



UNIVERSITÀ
POLITECNICA
DELLE MARCHE

Doctoral School of Civil, Building,
Environmental Engineering

XXXII Cycle
2016-2019

Artificial Intelligence assisted Building Digitization using Mixed Reality

PhD Candidate **Alessandra Corneli**

Academic tutor **Alessandro Carbonari**

ACKNOWLEDGEMENTS

This research has been a teamwork and for this reason I feel I have someone to thank.

First and foremost thanks to Professor Carbonari who firstly believed in me. He taught me that there are not problems but just question to be solved, it could seem banal but this let you see the world from a different point of view. He introduced me to the thrilling profession of the researcher sharing with me his experience, I could never thank him enough.

I want to say "thank you" also to Professor Naticchia. When thinking about his role in our research team there is an Italian term usually refers to football that comes to mind: *fantasista*. There would be also the English translation, playmaker, but in this case is not as powerful as the Italian one. The Italian word *fantasista* hold the word fantasy and fantasy link to imagination without which it would be impossible to do research. He can imagine and design research directions. His vision on future of construction industry is always fresh and fascinating. His guide has been fundamental for the successful of my work. There is another member of the team that played a crucial role in my work. Engineer Massimo Vaccarini gave me the most valuable support on technical developments. He shared with me his knowledge and he did with me many of the technical task to create the system proposed. Thank you Massimo.

Other thanks have to be said to my family. They supported me in every choice and they rejoice with me for my successes. Thanks to my mate who gave me all his help and support in these three years. I am grateful for him.

I want to say that I would start the PhD again and I consider it a unique experience. During these three years I have grown so much that I almost not recognize myself anymore. Using a too exploited quote I would say to three years ago version of myself *you know nothing Alessandra Corneli*.

ABSTRACT

Facility Management in complex buildings requires a large amount of information that can be stored in a functional building model. A functional building model is a structured representation of the building including information crucial for specific functions such as safety, refurbishment actions or operation and maintenance. Surveying this kind of data, such as technical properties of building components, is a costly process. For this reason, an advanced tool for engineering surveys is needed. Nowadays many studies still focus on capturing geometry, overlooking the fact that many recurring actions are conducted on assets inside buildings. Many systems proposed exploit highly accurate survey techniques, like laser scanning or photogrammetry, but they need long post-processing efforts to interpret data collected. Moreover, these operations are not pursued on site leading to inaccuracies for the incorrect interpretation of data. Under these circumstances, the possibility of performing the majority of operation on-site would definitely make the process more efficient and it would reduce errors. This research proposes a system for digitization exploiting man-machine intelligence collaboration without post-processing. To this aim, Mixed Reality with its capability of interacting with real world is applied giving an environment for man-machine collaboration. The capability of Mixed Reality of overlapping digital data to the real environment makes possible checking data directly on site. For the object recognition process the system proposed in this research make use of Neural Network. YOLO (You Only Look Once) Neural Networks has been chosen for its speed and multiple detection features, ideal for real-time applications. The system has been developed and its performance evaluated for the detection of fire protection system components. First single Neural Network have been tested reaching always more than 85%of F1 factor. Then the whole embedded system proposed has been tested on site to prove its feasibility in a real-world scenario.

TABLE OF CONTENTS

Acknowledgements	i
Abstract	ii
Table of contents	ii
List of tables	v
List of figures	vii
Acronyms	xi
1 Introduction	1
1.1 Background and Motivation	1
1.2 Facility Management: an overview	4
1.3 Goal and Overview: surveying of assets components	6
2 Literature Review	9
2.1 Introduction	9
2.2 Surveying	10
2.2.1 The widespread adoption of BIM paradigm in the AEC industry	10
2.2.2 Latest technologies that support building survey procedures	19
2.3 Machine Learning	29
2.3.1 Neural Networks for the recognition of objects	30
2.3.2 The use of NN for recognition in engineering	31
2.4 Mixed Reality	41
2.4.1 Definition of Mixed Reality	41
2.4.2 Uptake of MR to real-time problems	44
2.4.3 The use of Mixed Reality for building oriented applications	47
2.5 Conclusion	56
3 Methodology	57
3.1 Introduction	57
3.2 Use cases	58
3.2.1 Addressed issues	58
3.2.2 Automatic inventory/survey support	59
3.2.3 Diagnosis support	60
3.2.4 On-site operation support	60
3.3 Neural Networks	61
3.4 YOLO Neural Networks	71
3.5 YOLO training process	76
3.5.1 The training framework	77
3.5.2 The Dataset creation	80
3.5.3 Training the network	84

3.5.4	Validation process	88
3.6	Mixed reality	91
3.6.1	Holograms that overlap reality	92
3.6.2	The MR tool	96
3.7	Collected information storing	98
3.8	Conclusion	99
4	Object Recognition System	103
4.1	Introduction	103
4.2	Object Recognition System development	103
4.2.1	Mixed Reality environment	104
4.2.1.1	BIM model	105
4.2.1.2	MR platform	105
4.2.1.3	Hololens application	105
4.2.1.4	Database	108
4.2.2	Real environment	108
4.2.2.1	Microsoft Hololens	109
4.2.2.2	The embedded system	109
4.2.2.3	Raspberry	110
4.2.2.4	Movidius	110
4.2.2.5	On-site technician	113
4.2.2.6	Real world scene	113
4.3	Data transfer	113
4.4	Automatic survey process	115
4.5	Conclusion	119
5	Proof of concept	121
5.1	Introduction	121
5.2	Neural Network training	121
5.2.1	Datasets creation	122
5.2.2	Training sessions and results	124
5.2.3	Training sessions validation	127
5.3	Testing the Neural Network performances	135
5.4	Testing of the system	143
5.5	Discussion	146
5.6	Conclusion	147
6	Conclusions	153
6.1	Conclusions	153
6.2	Research Contributions	154
6.3	Suggestions for future RD	154

LIST OF TABLES

4.1	Performance tests, each one elaborate 150 photos. (a) Total time for sending raw photos (cropping and scaling timens are neglected, max 10 ms. (b) cropped from 869x504 photo, (c) scaled from 896x504.	114
5.1	Dataset composed by original images.	123
5.2	Emergency signs categories original images.	124
5.3	Dataset with both original and re-edited pictures.	124
5.4	Fire extinguisher training datasets from combination of original images dataset.	125
5.5	Training validations.	129
5.6	Training validations.	130
5.7	Training validations.	131
5.8	Training validations.	132
5.9	Neural Network performances test results (Ground and First floors).	138
5.10	Neural Network performances test results (Basement floor). . . .	139

LIST OF FIGURES

1.1	Building life cycle phases [Roper and Payant, 2014].	1
1.2	Building life cycle cost.	2
2.1	Developing of BIM and FM standards	11
2.2	Steps of the automated generation of parametric BIM objects from video and laser scanning data [Brilakis et al., 2010].	13
2.3	Steps of the recognition process through an image-driven fuzzy- system [Lu et al., 2018].	14
2.4	General framework of the DFO approach [Xue et al., 2018].	15
2.5	Classification of advanced building survey techniques.	20
2.6	Semi-automatic light inventory algorithm [Díaz-Vilariño et al., 2015].	22
2.7	Primitive modeling of a cylinder indicating a piece of a pipe [Rodríguez-gonzalvez et al., 2014]	23
2.8	Masonry point cloud processing phases [Valero et al., 2018].	24
2.9	Left column shows the reflectance images while the right col- umn shows the classification results from the surface algorithm [Xiong et al., 2013].	25
2.10	Starting raw data: point clouds [Chiabrando et al., 2016].	26
2.11	Final BIM model [Chiabrando et al., 2016].	26
2.12	Detection results for Simple Scenes. (Left) Original 4D orthoim- ages. (Right) Door detection [Quintana et al., 2018].	28
2.13	(a) Coloured point cloud, (b) depth image, (c) object detection for object recognition [Quintana et al., 2017].	29
2.14	Structure of the neural network with random generated structure: it presents 3 convolutional layers, followed by a dense layers that maps the output of the last convolutional layer to the three classes [Lamio et al., 2019].	32
2.15	Structure of a DBN [Zhao et al., 2015].	33
2.16	Crack detection with neural network [Cosenza et al., 2018]	34
2.17	Detected bounding box for formwork elements on a photography of a construction site [Braun et al., 2019].	35
2.18	Airplane aerial images for object detection with YOLO [Radovic et al., 2017].	36
2.19	Not hardhat use detection method framework [Fang et al., 2018].	38
2.20	YOLO Network structure [Tao et al., 2018].	39
2.21	Proximity monitoring proposed system [Kim et al., 2019].	40
2.22	Virtuality Continuum Spectrum by Milgram and Kishimo.	42
2.23	New propose for virtuality continuum spectrum.	43
2.24	Pure mixed reality explanation [Flavián et al., 2019].	44

2.25	hazard avoidance system [Kim et al., 2017].	45
2.26	BIM3R system architecture [Ammari and Hammad, 2014].	46
2.27	Example of on-site visualization with Mixed Reality [Chalhoub and Ayer, 2018].	48
2.28	Example of as-is model checking [Park et al., 2013].	49
2.29	Example of information displayed on site through Mixed Reality [Irizarry et al., 2014].	51
2.30	INSITER system structure [Riexinger et al., 2018].	53
2.31	Example of work orders displayed [Riexinger et al., 2018].	53
2.32	Training information shown directly on-site [Riexinger et al., 2018].	54
2.33	Information for site monitoring of working equipment [Wang et al., 2017].	55
3.1	Automatic inventory/survey support process.	60
3.2	Diagnosis on-site support process.	60
3.3	On-site critical operation support process.	61
3.4	A diagram showing the different layers in a CNN [Saha, 2018].	63
3.5	LeNet Network structure [Chatterjee, 2016].	64
3.6	AlexNet Network structure [Chatterjee, 2016].	65
3.7	VGGNet 16 Network structure [Das, 2017].	66
3.8	GoogleNet Network structure [Szegedy et al., 2015].	67
3.9	Skip-connection idea applied to ResNets [Chatterjee, 2016].	68
3.10	ResNets Network structure [Chatterjee, 2016].	70
3.11	YOLO Network structure [Redmon et al., 2016].	71
3.12	Performance comparison between Fast R-CNN and YOLO [Redmon et al., 2016].	73
3.13	YOLO v3 speed/accuracy tradeoff [Redmon and Farhadi, 2018].	74
3.14	Darknet-53 [Redmon and Farhadi, 2018].	75
3.15	YOLO Tiny v2 structure [Li et al., 2018].	75
3.16	Neural Network general structure [Moore, 2018].	76
3.17	Visual studio setting for Darkent compilation: x64 bit and Re- lease version.	78
3.18	Visual studio setting for Darkent compilation: including cudnn.lib.	78
3.19	Inserting OpenCV, CUDA and cuDNN in the darknet.exe folder.	79
3.20	Creation of the new Windows variable for cudnn.	79
3.21	Bounding box design with VoTT software.	84
3.22	.data file for network training	85
3.23	.names file containing name tags.	85
3.24	Training command line	85
3.25	Output chart of the training process	87
3.26	Output log file of the training process	88
3.27	Output of the validation process pursued on a desktop and fol- lowing evaluation for Precision and Recall calculation.	90
3.28	Command line for the validation procedure of the customized network.	90
3.29	Real-world validation of the customized network for fire extin- guisher recognition.	91
3.30	Mixed reality experience as the combination of three inputs: com- puter processing, human input, and environmental input [Microsoft, 2018e]	92

3.31	Differences between Virtual, Augmented and Mixed Reality with regard to the connection with real environment [Haeuss, 2017].	93
3.32	Optimal distance range for placing holograms [Microsoft, 2018a].	94
3.33	Microsoft Hololens [Microsoft, 2018c].	96
3.34	Hololens display, sensor and processor [Microsoft, 2018c].	98
3.35	Example of .xml format file for data transfer.	99
3.36	Example of fire extinguisher dataset as a set of pictures.	101
3.37	Example of signs dataset as a set of pictures.	102
4.1	System architecture.	104
4.2	Connection menu.	105
4.3	Excerpt of the script for the connection with the Raspberry inside the Unity development framework.	107
4.4	Script for FE hologram insertion in Visual Studio 2017.	108
4.5	Application menu.	109
4.6	Movidius processor[Intel, 2019a].	112
4.7	Movidius training and use [Intel, 2019b].	112
4.8	Image steps through the recognition process.	115
4.9	Position (the blue square) and rotation (the red circle) points.	117
4.10	Bounding box and object category at the end of the the recognition procedure.	117
4.11	Recognition process steps.	118
5.1	From left to right: emergency sign door, emergency sign man, emergency sign images categories.	123
5.2	Validation sheet reporting the bounding box, true positive, false positive, false negative and calculating performance indexes.	128
5.3	Maximum mAP for every network training (except Training 13).X axes is the number of images in the dataset used for the training.	133
5.4	Precision, Recall and F1 values coming from networks validation. Training number on the x axes.	134
5.5	BAS building.	135
5.6	First floor corridor showing the longitudinal development.	136
5.7	Embedded system including Movidius, Raspberry and Hololens.	137
5.8	Fire extinguishers at ground floor (in red) plus pictures points of view (in blue).	140
5.9	Fire extinguishers at first floor (in red) plus pictures points of view (in blue).	141
5.10	Fire extinguishers at basement floor (in red) plus pictures points of view (in blue).	142
5.11	Application interface for Hololens connection to embedded system.	143
5.12	Set position task.	144
5.13	Set rotation task.	144
5.14	Positioning of the hologram inside the scene.	145
5.15	Real-World scene referring to Fig 5.14	145

5.16	Fire extinguishers point of view during BAS test, basement floor. First row, left to right FE1, FE2, FE3. Second row FE4, FE5, FE6. Third row FE7.	148
5.17	Fire extinguishers point of view during BAS test, ground floor. First row, left to right FE1, FE2, FE3. Second row FE4, FE5, FE6. Third row FE7.	149
5.18	Fire extinguishers point of view during BAS test, ground floor. First row, left to right FE11, FE12, FE13. Second row FE14, FE15. . .	150
5.19	Fire extinguishers point of view during BAS test, first floor. First row, left to right FE1, FE2, FE3. Second row FE4, FE5, FE6. Third row FE7, FE8, FE9.	151
5.20	Fire extinguishers point of view during BAS test, first floor. First row, left to right FE10, FE11.	152

ACRONYMS

AECO	Architecture, Engineering, Construction and Owner-operated.	9
ANN	Artificial Neural Network.	29
AR	Augmented Reality.	42
AV	Augmented Virtuality.	43
BIM	Building Information Modeling.	3
CNN	Convolutional neural networks.	31
CWT	Continuous Wavelet Transform.	22
DBN	Deep Belief Network.	32
DFO	Derivative-Free Optimization.	14
FCN	Fully Convolutional Network.	37
FM	Facility Management.	3
FN	False Negative.	89
FOV	Field Of View.	21
FP	False Positive.	89
GPS	Global Positioning System.	50
HMD	Head Mounted Display.	41
HOG	Histograms of Oriented Gradients.	31
HPU	Holographic Processing Unit.	96
IFC	Industry Foundation Classes.	13
IFMA	International Association of Facility Managers.	4
IMU	Inertial Measurement Unit.	96
IOU	Intersection Over Union.	72

MR Mixed Reality. 41

MRTK Mixed Reality Toolkit. 94

NCS Neural Compute Stick. 106

NCSDK Neural Compute Software Development Kit. 112

NIST National Institute of Standards and Technology. 16

NN Neural Network. 9

PMR Pure Mixed Reality. 43

RANSAC Random Sample Consensus. 21

SVM Support Vector Machine. 31

TLS Terrestrial Laser Scanning. 22

TN True Negative. 89

TP True Positive. 89

UAS Unmanned Aerial System. 21

UAV Unmanned Aerial Vehicle. 37

VR virtual Reality. 41

YOLO You Only Look Once. 31

INTRODUCTION

1.1 Background and Motivation

The building life cycle consists of several stages: planning and design, construction and commissioning, operation, maintenance and renewal, revitalization, decommissioning and demolition (Fig. 1.1).

Life cycle is about taking a systematic approach to balancing maintenance costs, operating costs and replacement/refurbishment costs over the life of the asset. The concept is simple: life cycle costs include all costs associated with building assets from acquisition to disposal/replacement of the building itself.

For practical reasons, a typical lifespan is often used, such as 30 years or 50 years, depending on the organization and the type of facilities. Initial costs of a building represent only 10-20 percent of total cost, depending on the life span of the building.

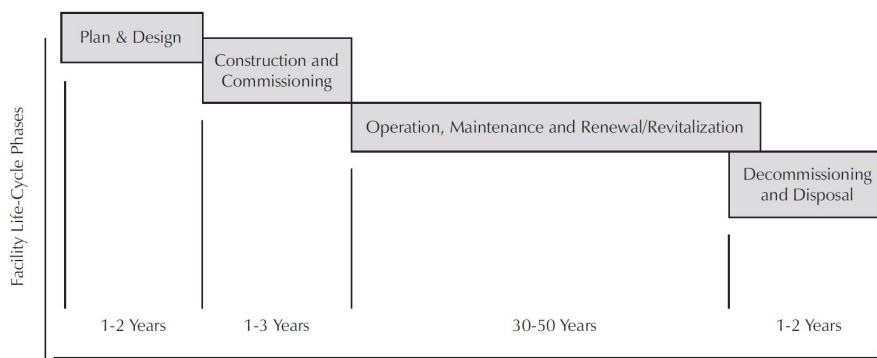


Figure 1.1: Building life cycle phases [Roper and Payant, 2014].

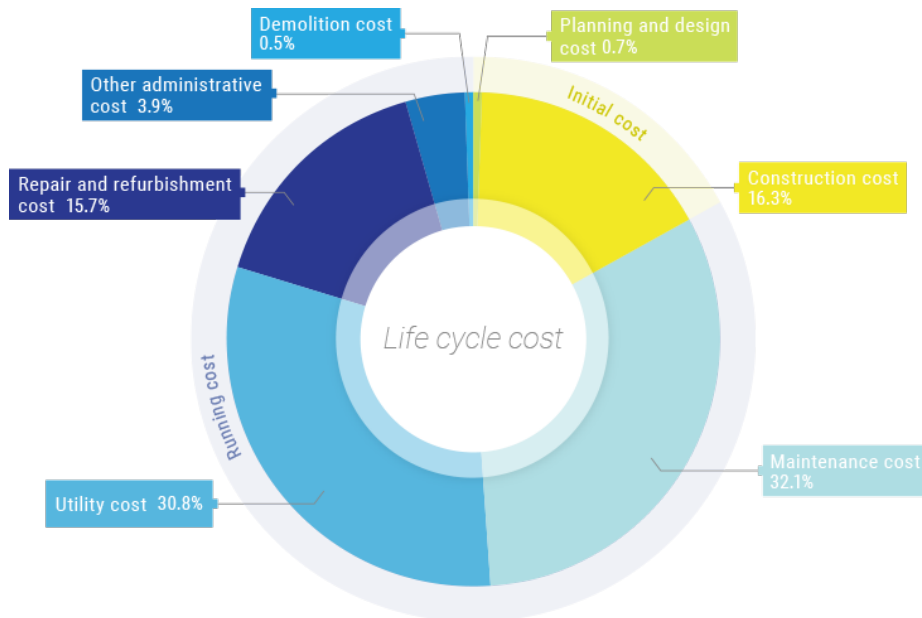


Figure 1.2: Building life cycle cost.

The rest of the cost, an astounding 80-90 percent, is for maintenance, operating and refurbishment/replacement of components over the building lifespan. Ignoring this large proportion of costs means wasting money (Fig. 1.2) [FMLink, 2018] [Devetakovic and Radojevic, 2007] [Pärn et al., 2017]. Some others estimates show that the life cycle cost is five to seven times higher than the initial investment cost and three times the construction cost [Lee et al., 2012] [Shen et al., 2010] [Akcamete and Akinci, 2010]. Referring to the third phase of building life cycle model it is the long duration that creates an information discontinuity.

When professionals are questioned about critical issues related to operational phase management more than 50% points out information accessibility [Liu et al., 2016]. In existing contexts in facts, as the one related to the management phase, information is often not available or not updated, while data coherent to the current situations are essential for every further process.

As a result, a building survey is usually necessary, either as an extension or validation of existing building documentation or to provide new documentation.

An essential prerequisite for FM tasks within existing contexts, is reliable functional data. A building survey which fulfils the needs of a planning task will be described as a planning-oriented building survey. Despite the many different fields of application for building surveying and the resulting different demands, the representation of the building geometry is typically the most investigated aspect of a building survey. However, without relationships to other kind of information, geometric information on its own can provide only limited

information for the planner [Donath and Thurow, 2007]. Functional models of buildings instead are crucial for O&M, refurbishment and especially emergency management.

Today, it is not uncommon for FM and Owner Organizations to have an incomplete concept of equipment inventories: their importance, use and how to maintain them. Equipment inventories affect facility safety, as well as how the facility is operated, maintained and forecasted. They also have a direct impact on facility costs. If the equipment inventory is not accurate, the facility and the organization will not be very effective. It is important to understand that accurate equipment inventories affect many different aspects of building management, including management of energy, projects, operations, maintenance, and customer service, and, therefore, they affect the overall finances of an organization. The emergency response includes the ability to identify replacement components, the ability to accurately scope the work to repair teams or contractors, and the ability to accurately estimate the project costs for management. The foundation to an effective O&M strategy is a component-level information inventory [Keady, 2013].

The Facility Management (FM) field relies heavily on getting usable data from a Building Information Modelling (BIM) model to do anything meaningful with it. All too often, this data are not really there or are inaccurate, as the model has not been updated with any design changes made after the design phase and is therefore not an accurate model of the facility as it is built [Kelly et al., 2013]. Clearly the role of the digital building must be to inform and interconnect the various activities that take place within these phases to enable more appropriate and longer-term decisions to be taken at each stage. Current BIM software is still mainly directed at the design phase where deployment is approaching a critical mass. As companies increasingly understand how best to exploit the software, and its capabilities continue to increase, rapid growth in deployment is anticipated. As the BIM vendor's focus starts to shift further down the building life cycle, the capability of BIM software is already moving from design into detailing and fabrication. Similarly, in the use phase BIM aware applications are already starting to be used to extract information from BIM models into facilities management software. For example, maintenance staff could access all relevant information via mobile devices with potential to also make use of augmented reality. More strategically, an anticipated development is the onward updating of the BIM to capture building usage, performance and maintenance information. Necessitating that an as-is model first be created (frequently from scratch). The effort this involves is a substantial barrier to the deployment of BIM software in the modify phase or the use phase of older buildings. The automatic acquisition of the existing geometry by laser scanning and point cloud technologies will soon help, but additional information acquisition tools will be needed to further reduce the barrier. [Watson, 2011].

Furthermore the process of information retrieval during surveys is trying to reach automation, both in the capture of information and in the passage of the latter to the BIM models.

1.2 Facility Management: an overview

Facility Management is an umbrella term under which a wide range of property and user related functions are brought together for the benefit of the organization and its employees as a whole. FM is holistic in nature, covering everything from real estate and financial management to maintenance and cleaning [Kelly et al., 2013].

According to the International Association of Facility Managers (IFMA), Facility Management is a profession that encompasses multiple disciplines to ensure functionality of the built environment by integrating people, place, process and technology [Devetakovic and Radojevic, 2007].

Therefore even if FM can comprise a huge variety of disciplines here there are some of the most common [Nazionale, 2010] [Camera di Commercio, 2012] [Roper and Payant, 2014]:

- buildings and infrastructures;
- utility services;
- environmental services;
- mobility services;
- technical scientific consulting services;
- ICT services;
- installation and maintenance of machinery, equipment and instruments;
- social health and educational assistance;
- administrative and legal services;
- logistic systems;
- cleaning services;
- procurement;
- waste disposal;
- financial services;
- security management;
- legal services.

Referring to building, information is critical for supporting efficient and effective management and day-to-day operations. However, the FM sector continues to grapple with information management, predominantly due to the peculiarity of information and its fragmentation [Pärn et al., 2017].

To take decisions related to building maintenance usually requires a high-level integration of various types of information generated by different persons at different times, such as maintenance records, work orders, causes and knock-on effects of failures, etc. In particular, while planning preventive maintenance actions, information flow through at least three different analysis nodes, respectively dealing with legal, technical and administrative aspects, each of which producing outputs that are necessary or considerably influential to others in order to correctly process and interpret data [Cesarotti et al., 2014].

Identifying the maintenance needs involves collecting and assimilating information from:

- regular condition surveys of the building stock;
- pre-acquisition surveys prior to any building purchase;
- the existing planned maintenance programme (or profile);
- faults and repairs notified by the building users;
- feedback from works of servicing, repairs and improvements in progress;
- existing building and service records.

Keeping track of all these required information in maintenance management requires careful handling to avoid errors, omissions or excessive bureaucracy [RICS, 2009]. The rapid advances in information and communications technology (ICT) in recent years have great significance in the management of maintenance. Computer databases, either networked or stand-alone, are increasingly used to store and manipulate such information [Roper and Payant, 2014]. For those of us who have a large inventory of older buildings with building drawings of uncertain validity, it is worthwhile to systematically have those buildings surveyed, their systems categorized and their drawings brought up-to-date.

The automation of processes could be the right means for the enhancement of efficiency. Thanks to the support given by machines to human tasks it is possible to save time and avoid errors. Anyway, the automation intended in this work is not the one obtained erasing human intervention but supporting it with the aid of tailor-made applications for the automation of some process operations instead.

1.3 Goal and Overview: surveying of assets components

The absence of a fully comprehensive picture of the present condition of buildings, especially for huge buildings stocks owner, represents a crucial issue. The current practice of asset management is paper-based consisting of manual inspection and proved to be time consuming, tedious, and prone to human error. An inventory of a water supply system, for instance, refers to detailed asset object list information of each facility and base information of each asset object that constitute an overall water supply system, and a group of such informational data can be called an inventory database for a water supply system. The collection of data for inventory became even more complex with the advent of BIM methodology. There is, on one hand the wish of describing the world exactly as it is, with all its details. On the other hand, every piece of information that has to be collected and modelled brings its own cost with it. There are a large number of components in a building and the data collection step consists of creating an accurate building inventory list, with various attributes that define the characteristics of these assets useful for further operations. Among the necessary data the location is fundamental but it is not the only one. There is a relevant amount of data that refer to the putting into service of components and to their functional data related to utilization. This information proves to be crucial during emergencies but also for daily operations. This is even more true when referring to complex buildings such as stations, airports, stadium, campus where the controls checks for the effective functioning of all safety devices is of primary importance. Not only this since in the aforementioned environments it is vital to maintain all the functions during operations.

The actual efforts in gathering data leads to the search for new methods for fastening this process. For this reason researches are focusing on automatic techniques for surveying and digitization. Among these techniques there are

- Image recognition;
- Laser scanning/lidar;
- Photogrammetry.

Anyway the majority of the researches still focus on geometry data. This kind of information is not the only one necessary to manage construction. Functional data are in fact essential to know the interactions between building, and more specifically between assets and their components without losing the connection with the spatial information. Very few studies investigate the possibilities for latest technology to support asset components inventory and they still share some drawbacks.

Current surveying techniques, even when using latest technologies, still comprise in their workflow long post processing phase to interpret data collected on site. Even when studies propose the use of powerful algorithms for semi-automatic interpretation of data this operation is still performed on a desktop avoiding the

possibility of concurrent verification of data collected. On-site operations happen before the post-processing phase so there is the impossibility of recovering data if missed.

The structure of this thesis is following:

- *Chapter 2* Literature Review - description of the latest surveying techniques is provided, then the use of recent techniques such as Machine Learning and Mixed Reality, especially referring to their potentialities in the construction industry, are presented.
- *Chapter 3* Methodology - in this chapter there is a deep description of all the technologies used in this research and the processes for their implementation inside the system proposed.
- *Chapter 4* Object Recognition System - chapter 4 explain all the different components of the Object Recognition System and its development.
- *Chapter 5* Proof of concept - this chapter presents all the test completed both related to the exploitation of Neural Network for performing object recognition and to the feasibility test of the whole system on-site.
- *Chapter 6* Conclusions

LITERATURE REVIEW

2.1 Introduction

This chapter provides background information on building surveying, Machine Learning and Mixed Reality technologies.

With the growing diffusion of BIM approach towards building data, thanks also to a boost provided by latest regulations issues principally in Europe, surveying needs are changed. Geometric data only are not sufficient anymore especially in complex buildings where the management of facilities requires a multidisciplinary knowledge.

The Architecture, Engineering, Construction and Owner-operated (AECO) sector is simultaneously living its transition to digitization of processes. The aim of this revolution is to increase efficiency in the construction industry where this parameter stopped its growth 40 years ago. With this perspective research is focusing on trying to automatize procedures exploiting new digital tools. The state of the art presented below finds its purpose in the analysis of current surveying techniques that aspire towards automatization of data collection and interpretation. This investigation allowed to highlight the opportunities for improvement in this field. As a result of this study it came out that all the proposed methods include post-processing phase to be performed separately. Furthermore most of the time these procedures do not gather functional data which are the most valuable for an efficient building management. Then the value of collecting and checking data directly on site and real time is outlined. Moreover according to FM needs identified through literature analyses one of the requirement for a profitable BIM model is, besides the exact localization of building components, the presence of technical properties and details useful for operations.

The state of the art referring to the use of Machine Learning and Neural Network (NN) summarizes some among the most interesting uses of image recognition in engineering fields.

Finally, Mixed Reality have been evaluated for its power in displaying information perfectly integrated in the real world. This technology applications showed that it is suitable for supporting operation on site leading to a man-machine efficient collaboration.

2.2 Surveying

2.2.1 The widespread adoption of BIM paradigm in the AEC industry

BIM is one of the most innovative and promising technologies that are making way in the construction industry. As far as the definition of the term is concerned there is no uniqueness; the one in the National Building dates back to 2008: "BIM is a process of programming, design, construction and maintenance that uses an informative model of a building, new or existing, which contains all the information concerning its entire life cycle ". The definition given here places the emphasis on the object that is created within the BIM framework. It is about a three-dimensional parametric model of the building, in which each element has property features and relates to other elements in compliance with certain rules.

The digital transition that is taking place in the construction industry represents a great opportunity to enhance the efficiency of this sector that shows a stationary if not even decreasing trend of the productivity in the last twenty years [McKinsey Global Institute, 2017].

In the UK and all over the world the race for better results in AECO industries has been supported by the adoption of BIM. In Europe, the adoption of BIM has increased considerably in the last five years as BIM implementation is seen as extremely beneficial to AECO clients [Codinhoto et al., 2013]. If its use has begun in the United States, lately thanks to the introduction of new regulations that require the use of electronic data formats, BIM is becoming known also in the Old World. To support productivity growth, regulators mandated the use of BIM to build transparency and collaboration across the industry; they tried to reshape regulations to support productivity, to create transparency on cost across the construction industry and to publish performance data on contractors (Fig. 2.1) [McKinsey Global Institute, 2017].

