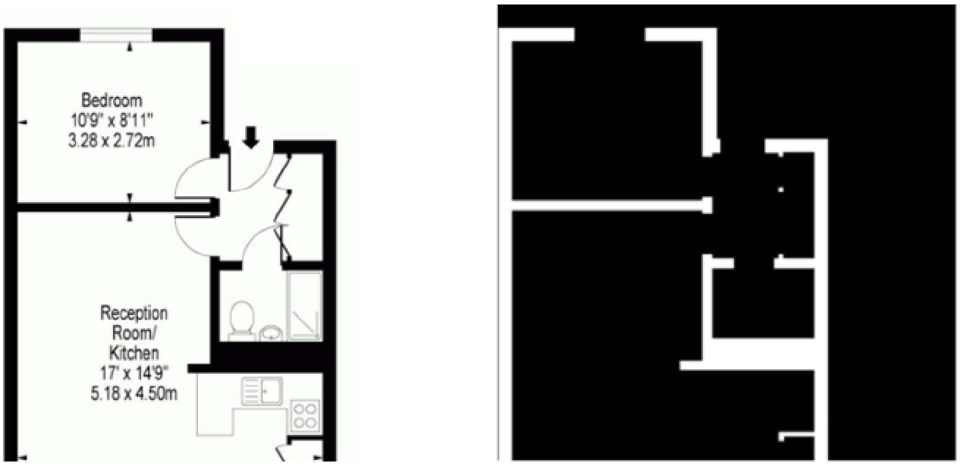


b)

Figure 6.5. Optimal fuzzy sets according to the BB-BC optimisation process applied on the data, these results were presented in [42].

6.3.3 Experiments and Results

This section discusses the experiments and results from [9], [42], [43] and how these results and process can be used to create the expected 3D building object from the 2D images. The type of images segmented by the different models tested is shown in Fig. 6.6. The dataset for training consisted only of 168 images like the one shown in Fig. 6.6. A visual comparison with the images used for the experiments in [97], [98] can conclude that the images used in this experiment are easier to handle for the models since there is much less noise (elements that are not part of a wall). However, there are still three challenges that the models need to overcome with those type of images: 1) identify the difference between text and wall, 2) furniture or decorative elements such as kitchen and bathroom elements, 3) remove structural elements that are not wall, like doors and windows.



a) b)

Figure 6.6. Example image in the training and testing dataset. a) shows the actual input and b) shows the expected segmented output for the image if it was processed by a perfect model.

The results shown in this section are the results of a neural network U-Net architecture [99], which is a convolution-based type of network considered part of the state-of-the-art for semantic segmentation tasks. The training dataset is too small to train a network from scratch but a pre-trained model can be used as the encoder part of the U-Net [99], in this experiments, pre-trained layers from the VGG-16 network are copied and used, this is known as transfer learning [100] and it is a common practice to avoid training a network from scratch when the computational resources or large numbers of data are not available. A visual representation of the architecture that was created for [9] is shown in Fig. 6.7 below.

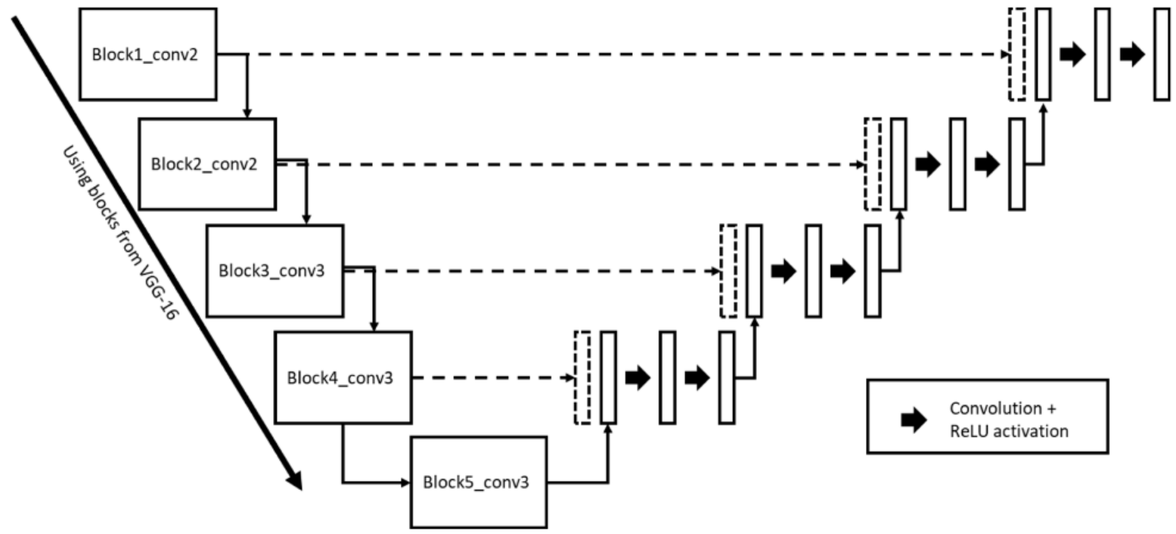


Figure 6.7. CNN architecture built in [9] for the segmentation task. It uses pre-trained layers from the VGG-16 network as the encoder.

Like chapter 5, the Intersection over Union (IoU) is used as the performance metric during the evaluation process. The optimised interval type-2 FRBS achieved a 97.5% IoU metric value, on the other hand, the U-Net deep learning approach achieved 99.3% IoU metric value. The visual results are shown in Fig. 8 for the proposed explainable approach using an interval type-2 FRBS and the visual results for the deep learning solution using a U-Net with VGG16 architecture is shown in Fig. 6.7.

Figure 6.8. Segmentation results of the optimised type-2 FRBS. a) shows the result of the segmentation process and b) shows the result after the noise removal process.

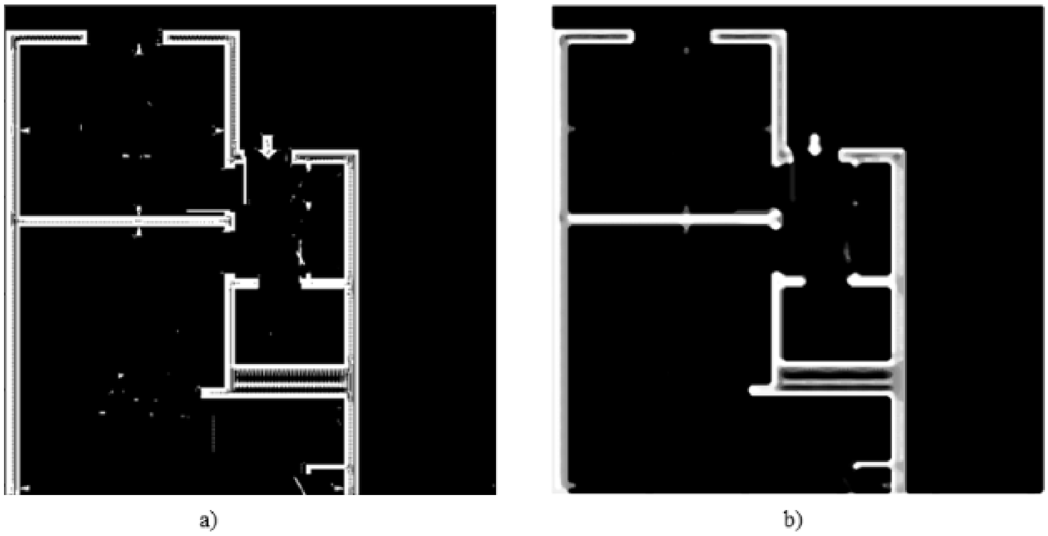


Figure 6.9. Results of the U-Net deep learning approach for the segmentation process. a) shows the segmentation result and b) shows the result after the noise removal process.

Although in the metric values the difference between the interval type-2 FRBS and the deep learning approach is not big, it is possible to see in Fig. 6.8 and 6.9 the better performance of the deep learning solution. The interval type-2 FRBS struggles to differentiate between the straight-line walls of the building floor structure and the straight-lines of the text in the floor plan drawings.

Our main conclusion from these visible results is that our proposed FRBS rules are still substantially based on colour, even though we included patch similarity information. This explains most of the visual errors of the FRBS in Fig. 6.9. However, CNNs are black box models that are not interpretable or augmentable, i.e., a human end-user will not understand the system 's decision process and will not be able to modify it [26]. Additionally, these models need large numbers of training data patterns to perform well. Obtaining data is costly, especially for semantic segmentation where the label needs to be pixel by pixel [79], [80]. On the other hand, the FRBS model has the advantage that a human end-user will be able to trace and understand the decision process and will also be able to modify the model by changing, adding, or removing antecedents in the rules (or even modifying complete rules) of the model. This allows the model to be improved by using expert knowledge without training (or optimising) it from scratch [26]. The interpretable and augmentable characteristics of the interval type-2 FRBS are not enough to choose that model over CNNs. However, it is a different approach that might be necessary due to internal or external policies related to the accountability of decisions and companies should be aware and choose the model that better fits their needs. An example scenario where an augmentable and interpretable model might be preferred over the higher performance of the CNNs is when there is a conflict between network assets and the BIM model created. An engineer will need to solve the conflicts, understanding why the model is built that way will help fix it, and it might also help the engineer improve the model by modifying the rules that were fired. However, a company might prefer to have the model with the highest possible performance deployed in a remote server and let the model do all the work. In this case, the CNN becomes a better option [9].

6.4 Creating a BIM Digital Model from Segmented Image Results

Once the segmentation process is completed and a segmented result mask (like the one shown in Fig. 6.8a and 6.9a) is obtained, it is now time to automatically create a BIM digital model for each of the wall elements detected. The first step is to process the output with some median blur, dilation, and erosion filters to remove isolated wrong predictions. The output of the noise removal filters can be seen in Fig. 6.8b and 6.9b, they visually look cleaner because the noise removal process works under the assumption that there are no single pixels that can be a wall, so this kind of pixels are removed, and the "salt and pepper" look is removed. This process was previously used in chapter 5 of this thesis.

The next step is to find the clusters of pixels that make up a wall element. The steps from chapter 5 when finding the clusters of tree pixels are repeated here but with a small difference. The assumption is that there are no curve walls in the floor plans that are being processed by the models so first the segmented output is separated into straight vertical and horizontal lines using Sobel filters. Now, it is possible to find the contours of the walls. Using a python library called IfcOpenShell [101], a standardised IFC tag is created for each element and all of the tags are compiled in a single file to create a BIM model of the floor. Information on additional floors in the same building can be added to the same BIM model file.

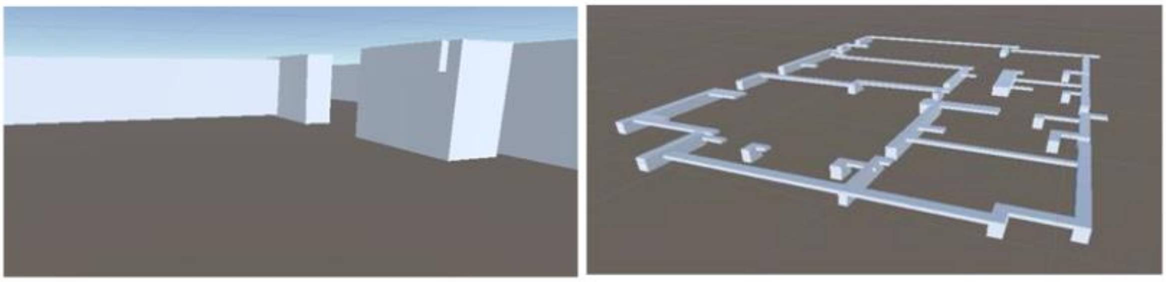


Figure 6.10. Visualisation of BIM models created from the floor plans. Left side image is a First-person view of the BIM using the Oculus Quest headset. Right-side image is a top view of the model in the same virtual environment.

In Fig. 6.10 the result of the created BIM models is shown, this is how they look from different perspectives in a virtual environment. Two important things to highlight as part of the conclusion of the process:

1) The height information of the walls and the material information cannot be extracted from floor plans, so it is not part of the BIM model, but it is possible to add the information. Every wall has its own IFC tag and a relationship between the wall tag and the material property type can be made. The BIM models are still not as data-rich as it would be if it was manually constructed but the structure is in a standard format and information can be added with other processes.
2) The BIM model as of now is Digital Model of a building floor in the levels of Digital Twin maturity level [49], i.e., it is just a virtual representation of the actual building, it still doesn’t have automated communication with the physical entity, so according to these levels of maturity from Kritzinger [49], it is still not a digital twin but it is the right first step to build a digital twin from 2D images.

Future work will explore how to improve the final 3D model output by including other sources of information and how to establish a communication link between the digital model and physical building.

6.5 Discussion

In this chapter, the concept of BIM was introduced, a process for different stakeholders to cooperate in a centralised model during the different stages of the building's lifecycle. The output of this process is a data-rich 3D model file known as a BIM model that contains the historical and current information of every single element in the building. The amount of information and relationships available in the model allow us to do smart and advanced tasks such as cost forecasting, maintenance prediction and construction scheduling. Utility companies such as British Telecom can benefit from using BIM models to understand the location and surroundings of their network assets within a building.

A BIM model is considered, at most a digital shadow of the actual building, it doesn't reach the digital twin level due to the lack of by-directional connection. However, the model can be placed in a virtual environment and can be connected to other digital twins and serve as contextual information for other objects.

In the previous chapter 5, the goal was to improve the network digital twin by adding context information about natural objects such as trees. To understand the surroundings of the outside network. In this chapter the goal is to understand the surroundings of network assets inside a customer premise by using BIM models. Since BIM has become a standard in the construction industry, it is expected that new buildings will have a BIM model associated with them. However, the information from existing buildings consists mostly of paper-based documentation or information from different locations. This represents a challenge for the utility companies that seek to connect all customer premises to their network detailed twin and create a fully connected network that contains both the outside and the in-building assets.

This chapter presented a process about how to transform the information of 2D floor plan images to an open-source standard format BIM file. The main stage of the process is the semantic segmentation of the floor plan images to automatically identify the pixels belonging to wall elements, the information on the identified pixels will then be used to create the BIM model. This chapter proposes the use of an interval type-2 FRBS as an explainable AI approach for the segmentation process and presents a comparison of this approach with a deep learning black box model. The results of the experiments can be summarised as follows:

- The interval type-2 FRBS outperforms the type-1 FRBS even after adding a noise removal stage in the process.
- The deep learning approach has the highest performance value. However, it is possible to remove most of the incorrect classifications from the interval type-2 FRBS around the text of the floor plan using the noise removal stage of the process. Thus, ending with a similar result as the black box alternative.

This chapter improved the interval type-2 FRBS for semantic segmentation presented in previous chapter 5. The proposed FRBS in this chapter includes the use of context information, which allows the FRBS to distinguish between pixels that have similar characteristics but belong to different objects, e.g., for example in the floor plans analysed wall and door elements are black colour, so distinguish between these two elements it is important to understand how the surrounding pixels are located. As this described in this chapter, it is achieved by using pre-labelled image patches and calculating the similarity between the input patch of pixels surrounding the pixel of interest and the pre-labelled list of patches.

The work presented in this chapter contributes to the idea of creating a network digital twin consisting of different levels of digital twins. Even though BIM models might not be considered complete digital twins, they can still be placed in a virtual environment and used as digital models that provide context information of a physical location. This chapter concludes the work related to the use of explainable AI alternatives for processing data to create digital models or digital shadows that will help improve the network digital twin.

The work in next chapter will focus on how to combine different sources of information in a mixed reality environment to understand what the user is doing and provide feedback on it. The mixed reality environment is a combination of the virtual world where the network digital twin exists and the real-world view of the user. This and previous chapters have discussed explainable AI alternatives to process data for creating digital twins, next chapter will focus on discussing an explainable AI alternative to process data for interacting with digital twins and real-world information.

Chapter 7. Fuzzy Rule-based Systems for Task Guidance in Mixed Reality Applications

7.1 An Overview of the Architecture of the System

In the previous chapters the focus has been on using interval type-2 FRBS to process real-world objects data to create digital representations of those objects in the virtual environments. In this chapter the focus is to apply a FRBS to process in real-time the data that is being captured by the mixed reality application. Mixed reality (MR) is the combination (it can be in different degrees) of both the real and the virtual world, the concept of augmented reality (AR) is part of MR and it is a view of the real world with a low number of virtual objects [51].

The proposed system is designed to work for an application used in human-centred AR assistance that provides field engineers with on-site support [102]. The system identifies all useful sources of information in the field engineers' view, combines them through an inference process and provides the field engineer with some guidance (or warning) while working on their task. From a utility company point of view, this will help reduce costs in training and revisits to on-site tasks, i.e., sending a second field engineer to complete the job because the first one could not or because it was not completed correctly. From a field engineer point of view this will help improve the quality of the work, reduce the time spent per task, and complete the task in a safe way since the system can alert of any potential hazard.

The use of a FRBS can handle the uncertainty in some of the input values, e.g., it is a complex task to define the limits of a target area in the 3D world, because different engineers will draw different area boundaries. However, a central target in the area can be specified and a fuzzy membership function can determine the degree of membership of a given location to that centroid, the higher the membership value, the higher the confidence of the field engineer's hand being the expected area. The handling of the uncertainty is a great advantage of using a

FRBS to combine the input information, however, the most important advantage of the FRBS is that it uses an interpretable inference engine for the decision making, which can be; 1) understood by the users creating the different use cases and 2) easily modified if needed by expert engineers.

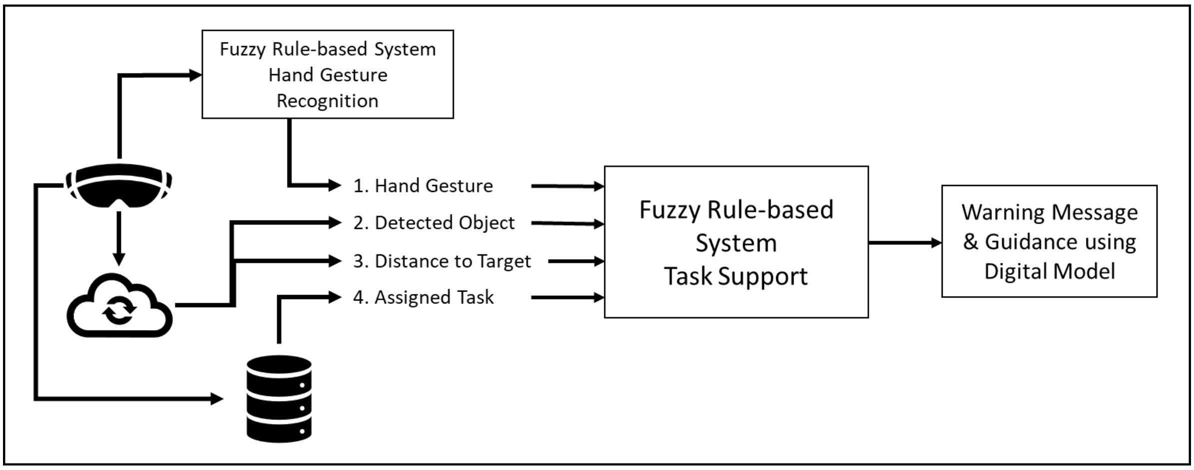


Figure 7.1. Architecture of the system using two fuzzy rule-based systems to support field engineers on the job.