The European Parliament in 2014 enacted a regulation in this regard.

DIRECTIVE 2014/24/EU, article 22, subsection 4 says:

For public works contracts and design contests, Member States may require the use of specific electronic tools, such as of building information electronic modelling tools or similar. In such cases the contracting authorities shall offer alternative means of access, as provided for in paragraph 5, until such time as those tools become generally available within the meaning of the second sentence of the first subparagraph of paragraph 1. [Dir. 24, 2014]

Italian regulation followed in 2016 with the D.M. n. 560 enacted on December 1st, 2017 [D.M. 560, 2017].

This regulation poses six progressive value threshold through which the Public Administration will be forced in the application of BIM methodology. This means that after 2025 all public contracts in the construction industry will be handled through with BIM.

Figure 2.1 shows also other European regulations that in last years pushed the

construction industry towards the adoption of BIM approaches. This generalized push towards the application of BIM depends upon the expectations that are attributed to this methodology. It is expected that, in the long run, BIM could lead to save cost caused by inadequate interoperability by offering owners and operators a powerful means to retrieve information from a virtual model of a building [Shirazi and Ashuri, 2018].

What is certainly true and to take into consideration is the fact that it is mirage to consider that the introduction of BIM solely, without changes in the organisation's processes and overall culture, will enhance communication and facilitate decision-making [Codinhoto et al., 2013].

Anyway the implementation of BIM within processes necessarily starts from the correct development of the information model and capturing a building's intricate and expanding portfolio of data requirements is complex [Pärn et al., 2017].

For existing buildings, it might be necessary just to update pre-existing BIM (if it had been created during Design and Construction), or to create a BIM anew. This second situation is revealed much more common in Europe, where 80 per-cent of residential buildings are built before 1990, but also much more perilous due to the inaccuracy of data that have to be gathered manually through a reverse engineering process.

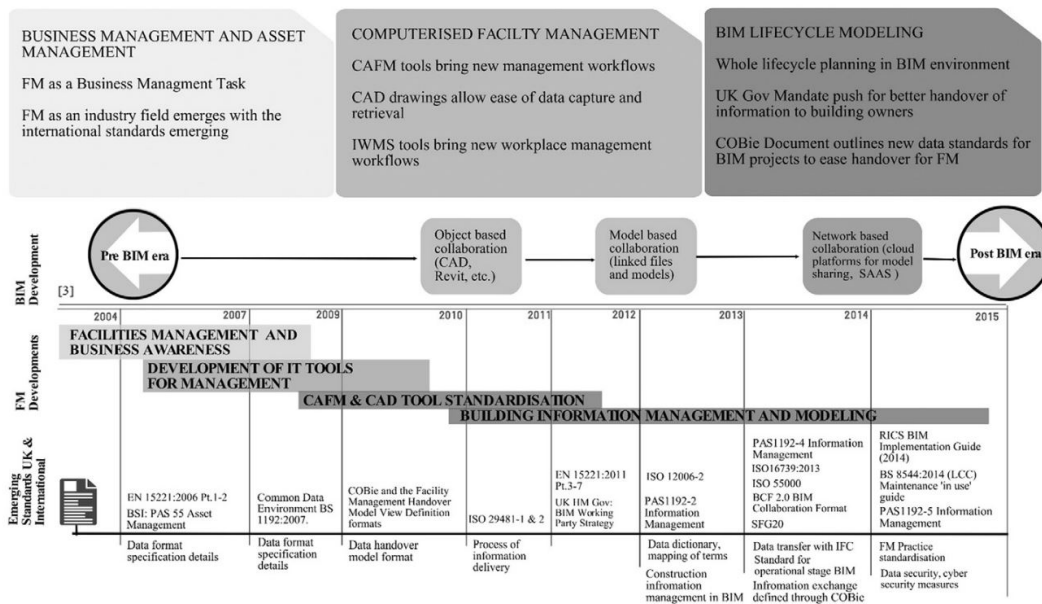


Figure 2.1: Developing of BIM and FM standards

Results show scarce BIM implementation in existing buildings yet, due to challenges of (1) high modeling/conversion effort from captured building data into semantic BIM objects, (2) updating of information in BIM and (3) handling of uncertain data, objects and relations in BIM occurring in existing buildings [Volk et al., 2014].

McArthur contends that identifying information required to inform operational decisions is critical to configuring data retrieval techniques at the post-construction stages [Mcarthur, 2015]. And it is well known that inconsistencies between demand and availability of particular information in an as-built model incur unnecessary expenditures. The necessity for an as-is model requires an effort that involves a substantial barrier to the deployment of BIM usage during refurbishment actions or management of older buildings [Watson, 2011].

The urgency of collecting data for the development of building model is witnessed by all the efforts that the research put into the accomplishment of this task.

The automatic acquisition of the existing data (geometric and semantic) is mostly pursued through point cloud collecting technologies, photogrammetry and image processing, but additional information acquisition tools will be needed to complete the necessary information framework.

The use of laser-scanning techniques has been one of the first way to automate the collection of data and it is still widely used [Xiong et al., 2013] [Ma and Sacks, 2016][Valero et al., 2018].

Supplementing the laser scanner data it is often image processing [Brilakis et al., 2010] worked with point clouds and images for the recognition of building components. Their process is made out of the following steps: 1.the spatial correlation of the collected data, from aligned point clouds and corresponding calibrated intensity images obtained by recording different scans. The output is a structured 3D surface model that describes the general topology of the scanned structures; 2. the recognition of visible attributes of objects using image processing tools; the input of this step is the rendered 3D surface while the output is the augmented 3D surface containing the recognized visual and spatial characteristics of potential objects and the background; 3. the classification of the objects and the adaptation of the dimensions. This process concerns the characterization and correspondence, based on the semantic labels provided, of an object with another, from a set of models in order to determine the type or class of this object. Finally, with this methodology a human assisted process is necessary, using a customized assembly interface built within a standard BIM application which incorporates the range of possible building object types in its internal object schema. Figure 2.2 shows the whole process proposed in this article.

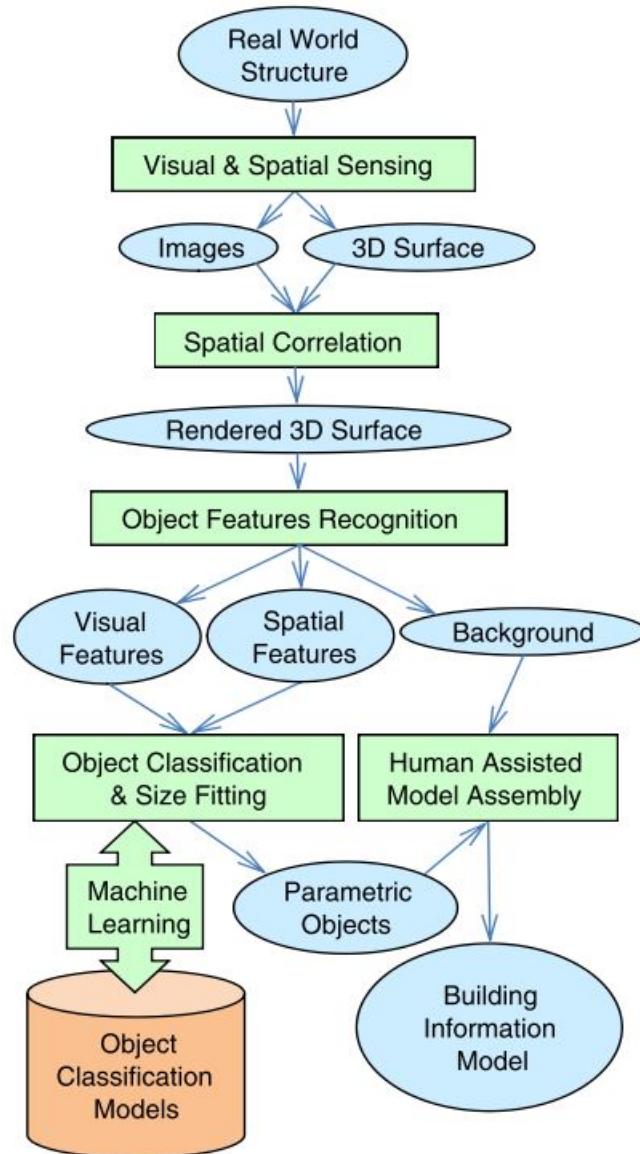


Figure 2.2: Steps of the automated generation of parametric BIM objects from video and laser scanning data [Brilakis et al., 2010].

[Lu et al., 2018] developed an image-driven system that seeks to effectively detect objects and their materials in complex environments with few features and to successfully create BIM objects represented in Industry Foundation Classes (IFC) (Fig. 2.3). This image-driven system contains three subsystems for different functions: an object recognition subsystem for recognizing

building components (i.e., columns, beams, windows, doors, and walls), a material recognition subsystem for recognizing surface materials, and an IFC BIM generation subsystem for creating BIM objects in IFC. The first steps for recognition are manual and applied by the operator who recognizes reference lines of known dimensions for the subsequent steps. To recognize objects in images, a sub-system based on the neuro-fuzzy framework was developed. The proposed material recognition subsystem subsequently recognizes the materials according to the classification procedure based on the extensible texture library. In this case the real IFC is generated using ifcengine, a .net library to implement ifc model parser. Anyway also in this case there is still need for a human intervention and all the data are also worked far from site.

Also the work by Xue et al. develops a Derivative-Free Optimization (DFO) approach for the automated generation of semantically rich as-built BIMs using 2D images (Fig. 2.4). Firstly, they formulate the as-built BIM as a constrained optimization problem. Secondly, the approach realizes an effective and segmentation-free BIM generation approach. Thirdly, the approach makes use

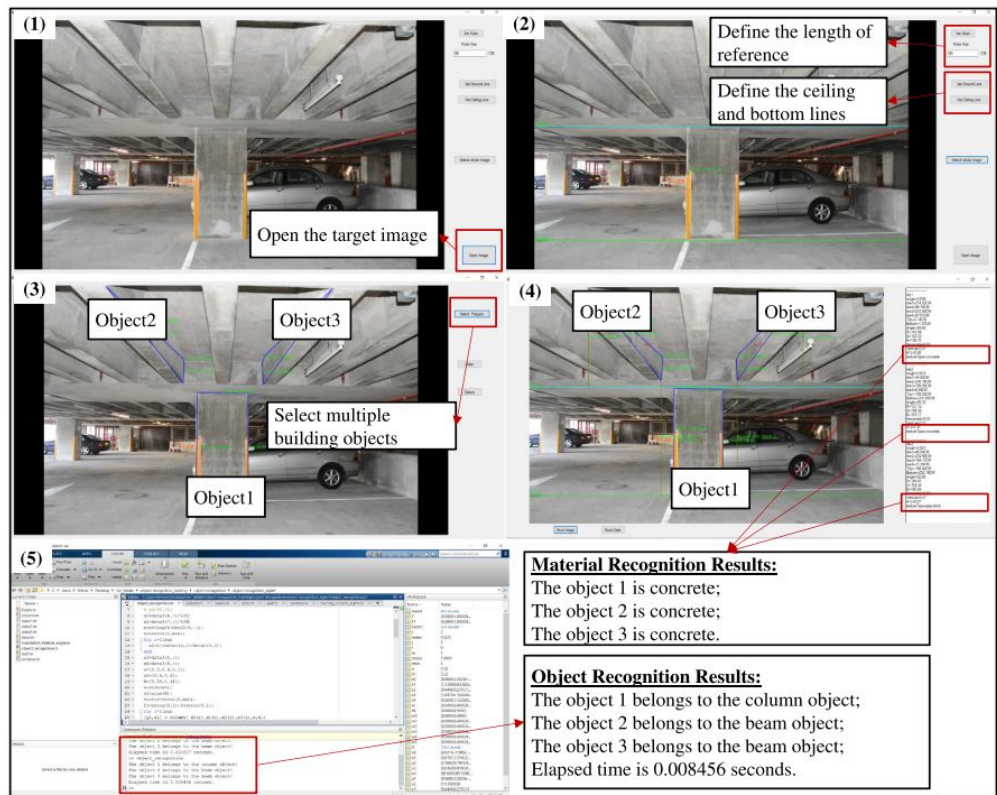


Figure 2.3: Steps of the recognition process through an image-driven fuzzy-system [Lu et al., 2018].

of the architecture domain knowledge and rich semantics in BIM component libraries that are available on the Internet or elsewhere. There are two inputs of the constrained optimization problem: the measurement data of as-built conditions and the BIM components contained in libraries. The measurement data, in this article, are 2D images that can be taken at a relatively low cost. Another input is the BIM components (e.g., facade, wall, windows, and doors) organized in one or multiple libraries. Then the generated BIM model is compared with the picture by means of a similarity function for the optimization of the problem. After an iteration process of comparison an optimized enriched as-is BIM model is obtained [Xue et al., 2018]. In this case, too, data are not processed on site. Furthermore the generation of the first BIM model becomes complex when the system is tested with a higher number of objects than the ones used here. Finally, creating libraries and rules for BIM generation seems a laborious process.

Despite all these efforts in the development of new procedures and methods for supporting and improving efficiency in the AECO industry another important problem to take into consideration is the fragmentation of this sector. Naturally, planning, construction and operation of a building are a high interdisciplinary task of different disciplines (architecture, constructional engineering, surveying and building services). This leads to difficulties in obtaining a smooth and continuous information flow among different parts in different stages. Notwithstanding the mandatory introduction of electronic tool for information management actually the current procedures show that information technology is limited in its use and application in construction, and most of the management work is done by human labor, which is inefficient and sometimes error-prone [Lin and Su, 2013].

The required exchange and adjustment of information between building construction services often is poorly geared to each other in practical terms (e.g. fragmented data, irregular modeling, media disruption, missing temporal agreement) which leads to errors, delays and finally to higher costs [Becker et al., 2018]. The building lifecycle phase mostly affected by this is the operation phase, which is the longest and for this reason the one that involves the highest cost.

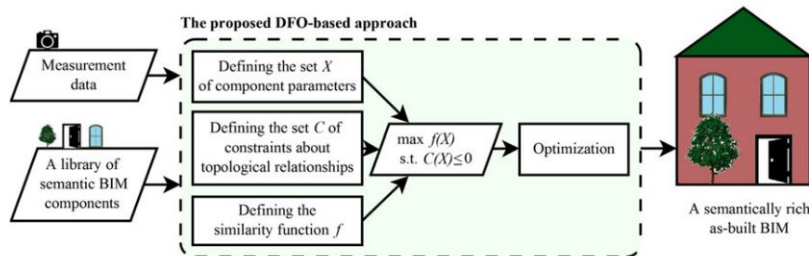


Figure 2.4: General framework of the DFO approach [Xue et al., 2018].

A study by the U.S. National Institute of Standards and Technology (NIST) showed that the annual costs associated with inadequate interoperability among software systems was \$15.8 billion [Gallaher M. and Gilday, 2004].

According to the FM representatives, data accessibility is a top problem and a considerable amount of time is spent in finding accurate information necessary to perform maintenance operations [Codinhoto et al., 2013][Liu et al., 2016] [Shirazi and Ashuri, 2018].

When facility maintenance contractors are paid to survey the existing building to capture as-built condition, owners are paying twice: once for the construction contractor to complete the documents at the end of construction and again for the maintenance contractor survey [East, 2007].

Data collection is furthermore a critical activity not only because of its high operational costs and low reliability, but also because of its impact on subsequent phases, i.e. data processing, object recognition and modeling, that are influenced by the data quality and level of detail [Cesarotti et al., 2014].

Anyway the information is not easy to be created in the model. In absence of the proper processes, in fact, it is hard to match model information with practice that is the real information needs [Feng and Lin, 2017].

The BIM methodology, which is spreading fast in the AECO industry, is recognized as an opportunity to increase efficiency and BIM models as a unique repository of building information. Consequently, BIM deployment becomes extremely invaluable to organisations that seek to reap inherent value and efficiency gains from the technology. The BIM capacity to harness valuable data and information throughout a building's life cycle is expressed by the following definitions provided by the UK Government and Succar. The first one defines BIM as: "a collaborative way of working, underpinned by digital technologies which unlock more efficient methods of designing, creating and maintaining assets" [task Group, 2012], whilst Succar [Succar, 2009] defines BIM as: "a set of interacting policies, processes and technologies producing a methodology to manage the essential building design and project data in digital format throughout the building's life-cycle." The aim behind the development of BIM tools was to create a building information system for information sharing regardless of software and data location. Towards this goal, the International Alliance of Interoperability proposed a standard that specifies object representations for construction projects: Industry foundation classes (IFCs), the basis of BIM methodology [Bonandrini et al., 2005].

BIM paradigm changes the way that information is managed, exchanged and transformed with the aim of stimulating greater collaboration between stakeholders via a single integrated model during the design and construction phases [Eastman et al., 2011]. This integrated approach to BIM ensures a smooth flow of information between all stakeholders. Building information modeling (BIM) has emerged as a disruptive innovation, showing great potential to mitigate many of the factors negatively affecting construction productivity.

[Poirier et al., 2015] tried to measure the productivity improvement given by the BIM use because despite the well-known benefits of BIM small organizations still hesitate to invest in such a change. This research's aim was to find a

productivity measure that could be taken into consideration by small realities, too. The findings of this research suggest an actual productivity rate that is superior to the estimated productivity rate across all systems and on all levels with the project, where BIM was used, having the greatest productivity ratio. Furthermore, they reviewed the key indicators of BIM's impact on productivity identified by Chelson [Fan et al., 2014] which are: quantity of request for information (RFI), amount of rework, schedule compliance and change orders due to plan conflicts. They identified that BIM did have an impact on two of these indicators for the scope that it was modeled and prefabricated for [Poirier et al., 2015].

BIM has been heralded as a facilitator for improvements also in FM efficiency by enhancing the integration of FM related information. These improvements are accrued during the generation and management of a facility's digital specification and characteristics data and cooperation between all parties involved in both building design and operation. Consequently, BIM can overcome some of the complexity and fragmentation experienced within the FM sector. In addition, a link from BIM models to FM databases could help to detect and diagnose building equipment based on necessary information such as specifications and maintenance history, which could be automatically associated with the located equipment and delivered to the on-site personnel. Moreover emergency management which depends on data from a variety of sources and needs spatial details, can find in BIM the perfect tool to efficiently store information. In the survey done by [Becerik-Gerber et al., 2011] the main task where BIM could favor FM have been identified as: locating building components, facilitating real-time data access, visualization and marketing, checking maintainability, creating and updating digital assets, space management, planning remodelling, renovating or demolishing, controlling and monitoring energy, personnel training and development.

[Ding and Drogemuller, 2009] further reinforces these findings and reveals that BIM enabled FM witnessed a 98% reduction in time used to update FM databases. With regards to BIM for FM in particular, the UK Government Strategy indicates that the relevant gains from BIM adoption will be perceived in the operational stages, where more efficient processes for managing the utilisation of public assets can be established (BIM Industry Working Group -IWG- 2011). In order to make BIM useful for facility managers or owners, project teams should define early on which FM information they need to include in their BIM models, and then establish a systematic process for capturing it during the design and construction phases [Pishdad-Bozorgi et al., 2018].

[Mcarthur, 2015] defined the following ones as the challenges that must be overcome to develop BIM models suitable for further operations: 1. identification of critical information required to inform operational decisions, 2. the high level of effort to create new or modify existing BIM models for the buildings, 3. the management of information transfer between real-time operations and monitoring systems and the BIM model, and 4. the handling of uncertainty based on incomplete building documentation.

As far as the first issue is concerned decisions for building maintenance require

integration of various types of information and knowledge created by different members of construction teams such as: maintenance records, work orders, causes and knock-on effects of failures, etc. Not only this, one of the key challenges in projects is the need to have sufficient information on products ready available for any maintenance operation. Among this information it is possible to find specifications, previous maintenance work, list of specialty professionals to conduct work. For this reason the FM sector continues to grapple with information management, predominantly due to the peculiarity of information and those fragmentation.

Trying to overcome this issue [Motawa and Almarshad, 2013] developed a knowledge-based library containing information about a building object subject to maintenance. They collected in the same library also the know-how so as it can be used for diagnosis and training as well.

BIM-FM integration represents a major challenge but it would be extremely beneficial for processing large sets of complex information typically associated with maintaining building assets [Love et al., 2014] [Pärn et al., 2017] [Yang and Ergan, 2017]. Several studies, in fact, posed the BIM as the fully information management tool for the development of FM support application [Kelly et al., 2013] [Lin and Su, 2013]. Most of them also combine the BIM database with the power of Augmented Reality for the display on site of a series of data, trying to accomplish the third of McArthur challenges [McArthur, 2015], that otherwise should be retrieved among several different documents. Furthermore, among the first features of a BIM model there is the management of space, since objects are located in a digital 3D copy of the building, and the visualization of objects in their 3D shape exactly as they are.

The second and forth issues expressed by McArthur [McArthur, 2015] can be considered together. They both refer to data collection and implementation that is an high demanding process in term of time and money. The FM field could rely heavily on getting usable data from a BIM model to do anything meaningful with it. However all too often, these data is not really there or is inaccurate, as the model has not been updated with any design changes made after the design phase and is therefore not an accurate model of the facility as it is built. Moreover most of the studies focus on the collection of maintenance data [Motawa and Almarshad, 2013][Kelly et al., 2013] but they do not mention the problem of having the knowledge about the status of the building which is crucial especially in countries where the number of aged building is much higher than the new construction rate.

It is believed that the main benefits from adopting BIM are yet to be seeing as a result of its application to FM as evidence that demonstrates its benefits is hard to produce [Codinhoto et al., 2013].

2.2.2 Latest technologies that support building survey procedures

As new construction rates in industrialized countries stagnate, planning and implementing refurbishment and retrofit measures in existing buildings gain in importance. The need to have structured and semantically enriched "as-is" 3D digital models of buildings in order to handle, more efficiently, projects of maintenance, restoration, conservation or modification is increasing in recent years. "As-built" BIM is a term used to describe the BIM representation of a building concerning its state at the moment of survey. It is usually a manual concept that involves three aspects: firstly, the geometrical modeling of the component, then the attribution of categories, material properties and functional data to the components and, finally the establishing of relations between them. Current literature proposes automatic "as-built" BIM approaches that could be classified into three main categories (Fig. 2.5):

- **Heuristic approaches:** in this field most of methods rely on a first segmentation of the scene. Those approaches use a human knowledge codification belonging to the architectural field. As a matter of example, doors and windows are always embedded in a wall class, roofs are always "hierarchically above" walls.
- **Approaches based on context:** they use relations between components.
- **Approaches based on prior knowledge:** this approach follows the principle of detecting differences existing between the conditions of the "as-built" and "as-designed". In this kind of approach, the recognition problem is reduced to a problem of fitting or matching between the entities of the scene and the point cloud.
- **Approaches based on ontologies:** this method uses a priori knowledge of objects and environment. This knowledge is extracted from databases, CAD drawings, GIS, technical reports or expert knowledge belonging to particular fields. Therefore, this knowledge constitutes the basis of a knowledge-based selective detection and recognition of objects in point clouds. In such a scenario, the knowledge of these objects must include detailed information on the geometry of the object structure, 3D algorithms, etc.

Especially as far as existing buildings are concerned it is necessary to develop an efficient approach, based on a first step of building survey, to develop a semantically enriched digital model. Various digital tools for building capture and auditing are available, such as 2D/3D geometrical drawings, tachometry, laser scanning or automatic locating of images (Fig. 2.6), but they need increased modeling and planning efforts of skillful personnel.

Semi-automated laser scanning with total stations is prevalent, although affected with disadvantages such as high equipment cost and fragility plus extensive data

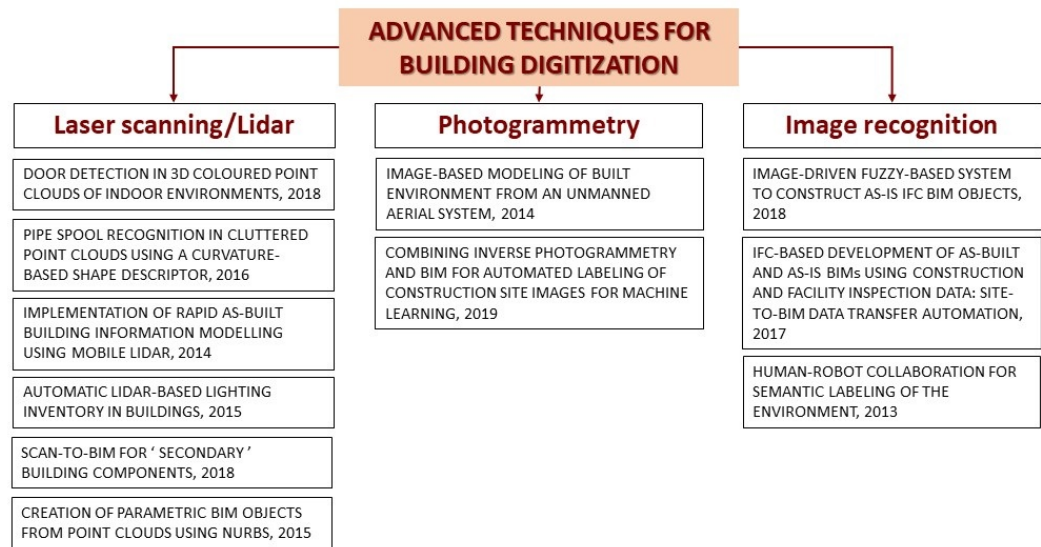


Figure 2.5: Classification of advanced building survey techniques.

processing and modeling steps [Volk et al., 2014]. As an example of the use of laser scanning techniques the survey work by [Mill et al., 2013] deals with data collection of the Tallinna Tehnikakõrgkool University of Applied Sciences (TTK/UAS) located in the capital city of Estonia. The case study provides a critical appraisal of the process of both collecting accurate survey data using a terrestrial laser scanner combined with a total station and creating a BIM model as the basis of a digital management model. Traditionally, a total station is used to record single points, even if this method is relatively time-consuming since points are recorded one by one. Terrestrial Laser Scanning data was acquired at 26 stations, to receive information from as many parts of the object as possible and to leave fewer hidden sections. To obtain a complete representation of the scanned object, the scans were combined into one dataset by directly georeferencing the point clouds into the predetermined geodetic reference frame. Since the level of interior detail was not high, the internal survey was accomplished using a total station Trimble M3. Post-processing of data is the main limit of this technique since it requires long time. In this study data processing was divided into three different phases, the first, exterior point cloud processing, the second, internal total station survey data processing and the third, processing data using BIM software to create the BIM model. The surface of the building facade was modelled and located manually entirely using the laser scanning point cloud data.

The approaches mentioned above may provide satisfactory results in the recognition of elements composing a scene. But in a BIM context and in order to

semantically enrich point clouds, it is not sufficient to detect their sub-parts as architectural components (walls, windows, doors, etc.). An important requirement is also to define the relations linking components to their attributes, in particular, spatial relations (topological, directional, etc.) between them. As example, if a wall is detected, it should be specified that it is connected to the ground, in a specific position, adjacent to other walls, these last ones having other positions, etc. And it is also necessary to specify, whether such wall is made of stone or bricks. In effect, attributes can vary according to the field, to the needs determined by management and to the use of the building [Hichri et al., 2013].

Other approaches using point clouds as a starting point is beginning to spread for the semi-automatic identification of objects. [Díaz-Vilariño et al., 2015] in their research used an algorithm (Fig. 2.7) for the automatic detection of ceiling lightings applied to the School of Mining Engineering at University of Vigo. The main sections of the algorithm consist of ceiling extraction, point cloud to image conversion, and luminaires detection. Data acquisition is performed using a Faro Focus 3D X 330 LiDAR. The scanner is a panoramic static terrestrial LiDAR system with a 360° horizontal field-of-view (FOV) and 270° vertical FOV. Although the Faro scanner can provide additional point information such as RGB and intensity properties based on an inner RGB sensor and LiDAR detected power, respectively, these attributes were not used in the present work where the algorithm is focused on geometric parameters. Ceilings are segmented from the rest of point cloud using plane detection by means of Random Sample Consensus (RANSAC) algorithm. In this case point clouds collected with the lidar are only the starting point since after this first process point cloud belonging are converted to images in order to apply image processing algorithms. From this binarized images two algorithms are manually applied by an operator: fluorescent lightings are distinguished using a refined Harris corner detector while a Hough transformation is applied to find circular low energy bulbs. Finally there are aforementioned systems like the one by [Lu et al., 2018] that exploit images rather than point clouds. Starting with the human intervention for the definition of reference lines the neuro fuzzy network developed is capable of recognizing building objects. Then a proposed material recognition subsystem recognizes the materials according to the classification procedure based on the extensible texture library.

Higher efforts for the automation of data interpretation have been done especially in recent years as can be seen from the following research studies.

[Rodríguez-gonzález et al., 2014] presented a methodology for automatically reconstructing 3D complex scenarios, particularly electrical substations, using images acquired from an unmanned aerial system (UAS). The case study was performed at an outdoor electrical substation located in Olloki, Pamplona, Spain, a complex scenario due to the large number of elements and the large area covered by the site. The use of a UAS permits the documentation of the elements completely by aerial images, guaranteeing a high spatial and temporal resolution. The Photogrammetry Workbench (PW) developed is a multiplatform software with a user-friendly interface that works with terrestrial

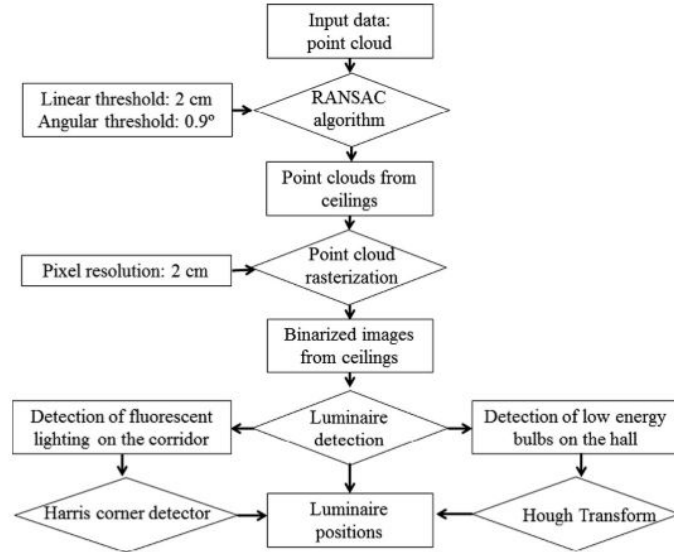


Figure 2.6: Semi-automatic light inventory algorithm [Díaz-Vilariño et al., 2015].

or aerial images and considers vertical or oblique geometries. The UAS Mikrokopter–Oktokopter platform was used. From the high level of detail of the elements, it is possible to model each element from the point cloud. Fig. 2.7 shows the process of modeling one of the objects existing (the intensity transformer) in the electrical substations. To obtain the final model, it is necessary to segment the element and apply a noise filter. The primitives are modeled by semi-automatic approaches of the reverse engineering field. Anyway we are still far from an effective automation of BIM model enriching process toward a reduction of post-processing phase of raw data.