Fig. 7.1 shows the proposed architecture of the complete system, the two main things to notice in this figure are: 1) there are two fuzzy rule-based systems and 2) the starting point of the decision process is the mixed reality headset which triggers the request for guidance and sends the sensors information to the different processing components. The mixed reality headset will trigger the following 3 actions:

1) Send to the first FRBS the available hand tracking information to recognise the field engineer's current hand gesture.
2) Send an image of the field engineer's current view to a cloud web service that detects the boundary of the equipment the engineer is working on. The cloud service is expected to return the coordinates of the bounding box, the label of the detected object and the confidence of the detection. These are inputs for the FRBS that makes the classification on whether the user is doing the right thing or not. It is important to notice

that if the headset device has the capability to run the object detection model, then there is no need to send the information to a cloud service.

3) Get the type of task that the field engineer is expected to do which should be registered in a system inside a company database. The headset can use the information of the user that is logged in to the device to create the query to the database.

The objective of the previous 3 actions is to get the input values for the FRBS to decide on whether the field engineer needs help or not. These inputs come from different sources, and they need to be pre-processed, the four inputs to be used are:

1) Label the detected hand gesture by one (or both) of the user's hand. Instead of the input being a crisp numeric value, it is just a linguistic label that comes from the output of a FRBS, therefore, this input is not fuzzified or mapped to another linguistic label. The main reason for having a separate FRBS for hand gesture recognition is to maintain the complete system explainable to the human user. This is further explained in the next section where a detailed description of the FRBS for hand gesture recognition is provided.
2) The image of the field engineer's view is analysed by an object detection model to find the location of the bounding box that surrounds the object of interest, i.e., the network asset where the field engineer is performing some actions. In a similar way to the first input, this is just a label that guarantees that the rules related to this object will only get activated when the object is in the engineer's view and the message will not get displayed under other circumstances.
3) The distance to the target is the only crisp numeric input that will get mapped to linguistic labels by the FRBS. It relates to the position of the user's hand with respect to a target area in equipment. Every action the field engineer needs to do in the equipment, must be completed in a specific area, e.g., put a screw in the top left corner

of the box, the top left corner becomes the target area, that is where the user's hands are expected to be. Section 3 of this chapter includes a further detailed description of how this input is calculated and used alongside the other inputs.

4) The last input of the FRBS is a linguistic label that specifies the type of task the field engineer is meant to be doing. On-site tasks are assigned to all engineers before the day starts through some digital system, the idea is for the system to use this available information to know the type of task. The reason is that it is possible to have rules that seem to be conflicting because they share the other 3 input antecedents, however, they do not conflict with each other because the expected task is different.

This is an overview of the system architecture and how the inputs are obtained after processing some of the different sources of information. The following sections discuss in detail the two FRBS used in the system. Section 7.2 describes how a FRBS is used to build an explainable and augmentable hand gesture recognition model. Section 7.3 discusses how all the inputs are combined and how the target area is used to capture the business knowledge on where the field engineer's hands are expected to be for each task. Finally, the last section of the chapter shows an example use case and some of the results of the system.

7.2 An Explainable AI Approach for Hand Gesture Recognition in Mixed Reality Environments

7.2.1 Overview of the Hand Recognition Systems in MR and VR Headsets

Mixed reality and virtual reality hardware (such as Microsoft HoloLens 2 and Oculus Quest 2) and applications have considerably progressed in recent years, and a broader audience is acquiring and interacting with these devices [103]. One of the key elements when interacting with these applications is the communication between the user and the extended reality displayed in the user's view. An important aspect for the user to be fully immersed in the mixed reality environment, is the interaction with the virtual objects, it must feel natural. To achieve a natural interaction between hardware and user, the headsets seek to use a combination of voice commands recognition and hand gestures recognition. However, the hand gesture recognition systems currently used are not flexible enough [104] to be adapted to new gestures, therefore, the users are limited to the gestures the system is trained to recognise.

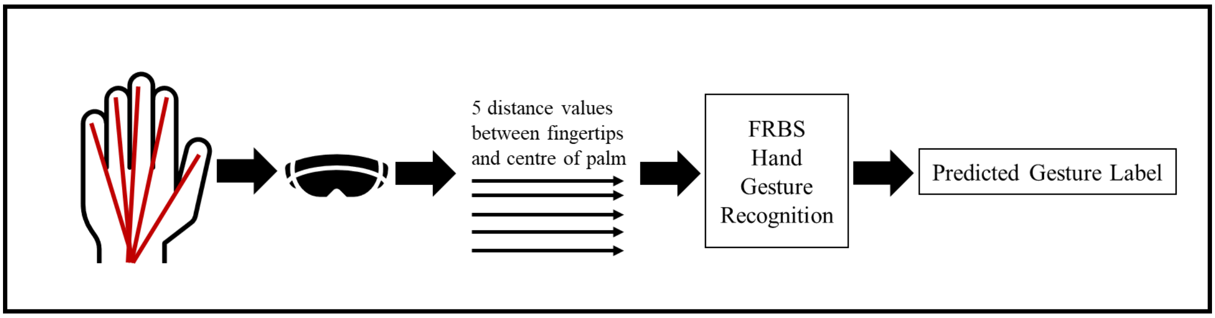


Figure 7.2 Diagram to show the process of capturing data from the hand using the HoloLens 2 to then output a gesture label using a FRBS.

Our work presented in [105], proposed a FRBS for detecting the user's hand gesture in extended reality environments that use the Microsoft HoloLens 2 or the Oculus Quest 2. The main advantage of the proposed method is the high degree of interpretability in the inference engine. The user can add, remove, or modify existing rules to change which hand gestures are detected. This way the new system provides the necessary flexibility for any company or user to adapt the inference engine and detect the needed hand gestures for their processes, instead of adapting the user's behaviour to fit with the available hand gestures. Fig. 7.2. shows how the information flows from the start, when position of the hand joints is captured by the HoloLens 2 and distances between them are calculated, until a hand gesture is predicted for that given point in time. The process to build the FRBS and how it is used in the headsets is described in the following sub sections 7.2.2 and 7.2.3.

7.2.2 Selected Features and Data

The Microsoft HoloLens 2 and the Oculus Quest 2 devices include a powerful hand-tracking system that provides information on the position of the hand joints, fingertips, and centre of the palm. Fig. 7.2, shows the different points that are tracked by the camera of the headsets.

Figure 7.3. The red dots highlight the points from each hand, as recorded by the hand tracking system in the Microsoft HoloLens 2 and Oculus Quest 2. The image is captured using Oculus Quest 2. Image from [106].

The idea of the hand gesture recognition FRBS is to take advantage of the available hand tracking system instead of extracting the hand information from scratch using the camera sensor. There is no need to duplicate work when the existing hand tracking systems are highly accurate. The FRBS seeks to create a flexible system that works on the already existing information instead of replacing the highly accurate systems.

Hand gestures are categorised as static or dynamic [107]. Static gestures are those in which the position of the hand remains the same (or very similar) over time. On the other hand, dynamic gestures are when some parts or all the hand changes over time, but the different positions of the hand are a single gesture, e.g., waving, the position of the hand changes over time, but the gesture is still waving. The scope and goal of this chapter are limited to detecting static gestures. The FRBS uses the distance between the fingertips and the centre of the palm as the input features. This means that there are 5 inputs (one for each finger), and therefore, 5 antecedents in each rule. Fig. 7.3 shows an example rule of the FRBS, in this rule, if the 5 fingertips are close to the centre of the palm, then the gesture is a “Fist”.

IF
thumb is close **AND** *index is close* **AND**
middle is close **AND** *ring is close* **AND**
pinky is close
THEN
hand gesture is a fist

Figure 7.4. Example of a rule in the FRBS.

Given that no public fingertips position datasets were found to train and evaluate different AI approaches, an application to capture data points [106] (i.e., the fingertips' position and palm's centre for both hands) was built using Unity3D. The application was executed in Oculus Quest 2 (virtual reality headset) to generate the training data for optimising the different approaches. However, the same application can be executed in the HoloLens 2 (mixed reality headset) since it is built in Unity 3D, which compiles for different devices. In this application, the user is requested to gesture with both hands whilst the data points are recorded in a CSV file. A data set of 10,000 training patterns was created with the application. The possible hand gestures in the data set are shown in Fig. 7.4. This file was used for optimising the type-1 FRBS and interval type-2 FRBS using the BB-BC algorithm described in chapter 3, and KNN algorithm. This data set was divided into 8,000 rows for training and 2,000 for validation. After completing the training, the three models were evaluated using a different data set created in the same way but with 3,000 input patterns. The reported accuracy values come from the predictions on this second dataset.

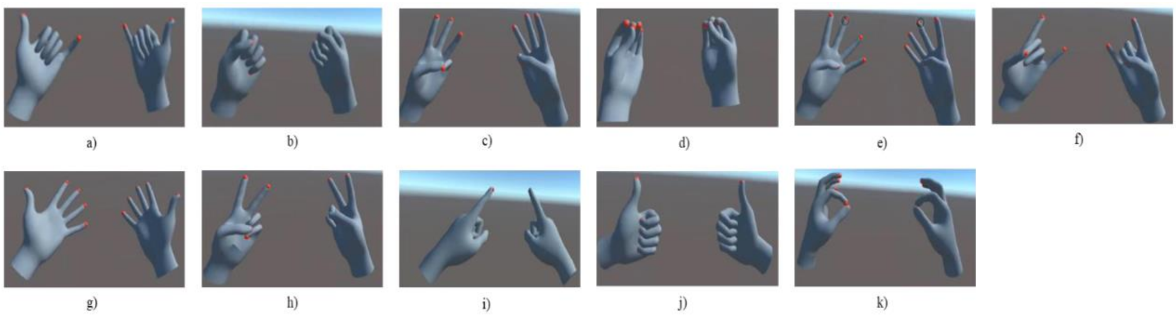


Figure 7.5. These are the available hand gestures in the dataset. The labels for the gestures are: a) shaka, b) fist, c) three, d) flower, e) four, f) horns, g) palm, h) two, i) point, j) ok, k) pinch.

7.2.3 Experiments and Results

As mentioned in previous section, the different models were evaluated using a testing dataset. However, the interval type-2 was also implemented in the HoloLens to evaluate the results visually, these results can be seen in Fig. 7.5.

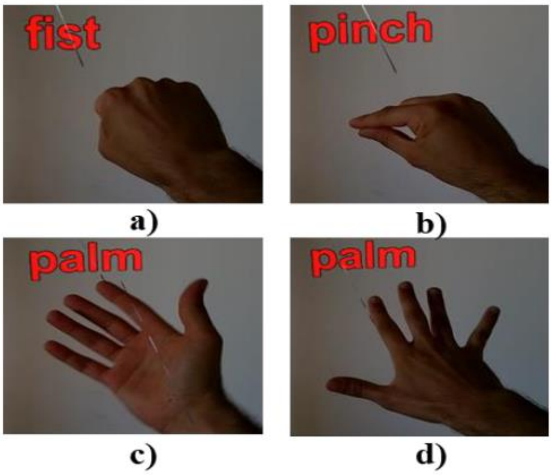


Figure 7.6. Real-time prediction of the interval type-2 FRBS in the Microsoft HoloLens 2 device. Predictions: a) fist, b) pinch, c) palm facing user, and d) palm not facing user.

In the evaluation process of the models, the KNN (black box solution) showed an accuracy of 98.9%, the type-1 FRBS had a 94.6% accuracy and the type-2 FRBS had a 96.4% accuracy. The difference between the black box solution and the proposed type-2 FRBS is 2.5% accuracy and in practice both have great accuracy as seen in Fig. 7.5, the difference comes in how both models handle the moments of transition between gesture, e.g., going from an open palm to a completely closed fist, the KNN did better at predicting the arbitrary moment in the dataset were it was no longer labelled as “palm” and it became “fist”. In a similar way to why the type-2 FRBS seemed to do better than the type-1 FRBS, by using type-2 membership functions to handle the changes in distance between fingertip and centre of palm.

The biggest value of the proposed system is not on its accuracy, it is a combination of a good accuracy and the interpretability of the inference engine that allows the human user to augment the rule base system. This way, the proposed system is highly flexible and can be adapted to predict any needed gesture if it can be described using the distance of the fingertips to the centre of the palm.

The disadvantage of the proposed approach is that it relies on the hand tracking information of the headset, so if the accuracy of the hand tracking is affected the prediction of the hand gesture might also get affected. From another perspective, this can also be an advantage because the FRBS utilizes the available information, and it is not using computing power to calculate the same information that the hand tracking system already provides.

Some other deep learning systems like a YOLOv3 network achieved a 97.68% accuracy [108], which is slightly worse than the KNN but still better than the type-2 FRBS. However, these deep learning solutions have additional disadvantages like the need for large volumes of training data, which tend to consume considerable computing power.

7.3 Proposed Fuzzy Rule-based System for Task Guidance using Mixed Reality Headsets

Digital twins are not meant to be just a centralised database of the physical object. The real benefits of a digital twin can be seen when the user interacts with and uses it to understand how to interact with the physical twin. Using mixed reality headsets users can interact with known physical objects and their digital twin to get valuable insight. The user will be capable of interacting with the digital model and getting the same feedback as if it was the physical object. Additionally, the user can interact with the physical object and get feedback through the digital model, e.g., the correct target area for the task is highlighted in the digital model.

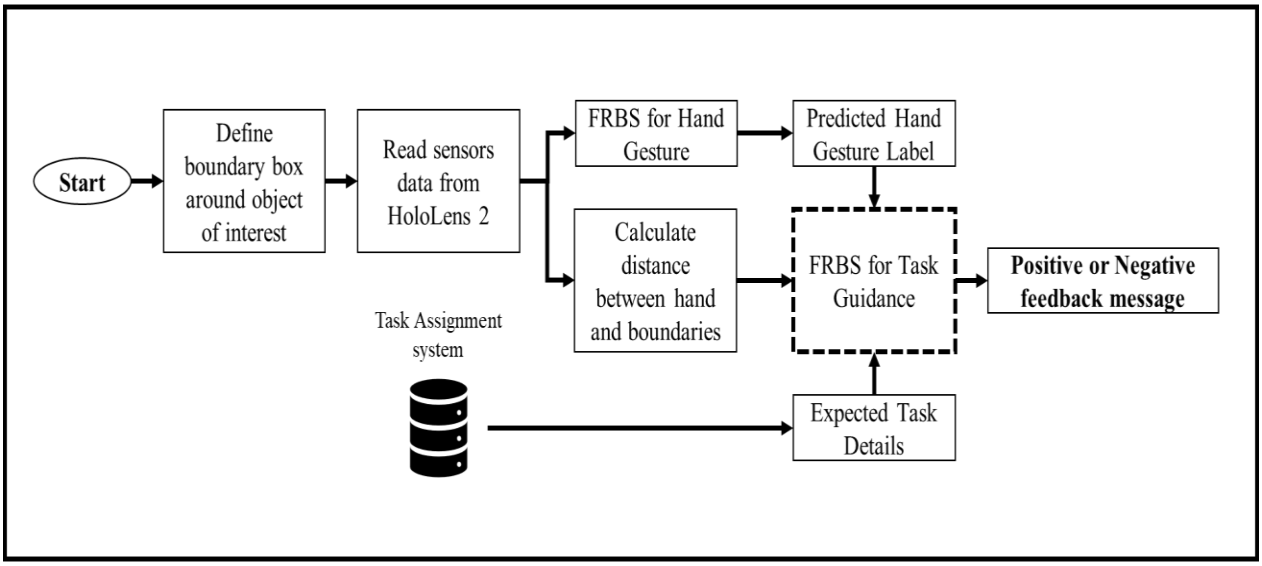


Figure 7.7. Flow chart to visualise to show the sources of information, the different processing steps and how all converges to the final FRBS (represented by the dashed line box) to output a positive or negative feedback that can guide the user.

Fig. 7.7 shows a detailed overview of what happens in each processing step from the start of the task until a feedback message is sent to the user. The previous section described how the FRBS hand gesture recognition system works, the output from that FRBS is used as an input for the second FRBS as it is shown in Fig. 7.7. This section will be focused on describing how the FRBS makes the final decision on whether the feedback should be positive or negative works.

7.3.1 Input Features and Rules

The antecedents of the rule are defined by the inputs the system uses. In section 7.1 of this chapter the overall architecture and inputs are described. The important new input feature is the distance to the target area.

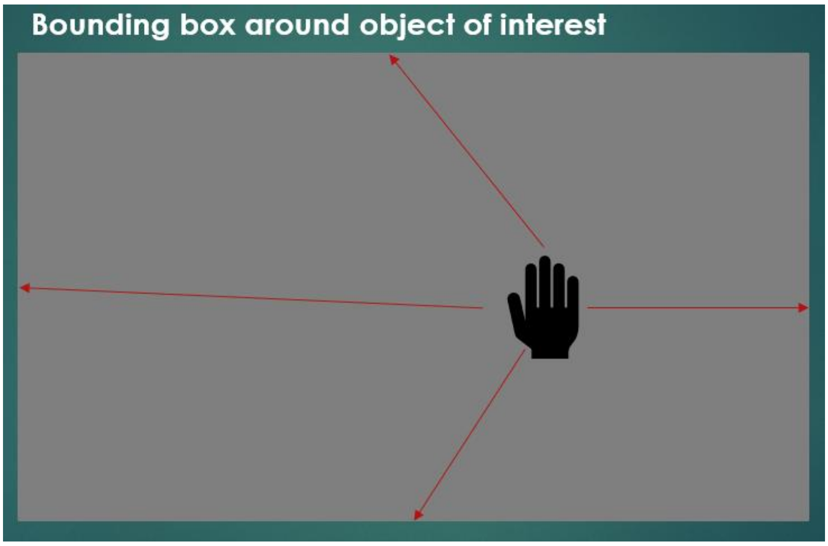


Figure 7.8. Describing the calculation of the distance to target input feature. The grey rectangle represents the bounding box of the object of interest, the red arrows represent the distance that need to be calculated. There are four distances.