Moving to more automation in data interpretation [Valero et al., 2018] presented an algorithm for automatic segmentation of individual masonry units and mortar regions in digitised rubble stone constructions, using geometrical and colour data acquired by Terrestrial Laser Scanning (TLS) devices. The algorithm is based on the 2D Continuous Wavelet Transform (CWT). The case study for testing the system proposed is the Craigmillar Castle and Linlithgow Palace. A Faro Focus 3D Laser Scanner digitised the scene, providing 3D and colour information, with a resolution of 3 mm and a Leica P40 Terrestrial Laser Scanner was also used for data acquisition, delivering colour and geometric information, with a resolution of 2 mm. Data acquisition and pre-processing of the data lead to coloured point clouds of the wall face. First, the data is converted into a 2D depth map by means of an orthogonal projection on a surface



Figure 2.7: Primitive modeling of a cylinder indicating a piece of a pipe [Rodriguez-gonzalvez et al., 2014]

grid. This grid is calculated following a strategy based on the RANSAC algorithm in the case of walls whose two principal curvatures are close to zero (i.e. planar walls). They then used the 2D Continuous Wavelet Transform which is a signal analysis method that is based on the convolution of the input signal with a wavelet function at different locations along the signal and at multiple scales. This enables the detection of the signal pattern of the wavelet function at potentially any scale and at any location. The 2D CWT is applied to the depth map using an estimate of the mortar joint width to define the scale of interest. The binary image delivered by the 2D CWT contains an approximated segmentation of the stones. The segmentation achieved with the described algorithm can be used for the evaluation of materials and their associated construction technologies. Point cloud processing phases in Figure 2.8. [Xiong et al., 2013] presented a method to automatically convert the raw 3D point data from a laser scanner positioned at multiple locations throughout a facility into a compact, semantically rich information model. This algorithm identify and model the main visible structural components of an indoor environment (walls, floors, ceilings, windows, and doorways) despite the presence of significant clutter and occlusion (Fig. 2.9). It begins by extracting planar patches from a voxelized version of the input point cloud. Patches are found using a region-growing algorithm to connect nearby points that have similar surface normals and that are well-described by a planar model.

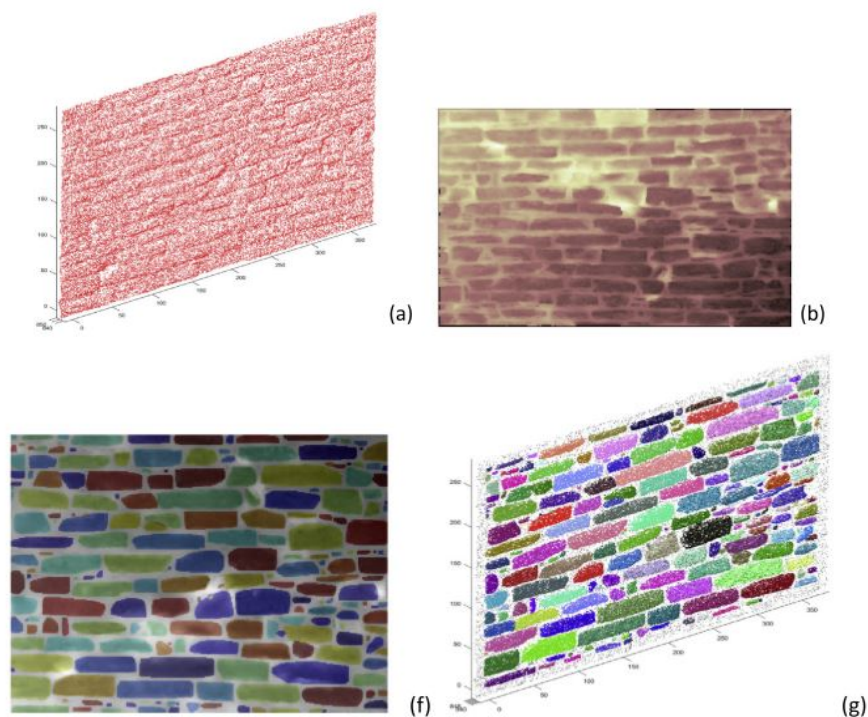


Figure 2.8: Masonry point cloud processing phases [Valero et al., 2018].

The algorithm learns the unique features of different types of surfaces and the contextual relationships between them and uses this knowledge to automatically label patches as walls, ceilings, or floors. Then, a learning algorithm is used to intelligently estimate the shape of window and doorway openings even when partially occluded. Finally, occluded surface regions are filled in using a 3D painting algorithm. To understand occlusions, a ray-tracing algorithm is used to identify regions that are occluded from every viewpoint and to distinguish these regions from openings in the surface. To compare the results from all algorithms they calculated the F1 score which is the harmonic means of precision and recall. The average F1 score achieved is of 0.85 over 4 classes of objects (Clutter, Wall, Ceiling, Floor) which is considered satisfactory. The semi-automatic data interpretation and BIM modelling of existing buildings have been discussed also by [Chiabrando et al., 2016] whose study uses cross sections and surface extrusion from the point clouds (semi-automatic). They worked on the digitization of a historical building, the Castle of Valentino in Turin. They worked partially with manual modelling of BIM objects (e.g. the foundation, walls, lunettes) starting from the basis of a mesh obtained through point clouds (Fig 2.10, 2.11).

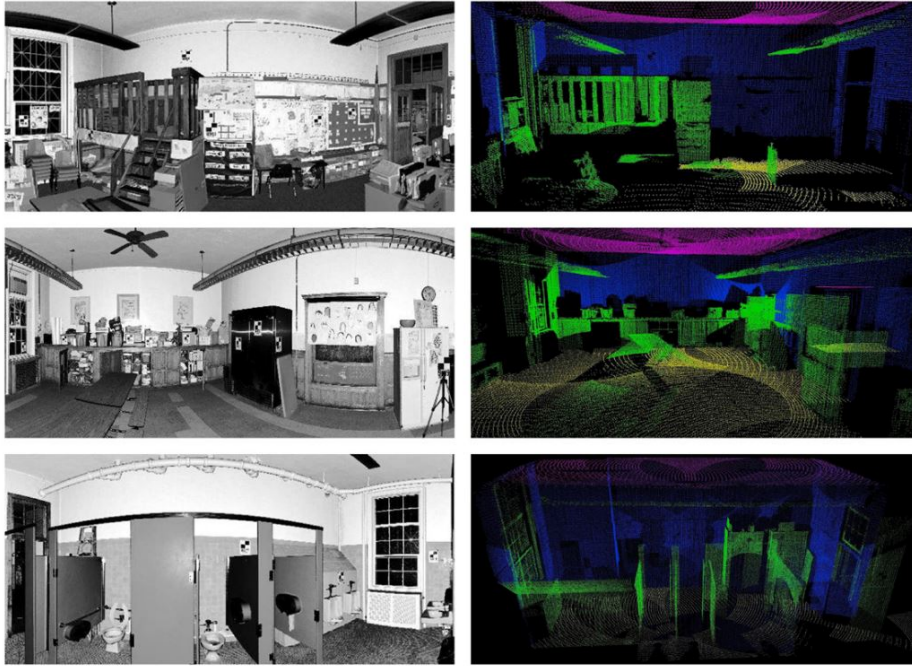


Figure 2.9: Left column shows the reflectance images while the right column shows the classification results from the surface algorithm [Xiong et al., 2013].

For the modeling it was used the Revit software which in its basic configuration does not read point clouds and does not use NURBS modeling. Surely this type of processing (semi-automatic) is time consuming and complex to realize. Other parts of the building were modelled semi-automatically using the plugin Scan to BIM (IMAGINiT Technologies) for Revit software, able to recognize surfaces described by clouds of points. Though a mesh model of the room was available, modeled from the point model into a software dedicated to the modeling of clouds (3D Reshaper), the parametric model in Revit has been built through the cloud interpretation. As for the wall surfaces most of the decorative elements have been extracted the profiles from the cloud and it is operated semi-automatic parametric modeling. The case of the columns has been treated with the loadable families.

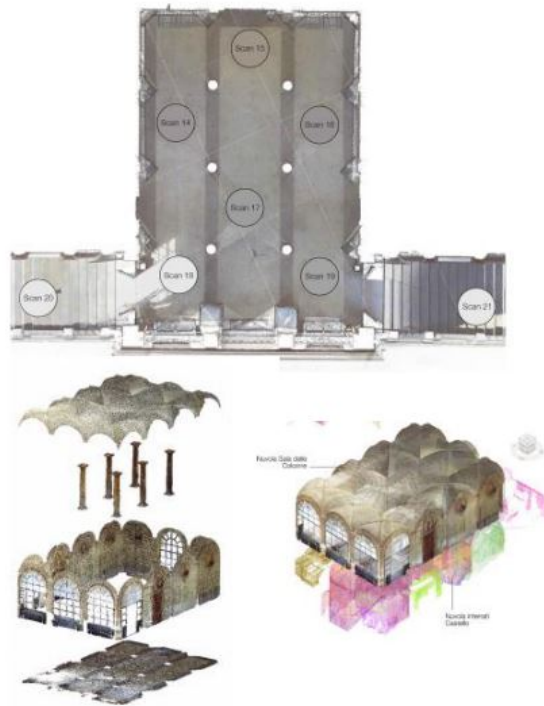


Figure 2.10: Starting raw data: point clouds [Chiabrandò et al., 2016].

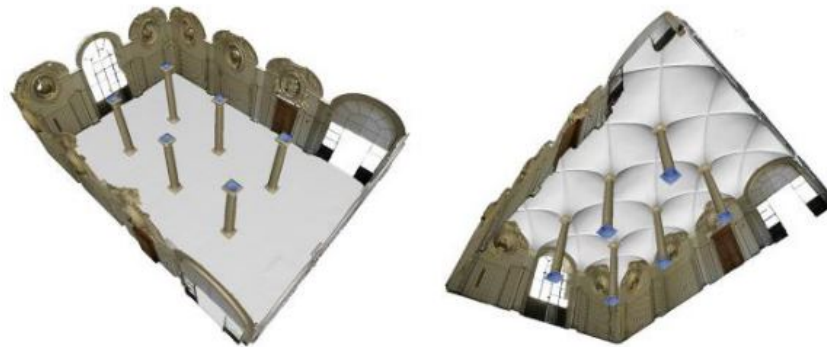


Figure 2.11: Final BIM model [Chiabrandò et al., 2016].

All of the aforementioned methods require long post-processing phase and none of them proposed the possibility to check data directly on site avoiding inaccuracies. Some efforts have also been put into the semi-automatic modelling of building objects rather than surfaces or global geometry. Most of the

recurrent maintenance actions are in fact conducted on assets inside buildings. This means that all the components need to be surveyed and their data properly stored and made available. One case of trying to automate the collection of building assets data is represented by [Quintana et al., 2018] who presented an approach that detects open, semi-open and closed doors in 3D laser scanned data of indoor environments. The proposed technique integrates the information regarding both the geometry (i.e. XYZ coordinates) and colour (i.e. RGB or HSV) provided by a calibrated set of 3D laser scanner and a colour camera. The work presented focused on door detection, that is performed once the scanning of a room has been completed. The output of the room scanning is composed of a dense 3D coloured point cloud, a labelled voxel model with associated 3D points from the point cloud; and a 3D boundary model of the room composed of planar rectangular patches (and their associated voxels) representing the walls, ceilings and floors. Wall elements have associated voxels that can be labelled as either:

- Occupied: the voxel contains at least one scanned point.
- Occluded: the voxel does not contain any point and was not visible from any of the scanning locations used to scan the room.
- Opening: the voxel does not contain any point, despite being visible from at least one scanning location.

The detection process considers the labelling and coloured 3D points associated with the voxels of each wall a rectangular segment. For each wall plane, a 4D orthoimage (Fig. 2.13) for each of the scanning locations is created, and then merged into a unified orthoimage JCD. The algorithm for detecting doors is divided into two steps, wall area detection and door detection. Taking the orthoimage as input they presented an approach based on discontinuities in the 4D RGB-D space and the knowledge of the wall area. Colour and depth components are processed separately. This is followed by an image binarisation process, using Otsu's global histogram threshold technique that selects the threshold to minimize the intraclass variance of the black and white pixels. White pixels in the orthoimage represent discontinuities in the colour-depth space, which enables the detection of door frames as discontinuities in the colour domain only, in the depth dimension only, or in both. Next all possible rectangles are defined by two pairs of horizontal and vertical lines. The pose and size of the recognized doors is evaluated by means of Precision, Recall and F-measure computed based on the overlap between the areas of the ground truth (that is the correct door placed in the true position) and recognized doors. For each of the aforementioned parameters the reached value overcome 90% proving the efficiency of the system proposed.

Finally a similar method by [Quintana et al., 2017] have been developed for the "Scan-to-BIM" recognition of small objects inside buildings. They proposed a system for the detection of "small components" based on coloured point clouds acquired by a 3D laser scanner calibrated with a digital camera and mounted



Figure 2.12: Detection results for Simple Scenes. (Left) Original 4D orthoimages. (Right) Door detection [Quintana et al., 2018].

on a mobile robot. This process started from the condition of having already modelled the basic geometry of the building. This way walls are already present in the information model and the recognition process can start from a coloured point cloud associated with a modelled wall. More precisely a 4D orthoimage is generated having for each pixel both colour (RGB) and depth. At this stage a double way of pursuing recognition is presented: recognition with geometry and recognition with colour. For detecting objects with geometry discontinuities the depth image is used (Fig. 2.12). In this case a Canny filter, for edge detection, is applied to the image. When the Canny threshold is below 0.05 a simple image cross-correlation algorithm is applied that assesses the correlation between the candidate regions with depth image models contained in the given database of objects to be recognised. On the other hand objects that are salient in the colour domain instead of the geometric one are detected by looking for colour discontinuities. The matching against objects in the database established a priori is conducted by matching SURF features extracted from the colour image models in the database with those extracted from the candidate region. With this method some object can be recognized both with depth and colour while for others only one kind of image processing is suitable. In the experimental test of this method they recognized sockets, switches, fire alarms, extinguishers and alarm signs and obtaining correlation parameters values higher than 0.55 which is the threshold to consider the detection as a potential candidate recognition. Even if laser scanning techniques have proven to be effective, delivering accurate 3D and colour measurements, the outcome always requires further processing to produce understandable semantically-rich information that can be interpreted by experts. As post-processing is performed off-site operators may be unaware in case some data are missing and unable to integrate missing data within the same survey mission.

Moreover recent research focused on capturing mainly geometric rather than

semantic representations of buildings and feeding point cloud data into BIM software. Finally most of the aforementioned techniques also are able to reconstruct only the geometry of objects without adding any semantic. New developments intensely research process models for automated BIM modeling from captured data ('scan-to-BIM') and improvements in LoD to enhance application in existing buildings [Volk et al., 2014].

Furthermore all the methods explained here consist of complex post processing operations that not only requires long time and therefore high costs but they are also pursued far from the site where data are collected. This lead to an error prone process and difficulties in the interpretation of gathered data.

2.3 Machine Learning

The problem of automatic programming is one of the central questions in computer science. Paraphrasing Arthur Samuel (1959), the question is 'How can computers learn to solve problems without being explicitly programmed?'. In other words, how can computers be made to do what needs to be done, without being told exactly how to do it?

Machine learning is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. With the emergence of deep learning, computer vision has proven to be useful for various applications. Deep learning is a collection of techniques from Artificial Neural Network (ANN), which is a branch of machine learning. ANNs are modelled on the human brain; there are nodes linked to each other that pass information to each other [Moore, 2018].

The first case of neural networks was in 1943, when neurophysiologist Warren McCulloch and mathematician Walter Pitts [McCulloch and Pitts, 1988] wrote a paper about neurons, and how they work. They decided to create a model of this using an electrical circuit, and therefore the neural network was born. Skipping some years already in 1958 we found the first example of pattern and shape recognition, Frank Rosenblatt designed the first artificial neural network. An-

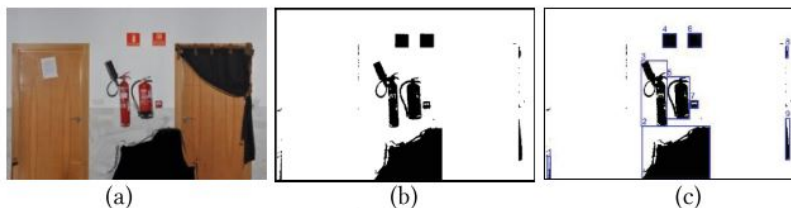


Figure 2.13: (a) Coloured point cloud, (b) depth image, (c) object detection for object recognition [Quintana et al., 2017].

other extremely early instance of a neural network came in 1959, when Bernard Widrow and Marcian Hoff created two models of them at Stanford University. The first was called ADELIN, and it could detect binary patterns. For example, in a stream of bits, it could predict what the next one would be. The next generation was called MADELINE, and it could eliminate echo on phone lines, so had a useful real world application. It is still in use today. Despite the success of MADELINE, there was not much progress until the late 1970s for many reasons. 1982 was the year in which interest in neural networks started to pick up again, when John Hopfield suggested creating a network which had bidirectional lines, similar to how neurons actually work. Neural networks use back propagation and this important step came in 1986, when three researchers from the Stanford psychology department decided to extend an algorithm created by Widrow and Hoff in 1962. This therefore allowed multiple layers to be used in a neural network, creating what are known as ‘slow learners’, which will learn over a long period of time. Since the start of the 21st century, many businesses have realised that machine learning will increase calculation potential. Machine learning algorithms were from the very beginning designed and used to analyze medical datasets [Kononenko, 2001]. From then machine learning techniques have been widely applied in a variety of areas such as pattern recognition, natural language processing and computational learning [Liu et al., 2017]. Game-playing applications offer various challenges for machine learning. A wide variety of learning techniques have been used for tackling these problems. Based on data/behavior observed in the past, machine learning methods can automate the process of building detectors for identifying malicious activities. Machine learning can be used to construct models for misuse as well as anomaly detection. [Lee et al., 1999] apply machine learning to detect attacks in computer networks. They first identify frequent episodes, associations of features that frequently appear within a time frame, in attack and normal data separately. Frequent episodes that only appear in attack data help construct features for the models. Ghosh and Schwartzbard [Ghosh and Schwartzbard, 1999] use neural networks to identify attacks in operating systems. Based on system calls in the execution traces of normal and attack programs, they first identify a number of “exemplar” sequences of system calls. For each system call sequence, they calculate the distance from the exemplar sequences.

2.3.1 Neural Networks for the recognition of objects

Particularly, the field of image recognition has seen an increase in development in the recent years. In the automotive industry, for example, the use of deep learning algorithm has allowed self-driving cars to recognize lanes and other obstacles without the need for more expensive and complex tools [Huval et al., 2015]. But the range of applications extends also to fields in which technology is not a key characteristic: for example in the field of arts, [Lecoutre et al., 2017], have used a residual neural network (ResNet) to build a model capable of detecting the artistic style of a painting with an accuracy of 62%, which could help in future the indexing of art collections. Object detection had an explosion concerning

both applications and research in recent years. Object detection is a problem of importance in computer vision. Similar to image classification tasks, deeper networks have shown better performance in detection. At present, the accuracy of these techniques is excellent. Hence it is used in many applications. Image classification labels the image as a whole. Finding the position of the object in addition to labeling the object is called object localization. Typically, the position of the object is defined by rectangular coordinates. Finding multiple objects in the image with rectangular coordinates is called detection [Moore, 2018].

Since the ML can be very versatile also its applications can be very diverse. It is possible to find example of human behaviors recognition through the use of You Only Look Once (YOLO) neural network and the LIRIS human activities dataset [Shinde et al., 2018].

This research focuses more on the performance of the network rather than its application but it is evident the use that could be done for instance for security purposes.

Another crucial use of object recognition with Neural Network is the one pursued by medicine. The support in the recognition of masses through medical images has been largely studied and it resulted in valuable applications [Al-masni et al., 2018] [Ali, 2019] [Wang et al., 2019a] [Wang et al., 2019b] [Nakagawa et al., 2009].

2.3.2 The use of NN for recognition in engineering

The use of Artificial Neural Network (ANN) had a wide spread in almost all engineering fields. Convolutional neural networks have proven to be valuable in many application fields but the AECO industry has not exploited this tool at its best yet [Braun et al., 2019].

[Lamio et al., 2019] studied an application of machine learning for the construction industry to categorize images of building designs into three classes: Apartment building, Industrial building or Other. No real images are used, but only images extracted from Building Information Model (BIM) software, as these are used by the construction industry to store building designs. The dataset consists of a total of 240 structural models, in which the images were extracted from their BIM virtual representations: 4 images for each of the 60 BIM representations have been extracted, showing completely different angles of the structures. Due to the low number of images available to validate the deep learning models, they have been augmented randomly generating samples, processed with a combination of random rotations, horizontal flips and vertical and horizontal shifts. They chose three ML methods: the classical machine learning method chosen for the problem described, is a combination of Histograms of Oriented Gradients (HOG) used for feature extraction and Support Vector Machine (SVM) for the classification task. Then they used two of the top performing deep neural networks architectures on the ImageNet database: the MobileNet and the ResNet. In addition to the two pre-trained networks just described, it was also used a Convolutional Neural Network (CNN) with a randomly generated structure and

randomly initialized weights (not pre-trained). The choice to randomly generate the network structure (Fig. 2.14) is

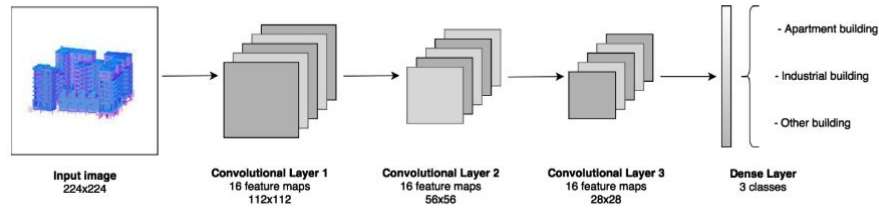


Figure 2.14: Structure of the neural network with random generated structure: it presents 3 convolutional layers, followed by a dense layers that maps the output of the last convolutional layer to the three classes [Lamio et al., 2019].

linked to one of the problems in building efficient neural network architecture: it is often difficult to optimize and fine tune its parameters. To evaluate their accuracy accuracy, the database of 240 images was randomly split in order to use 80% of the database to train the models and the remaining 20% to validate them. The results of the best model evaluation are shown: the best performing neural network was the ResNet50, with an accuracy of $97.92\% \pm 1.32\%$. The MobileNet as well scored an accuracy above 90%. Moreover, it can be seen that the CNN with randomly generated structure performed well, obtaining an accuracy of 89.60%. 3.39% lower than the ones obtained with the pre-trained network, but still acceptable. The worst performing model was the HOG + SVM approach, which scored an accuracy of only $57.19\% \pm 1.18\%$, much lower than the worst performing neural network. The results of this research show that despite the small size of the dataset, the deep learning models outperformed the classical machine learning model. Furthermore they overcome the problem of data collecting using artificial images and with further tests on real-world images this could prove to be a valuable method to build new datasets.

[Zhao et al., 2015] propose a similar approach towards the recognition of 3D BIM environments. To realize the retrieval and classification of a content-based 3D model, the key point is to require the extracted 3D model feature description is invariant and robust to translation, rotation, scale size and orientation transformation. They used ray-based feature extraction algorithm to extract features of a 3D model. The developed dataset included 1,814 models, covering most types of models common found in daily life. To prevent certain types of models from impacting the fairness of the assessment, the 1,814 models were divided into a training set and a testing set, each with 907 models. The deep belief network (DBN) (Fig. 2.15) constructed by restricted Boltzmann machines applies a features matrix and classifies the models, adopting the effective training process. The process of training DBN is layer by layer. The results show that compared with several commonly used classification methods, the method

proposed in this paper achieved good results in the 3D model classification for efficient BIM.

[Bloch and Sacks, 2018] used ANN for the automatic recognition of house spaces. The aim of their research is the automatic semantic enrichment of BIM models. Starting from the collection of the dataset, 150 spaces (10 examples for each possible label), they then chose the AZURE ML platform to implement the network. Five features have been identified after running a filter based feature selection module in AZURE. Filter based feature selection is a process of applying statistical measures to identify the features that are most relevant to a specific classification problem. The five features finally isolated were: area; number of doors; number of windows; number of room boundary lines; and floor level offset. The set of five features selected using this procedure achieved a maximum accuracy of 82%. They also compared the ML approach with another approach: rule-inferencing. What came out was that with rule inferencing they were able to recognize only five different types of room out of the 15 total number. This is due to the fact that trying to identify univocal rule is not an easy process. Working with a pairwise comparison of the spaces they built a matrix of possible result values for each rule taken into consideration and most of them are marked with "x" that means "may or may not be true". This big uncertainty lead to the not very flexible results that can be obtained by the rule-inferencing method in comparison with ML.

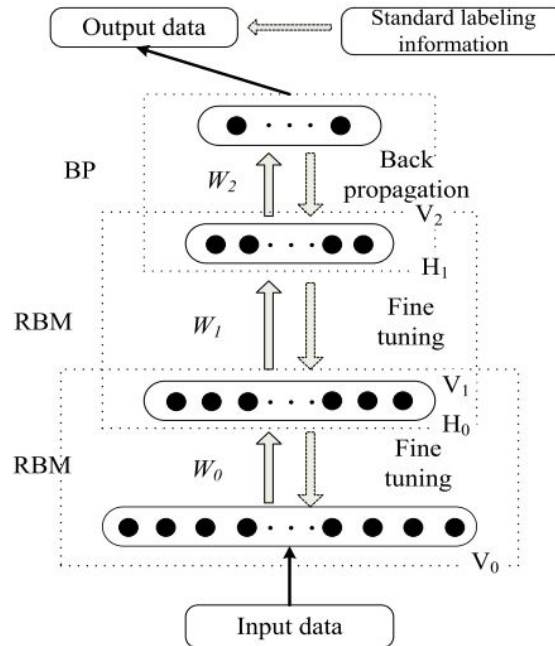


Figure 2.15: Structure of a DBN [Zhao et al., 2015].



Figure 2.16: Crack detection with neural network [Cosenza et al., 2018]

Some interesting applications of ML regards diagnostic issues such as the one from [Cosenza et al., 2018] who used semantic segmentation networks to develop a system able to recognize wall cracks on both stone and plastered walls (Fig. 2.16). Semantic Segmentation is the term that describes the process by which a ROI (region of interest) Label is associated with each pixel of the image that represents a label corresponding to a specific class. After having collected the relevant data to define a wall crack this research deals with modelling information through BIM approach. The aim of this research is the digitization of building current status. It is well known how expensive activities necessary to collect data on existing buildings are, both in terms of costs and in terms of human resources. This cost lead to the research for automation in these processes and object recognition is a valuable means for harvesting data just framing the environment.

Another growing field of action is the one regarding unmanned aerial vehicles. There are a wide range of applications for UAVs in the civil engineering field. A few applications include but are not limited to coastline observation, fire detection, monitoring vegetation growth, glacial observations, river bank degradation surveys, three-dimensional mapping, forest surveillance, natural and man-made disaster management, power line surveillance, infrastructure inspection, and traffic monitoring. As UAV applications become widespread, a higher level of autonomy is required to ensure safety and operational efficiency. Ideally, an autonomous UAV depends primarily on sensors, microprocessors, and on-board aircraft intelligence for safe navigation. For this reason there is a growing attention towards the automatic recognition of objects from UAV cameras.

This is the case of [Braun et al., 2019]. In their research a system for the automatic detection of formworks in construction sites is developed. Their input images are Unmanned Aerial Vehicle (UAV) photographs so as to integrate their work in a modern process of site inspection. They focus on two different image analysis tasks: image classification and object detection. Using a standard GoogLeNet CNN implemented in Caffe they developed the image classification task. To detect several formworks within an image of a construction site (Fig. 2.17), they used a CNN with DetectNet architecture, implemented in Caffe. They gathered 9,956 formwork elements, labeled manually on pictures of three construction sites.

Other studies focus on the differences in image perspective from UAV

[Radovic et al., 2017]. Having the right dataset in fact is crucial for the efficient recognition of objects. Considering that the perspective from a UAV is quite different from the normal capturing point of pictures it is evident the need for a specific dataset (Fig. 2.18). In their study [Radovic et al., 2017] focus on the recognition of planes from a top view. They collected a new dataset. These images consisted of a variety of airplane types and a wide range of image scales, resolutions, and compositions. The total number is 267 images containing a total of 540 airplanes. The CNN chosen, a YOLO network, was able to recognize “airplane” objects in the data set with 97.5% of accuracy. This study wants to pose the basis for commercial and military applications of object recognition from UAV for instance in transportation-related projects, construction site management and infrastructure asset inspections.



Figure 2.17: Detected bounding box for formwork elements on a photography of a construction site [Braun et al., 2019].



Figure 2.18: Airplane aerial images for object detection with YOLO [Radovic et al., 2017].

As stated before also safety is a field where the advantages of automatic detection technology can play an important role. Studies can be found on the automatic detection of individual protection systems in construction sites. [Fang et al., 2018] propose a system which uses surveillance cameras mounted in construction sites to check the correct use of hardhat (Fig. 2.19). More than 100,000 construction worker image frames were randomly selected from the far-field surveillance videos of 25 different construction sites over a period of more than a year. A total of 81,000 images from this dataset were randomly selected to comprise the training dataset. The research trains Faster R-CNN chosen for the mean precision values that are higher than other networks. They trained the network taking into consideration different parameters: impact of visual range, impact of weather, impact of illumination, impact of individual posture, impact of occlusions. For each of these parameters they tested precision, recall and missing rate, and the research proved that the network reaches good results in all cases (precision always higher than 90%). The experimental results demonstrate that the high precision, high recall and fast speed of the method can effectively detect construction workers' non hardhat use in different construction site conditions, and can facilitate improved safety inspection and supervision.

Methods like this offer a significant opportunity to contribute to real-time site monitoring and improve the safety management processes.

Enabling real-time application requires on one hand the possibility of having an embedded system performing processes directly on site. On the other hand the speed at which some tasks are accomplished marks the difference between

something that is usable in real-time, leading to a man-machine parallel working, and the impossibility of using some systems for their scarce efficiency.