The “distance to target area” input feature it is actually represented by 4 antecedents because the following 4 distance values are used: 1) distance from the hand to the top part of the bounding box of the object of interest, 2) distance from the hand to the left side of the bounding box, 3) distance of hand to the lower part of the bounding box, and 4) distance from hand to the right part of the bounding box. These 4 distances are represented in Fig. 7.6 by the 4 red arrows drawn in the four directions. With these 4 inputs it is possible to build the following type of rules.

Example rule 1)

- **IF** hand is close to bottom boundary
    - **AND** hand is close to right boundary
    - **AND** hand is far from left boundary
    - **AND** hand is far from top boundary
- **THEN** hand is close to the right corner of the equipment.

Using that kind of rules, the FRBS can understand the position of the hand within the boundaries of the object of interest. There is no available data for this kind of application, so the experiments for this section 7.3 assumes that the rules are defined by an expert field engineer that knows where the hands must be placed at any point during different tasks. It is important to clarify that for the presented experiments the fingertip from the index finger is used as the global position of the hand. A different point can be used as the location and the FRBS will still work, however, the index fingertip was preferred for the experiments because when reaching out to the object this point is one of the first ones to reach the object of interest, this helps calculate the distances more accurately.

To those 4 antecedents the other three labels are added as antecedents to have a final version of the rule as follows:

Example rule (2)

- **IF** hand is medium distance to bottom boundary,
    - **AND** hand is medium distance to right boundary,
    - **AND** hand is medium distance from left boundary,
    - **AND** hand is medium distance from top boundary,
    - **AND** equipment is a CSP box,
    - **AND** task is placing the entry cable,
    - **AND** hand action is grab/fist,
- **THEN** Error, hand is in the centre of the boundaries, and it is expected to be in the middle-bottom part.

7.3.2 Use Case - Customer Splicing Point Box Installation

To test this FRBS a proof of concept is developed for the use case of the customer splicing point (CSP) box equipment installation, a use case for British Telecom company. The CSP box is a piece of equipment installed outside a premise that is going to be supplied with fibre cable. The fibre from inside the premise goes through the wall and then inside the box through one of the entry ports, the outside fibre cable also goes inside the CSP box. Both parts of fibre cable are connected and inside the CSP box, this way the CSP box acts as the entry point from the outside network to the inside of the premise.

After mounting the CSP box to the wall outside the premise, the first task that needs to be done is to remove the rubber grommet from the second (from left to right) entry port, pass the fibre cable through it carefully, and then put the rubber grommet with the fibre cable back in the entry port. Some field engineers try to first grab the wheel in the centre of the CSP and place the fibre cable around it before passing it through the rubber grommet and entry port.

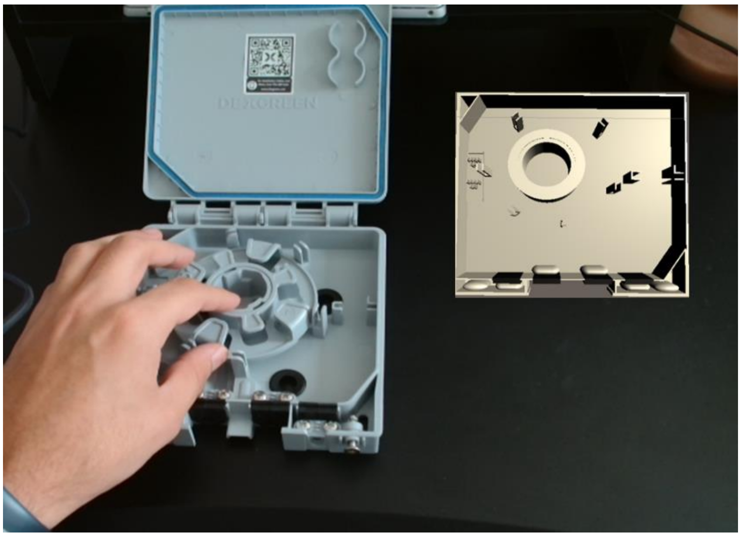


Figure 7.7. A proof of concept of the CSP task guidance process using a FRBS. The user is wearing a headset, and this is his view through the headset.

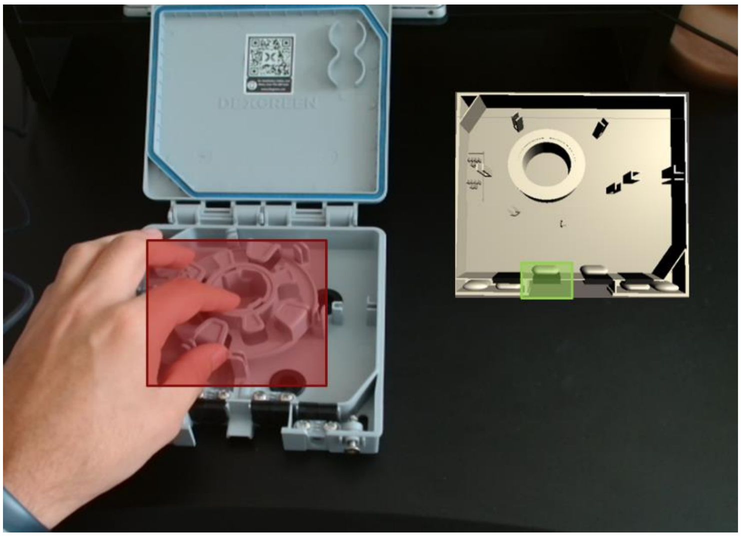


Figure 7.8. Visual example of what happens when one of the rules is triggered.

Fig. 7.7 shows the view of the user through the Microsoft HoloLens 2 just before any of the rules are triggered. Fig. 7.8 shows what happens when one of the rules is triggered. A detailed step by step description is provided below:

1) The task of the user is to pass the fibre cable through the entry port cable.
2) Through the Microsoft HoloLens 2 view the user can see a digital model of the CSP, object on the right side of Fig. 7.7 and Fig. 7.8. The digital model used for this proof of concept is not a detailed, it can be something to improve in future work.
3) Fig. 7.7 shows the moment just before example Rule 2) in previous section 7.3.1 is triggered and the warning boxes in Fig. 7.8 are shown.
4) The user is about to commit the common mistake between engineers, which is to skip step 1 and go straight to step 2 which is to place the fibre cable around the circle tray.
5) The example Rule 2) from previous section 7.3.1 is triggered. Two alert bounding boxes are placed in the user's view.
6) The red one on top of the fingertips is to indicate to stop the movement of the hand.
7) The green bounding box on top of the digital model indicates the target area where the engineer is expected to be working or performing an action.

This is an example proof of concept on how an explainable FRBS can be used to combine different sources of information to provide some guidance or advice to a field engineer during a task. The reason why not to include the antecedents of the hand gesture recognition in the rules of this second FRBS is to keep the model explainable by having fewer antecedents per rule, just one hand gesture antecedent instead of the 5 antecedents from the other FRBS.

7.4 Discussion

This chapter presented the use of two FRBS for combining the information from different sources to understand what the user is doing and what kind of feedback to provide. The idea is that field engineers can access the information stored in the network digital twin, e.g., location of an asset, the status of the asset, or area of interest according to task. In field service operations the preferred device for displaying virtual information to the user is mixed reality headsets because they allow the user to have their hands free to do any other action. Therefore, the focus of this chapter is to propose an explainable AI approach for handling the information coming from the sensors of a mixed reality headset when the user is performing an on-site task. The information needed to be processed is the user's view and the expected task, which means that the two sources of information are the headset's camera and the company's database containing the information of the assigned task. However, as pointed out in section 7.2 of this chapter, the raw frames of the camera are not used, instead these frames are processed by different systems and the FRBS then uses the output of those systems. The idea of this chapter is not to substitute all the image analysis with an explainable FRBS, instead the goal is to present how the business logic of the field engineer's task can be captured by a FRBS and display a message based on the scene from the user's view.

The first FRBS presented in this chapter is used for classifying the current hand gesture of the user using the distance between the fingertip and the centre of the palm. As mentioned before, the idea of this chapter is to use existing systems to process the user's view and then use a FRBS on top of those results. In this case, headsets such as the Microsoft HoloLens 2 and the Oculus Quest 2, already have a hand tracking system, the proposed FRBS and the KNN model used the information from these systems to compute the distance and then decide on the gesture. The results of the experiments showed that the KNN achieved a 98.9% accuracy (highest performance), the type-1 FRBS achieved 94.6% accuracy, and the interval type-2 FRBS achieved 96.4% accuracy. Although the KNN achieved the highest performance, the advantage of the explainable AI approach is that the inference engine can be easily modified by a user to accept new gestures or remove existing gestures. Therefore, it becomes a highly flexible system that can be adapted to different companies.

The second FRBS presented in this chapter is used for combining the different sources of information to decide if the user's current action is expected and if any feedback is needed. The main reason for two separate FRBS was:

- Maintain the interpretability of the system high. The hand gesture recognition uses 5 antecedents, this second FRBS uses 7 antecedents (including 1 for the output label of the first FRBS), if both rules are combined then there will be a total of 11 antecedents, and it will be more difficult for the user to read what is happening and to modify it, rules should be kept simple for interpretability purposes [26].

This shows how an ensemble of two FRBS can be used to keep the interpretability level of both high and achieve the same performance.

Additionally, the main goal of the second FRBS is to capture the business logic and provide feedback to the user based on their current view. This business logic can change and using an explainable AI model to combine the different sources of information has the advantage that it can be easily modified. There was no real-world data available for the second FRBS since there is not a known system that does a similar thing. Therefore, the system could not be compared and evaluated using a performance metric such as accuracy, but it was tested with a use case described in section 7.3.2. Fig. 7.7 shows how a digital model of the object of interest is displayed next to the physical object and Fig. 7.8 shows how feedback is provided through a bounding box on the fingertips and one on the digital model.

Chapter 8. Conclusions and Future Work

8.1 Conclusions

In this thesis, different fuzzy rule-based systems were proposed as an explainable AI model alternative for different applications related to the automated creation of 3D objects in virtual environments. The idea was to understand if these explainable AI approaches can compete against the highly complex black box solutions.

The following points summarise some of the advantages of choosing an explainable AI solution instead of a black box model.

- An explainable AI model has an inference process that can be understood and modified by a human user. Therefore, the main advantage is the flexibility of the model to change according to what the human expert is saying. It is now possible to capture the expertise of engineers with years of knowledge and put it in an AI model that combines the expertise with the available data.
- Additionally, explainable AI models tend to require lower volumes of data than deep learning solutions, mainly because the gaps in the data can be covered by the human expert knowledge.
- Explainable AI models provide accountability and a reason why a decision is being taken, which helps the user understand how the model works, when it is better to use the model, and when it is better to use human knowledge.

On the other hand, the black box models have the following advantages:

- Black box models are robust and complex models capable of handling high dimensional data and extracting complex features that help the model always have a high performance (or a low error).

On one side there is a black box model with the highest possible performance and on the other side there is a transparent model that a user understands, and the user can identify in which cases human intervention is needed. Which option will depend entirely on the needs of the company looking to integrate one of these models into their processes. If they want no human intervention a black box model is a better alternative, they want to use an explainable AI model as a tool to help users, then a model they can understand is likely a better option.

Chapters 5 and 6 presented an FRBS solution to process image data and generate 3D representations of the extracted information. The models were found to have worse performance than the black box model, 3% worse in accuracy for satellite image processing and 2.5% worse for floor plan processing. However, both have considerable good results and are a viable option as an alternative model.

Chapter 7 presented two FRBS, one used for hand gesture recognition and the second one used for combining different input sources to provide feedback to a user in a mixed reality environment. Both models showed the flexibility that an explainable AI model can have and how this flexibility can be used to build fuzzy logic systems that go on top of other systems to combine the different sources of information in an explainable way. For example, the hand gesture recognition interval type-2 FRBS using the hand tracking system of the mixed reality headsets achieved an accuracy of 96.4% accuracy, 1.5% accuracy lower than the opaque model. However, the FRBS rules can be easily modified by the engineer while other opaque models using the hand tracking system or deep learning hand gesture recognition models cannot.

8.2 Future Work

This section discusses the potential research routes that can be explore in the future to address the limitations that were encountered during this research work.

In relation to the segmentation of 2D images there are a couple of research routes that can be explored to expand the work presented in this thesis. First, is the implementation of a FRBS that uses context information for the segmentation of 2D coloured images. Chapter 5 presented the segmentation of coloured images using FRBS, and chapter 6 described how context information can be added to the FRBS when same colour information can mean two different things. However, images from chapter 6 were black and white, and just to complete the work, the same FRBS can be applied to coloured images and evaluate the performance of using visual word dictionary on this type of input data. Additionally, in both datasets (satellite and floor plan images) the resulting mask had a noise that was removed in a post-processing step. However, this noise removal step is not consistent across use cases, i.e., it needed to be adapted depending on the images and performance of the FRBS. A new research route that can be explored is the use of a second FRBS that handles the noise removal process, expanding on the existing solution and removing some of the limitations by using a hierarchical approach of two FRBS.

Furthermore, chapter 7 presented the use of a FRBS for guiding users during their assigned task in mixed reality applications, the FRBS used the sensors input data to identify hand actions at a given point in time and determine if it is an expected action or not. The limitation of this approach is that it only works with static hand actions or gestures, i.e., it doesn't consider the information across time. This is a possible future research work that can expand on the presented FRBS by using the information on how the input data changes over time, this way it will be possible to identify dynamic hand gestures instead of static.

Finally, the implementation and evaluation of the FRBS can also be further explored. In this thesis it was presented an approach for rule inference using the class dominance and crisp numeric values as inputs. A future research work that was not explored in this thesis, because it was out of scope, is to use intervals as inputs and different rule inference methods (e.g., by using descent method) and compare the performance against this work. Additionally, in this thesis the FRBS were used as an explainable AI approach, a future research work could be to evaluate the accountability, bias, and trustworthiness of these explainable AI solutions against their black box model counterpart.

These future work suggestions might inspire further exploration and improvement of fuzzy rule-based systems as an explainable AI solution.

References

[1] M. Grieves, ‘Digital Twin : Manufacturing Excellence through Virtual Factory Replication - A Whitepaper by Dr . Michael Grieves’, *White Paper*, no. March, pp. 1–7, 2014.

[2] M. Grieves and J. Vickers, ‘Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems’, in *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*, F.-J. Kahlen, S. Flumerfelt, and A. Alves, Eds., Cham: Springer International Publishing, 2017, pp. 85–113. doi: 10.1007/978-3-319-38756-7_4.

[3] Centre for Digital Built Britain, ‘National Digital Twin Programme’, 2021. https://www.cdbb.cam.ac.uk/what-we-do/national-digital-twin-programme (accessed Dec. 06, 2021).

[4] A. Bolton *et al.*, ‘Gemini Principles’, Apollo - University of Cambridge Repository, 2018. doi: 10.17863/CAM.32260.

[5] L. A. Zadeh, ‘Fuzzy sets’, *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965, doi: https://doi.org/10.1016/S0019-9958(65)90241-X.

[6] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems Introduction and New Directions*. Prentice-Hall, Inc., 2001.

[7] L. A. Zadeh, ‘Outline of a New Approach to the Analysis of Complex Systems and Decision Processes’, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 1, pp. 28–44, 1973, doi: 10.1109/TSMC.1973.5408575.

[8] H. A. Hagras, ‘A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots’, *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 524–539, 2004, doi: 10.1109/TFUZZ.2004.832538.

[9] H. Leon-Garza, H. Hagras, A. Peña-Rios, A. Conway, and G. Owusu, ‘A type-2 fuzzy system-based approach for image data fusion to create building information models’,

Information Fusion, vol. 88, pp. 115–125, 2022, doi: https://doi.org/10.1016/j.inffus.2022.07.007.

[10] D. Bernardo, H. Hagras, and E. Tsang, ‘A genetic type-2 fuzzy logic based system for the generation of summarised linguistic predictive models for financial applications’, *Soft Computing*, vol. 17, no. 12, pp. 2185–2201, 2013, doi: 10.1007/s00500-013-1102-y.

[11] 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, doi: 10.1109/FUZZ-IEEE.2013.6622310.

[12] A. Das and P. Rad, ‘Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey’. arXiv, Jun. 22, 2020. Accessed: Jan. 04, 2023. [Online]. Available: http://arxiv.org/abs/2006.11371

[13] A. Preece, D. Harborne, D. Braines, R. Tomsett, and S. Chakraborty, ‘Stakeholders in Explainable AI’. arXiv, Sep. 29, 2018. Accessed: Jan. 04, 2023. [Online]. Available: http://arxiv.org/abs/1810.00184

[14] C. Wierzynski, ‘The Challenges and Opportunities of Explainable AI’, 2018. https://www.intel.ai/the-challenges-and-opportunities-of-explainable-ai/

[15] A. B. Arrieta *et al.*, ‘Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI’, 2019, doi: 10.1016/j.inffus.2019.12.012.

[16] E. Tjoa and C. Guan, ‘A Survey on Explainable Artificial Intelligence (XAI): Towards Medical XAI’, *IEEE Trans. Neural Netw. Learning Syst.*, vol. 32, no. 11, pp. 4793–4813, Nov. 2021, doi: 10.1109/TNNLS.2020.3027314.

[17] N. Xie, G. Ras, M. van Gerven, and D. Doran, ‘Explainable Deep Learning: A Field Guide for the Uninitiated’, 2020, [Online]. Available: http://arxiv.org/abs/2004.14545

[18] A. Adadi and M. Berrada, ‘Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)’, *IEEE Access,* vol. 6, 2018, doi: 10.1109/ACCESS.2018.2870052.

[19] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, ‘Explaining explanations: An overview of interpretability of machine learning’, *Proceedings - 2018 IEEE 5th International Conference on Data Science and Advanced Analytics, DSAA 2018,* pp. 80–89, 2019, doi: 10.1109/DSAA.2018.00018.

[20] ‘AI in the UK: ready, willing and able’.

[21] D. Gunning and D. Aha, ‘DARPA’s explainable artificial intelligence (XAI) program’, *AI magazine,* vol. 40, no. 2, pp. 44–58, 2019.

[22] C. Mencar and J. M. Alonso, ‘Paving the way to explainable artificial intelligence with fuzzy modeling’, in *International Workshop on Fuzzy Logic and Applications*, Springer, 2018, pp. 215–227.

[23] M. T. Ribeiro, S. Singh, and C. Guestrin, ‘“Why Should I Trust You?”: Explaining the Predictions of Any Classifier’. arXiv, Aug. 09, 2016. Accessed: Jan. 04, 2023. [Online]. Available: http://arxiv.org/abs/1602.04938

[24] S. Lundberg and S.-I. Lee, ‘A Unified Approach to Interpreting Model Predictions’, Dec. 2017.