Object detection based on deep learning is a valuable means for real-time operations in the aspects of speed and accuracy. [Tao et al., 2018] developed a system called OYOLO, which stands for optimized YOLO. Inspired by Fully Convolutional Network (FCN), in order to simplify the structure of the convolutional neural network in YOLO and reduce the amount of computation to make object detection process faster, they remove the last two fully-connected layers, and add an average pool instead (Fig. 2.20). They trained the network for objects in the traffic scene (car, cyclist and pedestrian) and they add some images captured by roads' cameras into KITTI data set to get their own data set. According to studies that declare the optimal ratio of training data and testing data is 9:1, they divided the dataset in 8100 images in training set and 900 images in testing set. In the end the system the developed, the OYOLO system performs 44ms per image, which is faster than YOLO by 18%.

As example of using these methods one can suggest different security video systems which aims to detect violators in restricted areas, video monitoring in airports, public spaces, working places, offices etc. Human monitoring, in fact, is another skill that can be applied to a number of different tasks. However, where working with this kind of applications it became necessary to face problems connected with bad image quality, complex illumination conditions, PTZ cameras, sufficient enclosures of objects by the foreground, strict conditions imposed on computational and time resources given for particular algorithm [Molchanov et al., 2017].

A major research area for real-time application is represented by the prevention of accident in construction sites. Monitoring proximity between workers and equipment (or vehicles) enables the advanced detection of potential hazards, which allows for prompt feedback to involved workers. This proactive intervention can lead workers to prepare for evasive actions, thereby reducing the chance of an impending collision [Kim et al., 2019]. Further, systems that exploit computer vision can recognize multiple entities without installing any sensors and being a cost-effective and a non-invasive proximity monitoring while complementing existing sensing technologies. Kim et al. propose a system to check construction site (Fig. 2.21). Thanks to the mobility of Unmanned Aerial Vehicle (UAV) the monitoring of wide areas, not viable with conventional imaging devices such as surveillance or portable cameras, is enabled.

They chose a YOLO-V3 consisting of two main networks: the feature extractor and the object detector. The feature extractor network called darknet-53 has a deep architecture with successive 52 convolutional layers, which can extract fine-grained features from a coarse data. In particular, this network incorporates

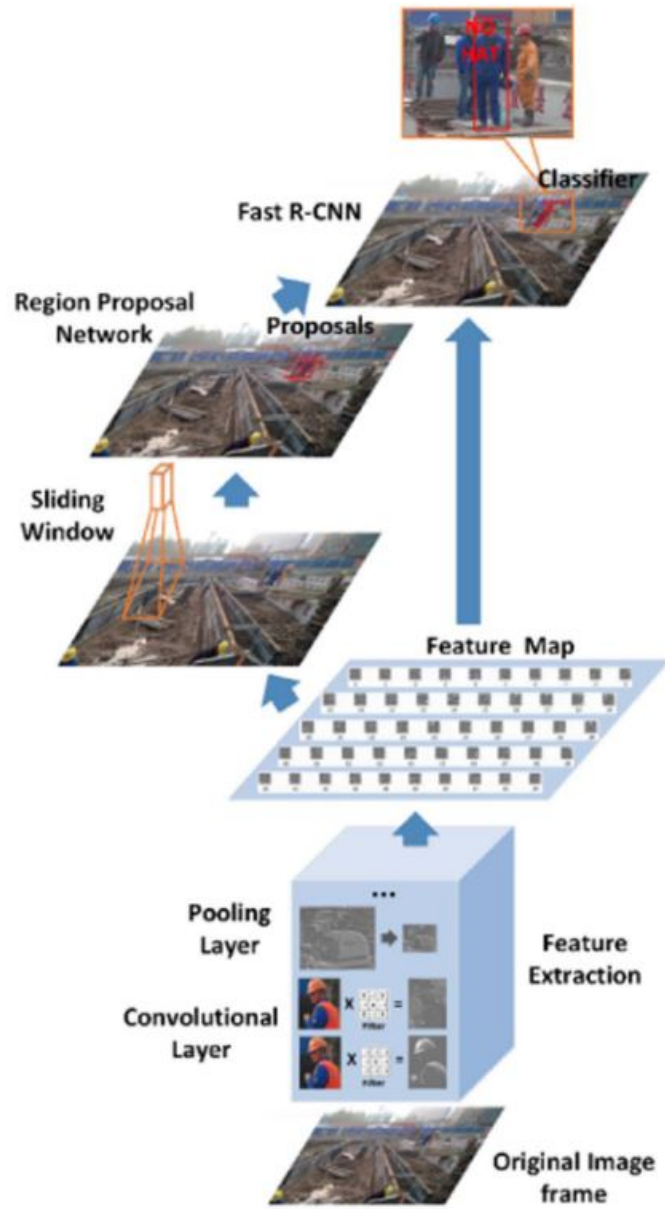


Figure 2.19: Not hardhat use detection method framework [Fang et al., 2018].

layer name	filter	size/stride	input	output
0 conv	64	7×7/1	448×448×3	448×448×64
1 maxpool		2×2/2	448×448×64	224×224×64
2 conv	192	3×3/1	244×224×64	244×224×192
3 maxpool		2×2/2	244×224×192	112×112×192
4 conv	128	1×1/1	112×112×192	112×112×128
5 conv	256	3×3/1	112×112×128	112×112×256
6 conv	256	1×1/1	112×112×256	112×112×256
7 conv	512	3×3/1	112×112×256	112×112×512
8 maxpool		2×2/2	112×112×512	56×56×512
9 conv	256	1×1/1	56×56×512	56×56×256
10 conv	512	3×3/1	56×56×512	56×56×512
11 conv	256	1×1/1	56×56×512	56×56×256
12 conv	512	3×3/1	56×56×256	56×56×512
13 conv	256	1×1/1	56×56×512	56×56×256
14 conv	512	3×3/1	56×56×256	56×56×512
15 conv	256	1×1/1	56×56×512	56×56×256
16 conv	512	3×3/1	56×56×256	56×56×512
17 conv	512	1×1/1	56×56×512	56×56×512
18 conv	1024	3×3/1	56×56×512	56×56×1024
19 maxpool		2×2/2	56×56×1024	28×28×1024
20 conv	512	1×1/1	28×28×1024	28×28×512
21 conv	1024	3×3/2	28×28×512	14×14×1024
22 conv	512	1×1/1	14×14×1024	14×14×512
23 conv	13	3×3/1	14×14×512	14×14×13
24 avgpool		2×2/2	14×14×13	7×7×13

Figure 2.20: OYOLO Network structure [Tao et al., 2018].

residual skip connections in the intervals of two convolutional layers. The object detector makes detection. The uniqueness of this network resides in its ability to achieve detection at three different scales, thereby improving scale invariance. The total dataset of 4,512 frames capturing construction workers and equipment were extracted from construction site videos and labeled. Of these, 4,114 images were used for the fine-tuning and the other data, 398 con-

secutive images were used for testing. This test considered the three types of object classes: construction worker; wheel loader and excavator. For the proximity measuring this research homogenizes the 3rd coordinates of points, thereby making distance measuring possible on a 2D image with a minimum computation. Along this way, this method leverages a reference object whose dimension is already known (e.g., a column foundation). After the rectification, the proximity can be measured in a metric unit, and the struck-by hazard can be visualized considering the unique scene scale. To evaluate the proposed method's accuracy in real-world applications, this research conducted two tests on real-site aerial videos. The first tests the ability for mobile construction entities to work a normal operation whereas the second test targets stationary entities in a controlled environment. It results that it was a challenge to measure the proximity on the field without interrupting the site operations, while also facing additional barriers to implementation.

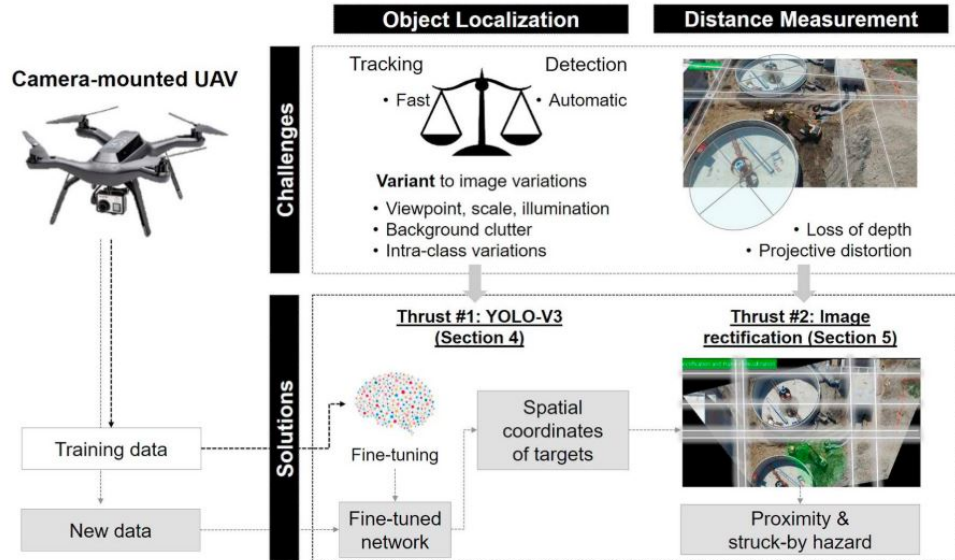


Figure 2.21: Proximity monitoring proposed system [Kim et al., 2019].

At the same time tests on real-site aerial videos showed a promising performance of the proposed method; the mean absolute distance errors were less than 0.9m and the corresponding mean absolute percentage errors were around 4%. Another issue, related with the real-time management, is the computational efficiency in proximity monitoring as it ultimately aims at timely intervention. The use in this research of a graphic process unit server with a 0.278s performance lead to the consideration that this is not sufficient for real time emergency management taking into consideration the construction site vehicles average speed. For this reason embedded systems capable of working data directly on site could prove to be faster and more efficient for real-time applications.

2.4 Mixed Reality

2.4.1 Definition of Mixed Reality

The aim of developing real-time applications can find a valuable support in the extended reality technologies. The arrival of Virtual-Reality, Augmented-Reality, and Mixed-Reality technologies is shaping a new environment where physical and virtual objects are integrated at different levels. Recent technological developments are changing the ways people experience the physical and the virtual environments. Specifically, Virtual Reality (VR) is likely to play a key role in several industries [Berg and Vance, 2017], such as retail [Bonetti et al., 2018] [Kerrebroeck et al., 2017], tourism [Griffin et al., 2017], education [Meißner et al., 2017], healthcare [Freeman et al., 2017], entertainment [Lin et al., 2018] and research [Bigne et al., 2016]. Recent reports show that sales of VR Head-Mounted Displays (HMD) have, for the first time, exceeded one million in a quarter [Canalys, 2017]; the value of VR devices sold is expected to increase from US\$1.5 billion in 2017 to US \$9.1 billion by 2021 [CCSInsight, 2017]. The never stopping releases of standalone Virtual Reality (VR), Mixed Reality (MR) Head Mounted Display (HMD) together with the declining prices of these devices, will determine the huge increasing usage of VR/MR.

The Real Environment is an actual setting where users interact solely with elements of the real world, whereas Virtual Environment is a completely computer-generated environment where users can interact solely with virtual objects in real-time. Between these extremes, we found technology-mediated realities where physical and virtual worlds are integrated at different levels. Some of these cutting-edge technological devices are not only smaller and portable, they are also wearable [Dieck et al., 2016] [Tussyadiah et al., 2018] and, in some cases, are integrated into the human body. These technologies are included in the users' personal space to improve their experiences and extend their sensory, cognitive and motor functions. According to Milgram and Kishimo [Milgram and Kishimo, 1994] the conventionally held view of a Virtual Reality (VR) environment is one in which the participant-observer is totally immersed in, and able to interact with, a completely synthetic world. Their definition of

Mixed Reality is the continuum in the middle between two extremes, the 100%

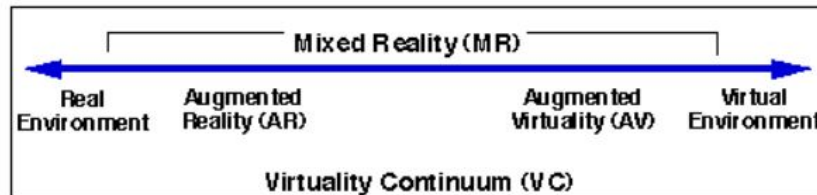


Figure 2.22: Virtuality Continuum Spectrum by Milgram and Kishimo.

virtual world and the 100% real world (Fig. 2.22).

Augmented Reality (AR) is then defined as any case in which an otherwise real environment is "augmented" by means of virtual objects. The system can also differentiate in immersive and non-immersive. Non-immersive systems are the simplest and cheapest type of VR applications that use desktops to reproduce images of the world. Immersive systems provide a complete simulated experience due to the support of several sensory outputs devices such as head mounted displays (HMDs) for enhancing the stereoscopic view of the environment through the movement of the user's head, as well as audio and haptic devices. In between semi-immersive systems such as Microsoft HoloLens.

Semi-immersive virtual reality refers to a specific type of VR that allows users to experience virtual three-dimensional environments while remaining connected to real-world surrounding visuals, auditory, smells, and haptics as well as keeping control over physical objects. With semi-immersive VR, you can see what's going on around you and interact with the objects you need.

[Flavián et al., 2019] tried to offer a better understanding of these concepts and integrate technological (embodiment), psychological (presence), and behavioral (interactivity) perspectives to propose a new taxonomy of technologies, namely the "EPI Cube", a three-dimensional cube which can classify all the current and potential new reality-virtuality technologies. In his theory of human-technology mediation, Ihde [Ihde, 1990] regarded embodiment as situations in which technological devices mediate the users' experience and, as a consequence, the technology becomes an extension of the human body and helps to interpret, perceive and interact with one's immediate surroundings. Presence is defined as the user's sensation of being transported to a distinct environment outside the real human body. Thus, they concur with previous research and consider the technological quality of the media as immersion (as a part of technological embodiment) and the psychological perception of the user as the sense of presence [Slater, 2003] [Thornson et al., 2009]. Interactivity is finally defined as the users' capacity to modify and receive feedback to their actions in the reality where the experience is taking place [Carrozzino and Bergamasco, 2010] [Muhanna, 2015]. They focused on what Hoffman and Novak [Hoffman and Novak, 1996] called human-machine interactivity, where the participants interact with the mediated

environment, which responds according to their actions.

Among all the revised taxonomies, the “Reality-Virtuality Continuum” proposed by Milgram and Kishino [Milgram and Kishino, 1994], has been the starting point for researchers to classify the wide variety of realities. Real Environments (RE) encompass the reality itself. MR was conceived as the different points of the continuum at which real and virtual objects were merged. Consequently, Augmented Reality (AR) and Augmented Virtuality (AV) are part of MR. MR must no longer be the broad part of the continuum that includes AR and AV, as noted by Milgram and Kishino (1994). It should be regarded as an independent dimension falling between AR and AV and characterized by the total blend of virtual holograms with the real world (Fig. 2.23). Thus the Reality-Virtuality Continuum had been adjusted by differentiating the independent dimension of Pure Mixed Reality (PMR) (Fig. 2.24).

Virtual content in PMR is not superimposed on the physical environment (as in AR) but virtual objects are rendered so that they are indistinguishable from the physical world. Visual coherence is a basic element of pure mixed reality [Collins et al., 2017]. Users can interact with both virtual and real objects in real-time and, simultaneously, these objects can interact with each other. This “environment awareness” implies that not only virtual objects can act in the real environment, but real objects can also modify the virtual elements, regardless of where the experience is taking place. For instance, in a pure MR, users would not be able to see a virtual box under a table unless they bent down to look at it; in an AR, the box would be overlaid and it would be unnecessary to bend down. Currently, the only technological developments that can truly be considered to be generating pure mixed realities are the holographic devices Microsoft HoloLens and the upcoming Magic Leap, which integrate virtual and real objects in a real-time display.

Reality – Virtuality Spectrum

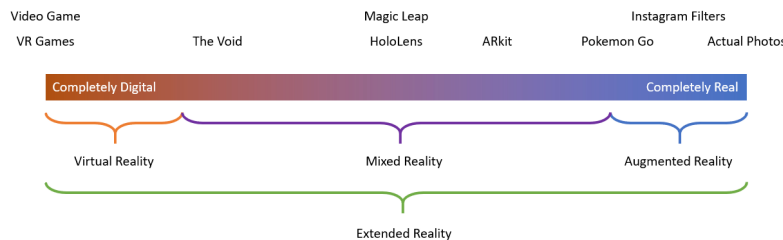


Figure 2.23: New propose for virtuality continuum spectrum.

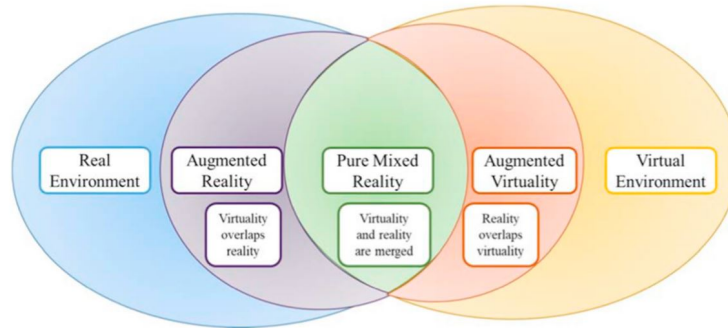


Figure 2.24: Pure mixed reality explanation [Flavián et al., 2019].

2.4.2 Uptake of MR to real-time problems

The real-time is a valuable feature that can be exploited in several fields, especially the ones related to safety and on-site applications. As far as safety is concerned [Kim et al., 2017] developed an application that exploits MR for proactive accident prevention (Fig. 2.25). Their system show safety information by means of a wearable device directly to site managers or worker. This safety information is derived from an image-based safety assessment system. The system comprises of three modules: vision-based site monitoring, safety assessment, and hazard information processing and visualization modules. In the monitoring module, global perspective images are acquired and utilized to locate moving objects in the construction site so that the spatial relationship of objects can be determined for calculating the safety information. The vision-based site-monitoring module collects raw image data using a stationary camera, specifically, a closed circuit television (CCTV), and a wearable device. On the basis of the images from the construction site, multiple objects are tracked using a motion-based object-tracking algorithm. Global perspective images captured from the CCTV are used to track multiple moving objects so that hazard information can be generated. Using images and the orientation data acquired from the wearable device, the location and heading direction of the worker are extracted. Then fuzzy inference-based safety assessment method is used to evaluate the safety of objects. The input variables are the proximity and crowdedness of each object, which are derived from the vision-based site-monitoring module.

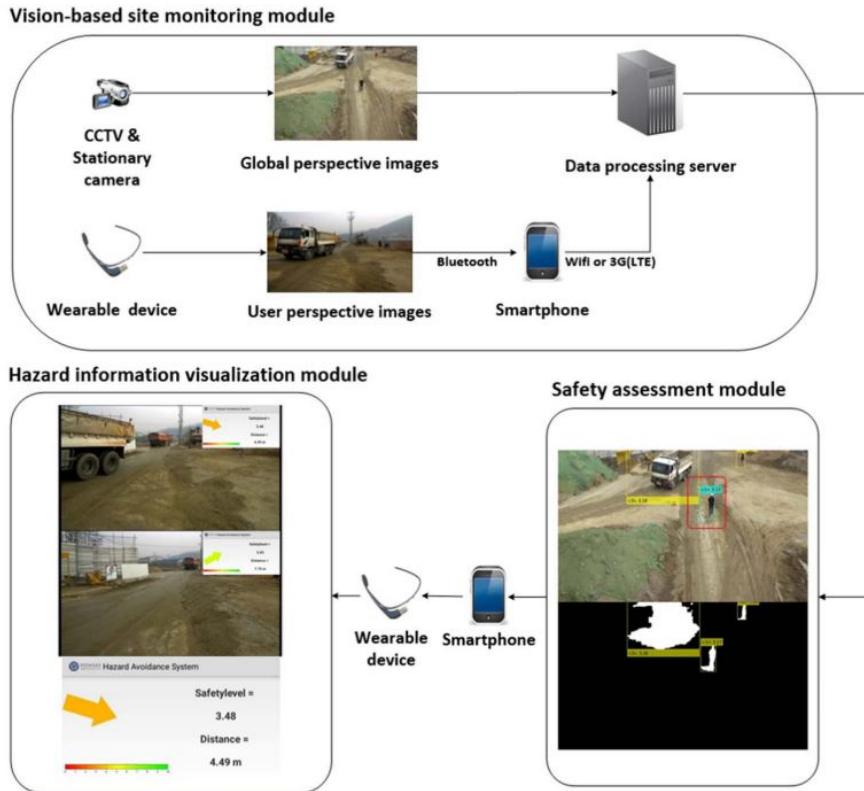


Figure 2.25: hazard avoidance system [Kim et al., 2017].

The output variables are the safety levels of each object, which range from zero (the most dangerous) to 10 (the safest). The visualization module displays the safety information in the form of augmented reality in the wearable device. Google Glass was utilized as the wearable device to capture the user perspective images.

Other application of the real-time management of data with MR is the construction site inspection.

[Zhou et al., 2017] proposed a MR application for rapid inspection of segment displacement during tunneling construction. The result shows that all inspections and analyses can be conducted on-site, in real-time, and at a very low cost. The live on-site scene can be captured by video camera, and the global coordinate and virtual camera coordinate can be acquired by a tracking sub-system. A virtual baseline model (3D CAD drawing) would be superimposed over an onsite image in real time. Then, the combined scene will be conveyed to the end users by a display subsystem. A widely used open source AR software package called ARToolKit™ was used in this project to develop the on-site segment displacement inspection system. The display of the displacement between two

segments starts from a marker placed beside one of the segments. This leads to the registration of a virtual model over the real displacement. The virtual models for the segment displacement are created in AutoCAD with the same dimensions as the designed shield segment and its threshold. Those models were used as the baseline models. Then, the CAD files were converted to VRML files to be loaded in the AR system.

Ammari and Hammmad [Ammari and Hammad, 2014] at the same time proposed a framework for a collaborative BIM-based Markerless Mixed Reality, focusing on the advantage of not having markers inside buildings. The framework integrates CMMS, BIM, and video-based tracking in a setting to retrieve information based on time and the location of the user, visualize maintenance operations, and support collaboration between the field and the office to enhance decision making (Fig. 2.26). With this system the detection and marking of a defected building element starts with the inspector walking around the facility for routine or scheduled inspection. After the inspector locates a defect, the broken element is located and its ID is shown within the AR scene. BIM3R then, gives the possibility of updating the BIM model with the information retrieved on site like: 1. type of the defect, 2. severity of the defect, 3. excepted consequences, and 4. any other notes or observations.

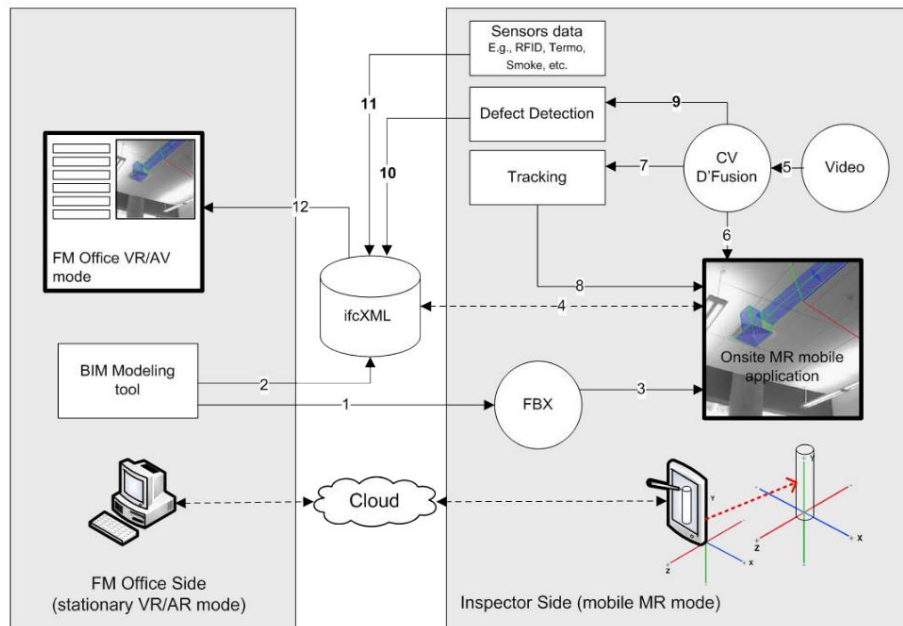


Figure 2.26: BIM3R system architecture [Ammari and Hammad, 2014].

All these data will be saved in XML-based database. BIM3R was developed using Autodesk Revit 2013 for BIM modeling, Unity3D 4.2 for system development, and D’Fusion Studio v 3.26 for creating markerless tracking scenarios. The BIM-based modeling tool (e.g., Autodesk Revit) is used to create the 3D BIM model, then converted to FBX format, supported by Unity3D Engine and with sufficient building information for the purpose of visualization and data retrieval. At the same time they saved the same BIM model as ifcXML file, using it as a database to facilitate the data entry for the FM inspection tasks. After that the two-way interaction relationship between the FBX file within the user mobile application and the ifcXML model located on the server side is necessary to update the scene with any added information collected on-site. Video streaming data (frames) are processed using computer vision (CV) and scenario manager tools within D’Fusion studio to create the tracking scenarios. Information is added to the database using XML parser leading to the possibility of augmenting the building model and sharing information in real-time.

2.4.3 The use of Mixed Reality for building oriented applications

This real-time on-site display potentiality of MR can represent valuable features also in the construction industry.

There are specific tasks that can benefit the most from this new technology. First of all the superimposed digital representation of the project can be used for design verification [Zhou et al., 2017][Kopsida and Brilakis, 2016]. Most projects still rely on 2D drawings, While Mixed Reality (MR) could theoretically be the primary means of communicating design content to on-site personnel in 3D through BIM methodology. The use of MR for design communication has been studied through several past efforts. In the construction industry, Feiner was the first to combine 3D Head Mounted Displays (HMDs) with mobile computing technologies, creating a prototype that overlaid campus information on top of an unobstructed view of a university campus [Feiner et al., 1997]. MR’s potential as an onsite model visualization tool has also been well studied. It has been used to visualize a 3D building model in its physical location [Honkamaa et al., 2007] [Kopsida and Brilakis, 2016] and objects hidden behind other existing structures [Smailagic and Siewiorek, 2004].

[Chalhoub and Ayer, 2018] proposed a study on the on-site display of electrical conduit through holograms (Fig. 2.27). The tool they chose for MR implementation is Microsoft HoloLens. They involved in the study a construction company working with traditional procedures for the transmission of design document to the construction site operators. Eighteen industry professionals participated in this study, including shop electricians, managers, and site electricians. Half of the participants had less than 1 year of experience assembling electrical conduit, and eight of the participants had not assembled conduit in the past year. The researchers aimed to compare the performance of each participant when using paper, and when using MR for design information delivery. The researchers

used a double-counterbalanced experimental design to make this comparison. Two conduit designs were engineered for this research. Both designs used the same prefabricated pieces in different order and orientation to create two unique conduits. This ensured that no participant would assemble the same conduit in both attempts, while ensuring that the assembly difficulty levels were comparable. If this approach had not been used and a participant would have assembled the same conduit twice, once using paper and once using MR, their performance could have been impacted by what they learned during their first attempt. Moreover, if all participants started with one information delivery method, the results could be subject to an order-induced error. Therefore, the researchers also varied which mode of visualization was provided to a participant first. Three key behaviors were identified to measure the performance of the participants, and enable direct comparison between the use of paper plans and MR for conduit assembly: (1) duration to assemble conduit, (2) duration looking at information and (3) duration to place conduit. The research proved a significant reduction in time according to all these three indicators. Furthermore in this study they also assessed the quality taking into consideration the information delivery related to mistakes and rework. Two metrics were used in this case: (1) the total number of mistakes; and (2) the total count of correct final assemblies. The results were the total number of mistakes reduced by 75%, but more importantly, reduced the amount of rework required by 72%. This research work helps to demonstrate the benefit that MR can offer for reducing construction errors. Moreover, after a questionnaire conducted at the end of the experience, it results that expectancy in the benefit of MR in the construction industry is high with more than the 50% of participants who agree or strongly agree on sentences related to the advantages in the use of this new technology.

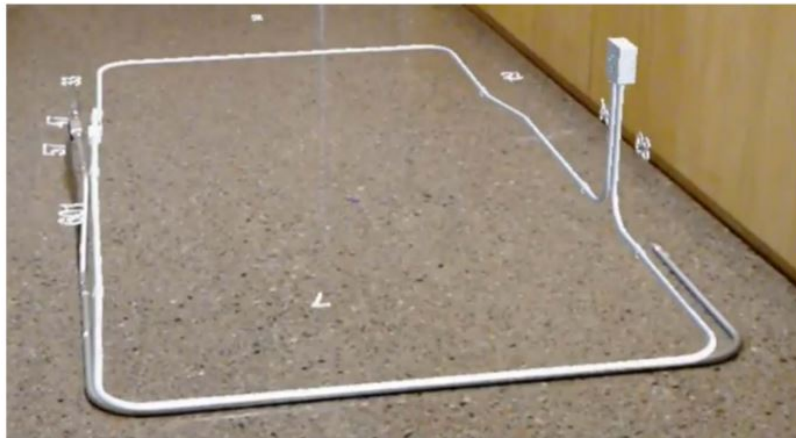


Figure 2.27: Example of on-site visualization with Mixed Reality [Chalhoub and Ayer, 2018].

Design verification also lead to proactive rather than reactive defect management plan, saving useless waste of time and money. [Park et al., 2013] developed a marker-based AR method to detect defects in buildings. Starting from a 3D model designed using ArchiCAD then it is converted to a WRL file, a virtual reality modeling language. A marker containing the 3D design model is made via a marker generator program, then the marker was attached on a specified location. At this stage it is possible to read the BIM information in the marker and to augment them onto real work place through mobile devices such as smartphone or tablet PC. The augmentation consists of a picture taken by the operator. This is compared with a 2D image generated from 3D BIM model to prevent and check the possible errors, for instance of the door and window openings as in Figure 2.28. The field application potentials of the AR and image-matching applied inspection system can be well understood with this research. Systems like this one are expected to enable managers to inspect and control worker's job performance more efficiently and also workers to confirm their works more readily leading to a reduction in errors and therefore extra time and unexpected cost.