[25] P. Dabkowski and Y. Gal, ‘Real Time Image Saliency for Black Box Classifiers’. arXiv, May 22, 2017. Accessed: Jan. 04, 2023. [Online]. Available: http://arxiv.org/abs/1705.07857

[26] H. Hagras, ‘Toward Human-Understandable, Explainable AI’, *Computer,* vol. 51, pp. 28–36, 2018, doi: 10.1109/MC.2018.3620965.

[27] D. Gunning, ‘Explainable artificial intelligence (xai)’, *Defense advanced research projects agency (DARPA), nd Web,* vol. 2, no. 2, p. 1, 2017.

[28] A. Fernandez, F. Herrera, O. Cordon, M. Jose Del Jesus, and F. Marcelloni, ‘Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to?’, *IEEE Computational Intelligence Magazine*, vol. 14, no. 1, pp. 69–81, 2019, doi: 10.1109/MCI.2018.2881645.

[29] R. Chimatapu, G. S. Member, H. Hagras, and A. Starkey, ‘Deep Type-2 Fuzzy Logic based System for Explainable Artificial Intelligence’, pp. 1–10.

[30] L. Veryard, H. Hagras, A. Conway, and G. Owusu, ‘A Type-2 Fuzzy Genetic Approach to Uncertain & Dynamic Resilient Routing within Telecommunications Networks’, in *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Glasgow, United Kingdom: IEEE, Jul. 2020, pp. 1–6. doi: 10.1109/FUZZ48607.2020.9177786.

[31] L. Veryard, H. Hagras, A. Starkey, and G. Owusu, ‘A Fuzzy Genetic System for Resilient Routing in Uncertain & Dynamic Telecommunication Networks’, in *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, New Orleans, LA, USA: IEEE, Jun. 2019, pp. 1–6. doi: 10.1109/FUZZ-IEEE.2019.8858788.

[32] L. Beasley, H. Hagras, A. Conway, and G. Owusu, ‘A Type-2 Fuzzy Based Multi-Objective Optimisation for Strategic Network Planning in the Telecommunication Domain’, in *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Luxembourg, Luxembourg: IEEE, Jul. 2021, pp. 1–7. doi: 10.1109/FUZZ45933.2021.9494555.

[33] E. Anas, *Edge Detection Techniques Using Fuzzy Logic*.

[34] B. Yao, H. Hagras, M. J. Alhaddad, and D. Alghazzawi, ‘A fuzzy logic-based system for the automation of human behavior recognition using machine vision in intelligent environments’, *Soft Computing*, vol. 19, no. 2, pp. 499–506, 2014, doi: 10.1007/s00500-014-1270-4.

[35] 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, doi: 10.1016/j.advengsoft.2005.04.005.

[36] H. Tang, J. Zhou, S. Xue, and L. Xie, 'Big Bang-Big Crunch optimization for parameter estimation in structural systems', *Mechanical Systems and Signal Processing*, vol. 24, no. 8, pp. 2888–2897, 2010, doi: 10.1016/j.ymssp.2010.03.012.

[37] T. Kumbasar, I. Eksin, M. Guzelkaya, and E. Yesil, 'Adaptive fuzzy model based inverse controller design using BB-BC optimization algorithm', *Expert Systems with Applications*, vol. 38, no. 10, pp. 12356–12364, Sep. 2011, doi: 10.1016/j.eswa.2011.04.015.

[38] T. Kumbasar and H. Hagras, 'A big bang-big crunch optimization based approach for interval type-2 fuzzy PID controller design', *IEEE International Conference on Fuzzy Systems*, 2013, doi: 10.1109/FUZZ-IEEE.2013.6622301.

[39] 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, doi: 10.1109/SMC.2013.491.

[40] R. Chimatapu, H. Hagras, A. Starkey, and G. Owusu, 'A big-bang big-crunch type-2 fuzzy logic system for generating interpretable models in workforce optimization', *IEEE International Conference on Fuzzy Systems*, vol. 2018-July, 2018, doi: 10.1109/FUZZ-IEEE.2018.8491662.

[41] J. Sanz, A. Fernández, H. Bustince, and F. Herrera, 'A genetic tuning to improve the performance of Fuzzy Rule-Based Classification Systems with Interval-Valued Fuzzy Sets: Degree of ignorance and lateral position', *International Journal of Approximate Reasoning*, vol. 52, no. 6, pp. 751–766, Sep. 2011, doi: 10.1016/j.ijar.2011.01.011.

[42] H. Leon-Garza, H. Hagras, A. Peña-rios, A. Conway, and G. Owusu, 'An Interval Type-2 Fuzzy-based System to Create Building Information Management Models from 2D Floor Plan Images', in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE),* 2021.

[43] H. Leon-Garza, H. Hagras, A. Peña-Rios, G. Owusu, and A. Conway, 'Type-1 Fuzzy Rule-based System using Patch-based Approach for Semantic Segmentation in Floor Plans', in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE),* 2021.

[44] H. Leon-Garza, H. Hagras, A. Peña-Rios, A. Conway, and G. Owusu, 'A Big Bang-Big Crunch Type-2 Fuzzy Logic System for Explainable Semantic Segmentation of Trees in Satellite Images using HSV Color Space', *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2020.

[45] Jin Zhao and B. K. Bose, 'Evaluation of membership functions for fuzzy logic controlled induction motor drive', in *IEEE 2002 28th Annual Conference of the Industrial Electronics Society. IECON 02*, Sevilla, Spain: IEEE, 2002, pp. 229–234. doi: 10.1109/IECON.2002.1187512.

[46] J. A. Sanz, D. Bernardo, F. Herrera, H. Bustince, and H. Hagras, 'A Compact Evolutionary Interval-Valued Fuzzy Rule-Based Classification System for the Modeling and Prediction of Real-World Financial Applications With Imbalanced Data', *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 4, pp. 973–990, Aug. 2015, doi: 10.1109/TFUZZ.2014.2336263.

[47] 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, doi: 10.1109/TFUZZ.2016.2578341.

[48] J. Vatn, 'Industry 4 . 0 and real-time synchronization of operation and maintenance', 2018.

[49] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, ‘Digital Twin in manufacturing: A categorical literature review and classification’, *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018, doi: 10.1016/j.ifacol.2018.08.474.

[50] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, ‘Characterising the Digital Twin: A systematic literature review’, *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36–52, 2020, doi: 10.1016/j.cirpj.2020.02.002.

[51] P. Milgram and F. Kishimo, ‘A taxonomy of mixed reality’, *IEICE Transactions on Information and Systems*, vol. 77, no. 12, pp. 1321–1329, 1994.

[52] R. T. Azuma, ‘A Survey of Augmented Reality’, 1997. [Online]. Available: http://www.cs.unc.edu/~azumaW:

[53] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, ‘Recent advances in augmented reality’, *IEEE Computer Graphics and Applications*, vol. 21, no. 6, pp. 34–47, Nov. 2001, doi: 10.1109/38.963459.

[54] D. W. F. Van Krevelen and R. Poelman, ‘A Survey of Augmented Reality Technologies, Applications and Limitations’, 2010. [Online]. Available: http://www.arvika.de/

[55] Feng Zhou, H. B.-L. Duh, and M. Billinghurst, ‘Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR’, in *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, Cambridge, UK: IEEE, Sep. 2008, pp. 193–202. doi: 10.1109/ISMAR.2008.4637362.

[56] E. Burke, P. Felle, C. Crowley, J. Jones, E. Mangina, and A. G. Campbell, ‘Augmented reality EVAR training in mixed reality educational space’, in *IEEE Global Engineering Education Conference, EDUCON*, 2017. doi: 10.1109/EDUCON.2017.7943058.

[57] T. W. Day, ‘Wheelchair-MR: A mixed reality wheelchair training environment’, in *Proceedings - 2017 International Conference on Cyberworlds, CW 2017 - in cooperation*

with: Eurographics Association International Federation for Information Processing ACM SIGGRAPH, 2017. doi: 10.1109/CW.2017.12.

[58] F. Ke, S. Lee, and X. Xu, ‘Teaching training in a mixed-reality integrated learning environment’, *Computers in Human Behavior*, 2016, doi: 10.1016/j.chb.2016.03.094.

[59] L. Buth, M. Juraschek, G. Posselt, and C. Herrmann, ‘Supporting SMEs towards adopting mixed reality : A training concept to bring the reality-virtuality continuum into application’, in *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, 2018. doi: 10.1109/INDIN.2018.8471990.

[60] G. Riexinger, A. Kluth, M. Olbrich, J. D. Braun, and T. Bauernhansl, ‘Mixed Reality for On-Site Self-Instruction and Self-Inspection with Building Information Models’, in *Procedia CIRP*, 2018. doi: 10.1016/j.procir.2018.03.160.

[61] X. Wang and P. E. D. Love, ‘BIM + AR: Onsite information sharing and communication via advanced visualization’, in *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2012*, 2012. doi: 10.1109/CSCWD.2012.6221920.

[62] P. Dahne and J. N. Karigiannis, ‘Archeoguide: System architecture of a mobile outdoor augmented reality system’, in *Proceedings - International Symposium on Mixed and Augmented Reality, ISMAR*, 2002, pp. 263–264. doi: 10.1109/ISMAR.2002.1115103.