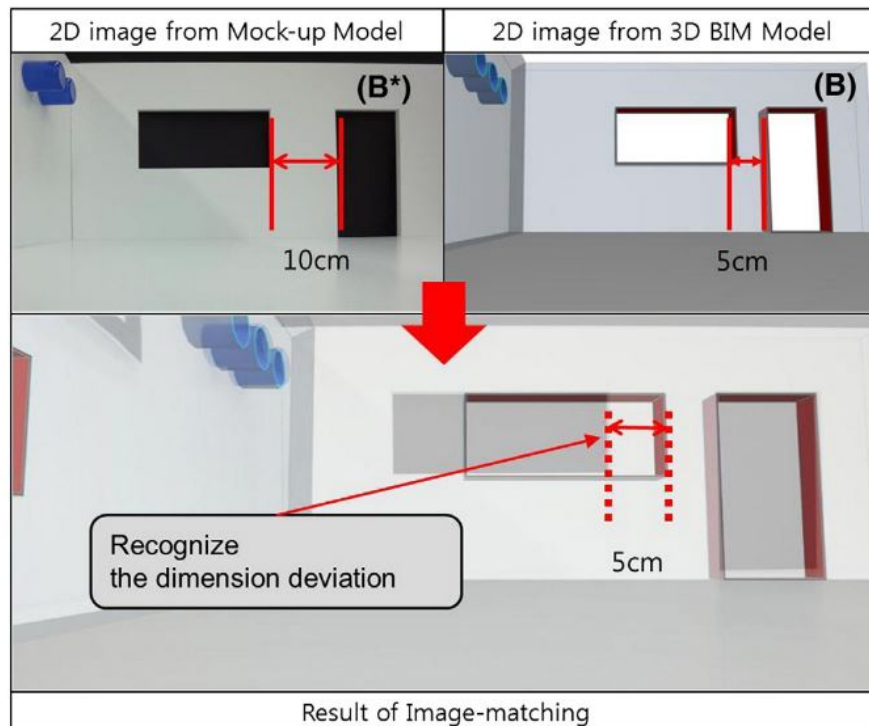


Figure 2.28: Example of as-is model checking [Park et al., 2013].

Augmented Reality have been used also to facilitate the inspection process. AR based systems can simplify and reduce the time of inspection by providing the inspector with instantaneous (real-time) access to the information stored in the Building Information Modelling (BIM). However, since precise alignment between the BIM model and the real world scene is still a challenge Kopsida and Brilakis compared three methods for estimating the position and orientation of the user [Kopsida and Brilakis, 2016]. Till now, for estimating the position and orientation of the user, methods have been proposed that either use markers or confine the user to a specific location, or use Global Positioning System (GPS) which cannot operate efficiently in an indoor environment. The aforementioned study presents an evaluation of different methods that could potentially be used for a marker-less BIM registration in AR. Project monitoring is conducted by visual inspections and the inspector needs to fill several forms, write reports and perform extensive information extraction from drawings. Interior inspections can be even more complex (e.g. installation of Mechanical, Electrical, and Plumbing, etc.). They compared 3 groups of methods. The first group uses 2D images taken from mobile devices and the model based AR framework, the second group uses 3D and pose estimation data acquired from 2D images using the LSD and ORB SLAM methods, and finally, the third group uses 3D and positioning data acquired from Microsoft Kinect and Google Project Tango which can provide RGBD data directly. In the end they stated that compared to the Kinect sensor, Project Tango offers a more robust motion tracking and although the 3D reconstruction is noisier than Kinect, it can capture larger scenes and operates more quickly, providing real-time advantages for AR inspection implementations required on busy construction sites.

The second task which can benefit from the real-time superimposed information provided by MR is the management of work orders also during the FM phase. Furthermore during operation speed and accuracy with which decisions can be made in dynamic environments can be the difference between success and catastrophic consequences, besides the efficiency improvement. According to this [Irizarry et al., 2014] presented a scenario for the integration of augmented reality (AR) and building information modeling (BIM) to build an ambient intelligent environment for facility managers where mobile, natural, user interfaces would provide the users with required data to facilitate operations. In this paper an ideal Ambient Intelligent environment has been proposed in which healthcare facility management operational data requirements would be automatically fetched from BIM databases of the facility, and would be augmented as a layer of information over the real world view of the facility manager. Since facility managers are often required to relate physical objects to database-like text-based information, AR represents a good candidate to aid facility managers with their routine tasks because their live view of a space could now be supplemented by the database information they need, all in one interface. Moreover, since facility managers are constantly moving through the spaces they manage, having a portable, mobile device would be beneficial if they were to employ AR in their tasks.

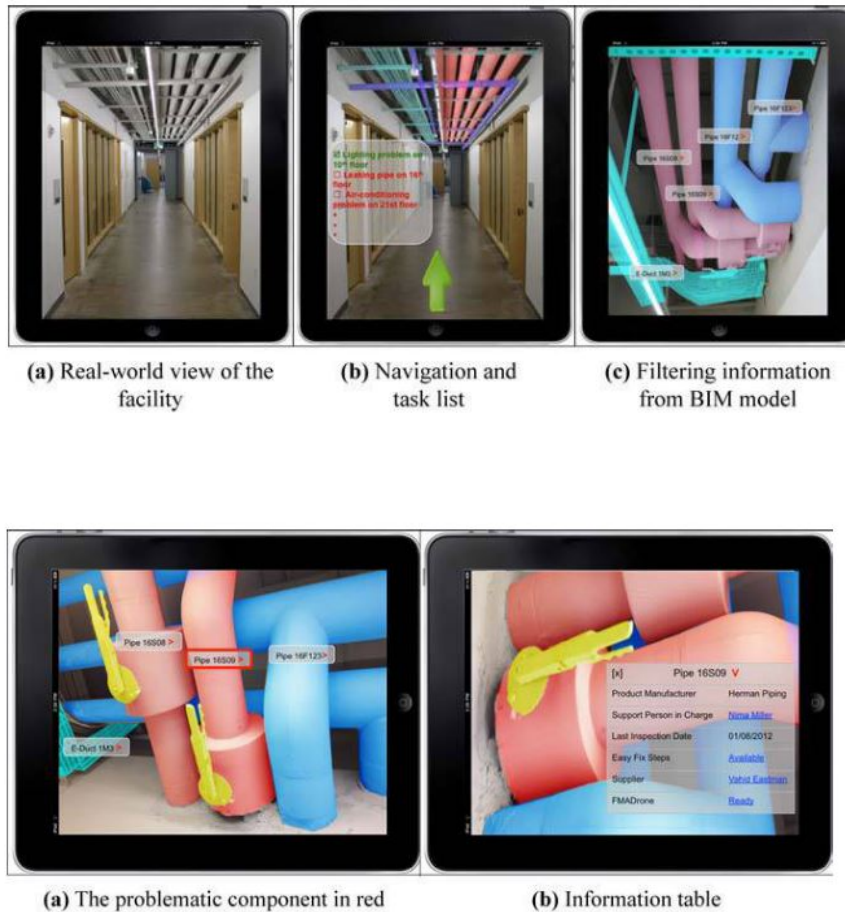


Figure 2.29: Example of information displayed on site through Mixed Reality [Irizarry et al., 2014].

As shown in Figure 2.29 only the specific part of the information that is required for performing the current task is displayed in the user interface. All this information is provided from the BIM model of the facility that is the main data repository of all the objects in the facility. A set of augmented visual steps for fixing the leaking problem would be displayed on the tablet. The system is BIM-based but it is also sensitive to changes in the environment and uses image recognition to report those changes.

This system is thought to be built on a framework (based on BIM or some other technology) that not only has been populated with all the required information but also possesses the capability to be constantly updated to provide real-time visual geometry guidance in the form of augmented reality. At the

same time the European project INSITER focused on the support given to planning processes as well as production and construction workflows by interactive MR visualization solutions and prototypes [Riexinger et al., 2018]. This project proposes the utilization and development of MR solutions, which connect the virtual and physical environment for self-inspection and self-instruction. Self-inspection functionalities should enable and encourage workers and stakeholders on site to check their own working processes and the results respectively, both individually as well as in collaboration with others. Self-instruction features on the other hand provide interactive guidance to any actors on site during their working processes, preventing incorrect actions and helping the workers to rectify any error immediately. The developed prototypes connect virtual models and digital planning information based on Building Information Models with the physical building or production site, to provide relevant data for different stakeholders on-site. Digital planning data, such as 3D objects, BIM models and BIM-based simulations with all its parts and assembly workflows are going to be superimposed into the field of vision of the user to expand the perception of reality. Within INSITER, 4D step-by-step simulations of on-site construction processes and product assembly steps within production environments have been developed. After the creation of 4D simulations, the data can be made available via export of the 4D simulations to the BIM or standardized file formats to become available for self-instruction with the help of MR. The INSITER IT environment includes four different layers (Fig. 2.30):

- the acquisition layer represents the gathering of on-site information.
- the adoption layer that will transfer the information into appropriate formats to be stored on the INSITER BIM platform.
- the BIM layer which contain the BIM platform, combination of PostgreSQL database, an Open BIM server and a SharePoint server.
- the application layer provides the whole INSITER toolset to interact with the collected information to create benefits

Part of the applications are MR or AR solutions for the in-situ visualization of digital planning or process data. Within this MR use case application, a guided process workflow and assembly process with detailed and dynamic work instructions has been developed and implemented to be used by assembly and construction workers on site, see Fig 2.31. Enhanced MR visualizations, including assembly process information, allow a high flexibility and productivity, supporting the error-free assembly of components not only in the manufacturing domain but also in the field of building construction [Riexinger et al., 2018]. With this method construction workers can be provided with detailed 4D BIM-based on-site instructions. Within the described MR use case, a BIM-based simulation of construction or refurbishment processes has been developed and integrated.

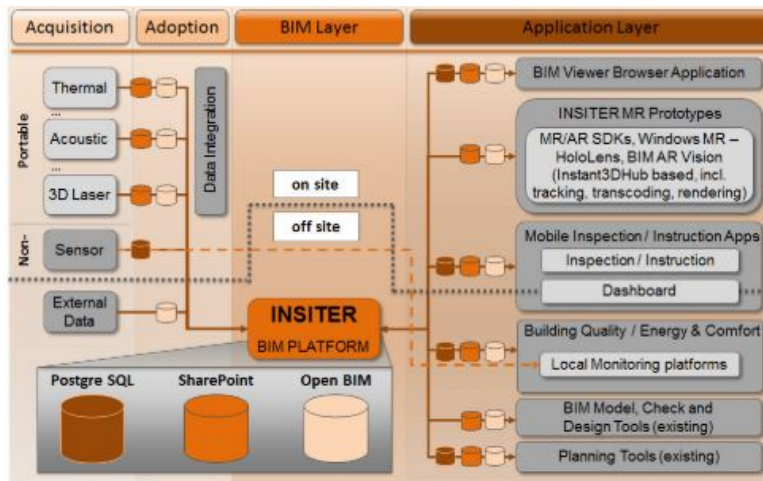


Figure 2.30: INSITER system structure [Riexinger et al., 2018].

The developed MR prototype enables detailed 3D scenes evaluation for any production or construction element. Workers on site can visualize in-situ what elements have e.g. to be attached, installed or removed for refurbishment works or installations. Also project managers can monitor the installation work on site and check if new machines, construction elements or MEP systems are correctly installed.



Figure 2.31: Example of work orders displayed [Riexinger et al., 2018].

Furthermore, one of the increasingly important application areas of AR is in training and education (Fig. 2.32). This kind of BIM-AR integrated environment can be used for education purposes and training of the facility management practitioners as well as students. Simulation can be used to illustrate concepts and provide exercises that allow the learner to train in a realistic environment. Training-based scenarios can be defined for different maintenance-related tasks and trainees can be walked through performing them.

[Wang et al., 2017] proposed a training system called Augmented Reality operator training system (Fig. 2.33). With their system they try to overcome the limits of the actual process of information retrieval such as not finding the right information in a timely manner or the critical information that exists on different platforms (e.g., paper-based specification, standalone-computers) characterized by a lack of mobility. The technology platform that has been conceptualized has a total of seven separate modules.

1. Remote Server Module: Remote server module consists of several experts and a central database server.
2. Identification Module: This module uses RFID technology to identify equipment.
3. Display module: ARvision-Stereo HMD, a prototype manufactured by Trivisio Company, is used for display. A computer-integrated haptic device can provide force feedback to the operator, which along with the force-control algorithm can enable the novice to feel the simulated force.
4. Mobile Computer Module: This module is located on a high-performance, light-weight laptop and consists of a local database, RFID antenna with reader, and AR software.

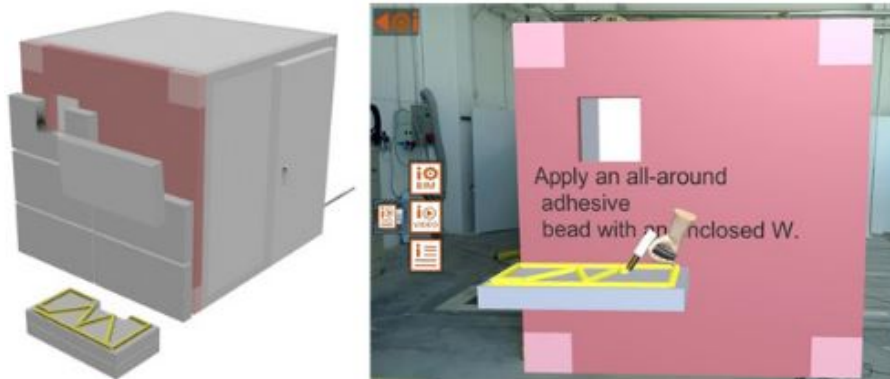


Figure 2.32: Training information shown directly on-site [Riexinger et al., 2018].

5. Tracker Module: AR EMS requires a means to track the user and equipment's locations within the construction site. The panel tags, static objects, and dynamic objects require multiple-trackers to be combined.
6. Input module: This module handles manipulation of the displayed digital content (e.g., clicking hypertext to browse details) and annotates the comments. A data glove is used to manipulate a virtual 3D cursor and yet not negatively alter equipment operation.
7. Representation module: Three types of digital content are defined—panel tags, static object, and dynamic object.

The local database can be accessed by Sybase SQL anywhere as the database query. All of the management data can be stored in the Sybase database. Technical imagery and data about the equipment fleet should be collected in order to compile a comprehensive, equipment-specific database .

MR has been shown to enhance the spatial abilities among students [Eh Phon et al., 2014] [Dünser et al., 2006]. MR was also used to teach engineering students the relationship between 3D objects and their projections in engineering graphic classes [Chen et al., 2011] and allowed students to better understand the construction site by site condition simulation [Shanbari et al., 2016] [Issa and R.A., 2014]. MR was also used for workforce training purposes. Wang and Dunston designed two MR training systems, one for operation and one for maintenance of heavy construction equipment [Wang et al., 2017] [Wang and Dunston, 2007]. MR was deployed to also train crane operators [Juang et al., 2013] and for providing spatially relevant data for training architects, construction crews and fireman on operation in large wooden buildings [Phan and Choo, 2010].

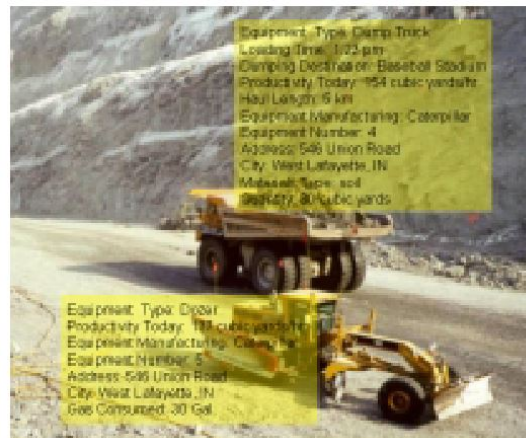


Figure 2.33: Information for site monitoring of working equipment [Wang et al., 2017].

The work by [Chalhoub and Ayer, 2018] too demonstrates the value of MR for information transfer to new workers. At the end of their experiment, in fact, it was noteworthy to see that participants with no conduit assembly experience achieved the best times using MR, and they were also faster than the most experienced participants who used traditional paper plans.

Compared to real exercises, AR offers reduced costs and hazards and unlimited training conditions/scenarios. This is particularly attractive in heavy construction equipment training. For example, a computer-generated virtual stockpile or piping materials will incur only computational resource costs. In addition, AR allows the creation of unlimited training scenarios (e.g., various terrain, shapes and sizes of virtual stockpiles) providing flexibility without increasing costs. The safety advantage can be realized in many situations such as the simulated fall of elevated virtual materials or equipment tipping due to lifting errors.

2.5 Conclusion

At the end of the analysis of this state of the art some drawbacks in current surveying techniques are outlined. Despite the advent of BIM delineates new information requirements for a profitable building modeling it is still hard finding real digital twins. This may be partially attributable to the high costs related to the collection of data. Even the latest procedures defined by researchers still require long post-processing phase. In addition they easily lead to inaccuracies due to the impossibility of checking collected data real-time.

The potentialities of ML and object recognition through NN can represent a valuable solution for performing data interpretation immediately on-site. It came out from this literature review the assessment that YOLO neural networks are the most versatile for fast object recognition and customization, even if they have not been exploited in construction industry yet. Furthermore this technology would allow the identification of object type leading to a more straightforward harvesting of functional data going beyond component shape design on which previous research has often been focused.

The MR on the other hand with the capacity of displaying information on-site finds its best application for real time data management. This is even more true thanks to the interaction that technician can have with holograms supporting efficient building component survey.

The innovation proposed by this research lies in the possibility of performing all the surveying tasks directly on site. This leads to the possibility of avoiding post-processing phase for data interpretation. Furthermore this work focuses on the collection of functional data rather than geometric ones. It also refers to assets that are usually not taken into consideration by the majority of previous research but which represent building components most affected by operational management.

METHODOLOGY

3.1 Introduction

In this chapter methods and technologies supporting the system development will be described. Starting from the addressed issues of this research the specific needs satisfied by the different tools will be outlined.

As far as Neural Network is concerned a brief explanation of all the most widespread neural networks will follow. It follows the reasons for choosing YOLO neural networks and all the details for the accomplishment of the training process for custom recognition will be reported. The Darknet training framework installation will be explained, including the dataset creation and finally the setting for beginning the training phase. After these passages the validation procedure is outlined both using the computer, as a tool for testing the goodness of the network training, and the real-world one which includes the use of the whole embedded system.

Later in this chapter the reader will deal with Mixed Reality technology. The differences between Virtual, Augmented and Mixed Reality will be clarify. The holograms features and development have been reported. In the end, the choice of the MR tool is explained and its characteristics enlisted.

3.2 Use cases

3.2.1 Addressed issues

Existing buildings management requires extensive information requirements on a wide variety of different subjects. For this reason Facility Management personnel faces huge challenges in the retrieval of information. They usually on-site rely on paper-based blueprints or on their experience, intuition, and judgment in finding and locating building equipment such as HVAC systems and electrical, gas, and water lines, which are located in places not readily visible. Finding the right location of an equipment during operations is a time and labor consuming action especially with newly assigned personnel and/or when an outsourced FM group takes over responsibility for the facility, or when equipment has been replaced or removed without the awareness of the FM personnel in charge. The locating building components issue then becomes even more critical during an emergency when also real-time management of information can mark the difference between a good or bad resolution of the problem [Phan and Choo, 2010] [La Delfa et al., 2016]. Effective and immediate access of information would minimize time and labor needed for retrieving it and help avoid ineffective decisions made in the absence of information [Becerik-Gerber et al., 2011]. This context brings out the need for building functional models. Functional models can be digital building models including information crucial for specific management purposes that could be safety inside buildings, refurbishment action needs, operation and maintenance just to name a few. As a consequence it is emerging that the detailed survey of facilities components for the semantic enrichment of geometric 3D models is urgent. Anyway data are still often collected manually or by means of semi-automatic techniques for data collection that lead to long post-processing for data interpretation. On the other hand, however, data collection process as expressed before must be performed by expert technician and this made it costly and time consuming. The digitization that the construction industry is facing in recent years instead has led to a growing interest in one of the major benefits of this change: the automation of processes. With the aim of reducing cost and trying to move towards the digitization of processes this research proposes an advanced system for information elicitation and engineering survey. This has been done exploiting the advantages of three different technologies: Mixed Reality, BIM and Neural Network having the aim of reducing post-processing effort in the interpretation of data thanks to the automation of some processes and an efficient human-machine collaboration [Naticchia et al., 2019].

Since BIM is spreading as a standard for information management, the aforementioned semantic enrichment is expected to be conducted on the Asset Information Model (AIM) populated from the as-built BIM model which would play a beneficial role in many FM practices [Becerik-Gerber et al., 2011]. The core innovation proposed in this research lays in the integration of several hardware and software innovations into one system architecture:

- Neural Networks, in the form of a trained Deep Learning Neural Network which performs the recognition providing the operator with objects features and position;
- Mixed Reality (MR) which operates as an interface between the user and the digital information just created;
- BIM enrichment, finally data collected is translated into IFC format so as to be added to a BIM model;

The proposed system develops efficient human-machine collaboration, employing MR as a powerful medium between human, reality and data [Corneli et al., 2019]. The visually supporting information provided by the MR tool, the possibility of working on data directly on site and the portability of the system represent means for increasing efficiency in survey operations.

3.2.2 Automatic inventory/survey support

The creation of BIM models of buildings, especially when talking about complex structures, demands a lot of time and is also expensive. When it comes to existing buildings, a large amount of multidisciplinary information essential for efficient FM must be collected. This turns out to be a hard task, since usually as built documents do not exist or are unreliable. In addition to this, some information is difficult or impossible to find out [Scherer and Katranuschkov, 2017] [Oesau et al., 2014]. All these aforementioned issues refer to the availability of data that is a well-known problem especially in existing buildings [Yang and Ergan, 2017]. For all these reasons, the first use case of this system is the one regarding the automated acquisition of data (Fig. 3.1). In this first case the operator on-site sends data about geometric features and building components to the BIM database. With these data the BIM model can be updated. The innovative contribution of this research in the survey is trying to minimize the post-processing efforts. Not only does the proposed procedure want to make building surveys faster but it also assists in resolving the information availability issue. The support provided in this case by the MR tools resides in the possibility of automatically detecting information on site and being able to transmit it in real time to the database.

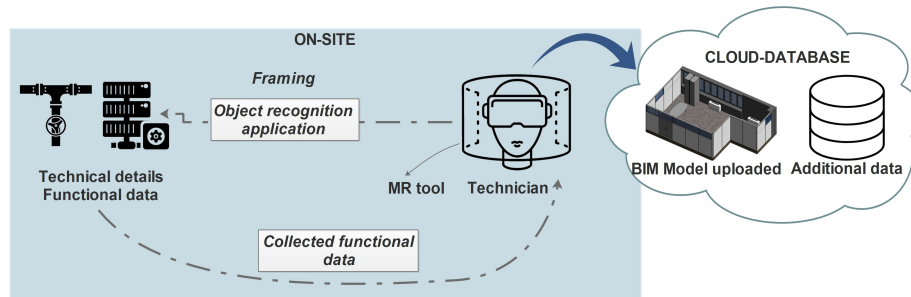


Figure 3.1: Automatic inventory/survey support process.

3.2.3 Diagnosis support

The second task this system could bring benefits to is that one of diagnosis performed on-site (Fig. 3.2). Correct and immediate localization of objects whose conditions need to be assessed requires considerable effort. The mixed reality display viewer avoids having to rely on paper documents, thus lowering the likelihood of introducing errors, helping in locating objects and providing all the necessary information displayed on-site. Moreover, the MR capability to overlap virtual reality to real things allows maintenance personnel to display attributes, technical properties and details (e.g. component type, last maintenance operation, etc.). The diagnosis task could receive a great support also from the on site visualization of causes analysis of defects. In this regard already exist in literature case based libraries like the one by [Motawa and Almarshad, 2013] which could be linked to building components and displayed on site. All the aforementioned data start from building information databases and can then be visualized thanks to a head-mounted display, directly on site.

3.2.4 On-site operation support

The last scenario is that of on-site operations support from expert technician. This scenario is very similar to the diagnosis support adding the fact that during

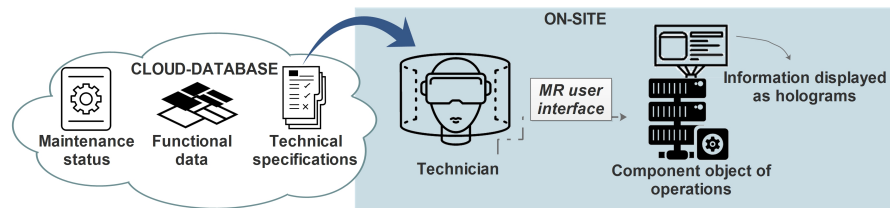


Figure 3.2: Diagnosis on-site support process.

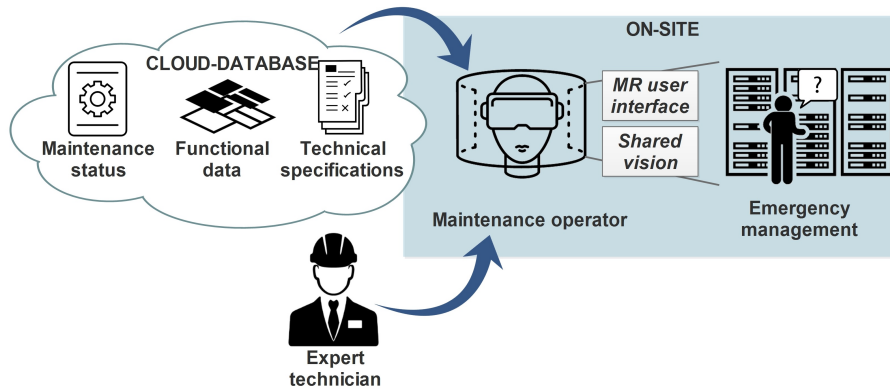


Figure 3.3: On-site critical operation support process.

critical operations the maintenance operator is given the possibility of consulting experts, who are off-site experts, in real time with the possibility of sharing his view too (Fig. 3.3). In the case of standard operations, procedures can be shown through the MR on-site. This initially implies a careful collection of the currently in force procedures. This information in fact is part of the background to be included in the database. Having the possibility to consult the standard procedures with this method on site, as well as reducing errors, also shorten training time for new personnel. However sometimes it is not possible to reduce operations to a standard procedure displayed as an object property. For instance, in case of emergency, a standard procedure to be performed is not always possible. These are the circumstances in which real time decision support can be fundamental for the good success of operations. Being advised by experienced personnel can be decisive and the powerful visualization capabilities of the digital viewer allow to share information and images in real time.

3.3 Neural Networks

Since the start of the 21st century, many businesses have realised that Neural Networks for computer vision will increase calculation potential. They span from the automotive industry for cars self-driving to language recognition, from face detection in social network to application that are able to recognize art styles. As another significant application of computer vision, image change detection plays an important role in not only civil but also military fields. The image detection has been widely employed in remote sensing, medical diagnosis, disaster evaluation, and video surveillance. The countless fields of application of recognition with Neural Networks make the research on this topic more fervent every day .

Companies struggle to stay ahead of the competition and these are some of the

bigger projects now on the market [Morrison, 2018]:

- GoogleBrain (2012) - a deep neural network created by Jeff Dean of Google, which focus on pattern detection in images and videos.
- AlexNet (2012) - this won the ImageNet competition by a large margin in 2012, which led to the use of GPUs and Convolutional Neural Networks in machine learning.
- DeepFace (2014) - created by Facebook for people recognition.
- OpenAI (2015) - a non-profit organisation created by Elon Musk and others, to create safe artificial intelligence that can benefit humanity.
- Amazon Machine Learning Platform (2015) - part of Amazon Web Services, it shows how most big companies want to get involved in machine learning.
- ResNet (2015) - a major advancement in CNNs.
- U-net (2015) - a CNN architecture specialised in biomedical image segmentation.

Convolutional Neural Network, the kind of network chosen for this research work, have shown satisfactory performance in processing two-dimensional data with grid-like topology, such as images and videos. The architecture of CNNs is inspired by the animal visual cortex organization. In the 1960s, Hubel and Wiesel [Hubel and Wiesel, 1962] proposed a concept called receptive fields. They found that the complex arrangements of cells were contained in the animal visual cortex in charge of light detection in overlapping and small sub-regions of the visual field. The CNN topology leverages spatial relationships so as to reduce the number of parameters in the network, and the performance is therefore improved using the standard backpropagation algorithms. Another advantage of the CNN model is that it requires minimal pre-processing. The training procedure for a CNN is similar to that for a standard NN using backpropagation. Instead of setting parameters, as is the case with traditional NNs, it is only necessary to train the filters in CNNs. Moreover, in feature extraction, CNNs are independent of prior knowledge and human interference. The CNN structure is shown in Figure 3.4. It is composed by a deep stacking portion and a classification portion. The first one is an alternation of convolution and pooling layers. The objective of the Convolution Operation is to extract the high-level features such as edges, from the input image. Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training of the model. The second portion of the network starts with a Flattening Layer which transforms a two-dimensional matrix of features into a vector that can be fed into a fully connected neural network classifier.

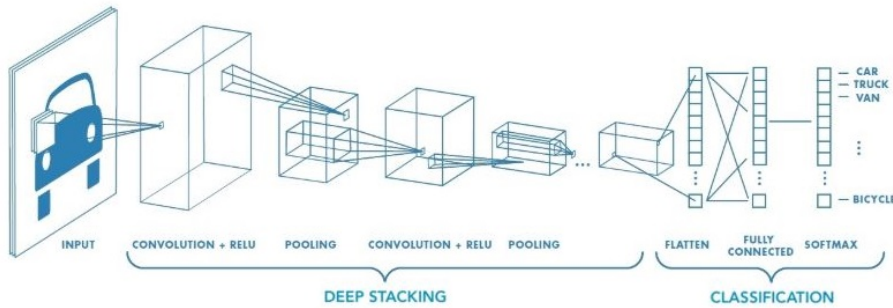


Figure 3.4: A diagram showing the different layers in a CNN [Saha, 2018].

Then there is a fully connected layer also known as the dense layer, in which the results of the convolutional layers are fed through one or more neural layers to generate a prediction. Finally a softmax layer, allows the neural network to run a multi-class function for establishing the probability that each detected class is present in the image. CNN has become a popular research topic in the past few years. Great success has been achieved when CNNs are applied to the research of computer vision. Detection is one of the most widely known sub-domains in computer vision and the one exploited in this research work. It seeks to precisely locate and classify the target objects in an image. As demonstrated in [Szegedy et al., 2015], due to their strong abilities to capture the geometric information such as object locations, DNNs have been widely used for detection and have shown outstanding performance [Liu et al., 2017].