[63] H. Leon-Garza, H. Hagras, A. Peña-Rios, G. Owusu, and A. Conway, ‘A Fuzzy Logic Based System for Cloud-based Building Information Modelling Rendering Optimization in Augmented Reality’, in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2020.

[64] D. Gerber, B. Nguyen, and I. Gaetani, ‘Digital twin: towards a meaningful framework’, p. 160, 2019.

[65] National Research Foundation (NRF), 'Virtual Singapore', 2018. National Research Foundation (NRF)

[66] E. Esekhaigbe, E. Kazan, and M. Usmen, 'Integration of Digital Technologies into Underground Utility Asset Management', *Open Journal of Civil Engineering*, vol. 10, no. 04, pp. 403–428, 2020, doi: 10.4236/ojce.2020.104030.

[67] K. Hribernik, G. Cabri, F. Mandreoli, and G. Mentzas, 'Autonomous, context-aware, adaptive Digital Twins—State of the art and roadmap', *Computers in Industry*, vol. 133, p. 103508, 2021, doi: 10.1016/j.compind.2021.103508.

[68] A. Peña-Rios, C. Di Cairano-Gilfedder, L. Krug, T. Glover, A. Conway, and G. Owusu, 'Uncovering the Power of Digital Twins for Telcos', vol. 16, pp. 38–44, Apr. 2022.

[69] S. Kahn *et al.*, 'Beyond 3D "as-built" information using mobile AR enhancing the building lifecycle management', in *Proceedings of the 2012 International Conference on Cyberworlds, Cyberworlds 2012*, 2012. doi: 10.1109/CW.2012.12.

[70] G. Schall *et al.*, 'Handheld Augmented Reality for underground infrastructure visualization', *Personal and Ubiquitous Computing*, vol. 13, no. 4, pp. 281–291, 2009, doi: 10.1007/s00779-008-0204-5.

[71] 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, doi: 10.1117/1.jrs.10.010501.

[72] 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'.

[73] M. Volpi and V. Ferrari, 'Semantic segmentation of urban scenes by learning local class interactions'.

[74] 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, doi: 10.3390/ijgi7030110.

[75] 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, doi: 10.1109/TGRS.2017.2740362.

[76] 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, doi: 10.1109/IGARSS.2017.8127684.

[77] M. M. Alam, Z. Zhu, B. Eren Tokgoz, J. Zhang, and S. Hwang, ‘Automatic Assessment and Prediction of the Resilience of Utility Poles Using Unmanned Aerial Vehicles and Computer Vision Techniques’, *Int J Disaster Risk Sci*, vol. 11, no. 1, pp. 119–132, Feb. 2020, doi: 10.1007/s13753-020-00254-1.

[78] O. Jumbo and R. Moghaddass, ‘Resource optimization and image processing for vegetation management programs in power distribution networks’, *Applied Energy*, vol. 319, p. 119234, Aug. 2022, doi: 10.1016/j.apenergy.2022.119234.

[79] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, ‘What’s the Point: Semantic Segmentation with Point Supervision’, Jun. 2015, [Online]. Available: http://arxiv.org/abs/1506.02106

[80] P. Tokarczyk, J. D. Wegner, S. Walk, and K. Schindler, ‘Features, color spaces, and boosting: New insights on semantic classification of remote sensing images’, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 1, pp. 280–295, 2015, doi: 10.1109/TGRS.2014.2321423.

[81] 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, doi: 10.1007/978-3-319-10590-1_53.

[82] C. Amza, ‘A Review on Neural Network-Based Image Segmentation Techniques’.

[83] C.-Y. Wen and C.-M. Chou, ‘Color Image Models and its Applications to Document Examination’, 2002.

[84] 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’, *International Journal of Emerging Technology and Advanced Engineering*, vol. 5, no. 2, 2015, [Online]. Available: www.ijetae.com

[85] S. Ohleyer, ‘Building segmentation on satellite images’. 2018.

[86] Associated General Contractors of America, ‘The Contractor’s Guide to BIM’.

[87] Cooperative Research Centre for Construction Innovation, ‘Adopting BIM for facilities management’, 2007.

[88] Y. Y. Al-Ashmori *et al.*, ‘BIM benefits and its influence on the BIM implementation in Malaysia’, *Ain Shams Engineering Journal*, vol. 11, pp. 1013–1019, 2020, doi: 10.1016/j.asej.2020.02.002.

[89] C. Boje, A. Guerriero, S. Kubicki, and Y. Rezgui, ‘Towards a semantic Construction Digital Twin: Directions for future research’, *Automation in Construction*, vol. 114, no. January, p. 103179, 2020, doi: 10.1016/j.autcon.2020.103179.

[90] Y. Arayici, ‘Towards building information modelling for existing structures’, *Structural Survey*, vol. 26, no. 3, pp. 210–222, 2008, doi: 10.1108/02630800810887108.

[91] A. Dakhil, M. Alshawi, and J. Underwood, ‘BIM Client Maturity: Literature Review’.

[92] L. Gimenez, J. L. Hippolyte, S. Robert, F. Suard, and K. Zreik, 'Review: Reconstruction of 3D building information models from 2D scanned plans', *Journal of Building Engineering*, vol. 2, pp. 24–35, 2015, doi: 10.1016/j.jobe.2015.04.002.

[93] D. Larlus, J. Verbeek, and F. Jurie, 'Category level object segmentation by combining bag-of-words models with dirichlet processes and random fields', *International Journal of Computer Vision*, vol. 88, no. 2, pp. 238–253, 2010, doi: 10.1007/s11263-009-0245-x.

[94] T. J. Burns and J. J. Corso, 'Robust unsupervised segmentation of degraded document images with topic models', *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, vol. 2009 IEEE, pp. 1287–1294, 2009, doi: 10.1109/CVPRW.2009.5206606.

[95] L. P. D. L. Heras, J. Mas, G. Sánchez, and E. Valveny, 'Wall patch-based segmentation in architectural floorplans', *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, no. September, pp. 1270–1274, 2011, doi: 10.1109/ICDAR.2011.256.

[96] L. P. De Las Heras, J. Mas, G. Sánchez, and E. Valveny, 'Notation-invariant patch-based wall detector in architectural floor plans', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7423 LNCS, pp. 79–88, 2013, doi: 10.1007/978-3-642-36824-0_8.

[97] S. Dodge, J. Xu, and B. Stenger, 'Parsing floor plan images', *Proceedings of the 15th IAPR International Conference on Machine Vision Applications, MVA 2017*, no. December, pp. 358–361, 2017, doi: 10.23919/MVA.2017.7986875.

[98] J. H. Yang, H. Jang, J. Kim, and J. Kim, 'Semantic Segmentation in Architectural Floor Plans for Detecting Walls and Doors', *Proceedings - 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2018*, 2019, doi: 10.1109/CISP-BMEI.2018.8633243.

[99] O. Ronneberger, P. Fischer, and T. Brox, ‘U-net: Convolutional networks for biomedical image segmentation’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9351, pp. 234–241, 2015, doi: 10.1007/978-3-319-24574-4_28.

[100] A. A. Pravitasari *et al.*, ‘UNet-VGG16 with transfer learning for MRI-based brain tumor segmentation’, *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 3, pp. 1310–1318, 2020, doi: 10.12928/TELKOMNIKA.v18i3.14753.

[101] T. Krijnen, ‘IfcOpenShell’, 2017. http://ifcopenshell.org/ (accessed Sep. 20, 2020).

[102] A. Pena-Rios, H. Hagras, G. Owusu, and M. Gardner, ‘Furthering Service 4.0: Harnessing Intelligent Immersive Environments and Systems’, *IEEE Systems, Man, and Cybernetics Magazine*, 2018, doi: 10.1109/MSMC.2017.2769199.

[103] J. Huo, K. L. Keung, C. K. M. Lee, and H. Y. Ng, ‘Hand Gesture Recognition with Augmented Reality and Leap Motion Controller’, in *2021 IEEE Internation Conference on Industrial Engineering and Engineering Management*, 2022, pp. 1015–1019. doi: 10.1109/ieem50564.2021.9672611.

[104] Z. Hu, Y. Hu, J. Liu, B. Wu, D. Han, and T. Kurfess, ‘3D separable convolutional neural network for dynamic hand gesture recognition’, *Neurocomputing*, vol. 318, pp. 151–161, 2018, doi: 10.1016/j.neucom.2018.08.042.

[105] H. Leon-Garza, H. Hagras, A. Pena-Rios, O. Bahceci, and A. Conway, ‘A Hand-Gesture Recognition Based Interpretable Type-2 Fuzzy Rule-based System for Extended Reality’, in *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Prague, Czech Republic: IEEE, Oct. 2022, pp. 2894–2899. doi: 10.1109/SMC53654.2022.9945407.

[106] O. Bahceci, A. Pena-Rios, G. Buckingham, and A. Conway, ‘Supervised Machine Learning Hand Gesture Classification in VR for Immersive Training’, in *IEEE Conference on Virtual Reality & 3D User Interfaces*, doi: 10.1109/TIM.2021.3077967.D.

[107] P. K. Pisharady and M. Saerbeck, ‘Recent methods and databases in vision-based hand gesture recognition: A review’, *Computer Vision and Image Understanding*, vol. 141, pp. 152–165, 2015, doi: 10.1016/j.cviu.2015.08.004.

[108] A. Mujahid *et al.*, ‘Real-Time Hand Gesture Recognition Based on Deep Learning YOLOv3 Model’, *Applied Sciences*, vol. 11, no. 9, 2021, doi: 10.3390/app11094164.