The first major success of convolutional neural networks was AlexNet, developed by Alex Krizhevsky, in 2012 at the University of Toronto. Convolutional neural networks (CNN) are similar to other neural networks, they have weights, biases, and outputs through a nonlinear activation. Regular neural networks take inputs and the neurons fully connected to the next layers. Neurons within the same layer don't share any connections. Using regular neural networks for images will involve a very large in size due to a huge number of neurons, resulting in overfitting. This cannot be used for images, as images are large in size. An image can be considered a volume with dimensions of height, width, and depth. Depth is the channel of an image, which is red, blue, and green. The neurons of a CNN are arranged in a volumetric fashion to take advantage of the volume. Each of the layers transforms the input volume into an output volume. From the AlexNet breakthrough, many new uses have arisen for CNNs, many of which go beyond image classification and rely on segmentation. Fully Convolutional Networks (FCNs) represent the underlying model of recent attempts to solve semantic segmentation using CNNs. These architectures omit the use of fully connected layers. As well as being faster, this approach generates segmentation maps from images of any size, (as opposed to the fixed-size constraint of fully connected layers) [Moore, 2018] [Abiodun et al., 2018].

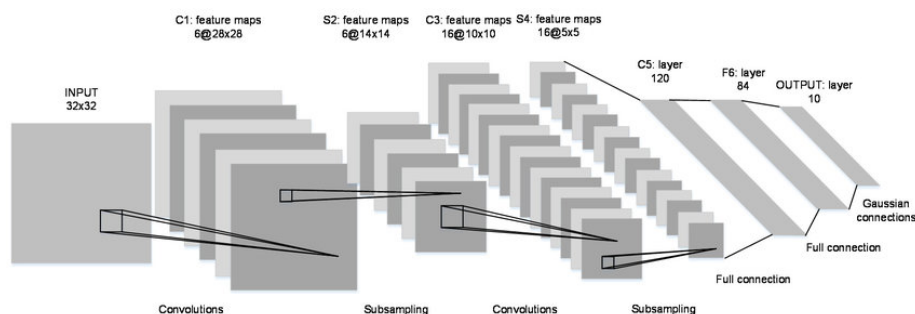


Figure 3.5: LeNet Network structure [Chatterjee, 2016].

The subsequent is a list of the most famous Fully Convolutional Neural Network:

- LeNet

No discussion of the CNN architectures can begin without this. A groundbreaking algorithm that was the first of its kind and capability, in-terms-of object classification. Originally trained to classify hand written digits from 0~9, of the MNIST Dataset. It comprises of 7 layers (Fig. 3.5), all made of trainable parameters. Its input is a 32×32 pixel image, which is comparatively large in size with regards to the images present in the data sets on which the network was trained. The activation function applied is Rectified Linear Unit (σ) function. The layers are arranged in the following manner: The First Convolutional Layer consist of 6 filters of size 5×5 and a stride of 1 (stride is the number of pixels shifts over the input matrix). The Second Layer is a “sub-sampling” or average-pooling layer of size 2×2 and a stride of 2. The Third Layer is also a Convolutional layer consisting of 16 filters of size 5×5 and stride of 1. The Fourth Layer is again an average-pooling layer of size 2×2 and stride of 2. The Fifth Layer is connecting the output of the fourth layer (400 parameters) to a fully connected layer of 120 nodes. The Sixth Layer is a similarly fully-connected layer consisting of 84 nodes, deriving from the outputs of the 120 nodes of the fifth-layer. The Seventh Layer (or the last layer) consist of classifying the output of the last layer into 10 classes related to the 10-digits that it was primarily trained to classify. Implementation of this architecture, on the data sets, using various libraries, would reach an accuracy of around 98.9%. However, when it came to processing large size image and classifying among a large number of classes of objects, this network fails to be effective in terms of computation cost or accuracy [LeCun et al., 1998].

- AlexNet

AlexNet, the winner of the ImageNet ILSVRC-2012 competition, was designed by Alex Krizhevsky, Ilya Sutskever and Geoffery E. Hinton (Fig. 3.6). It is able to reduce the top-5 error rate (when the target label is one of the top 5 predictions, the 5 ones with the highest probabilities) to 15.3 % compared to the

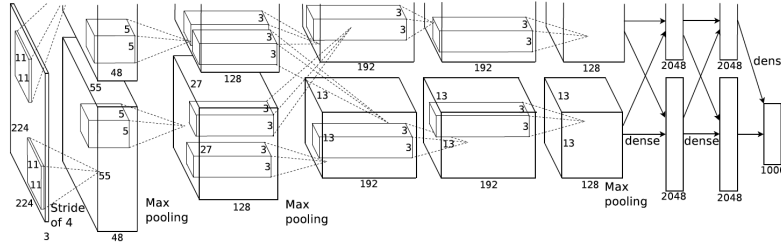


Figure 3.6: AlexNet Network structure [Chatterjee, 2016].

error rate of the runners-up of that competition which attained an error rate of 26.2%. The network is similar to the LeNet Architecture, but has a large number of filters compared to the original LeNet, and thus was able to classify among a large class of objects. Moreover, it used “dropout” instead of regularization, to deal with overfitting. Dropout essentially decreases the size of the number of parameters to be accounted for during the process of training/learning. Let us define the layers in short. It takes in input a color (RGB) image of dimension 224 X 224. First, a Convolution Layer (CL) of 96 filters of size 11 X 11 and stride 4. Next, a Max-Pooling Layer (M-PL) of filter size 3 X 3 and stride = 2. Again, a CL of 256 filters of size 5 X 5 and stride = 4. Then, a M-PL of filter size 3 X 3 and stride = 2. Again, a CL of 384 filters of size 3 X 3 and stride = 4. Again, a CL of 384 filters of size 3 X 3 and stride = 4. Again, a CL of 256 filters of size 3 X 3 and stride = 4. Then, a M-PL of filter size 3 X 3 and stride = 2. The output of the last layer, when converted into input-layer like for the Fully Connected Block consists of 9261 nodes, fully connected to a hidden layer with 4096 nodes. The first hidden layer is again fully connected to another hidden layer consisting 4096 nodes. This last hidden layer is fully connected to the output layer implementing “softmax regression” of 1000 nodes [Jansen and Zhang, 2007].

- VGGNet 16

This particular network architecture was the runners up of the ILSVRC-2014 competition, designed by Simonyan and Zisserman (Fig. 3.7). It was able to achieve a top-5 error rate of 5.1%. Though it might look complicated with a whole bunch of parameters to be taken care of, it is actually very simple. Developers prefer it highly, when it comes to feature extraction because of the simple pattern that it follows. The basic hyperparameters regarding the filter size and the strides for both of the convolution layer and the pooling layer are constant: CONVOLUTION LAYER has filters of size 3 X 3 and stride = 1 and the MAX-POOLING LAYER has filters of size 2 X 2 and stride = 2. These layers are applied in a particular order throughout the network. Only the number of filters defined for each convolution block differs.

It takes in a color (RGB) image of 224 X 224 dimensions. As shown in Figure

3.7 the structure of this network is an alternation of convolutional and max-pooling network growing in dimension. The output of the last Pooling Layer is fed into a fully connected hidden layer consisting of 4096 nodes. This is again fully connected to another hidden layer consisting again of 4096 nodes. This is fully connected to an output layer implementing “softmax regression”, classifying among 1000 classes of objects. That was a lot of layers. It thus has nearly 140 millions parameters to handle, which makes the task, of implementing this network, challenging.

However, weights of pre-trained VGGNet are easily available, and can be used by developers in their project [Simonyan and Zisserman, 2014].

- GoogleNet / Inception

The GoogleNet or the Inception Network was the winner of the ILSVRC 2014 competition, achieving a top-5 error rate of 6.67%, which was nearly equal to human level performance. The model was developed by Google and includes a smarter implementation of the original LeNet architecture (Fig. 3.8). This is based on the idea of inception module. The basic idea behind the modules is that, instead of implementing convolutional layers of various hyperparameters in different layers, we do all the convolution together to output a result containing matrices from all the filter operations together. This is an image of a simple inception module with various convolutional layer implemented together. The concatenated output consists of results from all the convolution operation. Notice that one layer of convolution containing filters of size 1 X 1 is implemented. This reduced the size of the image on which a further convolutional layer, containing filters of size 5 X 5, is applied. The reason behind this is that, the total number of computation units is reduced to a large extent. With the GoogleNet network a Convolutional Layer of 16 filters of size 1 X 1 is applied first, before the implementation of the Convolutional Layer of 32 filters of size 5 X 5, the size of the matrices decreases to 28 X 28 X 16 and then the second convolution is done. Thus the total number of computations is:

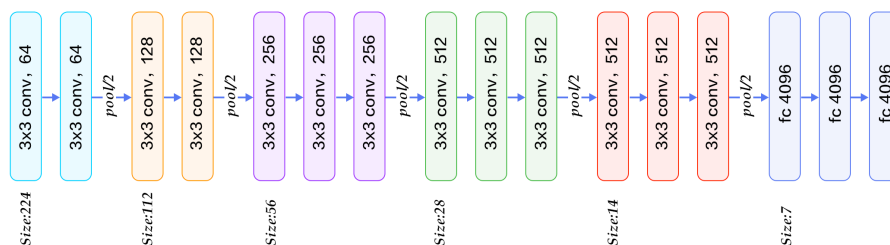


Figure 3.7: VGGNet 16 Network structure [Das, 2017].



Figure 3.8: GoogleNet Network structure [Szegedy et al., 2015].

$$\begin{aligned}
& 28 \times 28 \times 16 \text{ (output of first convolutional layer)} * \\
& 1 \times 1 \times 192 \text{ (size of the weight matrices of the first convolutional layer)} \\
& + 28 \times 28 \times 32 \text{ (output of the second convolutional layer)} \\
& * 5 \times 5 \times 16 \text{ (size of the weight matrices of the second convolutional layer)} \quad (1) \\
& \approx 2.4 \text{ million} + 10.0 \text{ million} \\
& \approx 12.4 \text{ million}
\end{aligned}$$

This number is significantly lower than than the 120 million weights obtained without the first 1x1 filter. Thus, over all the total cost decreases. The last layers are fully connected network layers followed by “softmax regression” for classification in the output layer [Szegedy et al., 2015].

- ResNets

Probably after AlexNet, the most ground-breaking development in the field of CNN architecture development occurred with ResNet or Residual Networks. This is based on the idea of “skip-connections” (Fig. 3.9) and implements heavy batch-normalization, that helps it in training over thousands of layers effectively, without degrading the performance in the long run. The problem rose with the training of deeper networks. The issue of “vanishing gradient” where repeated multiplication being done, as the gradient is being back-propagated, makes the gradient infinitely small. This results in degradation of performance. The idea that was infused in this architecture was “identity shortcut connection” that implies transferring the results of a few layers to some deeper layers skipping some of the other layers in between. Figure 3.10 shows this network structure.

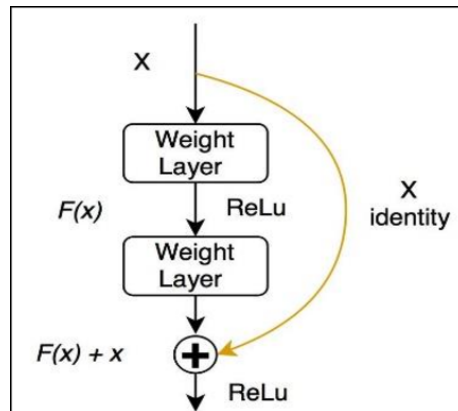


Figure 3.9: Skip-connection idea applied to ResNets [Chatterjee, 2016].

The intuition behind it, was that the deeper layers should not produce higher training errors than its shallower counterparts. The skip-connections were done to implement this idea. This 1001 layer deep ResNet achieved a top-5 error rate of 3.57%, which actually beats human level performance on the dataset.

- You Only Look Once (YOLO)

The network structure looks like a normal CNN, with convolutional and max pooling layers, followed by 2 fully connected layers in the end. Some comments about the architecture (Fig. 3.11): it was crafted to evaluate PASCAL VOC, YOLO uses 7x7 grids (SxS), 2 boundary boxes (B) and 20 classes (C). This explains why the final feature maps are 7x7, and also explains the size of the output ($7 \times 7 \times (2 \times 5 + 20)$). Use of this network with a different grid size or different number of classes might require tuning of the layer dimensions. The authors mention that there is a fast version of YOLO, with fewer convolutional layers. The sequences of 1x1 reduction layers and 3x3 convolutional layers were inspired by the GoogLeNet (Inception) model. The final layer uses a linear activation function. All other layers use a leaky RELU [Redmon et al., 2016].

The recognition of small objects pursued by this research takes place by means of pre-trained neural networks. The objectives of this recognition process are:

- detecting the right location of the object;

- identifying the right type of object (e.g. fire extinguisher size).

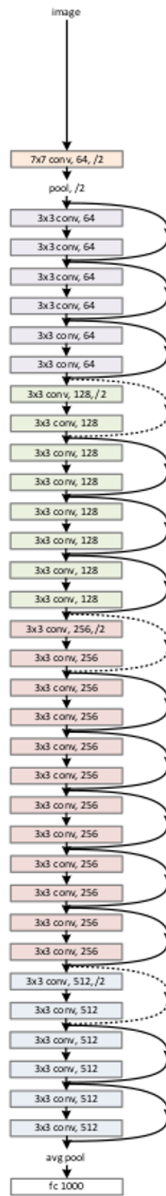


Figure 3.10: ResNets Network structure [Chatterjee, 2016].

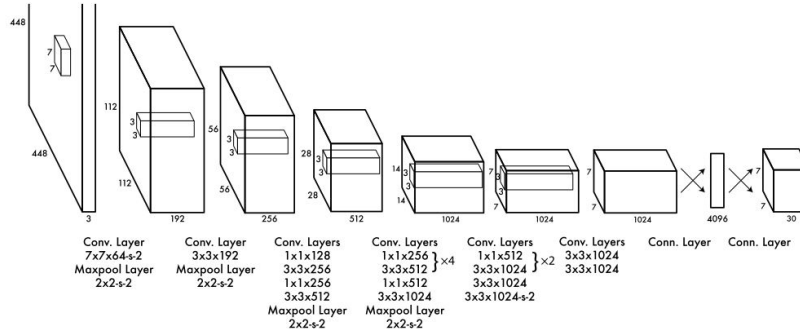


Figure 3.11: YOLO Network structure [Redmon et al., 2016].

Several types of neural networks exist and the YOLO is the one chosen for this project. The choice of using YOLO networks depends on some features of this type of network:

- the speed which is 45 frames per second;
- the simultaneous prediction of multiple bounding boxes;
- the simultaneous prediction of multiple label confidence score;
- it is open source.

Among all the types of neural networks that exist the YOLO, which are able to perform classification and localization in one-step, is the one chosen for this project. This choice depends upon the speed of this kind of NN which is 45 frames per second; making the snapshots processed in real-time, with negligible latency of a few milliseconds. Furthermore, the YOLO can predict multiple bounding boxes and scores simultaneously. Finally, it is an open source solution. In this project, a pre-trained YOLO is used. In order to customize the recognition process it is possible to re-train the last level of the network [Naticchia et al., 2019].

3.4 YOLO Neural Networks

The YOLO project started in 2016 with the aim of reframing object detection as a single regression problem, straight from image pixels to bounding box coordinates and class probabilities. This was in contrast to the previous complex pipeline of R-CNN that first generate potential bounding boxes in an image and then run a classifier on these proposed boxes. The YOLO network uses features from the entire image to predict each bounding box. It also predicts all bounding boxes for an image simultaneously. This means the network reasons

globally about the full image and all the objects in the image. The image is divided into a $S \times S$ grid. If the center of an object falls into a grid cell, that grid cell is responsible for detecting that object. Each grid cell predicts B bounding boxes and confidence scores for those boxes. These confidence scores reflect how confident the model is that the box contains an object and also how accurate it thinks the box is that it predicts. If no object exists in that cell, the confidence scores should be zero. Otherwise the confidence score is equal to the Intersection Over Union (IOU) between the predicted box and the ground truth. Each bounding box consists of 5 predictions: x , y , w , h , and confidence. The (x, y) coordinates represent the center of the box relative to the bounds of the grid cell. The width and height are predicted relative to the whole image. Finally the confidence prediction represents the IOU between the predicted box and any ground truth box. Each grid cell also predicts C conditional class probabilities, $\Pr(\text{Class}|\text{Object})$. These probabilities are conditioned on the grid cell containing an object. For evaluating YOLO on PASCAL VOC, the parameters are $S = 7$, $B = 2$. PASCAL VOC has 20 labelled classes so $C = 20$. Yolo architecture is more like FCNN (fully convolutional neural network) and passes the image $(n \times n)$ once through the FCNN and output is $(m \times m)$ prediction [Morrison, 2018]. YOLO is a single convolutional network that simultaneously predicts multiple bounding boxes and class probabilities for those boxes. YOLO trains on full images and directly optimizes detection performance. This unified model has several benefits over traditional methods of object detection. First, YOLO is extremely fast. This network runs at 45 frames per second with no batch processing on a Titan X GPU and a fast version runs at more than 150 fps. This means it can process streaming video in real-time with less than 25 milliseconds of latency. Furthermore, YOLO achieves more than twice the Mean Average Precision (mAP) of other real-time systems. Secondly, YOLO reasons globally about the image when making predictions. Unlike sliding window and region proposal-based techniques, YOLO sees the entire image during training and test time so it encodes contextual information about classes as well as their appearance. YOLO makes less than half the number of background errors compared to Fast R-CNN (Fig. ??). Third, YOLO learns generalizable representations of objects. When trained on natural images and tested on artwork, YOLO outperforms top detection methods like DPM and R-CNN by a wide margin. Since YOLO is highly generalizable it is less likely to break down when applied to new domains or unexpected input [Redmon et al., 2016].

After the first development of YOLO the same developers worked to improve some shortcomings focusing mainly on improving recall and localization while maintaining classification accuracy. With YOLOv2 they were looking for a more accurate detector that was still fast. Instead of scaling up the network, they simplified the network and then made the representation easier to learn. The new features of YOLO V2 were:

- Batch Normalization. It leads to significant improvements in convergence while eliminating the need for other forms of regularization. By adding

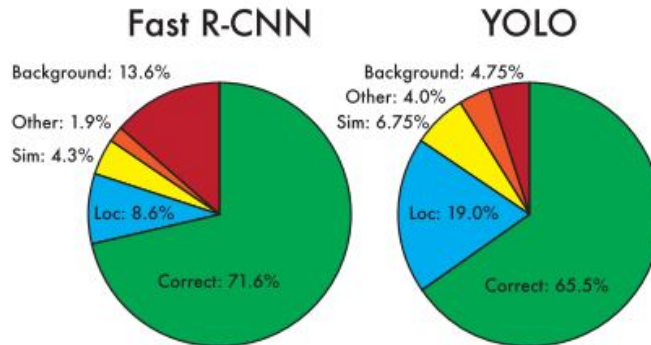


Figure 3.12: Performance comparison between Fast R-CNN and YOLO [Redmon et al., 2016].

batch normalization on all of the convolutional layers in YOLO they get more than 2% improvement in mAP. Batch normalization also helps regularize the model.

- **High Resolution Classifier.** All state-of-the-art detection methods use classifier pre-trained on ImageNet. For YOLOv2 they first fine tune the classification network at the full 448x448 resolution for 10 epochs on ImageNet. This gives the network time to adjust its filters to work better on higher resolution input. This high resolution classification network gives us an increase of almost 4% mAP.
- **Convolutional With Anchor Boxes.** YOLO predicts the coordinates of bounding boxes directly using fully connected layers on top of the convolutional feature extractor. They remove the fully connected layers from YOLO and use anchor boxes to predict bounding boxes.

First we eliminate one pooling layer to make the output of the network’s convolutional layers higher resolution. They also shrink the network to operate on 416 input images instead of 448x448. YOLO’s convolutional layers downsample the image by a factor of 32 so by using an input image of 416 the output feature map is of 13x13. Using anchor boxes there is a small decrease in accuracy. YOLO in its first version only predicted 98 boxes per image but with anchor boxes YOLO v2 model predicts more than a thousand. Without anchor boxes our intermediate model gets 69.5 mAP with a recall of 81%. With anchor boxes our model gets 69.2 mAP with a recall of 88%. Even though the mAP decreases, the increase in recall means that our model has more room to improve [Redmon and Farhadi, 2017]. They propose a new classification model to be used as the base of YOLOv2. It is called Darknet-19, has 19 convolutional layers and 5 maxpooling layers. Darknet-19 only requires 5.58 billion operations

to process an image yet achieving 72.9% top-1 accuracy and 91.2% top-5 accuracy on ImageNet.

Then in 2018 they released YOLOv3 that is able to predict an objectness score for each bounding box using logistic regression with the performances expressed in Figure 3.13. This should be 1 if the bounding box prior overlaps a ground truth object by more than any other bounding box prior. If the bounding box prior is not the best but does overlap a ground truth object by more than some threshold they ignore the prediction. Each box predicts the classes the bounding box may contain using multilabel classification. They do not use a softmax as we have found it is unnecessary for good performance, instead they simply use independent logistic classifiers. During training we use binary cross-entropy loss for the class predictions. This formulation helps when we move to more complex domains like the Open Images Dataset. In this dataset there are many overlapping labels (i.e. Woman and Person). Using a softmax imposes the assumption that each box has exactly one class which is often not the case. A multilabel approach better models the data. Their network uses successive 3x3 and 1x1 convolutional layers but now has some shortcut connections as well and it is significantly larger. It has 53 convolutional layers so they call it Darknet-53 (Fig. 3.14) [Redmon and Farhadi, 2018].

There have developed also a fast version of YOLO designed to push the boundaries of fast object detection. Fast YOLO uses a neural network with fewer convolutional layers (9 instead of 24) and fewer filters in those layers. Other than the size of the network, all training and testing parameters are the same between YOLO and Fast YOLO. The final output of our network is the 7 x 7 x 30 tensor of predictions [Redmon et al., 2016]. The YOLO chosen for this research is the Yolo v2 Tiny (Fig. 3.15) which has fewer parameters than Yolo v1. Its network structure is composed of 9 convolution layers and 6 maximum pooling layers [Li et al., 2018].

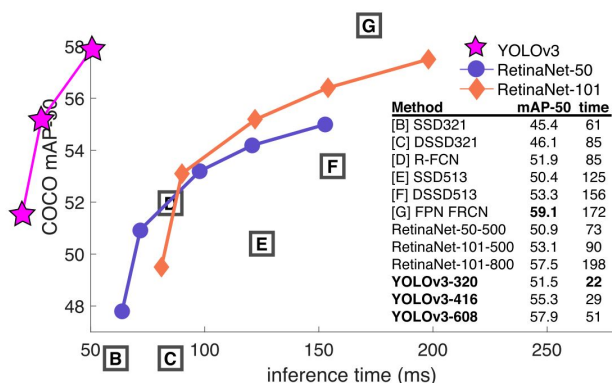


Figure 3.13: YOLO v3 speed/accuracy tradeoff [Redmon and Farhadi, 2018].

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	128 × 128
	Convolutional	64	3 × 3	
	Residual			
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	64 × 64
	Convolutional	128	3 × 3	
	Residual			
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	32 × 32
	Convolutional	256	3 × 3	
	Residual			
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	16 × 16
	Convolutional	512	3 × 3	
	Residual			
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	8 × 8
	Convolutional	1024	3 × 3	
	Residual			
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 3.14: Darknet-53 [Redmon and Farhadi, 2018].

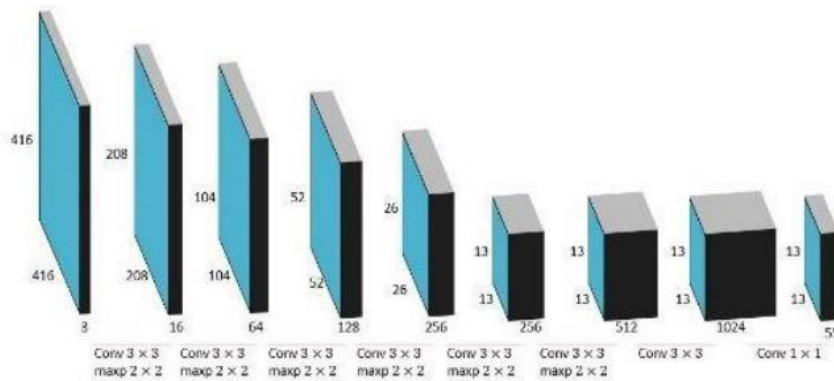


Figure 3.15: YOLO Tiny v2 structure [Li et al., 2018].

3.5 YOLO training process

The inputs of a Neural Network are weighted and summed as shown in Figure 3.16. The sum is then passed through a unit step function, in this case, for a binary classification problem. A perceptron can only learn simple functions by learning the weights from examples. The process of learning the weights is called training. The model values are initialized with random values during the beginning of the training. The error is computed using a loss function by contrasting it with the ground truth. Based on the loss computed, the weights are tuned at every step. The training is stopped when the error cannot be further reduced. The training process learns the features during the training [Moore, 2018].

In order to customize a neural network for the recognition of a specific object this needs to be re-trained. Two processes can be followed:

- doing a training from scratch;
- using a pre-trained network and exploiting the transfer learning.

As the deep neural network is not sensitive to the features, the models trained for some special purposes can be used or partially integrated into the new model with the help of transfer learning. In this way, the time and hardware cost would be significantly reduced, making the deep neural network more practical. Thus, researchers and professionals introduce transfer learning to store the knowledge gained from solving one problem and applying it to a different but related problem. Convolutional neural networks can be designed from scratch and subsequently trained on various datasets to achieve optimal performance. This approach requires a large amount of time, even when there are sufficient hardware resources. Transfer learning uses a machine learning algorithm (e.g. CNN) as an extractor of features which are then fed into another classifier [Kolar et al., 2018].

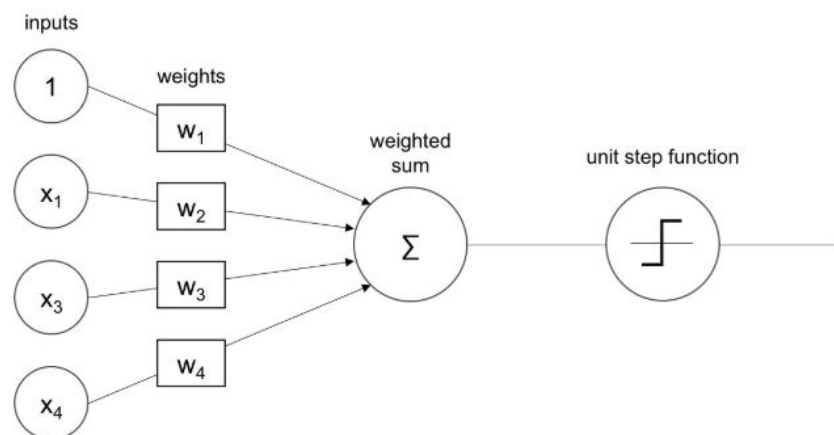


Figure 3.16: Neural Network general structure [Moore, 2018].

For training process in this research the Darknet neural network framework has been used as advised by YOLO developers [Redmon and Farhadi, 2018]. In the following paragraphs the training process will be explained starting from the essential tool of the training framework.

3.5.1 The training framework

In order to train the network it is necessary to have a training environment. The following ones are all the necessary requirements [AlexeyAB, 2019]:

- Windows or Linux;
- CMake ≥ 3.8 for modern CUDA support;
- CUDA;
- OpenCV ≥ 2.4 ;
- cuDNN ≥ 7.0 ;
- GPU with CC ≥ 3.0 ;
- on Linux GCC or Clang, on Windows MSVC 2015/2017/2019.

Since this project involves the use of YOLO neural network it has been decided to use the training platform advised by the developer of the network itself: Darknet-19 [Shinde et al., 2018]. Darknet is an open source neural network framework written in C and CUDA. It supports CPU and GPU computation [Redmon, 2018].

In order to install Darknet it is necessary to set the proper environment. This consists mainly in the installing of a development system and the aforementioned dependencies [Redmon, 2016]. The development system is Visual Studio installed with its default options. The dependencies are CUDA, cuDNN and OpenCV.

Starting from CUDA, the version installed is the 9.1. This installation requires also the installation of the NVIDIA Graphics Drivers if not yet on the pc.

The second installation to be done is cuDNN version 7.0.

Finally it follows the installation of OpenCV 3.4.0.

After having done all this installations Darnet needs to be compiled with the following procedure:

1. Start Microsoft Visual Studio

2. Open the darknet.sln

3. set x64 and Release (Fig. 3.17)

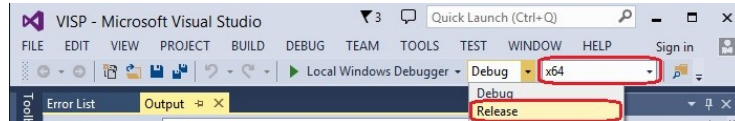


Figure 3.17: Visual studio setting for Darkent compilation: x64 bit and Release version.

4. Include cudnn.lib in your Visual Studio project (Fig. 3.18).

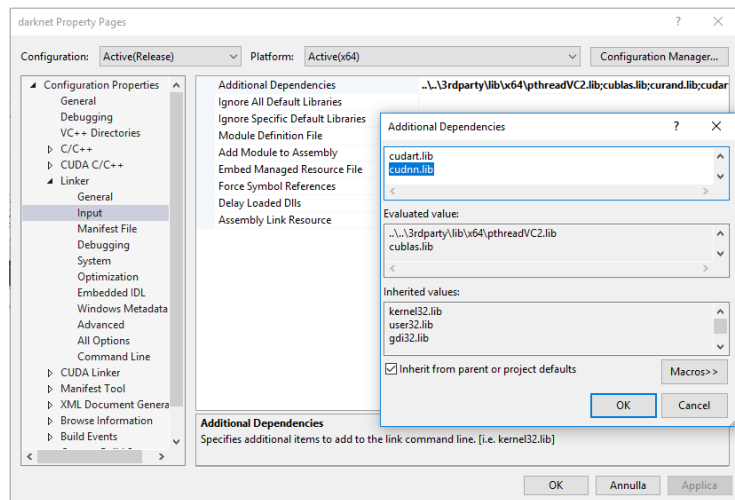


Figure 3.18: Visual studio setting for Darkent compilation: including cudnn.lib.

5. Build > Build darknet.

At this moment the darknet.exe is generated inside the folder.

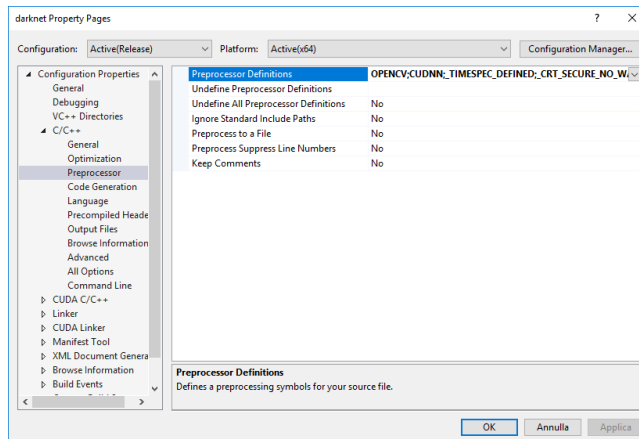


Figure 3.19: Inserting OpenCV, CUDA and cuDNN in the darknet.exe folder.

Finally darknet needs to be prepared for using OpenCV, CUDA and cuDNN. The bin file has to be placed in the same folder of darknet.exe (Fig. 3.19). Bin and include folders have to be inserted also in CUDA folder if they are not already there. Finally a new Windows variable cudnn has to be created (Fig. 3.20).

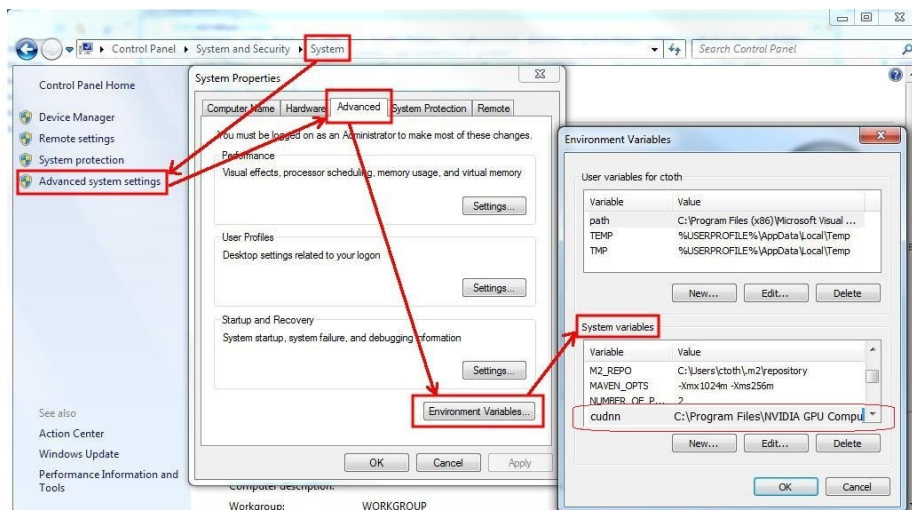


Figure 3.20: Creation of the new Windows variable for cudnn.

3.5.2 The Dataset creation

The dataset to train the network to recognize a specific object must have specific features. With the aim of identifying only the right object (e.g. fire extinguisher) the dataset will include at least one image for every existing type of fire extinguisher, considering that, to be efficient, the network should be able to recognize the object in every situation. This means that if for the same object it is possible to find different shapes they must be all included into the dataset. Furthermore it has to be taken into consideration that the recognition is performed according to different environmental conditions such as lighting or view point. All these factors must be considered when collecting pictures for a new dataset. In order to successfully recognize the object at the first attempt the dataset should contain a high number of pictures of the same object. It is a hard task to define a minimum number because it depends also upon the object specific features. Something red usually on top of a white surface, like a fire extinguisher, would be easier to recognize in comparison to a white socket on a white wall. Furthermore the network would not be able to recognize the type of object only by the overall appearance or shape. For this reason, identifying object components that determine its type and working on the recognition of them could represent a valuable solution. According to the COCO dataset approach choosing images with the object in context improve the recognition of it in real scenarios [Lin et al., 2014]. For this reason pictures with the objects in their common context have been preferred in this research.

The presence of multiple objects in the same pictures is another parameter that improves the performance of the network. For instance it is common in some countries (e.g. UK) that the most common configuration requires two fire extinguishers one next to the other. The same reasoning can be done for emergency signals since there could be more than one in a corridor and also it is plausible to have different signals indicating different kind of exit close to each other.

Another parameter taken into consideration is the perspective. As pointed out by Radovich et al. [Radovich et al., 2017] even if the object is present in a dataset the point of view is a fact that counts. For this reason the pictures chosen in this research represent a plausible perspective from an operator walking inside a building. At the end of this chapter it is possible to see some pictures of the datasets used in this work.

The chose of starting with fire extinguishers and emergency signs depends only upon the fact that the fire protection system is one of the facility most subjected to periodic checkings. The system proposed can be generalized for every equipment asset inside buildings. Collecting pictures for the dataset can be a tough task. As stated before it is not only having images of the objects themselves but it involves also inserting them in the right context. Moreover collecting original images is always a time consuming task. There have been defined different techniques to gather pictures, among them the most used are the following ones:

1. crowdsourcing;

2. web scraping.

Generally speaking the crowdsourcing regards all the possible processes to obtain information or input into a particular task or project by enlisting the services of a large number of people, either paid or unpaid, typically via the Internet. The benefits of this kind of process as far as the dataset creation for NN training is concerned are more than one. Unlike datasets shot in controlled environments crowdsourcing brings in diversity which is essential for generalization [Laptev and Gupta, 2016]. In fact collecting images from people enhances the possibilities of having different context, lighting and also object features. Moreover crowdsourcing splits the task of collecting images among several participants and this means an easier task to accomplish but also a bigger number of items collected with less effort. In this research the method used was more inspired by the crowdsourcing than a real implementation of it. In this case the task was not spread by internet but among people in the department and acquaintances outside the university.

On the other hand the digital era has brought with it an enormous explosion of data. The latest estimations put a number of more than 3 billion photos on Flickr, a similar number of video clips on YouTube and an even larger number for images in the Google Image Search database [Deng et al., 2009]. For these reasons another common technique used to increase the number of pictures is the web scraping. In this case three popular sites have been exploited: 1. Google; 2. Flickr (already used in other research like [Everingham et al., 2015]); 3. Instagram. As far as Google is concerned a python script has been used for web scraping images [Github, 2019].

The script needs one or more keyword for the research and the desired number of images has to be set. Since search engines usually limit the number of images retrievable one method to expand the query is the use of synonyms. To further enlarge and diversify the candidate pool, it is also advisable to translate the queries into other languages and in this case we used both Italian and English definitions [Deng et al., 2009]. After performing several Google queries the process of filtering the noisy results have been performed [Ordonez et al., 2011]. In this way only the more relevant pictures have been kept. The other contribution for the harvesting of a large set of images has been the searching and downloading of images on Flickr. Flickr is an image and video hosting service with more than 3.5 million new images uploaded daily. Also in this case the expedient of using translation in different languages of the keyword have been used. The exact same thing has been pursued with Instagram which is a photo and video-sharing social networking. The only difference in this case was that looking for images in Instagram requires the use of the hashtag, a metadata tag for social networks.

As the dataset must include thousands of pictures, a common technique consist in the usage of both original pictures, from real buildings in this case, and graphically re-edited photos. The process of editing the original photos for the dataset is called augmentation. Data augmentation gives ways to increase the size of the dataset. Data augmentation introduces noise during training, producing

robustness in the model to various inputs. This technique is useful in scenarios where the dataset is small and can be combined and used with other techniques. The percentage of original images and modified images has been studied in order to obtain a trained network with good performances [Montserrat et al., 2017]. After a comparison between the result of different training done with different dataset that are 100% , 50% and 25% made out of augmented images the result is that, if the number of object poses is the same, the 50% is the best amount of augmented images. On the other hand when the comparison is among different numbers of poses of the object the higher the number of images (even if all augmented) the better the performance. These are matters to take into consideration when deciding the number, and therefore the percentage, of augmented images.

There are various ways to augment the images as described in the following list [Moore, 2018]:

- Flipping: the image is mirrored or flipped with respect to a horizontal or vertical direction;
- Random cropping: Random portions are cropped, hence the model can deal with occlusions;
- Shearing: the images are deformed to affect the shape of the objects;
- Zooming: zoomed portions of images are trained to deal with varying scales of images;
- Rotation: the objects are rotated to deal with various degrees of change in objects;
- Whitening: the whitening is done by a Principal Component Analysis that preserves only the important data;
- Normalization: normalizes the pixels by standardizing the mean and variance;
- Channel shifting: the color channels are shifted to make the model robust against color changes caused by various artifacts.

There can be also other variations not mentioned like for instance the resize and the distortion. Other methods also exist that can be considered a way of augmenting the data. These involve the generation of fake images overlapping real images of the object to recognize on top of an image of a plausible context [Jeong et al., 2018].

In this work the augmentation involves only the modification of the whole picture according to the list of possible operations mentioned above. These transformations have been performed with the help of a custom MatLab script. This tool gives the possibility to automatically modify the photos, choosing what transformations must be performed, the starting dataset and the number of images to be created. Not only this, the script also automatically modifies the bounding box, an instance that will be explained later in this section. The custom script applies the following transformations with these parameters:

- Pixels transformation
 - Resize: image, minValue, maxValue, probability
 - Shifting: image, hueShift, saturationScale, saturationShift, valueScale, valueShift, probability
 - Additive noise: image, percentagePixels, probability
- Geometric transformations
 - Rotate: image, annotations, minValue, maxValue, objectCornerRadius, probability
 - Zoom: image, annotations, minValue, maxValue, probability
 - Crop: image, annotations, minValue, maxValue, probability

There are also ready available scripts to run the augmentation [Augmentor, 2019] but the reason why a customized one has been used is that with it also the bounding box are coherently modified avoiding the need of labelling a huge amount of pictures afterwards.

The creation of the dataset involves also labelling all the images. Labeling defines the approach of marking all regions of interest in a set of pictures and defining the type of marked region [Braun et al., 2019]. Creating the label involves both the design of the bounding box around the object to recognize and attaching the correct label to it. This operation could be performed in many different ways both manually and with the aid of software tools.

In the case performed in this thesis we worked using a tool that supports the manual drawing of the bounding box and which gives the possibility of adding a label to every single box according to the category of the object involved in the recognition (Figure 3.21) [VoTT, 2019].

This tool gives the possibility to choose the right output format according to the kind of network chosen. The output for the YOLO network is a .txt file, with the coordinates of the boxes and the label attached to them as shown below.

0 0.550595 0.428943 0.558862 0.796875
↑ ↑ ↑ ↑ ↑
Label x coordinate of the centre y coordinate of the centre Width Height

The final task to complete the dataset is the definition of the train and test sets of images. The train set is defined by a .txt file listing all the images that will be used for training. On the other hand the test list of images, also defined by a .txt file, will be used after the training phase to verify the performances of the network.

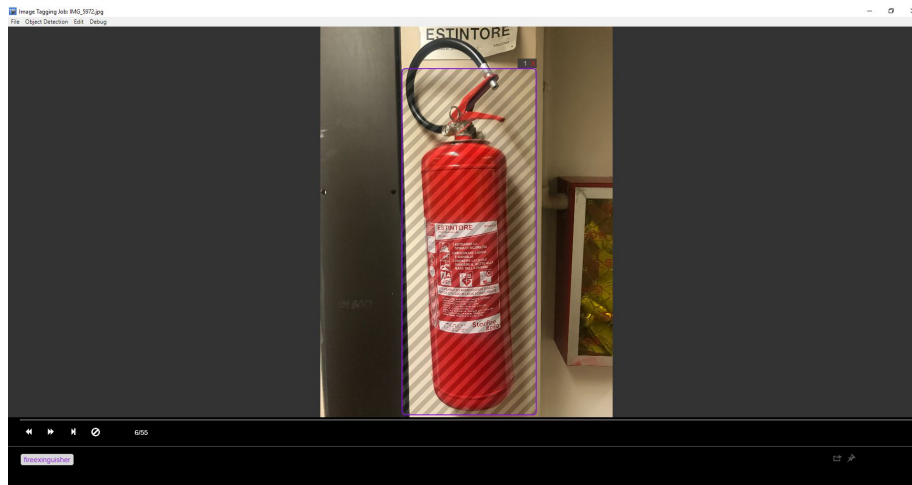


Figure 3.21: Bounding box design with VoTT software.

3.5.3 Training the network

After having created the dataset the training session can start. There are some files to set before starting the process:

- the .cfg file of the network chosen
- the .data file
- the .names file
- the .weights file

As far as the .cfg file is concerned some parameters have to be taken into consideration for training a customized network. The parameters to check are [Tijtgat, 2017] [AlexeyAB, 2019]:

- batch = 64, this means we will be using 64 images for every training step;
- subdivision = 8, the batch will be divided by 8 to decrease GPU VRAM requirements. If you have a powerful GPU with loads of VRAM, this number can be decreased, or batch could be increased. The training step will throw a CUDA out of memory error so you can adjust accordingly;
- classes = 1, the number of categories we want to detect;
- filters = (classes + 5)*5.

The .data file is the file that the software reads to train the network. It contains all the paths to the other necessary files for the training process as show in Figure 3.22.

```
1 classes = 1
2 train  = TRAINING_Dataset_SIGN_FE/obj/train.txt
3 valid  = TRAINING_Dataset_SIGN_FE/obj/test.txt
4 names  = data/obj4.names
5 backup = backup/
```

Figure 3.22: .data file for network training

The classes parameter is the number of different objects for the training process, the train is the path of the train list of images, the valid is the path for the test list of images, names is the path for the .names file and finally the backup define the name of the backup folder where all the weights created will be saved.

The .names file (Fig. 3.23) is the file that contains the name of the tag inside the images of the dataset. The tags are identified by the row number inside the file.

```
1 fireextinguisher
2 emergencysigndoor
3 emergencysignman
4 emergencysign
```

Figure 3.23: .names file containing name tags.

The weights have to be chosen according to the network that one wants to train. Choosing the weight file means that the network used is a pre-trained network with a general dataset (CoCo, PascalVoc or others). It would be possible also not to choose any weights file and in that case the network will be trained from scratch and it will not profit from the transfer learning process. The training command line can be sent from Windows Power Shell, a command-line shell with associated scripting language. To be in power of letting the train start the command has to be sent inside the folder where the executable of Darknet is located. The the command line will be as follows (Fig. 3.24):

```
PS C:\darknet\build\darknet\x64> ./darknet.exe detector train data/obj.data cfg/yolov2-tiny.cfg yolov2-tiny.weights -map > log.txt
```

Figure 3.24: Training command line

where the paths of the .data file, the .cfg file and .weights file are defined. The `-map > log.txt` command is an output request so as to have the trend of the mean average precision and the loss mapped on a chart. During the training of the network on one hand it is possible to see in a chart (Fig. 3.25) the mean average precision (mAP) increasing in value. On the other hand, there is the calculation of the loss function by means of Equation 1. The loss function only penalizes classification error if an object is present in that grid cell. It also penalizes bounding box coordinate error only if that predictor is “responsible” for the ground truth box (i.e. has the highest IOU of any predictor in that grid cell).

$$\begin{aligned}
& \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \\
& + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} (\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} (C_i - \hat{C}_i^2) \\
& + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{noobj} (C_i - \hat{C}_i^2) \\
& + \sum_{i=0}^{S^2} 1_{ij}^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2
\end{aligned} \tag{2}$$

The output of the training process therefore are:

- the chart with the mAP progress in red and the Avg progress in blue (Fig. 3.25);
- the log file with all the operations executed to train the network (Fig. 3.26), it contains the report of all the epoch, with the avg and mAP values;
- the backup folder which contains the weights of the trained network saved at predefined stages (1000 epochs this case).

Looking at Figure 3.26 line third:

- 411267 indicates the current training iteration/batch;
- 1.623754 is the total loss;

- 2.182349 avg is the average loss error, which should be as low as possible;
- 0.000100 rate represents the current learning rate, as defined in the .cfg file;
- 0.469000 seconds represents the total time spent to process this batch;
- The 26321088 images at the end of the line is nothing more than $9778 * 64$, the total amount of images used during training so far.

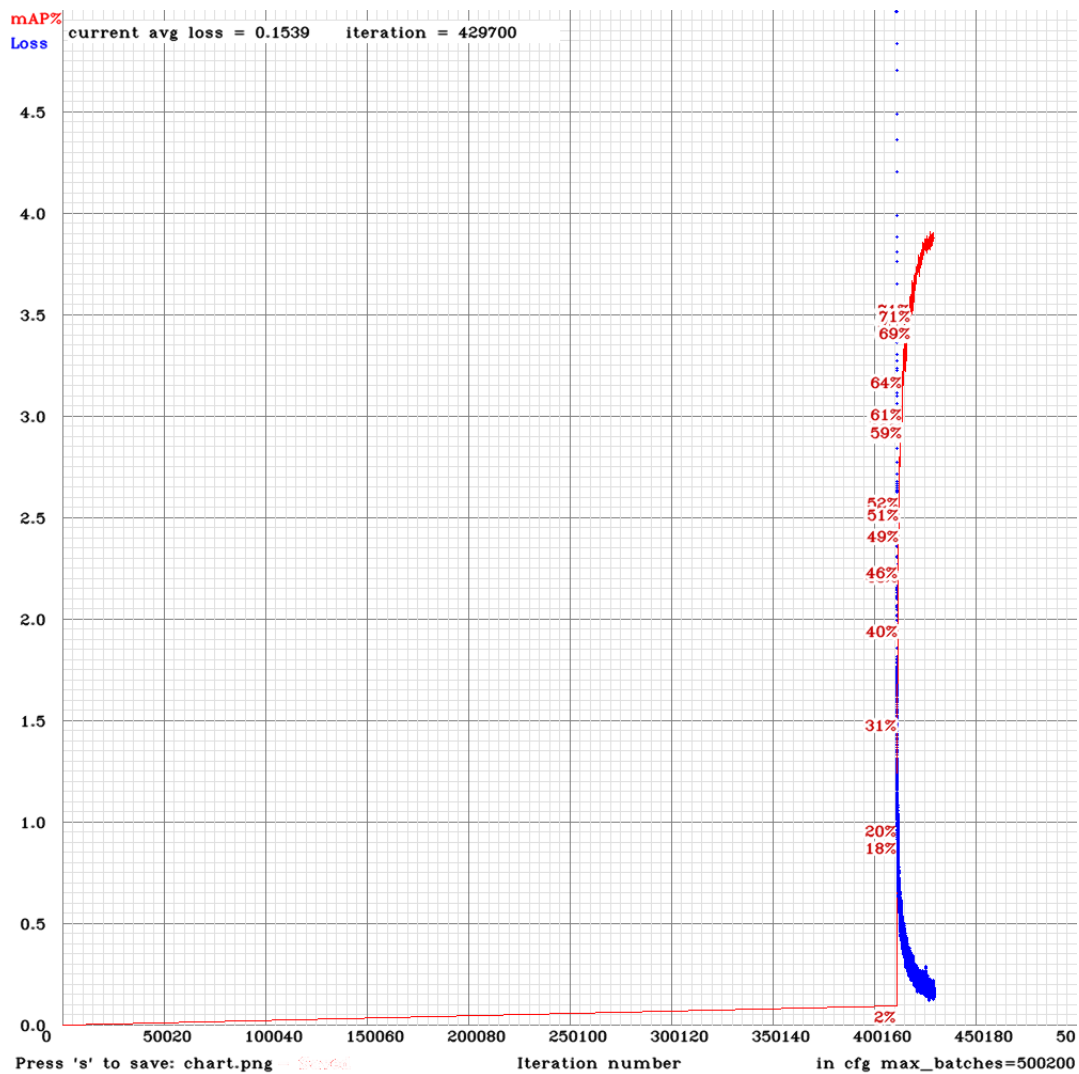


Figure 3.25: Output chart of the training process

```

(next mAP calculation at 411270 iterations)
Last accuracy mAP@0.5 = 3.33 %
411267: 1.623754, 2.182349 avg loss, 0.000100 rate, 0.469000 seconds, 26321088 images
Loaded: 1.515000 seconds
Region Avg IOU: 0.428098, Class: 1.000000, Obj: 0.329525, No Obj: 0.006018, Avg Recall: 0.347826, count: 23
Region Avg IOU: 0.415163, Class: 1.000000, Obj: 0.332707, No Obj: 0.005271, Avg Recall: 0.347826, count: 23
Region Avg IOU: 0.420006, Class: 1.000000, Obj: 0.230340, No Obj: 0.005023, Avg Recall: 0.250000, count: 24
Region Avg IOU: 0.400327, Class: 1.000000, Obj: 0.509652, No Obj: 0.007539, Avg Recall: 0.333333, count: 27

(next mAP calculation at 411270 iterations)
Last accuracy mAP@0.5 = 3.33 %
411268: 1.655088, 2.129623 avg loss, 0.000100 rate, 0.469000 seconds, 26321152 images
Loaded: 1.328000 seconds
Region Avg IOU: 0.386825, Class: 1.000000, Obj: 0.462841, No Obj: 0.007151, Avg Recall: 0.333333, count: 24
Region Avg IOU: 0.422769, Class: 1.000000, Obj: 0.279391, No Obj: 0.005894, Avg Recall: 0.300000, count: 20
Region Avg IOU: 0.361742, Class: 1.000000, Obj: 0.301057, No Obj: 0.006179, Avg Recall: 0.250000, count: 24
Region Avg IOU: 0.411956, Class: 1.000000, Obj: 0.381415, No Obj: 0.006649, Avg Recall: 0.290323, count: 31

(next mAP calculation at 411270 iterations)
Last accuracy mAP@0.5 = 3.33 %
411269: 1.732979, 2.089958 avg loss, 0.000100 rate, 0.469000 seconds, 26321216 images
Loaded: 1.296000 seconds
Region Avg IOU: 0.334095, Class: 1.000000, Obj: 0.289606, No Obj: 0.006712, Avg Recall: 0.142857, count: 49
Region Avg IOU: 0.324033, Class: 1.000000, Obj: 0.435073, No Obj: 0.006620, Avg Recall: 0.194444, count: 36
Region Avg IOU: 0.407480, Class: 1.000000, Obj: 0.282529, No Obj: 0.006685, Avg Recall: 0.281250, count: 32
Region Avg IOU: 0.462179, Class: 1.000000, Obj: 0.246844, No Obj: 0.006014, Avg Recall: 0.500000, count: 22

```

Figure 3.26: Output log file of the training process

Looking then at the Region Avg line the meaning of the values is the following:

- Region Avg IOU: 0.428098 is the average of the IOU of every image in the current subdivision. A 32,66% overlap in this case, this model still requires further training.
- The Avg Recall: 0.347826 is defined in code as $\text{recall}/\text{count}$, and thus a metric for how many positives YOLOv2 detected out of the total amount of positives in this subdivision. In this case only one of the eight positives was correctly detected.
- count: 23 is the amount of positives (objects to be detected) present in the current subdivision of images (subdivision with size 8 in our case). Looking at the other lines in the log, it can be noticed that there are also subdivisions that only have 6 or 7 positives, indicating there are images in that subdivision that do not contain an object to be detected.

The training process can be stopped when the average loss reaches an acceptable value. In that case even if the mAP has not reached good results going on with the training will not lead to better performances. It is hard to find in literature the exact value under which the average loss can be considered low enough. As a rule of thumb, according to what found in other customize network experiences, over one zero after the point everything is considered acceptable (e.g. 0.02 or less).

3.5.4 Validation process

The most common indexes used for the evaluation of Neural Network performances are precision (Eq. 2) and recall (Eq. 3) [Li et al., 2018] [Hamledari et al., 2017]

[Deng et al., 2009] [Montserrat et al., 2017] [Quintana et al., 2018] [Everingham et al., 2015].

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

True Positives (TP) and False Positives (FP) are the number of objects correctly and incorrectly predicted, respectively, as the object of interest. Similarly, True Negatives (TN) and False Negatives (FN) are the number of objects correctly and incorrectly recognized as background. Because the precision and recall rates cannot be reported for scenes without any actual positives, the images taken into consideration contain at least one instance of the objects of interest [Hamledari et al., 2017].

To compute the mean Average Precision (mAP), the Precision and Recall are required. Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. Recall is the ratio of correctly predicted positive observations to all the observations in actual class. For each class, a Precision/Recall curve is obtained by varying the threshold parameter from 0 to 1. The average precision is defined as the area under the curve. The Mean Average Precision (mAP) is computed by averaging the AP value for all classes. The previous process is repeated obtaining the AP for each class [Montserrat et al., 2017].

Besides the calculation of Precision and Recall also the F1 parameter has been measured. This metrics is frequently used in pattern recognition performance assessment [Quintana et al., 2018]. F1-measure is a measure that combines Precision and Recall, using a kind of weighted average (Eq. 4).

$$F = 2 \frac{Precision * Recall}{Precision + Recall} \quad (5)$$


ALL_1000_ORIGIN/obj/IMG_0884.JPG		60,00% TP 100,00% TP
		94,00% FP

Figure 3.27: Output of the validation process pursued on a desktop and following evaluation for Precision and Recall calculation.

The validation for all the training processes pursued has been done using the same test dataset so as to make the tests comparable. Moreover the first tests have been conducted on a desktop so as to assess the performance of the custom neural network with the same kind of input data as the training one. Figure 3.27 shows the output of the test and the evaluation about the result for the calculation of precision and recall. The third column shows the confidence score for every single object detected marked by its bounding box.

Figure 3.28 shows the command line for the validation of the customized network. Within the line it is possible to find the path to the .data file; the path to the YOLO network structure; the path to the chosen weights for the validation process. As stated before in this chapter, in fact, the training produces a number of weights file according to the saving settings of the Darknet framework. In this research two Darknet frameworks have been set, one saved the weights every 1000 iterations, the other did it every 100 iterations. This depended upon the possibility of reaching the weights closest to the maximum mAP obtained by the network training. With this aim for every validation it has been selected the weights closer to the number of iteration that had obtained the higher mAP. Finally in the command line it is possible to see a threshold setting. This is the minimum confidence score to be taken into consideration for showing the recognition result.

Following the validation done on a desktop a real-world scenario validation was unavoidable to start testing the feasibility of the system. Figure 3.29 shows one of this tests conducted in the DC3 laboratory at Politechnic University of Marche. In this case in addition to confidence score, TP, TN and FP and FN another parameter has been registered. The lag between sending the picture to the embedded system and the recognition performed. This is fundamental in order to assess the possibility of using the system for real-time applications directly on site.

```
PS C:\darknet\build\darknet\x64> ./darknet.exe detector test data/fe.data cfg/yolov2-tiny-c1.cfg yolov2-tiny-c1_425200.weights -i 0 -thresh 0.25
```

Figure 3.28: Command line for the validation procedure of the customized network.



Figure 3.29: Real-world validation of the customized network for fire extinguisher recognition.

3.6 Mixed reality

Mixed reality is the result of blending the physical world with a digital representation. Advancements in sensors and processing are giving rise to a new area of computer generated input from environments. The interaction between computers and environments is effectively environmental understanding or perception. Environmental input captures things like a person's position in the world (e.g. head tracking), surfaces and boundaries (e.g. spatial mapping and spatial understanding), ambient lighting, environmental sound, object recognition, and location. Now, the combination of all three, computer processing, human input, and environmental input, sets the opportunity to create true mixed reality experiences (Fig. 3.30). Movement through the physical world can be transferred to movement in the digital world. Boundaries in the physical world can influence application experiences in the digital world. Without environmental input, experiences cannot blend between physical and digital realities. The experiences that overlay graphics on video streams of the physical world are augmented reality. The experiences that occlude your view to present a digital experience are virtual reality. As you can see, the experiences enabled between these two extremes is mixed reality [Microsoft, 2018e]:

- starting with the physical world, placing a digital object, such as a hologram, as if it was really there.
- starting with the physical world, a digital representation of another person shows the location where they were standing when leaving notes. In other

words, experiences that represent asynchronous collaboration at different points in time.

- starting with a digital world, physical boundaries from the physical world, such as walls and furniture, appear digitally within the experience to help users avoid physical objects.

In Pure Mixed Reality (PMR), users are placed in the real world and digital content is totally integrated into their surroundings, so that they can interact with both digital and real contents, and these elements can also interact among them (Fig. 3.31) [Flavián et al., 2019].

3.6.1 Holograms that overlap reality

Holograms are objects made of light and sound that appear in the world, just as if they were real objects. Holograms add light to the world, which means that a person sees both the light from the MR tool display and the light from your surroundings. Holograms can have many different appearances and behaviors. Some are realistic and solid, and others are cartoonish and ethereal. Holograms can highlight features in your surroundings, and they can be elements in an app's user interface.

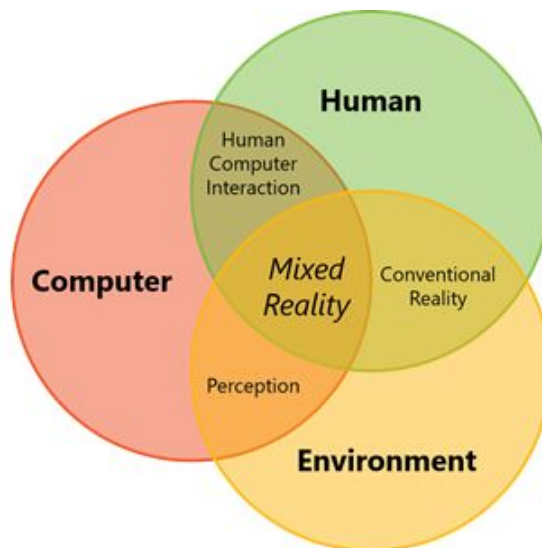


Figure 3.30: Mixed reality experience as the combination of three inputs: computer processing, human input, and environmental input [Microsoft, 2018e]

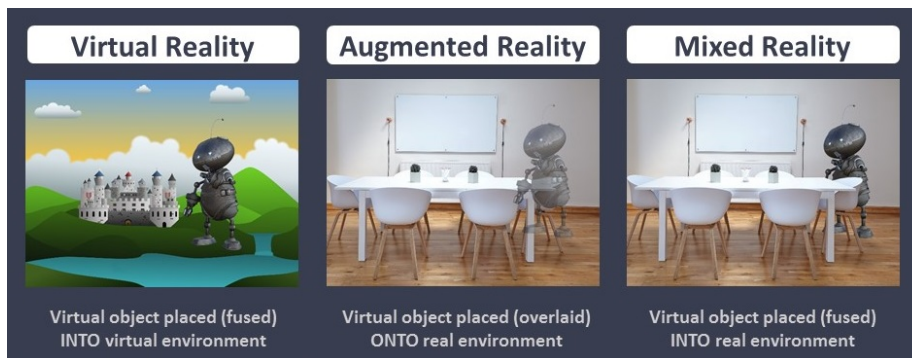


Figure 3.31: Differences between Virtual, Augmented and Mixed Reality with regard to the connection with real environment [Haeuss, 2017].

Holograms can also make sounds, which will appear to come from a specific place in the surroundings. Not only an hologram can be placed precisely in the world and walking around it, also it will appear stable relatively to the surrounding world. Using a spatial anchor to pin that object firmly to the world, the system will even remember where it has been left once back. Holograms can also be set to follow the user. Two meters is usually the most optimal distance between the user and the hologram (Fig. 3.32).

Holograms are not only about light and sound; they are also an active part of the world since they can even interact with the surroundings. For example, it is possible to place a holographic bouncing ball above a real table. Holograms can also be occluded by real-world objects. For example, a holographic character might walk through a door and behind a wall, out of the user sight [Microsoft, 2018a].

There are some tips that developers must follow for integrating holograms and the real world:

- Aligning to gravitational rules makes holograms easier to relate to and more believable.
- Many designers have found that they can even more believably integrate holograms by creating a "negative shadow" on the surface that the hologram is sitting on.

To ensure maximum comfort on head-mounted displays, it is important for designers and developers to create and present content in a way that mimics how these cues operate in the natural world. Users see the world of mixed reality through lenses working as displays powered by the headset. Developing Mixed Reality apps mostly means placing holograms in your world that look and sound like real objects. This involves precisely positioning those holograms at places in the world that are meaningful to the user. Spatial mapping makes it possible to place objects on real surfaces.

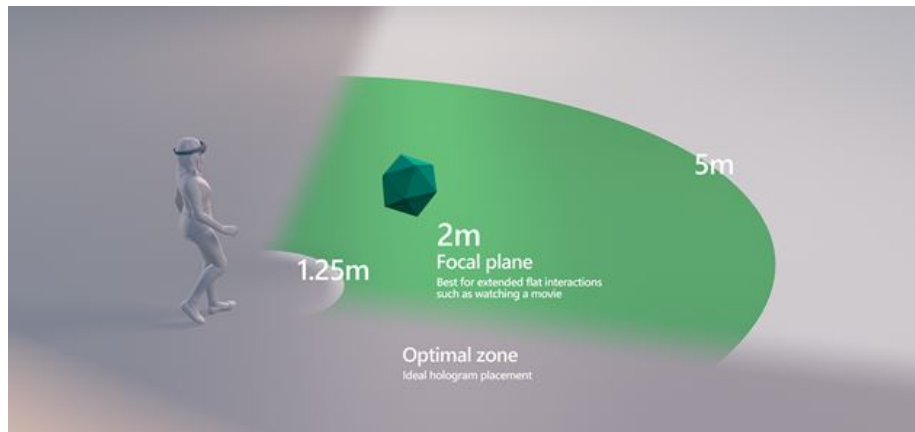


Figure 3.32: Optimal distance range for placing holograms [Microsoft, 2018a].

This helps anchor objects in the user's world and takes advantage of real world depth cues. Users can touch and manipulate holograms directly with one or both hands much like they do with real world objects. Designing content for mixed reality requires careful consideration of color, lighting, and materials for each of the visual assets used. This means incorporating as many of the visual cues as we can that help us (in the real world) understand where objects are, how big they are, and what they are made of [Microsoft, 2018b].

One possible tool for the creation of Mixed Reality applications is Unity. Unity is a cross-platform game engine developed by Unity Technologies and used to develop video games for PC, consoles, mobile devices and websites. In order to work with Mixed Reality inside a Unity project there are a small set of Unity settings that need to be manually changed for Windows Mixed Reality. These are broken down into two categories: per-project and per-scene. Otherwise Microsoft developed the Mixed Reality Toolkit (MRTK) which is now at his version no. 2. The MRTK v2 with Unity is an open source cross-platform development kit for mixed reality applications. All of the core building blocks for mixed reality applications are exposed in a manner consistent with other Unity APIs. They are:

- Camera: when you wear a mixed reality headset, it becomes the center of your holographic world. The Unity Camera component will automatically handle stereoscopic rendering and will follow your head movement and rotation when your project has "Virtual Reality Supported" selected with "Windows Mixed Reality" as the device.
- Coordinate systems: which is fundamental for precisely positioning and orienting holograms at places in the world that are meaningful to the user.
- Gaze: conceptually, head-gaze is implemented by projecting a ray from the user's head where the headset is, in the forward direction they are

facing and determining what that ray collides with. In Unity, the user's head position and direction are exposed through the Unity Main Camera, specifically.

- Gestures and motion controllers: there are two key ways to take action on your gaze in Unity, hand gestures and motion controllers in HoloLens and Immersive HMD. You access the data for both sources of spatial input through the same APIs in Unity.
- Voice input: with the KeywordRecognizer, your app can be given an array of string commands to listen for. With the GrammarRecognizer, your app can be given an SRGS file defining a specific grammar to listen for. With the DictationRecognizer, your app can listen for any word and provide the user with a note or other display of their speech.
- Persistence: The WorldAnchorStore is the key to creating holographic experiences where holograms stay in specific real world positions across instances of the application. This lets your users pin individual holograms or a workspace wherever they want it, and then find it later where they expect over many uses of your app.
- Spatial sound: spatial Sound, in Unity, is enabled using an audio spatializer plugin.
- Spatial mapping: this topic describes how to use spatial mapping in a Unity project, retrieving triangle meshes that represent the surfaces in the world around a HoloLens device, for placement, occlusion, room analysis and more.

There are other key features that many mixed reality applications will want to use that are also exposed to Unity apps:

- Shared experience
- Locatable camera
- Focus point
- Tracking loss
- Keyboard

Once the holographic Unity project is ready for testing, the following step is to export and build a Unity Visual Studio solution. With that VS solution in hand, it is possible to run the application in one of three ways, using either a real or simulated device [Microsoft, 2018d]:

- Deploy to a real MR device;
- Deploy to an emulator;

- Deploy to a simulator.

Whether you're creating a tailored experience just for Mixed Reality or porting an existing VR game, Unity has also unlocked access to an entirely new range of Mixed Reality devices.

3.6.2 The MR tool

The MR tool chosen for this project is the Microsoft HoloLens (Fig. 3.33). This device is a wearable computer that draws holograms on the environment where the user is. User interaction is quite realistic because it is done through gestures and voice commands. All features of the glasses are inside the device, including CPU, GPU and an Holographic Processing Unit (HPU), letting the user move freely in any environment. The holography that HoloLens® produces can be classified as mixed reality (MR) [Adams and Hannigan, 2018].

The list of all the technical features of the HoloLens is following [Microsoft, 2018c]:

- Display

HoloLens has see-through holographic lenses (Fig. 3.34)

- Optics See-through holographic lenses (waveguides);
- Holographic resolution 2 HD 16:9 light engines producing 2.3M total light points;
- Holographic density >2.5k radiants (light points per radian);
- Eye-based rendering Automatic pupillary distance calibration.

- Sensors

HoloLens has sensors for understanding its environment and user actions (Fig. 3.34)

- 1 inertial measurement unit (IMU);



Figure 3.33: Microsoft HoloLens [Microsoft, 2018c].

- 4 environment understanding cameras;
- 1 depth camera;
- 1 2MP photo / HD video camera;
- Mixed reality capture;
- 4 microphones;
- 1 ambient light sensor.

- Power

- Battery Life;
- 2-3 hours of active use;
- Up to 2 weeks of standby time;
- Fully functional when charging;
- Passively cooled (no fans).

- Processors

- Intel 32-bit architecture with TPM 2.0 support (Fig. 3.34);
- Custom-built Microsoft Holographic Processing Unit (HPU 1.0) (Fig. 3.34).

- Memory

- 64 GB Flash;
- 2 GB RAM.

Even though Microsoft HoloLens is an outstanding piece of technology with very robust computer vision techniques to position virtual object in the real world and a system that reconstruct the mesh of our environment in real-time, it is only at his first developer version and so designers have to cope with a few limitations while designing an UI/UX for it. For instance, the device has a very narrow field of view, so content is rapidly disappearing from the user frustum view. It is important to provide feedback in the main view of where the content has been left out or use tag along techniques to ensure that it is always a glance away from users. Moreover, HoloLens is not suited to display content closer than 85 centimeters, which prevents users from observing holograms up close. In the case of our application this only have a small impact on the user comfort but nothing critical [Fonnet et al., 2017].



Figure 3.34: HoloLens display, sensor and processor [Microsoft, 2018c].

3.7 Collected information storing

Data collected during the survey must be stored with a format which allow their further use. After the lately widespread adoption of BIM approach it is essential that data can be linked or returned to BIM objects. At the same time in the case of information production an attempt to replicate the process of writing an .ifc file from skratsh can be complex because of relations between entities and the need to use Express schema. Moreover it happens often that specific object inside buildings are not linked to a specific IFC entity and this lead to difficulties in classifying the information.

For this reason and in order to have more freedom in classes definition the process proposed in this research take into consideration the use of .xml format. Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents/information in a format that is both human-readable and machine-readable. The W3C XML (Extensible Markup Language) is a profile (subset) of SGML (Standard Generalized Markup Language, ISO 8879 - 1986) designed to ease the implementation of the parser compared to a full SGML parser, primarily for use on the World Wide Web. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services. XML uses tags to label, categorize, and organize information in a specific way. Markup describes document or data structure and organization. Content, such as text, images, and data, is that part of the code that the markup tags contain. XML is not limited to a particular set of markup — you create your own markup to suit your data and document needs. The flexibility of XML has led to its widespread use for exchanging data in a multitude of forms. One of the most useful functions of XML involves classifying information. Giving your tags meaningful names that actually reflect the content makes it easier to work with the information.

An example of an .xml file collecting data on a detected fire extinguisher is reported in Figure 3.35. Among all the data contained in the file the one pointed out by the red arrows are the most significant. They are the confidence score reached by the recognition process, the type of object recognized, the coordinates for inserting the object inside the building according to the process and the reference point that will be described in Chapter 4, Section 4.4.

Classifying the information as shown here makes it possible the storage, the

```

<?xml version="1.0" encoding="UTF-8"?>
<data>
  <message>
    <extras>
      <_msgid>b8ba367a</_msgid>
    </extras>
    <time>2019-05-16T13:06:39.536Z</time>
    <topic>holo/holo05/fireProtection/assets</topic>
  </message>
  <node>
    <name>holo/holo05</name>
    <type>mixedRealityDevice</type>
  </node>
  <space>
    <name>Q145030_A</name>
    <building>Belluschi-3B</building>
  </space>
  <sensor>
    <name>HoloFireProtectionApp</name>
    <observation>
      <confidence>0.9</confidence>
      <sequence>266</sequence>
      <objecttype>fireextinguisher</objecttype>
      <position>insertionpoint</position>
      <unit></unit>
      <value>
        <x>3.36</x>
        <y>4.43</y>
        <z>1.51</z>
      </value>
      <valueTime>2019-05-16T13:06:39.536Z</valueTime>
    </observation>
  </sensor>
</data>

```

Figure 3.35: Example of .xml format file for data transfer.

search and the retrieval with ease. Moreover, XML excels at allowing to create rules for the format of your data. Using either Document Type Definitions (DTDs) or XML Schemas it is possible to validate data, to ensure the accuracy of the collected information, ensure that the information gathered is in the most usable format for our needs.

Finally .xml files can be made to converge into a BIM server where all data can be gathered, including BIM models, and they can be queried for further operations.

3.8 Conclusion

This chapter starts from the definition of AECO industry issues addressed by this research and the explanation of use cases that can benefit from the system proposed. Among them the first scenario is the one developed with this research since it represents also the first necessary step so as to collect all the functional data fundamental to pursue the following operations.

With the aim of collecting building components data the use of neural network for object recognition has been investigated. YOLO neural networks have been

chosen for their good results in multiple object detection and their speed, which represents a crucial feature in real-time applications.

Among those networks the one used is YOLO tiny V2 which is more efficient with small datasets. The training framework for the neural network is Darknet-19, following the network developer directions. Starting from the dataset collected from both shot and collected pictures in this case the images need to be prepared for the training. The operations to pursue are the augmentation, if necessary, and drawing the bounding box and assigning the class to the objects in the pictures that one wants to recognize. The following step is the preparation of four files essential for training: .data file, .names file, .cfg file and .weights file only in case of using pre-trained network. At the end of the training the customized network is tested and precision, recall and F1 performance parameters calculated.

Finally in this chapter the role and features of MR have been explained. The possibilities provided by the holograms capability of overlapping and interacting with real world objects gives a big visualization power on-site. This means on one hand that information can be easily displayed to technicians, on-site and in real-time. On the other hand the interaction with real object and surfaces brings the user into a MR immersive realistic experience keeping the contact with the surrounding environment. These features lead to a profitable man-machine collaboration that this research aims to exploit in order to improve efficiency during building survey operations.



Figure 3.36: Example of fire extinguisher dataset as a set of pictures.



Figure 3.37: Example of signs dataset as a set of pictures.

OBJECT RECOGNITION SYSTEM

4.1 Introduction

In this chapter the developed methodology of the whole system will be detailed. Starting from the system architecture every component will be described both in terms of its technical features and in terms of the role it plays. Everything has been divided into two main environments: the MR environment and the Real environment. This division depends upon the different nature of the different parts. Some pieces of the architecture are hardware, devices like the Hololens, some others are software or applications developed through scripts, such as the recognition application or the MR. For this reason and in order to facilitate the understanding the differences between what is real and what is digital this division has been preferred.

Then in this chapter the communication between different hardware components will be detailed. The Hololens and the Raspberry communicate by means of a network socket. The Hololens is connected directly to the wireless hotspot created by the Raspberry. The piece of information that flows from the hololens to the Raspberry consists in the image containing the object that has to be recognized and the information retrieved by the recognition procedure response is the class and location of the detected objects. The recognition process converts pictures into information that can enrich a BIM model.

Finally the simulation of the steps performed by a technician has been given. The basic assumption is that the technician performs an indoor on-site survey.

4.2 Object Recognition System development

The development of the object recognition application started from the definition of all the components of the system. Defining the system architecture entails in the first place the identification of both the hardware parts and the information flow.

In this research, the field of action was divided into two different environments: mixed reality environment and real environment (Fig. 4.1). The first one refers to all the software components while the second is composed by the hardware units and the real world objects. They will be deeper described in the following

paragraphs.

One of the most valuable advantage of the system proposed is its feature of being wearable on-site. Thanks to the Neural Compute Stick Movidius, which gives the possibility to run neural networks locally, and the mixed reality, which allows the visualization and verification of data in real-time, the post-processing phase for interpretation of collected data is eliminated.

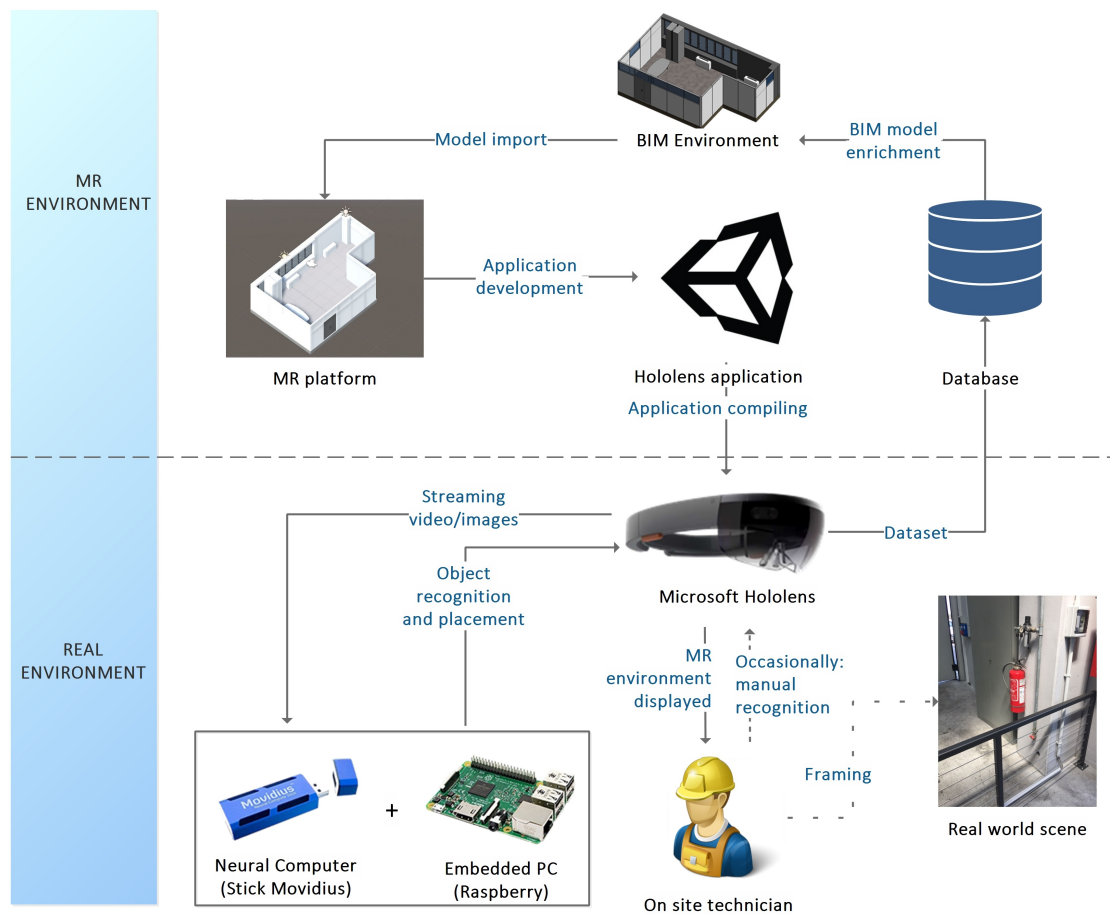


Figure 4.1: System architecture.

4.2.1 Mixed Reality environment

The MR environment (Fig. 4.1) comprises all the software elements and interfaces of the system.

The approach proposed in this research starts from the hypothesis that a geometric BIM model of the building already exists. This could be the product of a translation in a 3D building information modeling of previous CAD drawings,

for instance in existing buildings, or coming from advanced techniques for geometric surveys like the ones presented in the literature review.

4.2.1.1 BIM model

The system starts with the geometric BIM model of the environment. This model is supposed to be lacking of semantic and functional data, assets and components of technical details that must be automated generated by the survey procedure.

4.2.1.2 MR platform

The Mixed Reality environment is setup and configured by means of the software development platform, Unity in this case, that allows the development of applications for MR devices.

The recognition application was developed in Unity, using the programming language C#. Two tools are necessary in order to develop and install the application for the Hololens. The Mixed Reality Toolkit has to be installed in Unity. This allows to develop a Mixed Reality Environment for Hololens in Unity. The second necessary tool is Visual Studio which is used both for editing and debugging C# scripts and for deploying the application on the Hololens.

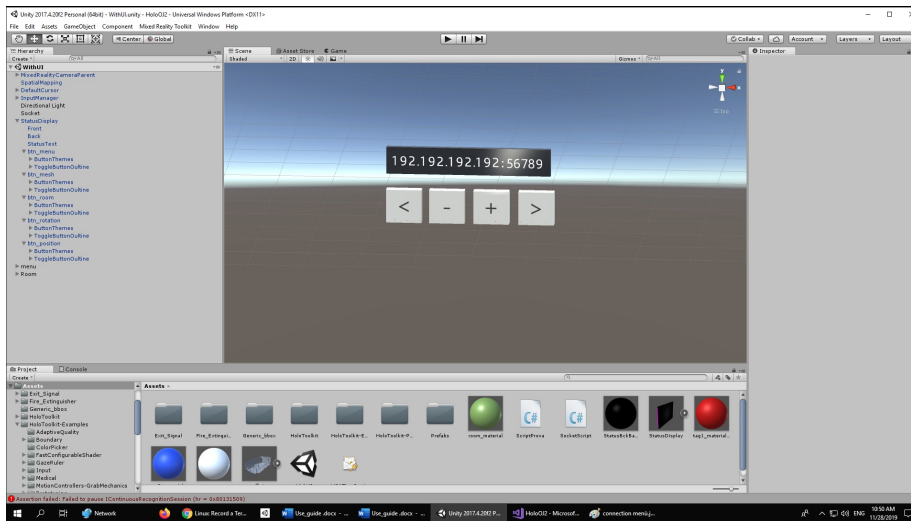


Figure 4.2: Connection menu.

4.2.1.3 Hololens application

The developed recognition application pursues the following tasks:

1. it connects to the Raspberry equipped with Movidius Neural Compute Stick (NCS);
2. it reads and interprets the data about the position of the operator in the virtual scene which is represented by the virtual model of the building;
3. it keeps track of the position of the gaze (the point that the operator is looking at) in order to place the identified component;
4. it records a list of positions of the operator, because of the time interval between the dispatch of the image and the information comeback. With the list of positions it would be possible to locate the object in the right position. The time interval has to be set with a predefined value so as it would be possible to know exactly which position in the list should be taken;
5. it sends the streaming video (or snapshots to the embedded system);
6. it reads the recognition response provided by the neural network (bounding box coordinates, object type and photo id);
7. it selects among many objects in a predefined library the correct type according to the type specified in the recognition response;
8. it locates the recognized object as hologram inside the scene;
9. it provides the possibility of modifying the object type or its position manually;
10. it provides the possibility of adding objects manually.

The scene inside the application includes the imported building model. Despite Autodesk Revit provides the common interchange FBX format, in this case a seamless integration has been realized among the BIM model and the serious virtual game engine, building up a specific tool, defined “IFC loader”. It allows importing contextual, geometrical and material properties data from the BIM model once exported in IFC XX format. It consists of an Asset for the virtual reality engine, constituted by several C# scripts aimed at importing the main data from the IFC model. This tool allows also the automatic update of the data in the virtual engine whenever a modification is made in the original Revit model. Once added the building model, the user interface and the application operations must be developed. Game objects, drop-down lists, menus, rules and behaviors have been added. The first menu for the connection between the MR tool (Hololens) and the Embedded system as explained in detail in section 4.3 and shown in Figure 4.2. In Figure 4.5 it is possible to see the main menu of the application. Since the capture of snapshots from the Hololens the quality of the image is not very high the menu options are explained below:

- "Connection information" the fuchsia box contains information about the connection with the embedded system;

- "Hide menu" for showing or not the menu in the scene;
- "Show meshes" for showing or not the result of the Hololens spatial mapping process;
- "Show room", this shows the user the digital model of the room where the technician is performing the recognition process displaying modeled spaces (walls, floors and ceilings);
- "Catch rotation";
- "Catch Position", this and the previous options are used to locate the user with precision inside the room; the complete process will be explained in section 4.4.

Game objects can be visualized through holograms of the object classes detected by the recognition process. Rules and Behaviors are used to develop the steps of the recognition procedure and to assign features to the holograms. An example of rule is that the fire extinguisher hologram can be located only on a wall and at a reasonable height from the ground.

The language to develop the application is C# (Fig. 4.3 and Fig. 4.4). Once the application is ready it has to be deployed inside the MR, i.e. Hololens. The tool for textual editing, compiling and deploying the application to be used with the Hololens is Visual Studio. In this case Unity2017 and Visual Studio 2017 have been used.

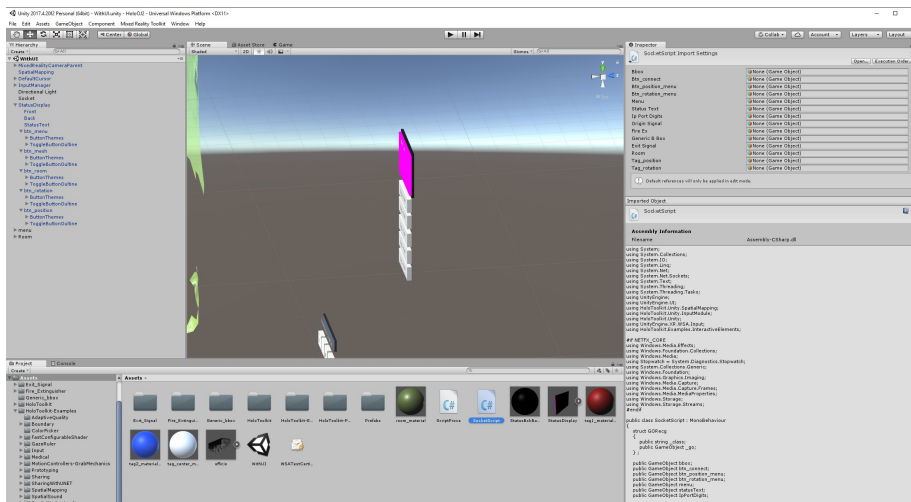


Figure 4.3: Excerpt of the script for the connection with the Raspberry inside the Unity development framework.

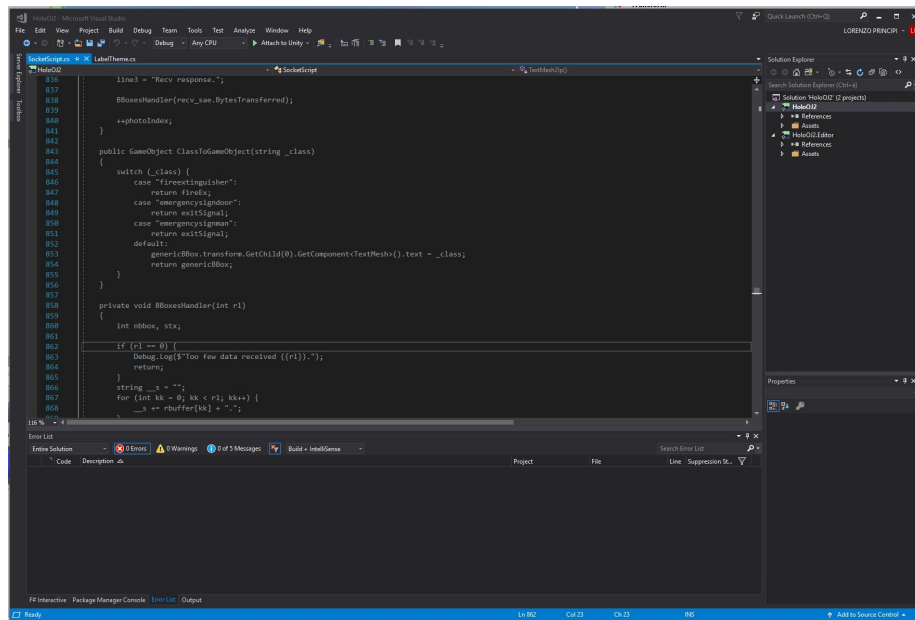


Figure 4.4: Script for FE hologram insertion in Visual Studio 2017.

4.2.1.4 Database

The same digital environment is the place where data are transferred to a BIM server, the database. As far as the BIM model enrichment is concerned it depends upon the kind of information. It could be not necessary to recreate the IFC format of information with some data that could be exploited better inside a DB. On the other hand other data could be more useful if inserted in the BIM model, especially for visualization inside spaces. However the data translation to the IFC format has not to be considered essential in order to use information efficiently. On the other hand the method chosen for storing collected information as been expressed in 3.7.

4.2.2 Real environment

The real environment (Fig. 4.1) is composed by all the hardware tools exploited in this research plus the possible real world scene the user would see.

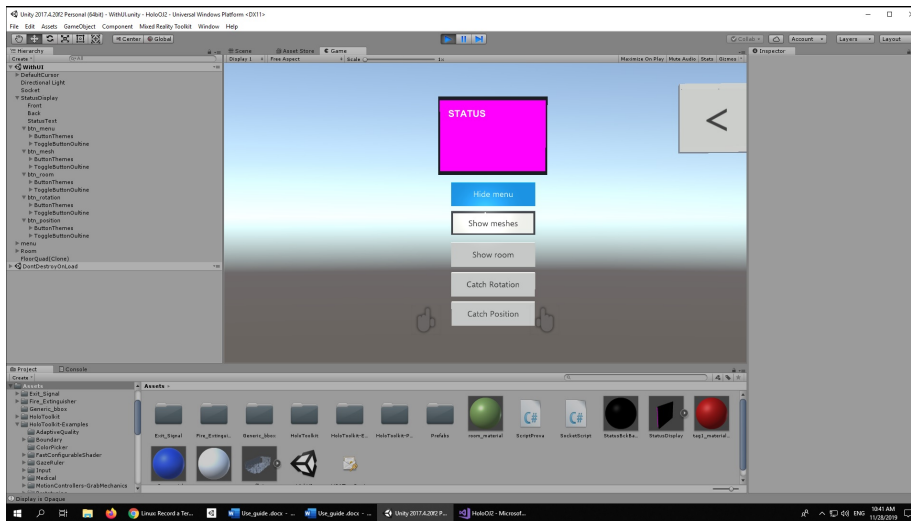


Figure 4.5: Application menu.

4.2.2.1 Microsoft Hololens

Microsoft Hololens is the head-mounted display chosen to show the MR environment on site and to act as an interface between the Mixed Reality displayed and the operator. Its technical features have been deeply explained in paragraph 3.6.2. The capabilities provided by Microsoft Hololens and exploited by the proposed system are resumed here:

1. displaying information on site through the overlapping of the virtual word and the real one;
2. providing spatial information thanks to its sensors;
3. allowing to perform the data enrichment process on site and in real time;
4. giving the possibility of checking the data directly on site.

Another task performed by the Hololens is to take pictures of the real world, framing the object that must be recognized. Then the pictures was sent to the embedded system by means of the application inside the device. Finally, once the response comes back the application running on the Hololens it places the hologram of the recognized component inside the space.

4.2.2.2 The embedded system

The choice of performing the recognition process through an embedded system comes from the will of avoiding the use of building Wi-Fi on one hand and the corresponding increased latency when for a response from a server system on the

other.

Despite the fact that the majority of buildings have Wi-Fi networks installed that can be exploited by this system, one of the aim of this research was not to rely on systems already present nor to equip the building with any kind of sensors or tags. The hypotesys is that the process can be pursued even if the technician is entering the building for its first time and without the need for preliminary operation inside the spaces.

The embedded system developed by this research is made of two components: the Raspberri Pi 3 B+ and the Intel Movidius Neural Compute Stick (version 1).

4.2.2.3 Raspberry

The Raspberry Pi 3 B+ belongs to a series of small single-board computers. All models feature a Broadcom system on a chip with an integrated ARM-compatible central processing unit (CPU) and on-chip graphics processing unit (GPU). The Raspberry Pi 3 Model B+ has the following technical features:

- Quad core 64-bit processor clocked at 1.4GHz;
- 1GB LPDDR2 SRAM;
- Dual-band 2.4GHz and 5GHz wireless LAN;
- Bluetooth 4.2 / BLE;
- Higher speed ethernet up to 300Mbps;
- Power-over-Ethernet capability (via a separate PoE HAT).

The operating system installed is Raspbian 14. In this system the Raspberry works mainly as an interface between the Hololens and the Movidius, and as a hardware support for the latter. It is the means that allows the transfer of images taken on-site to the Movidius to be processed.

Furthermore the Raspberry Pi can be used as a wireless access point, running a standalone network. This can be done using the inbuilt wireless features of the Raspberry Pi 3. This feature has been used for avoiding the use of a local Wi-Fi inside the building. The hololens, infact, is directly connected to the wireless created by the Raspberry and this is their means of communications as explained later in section 4.3.

4.2.2.4 Movidius

Intel Movidius Neural Compute Stick (NCS) is a tiny fanless USB drive designed to learn efficiently implement and run deep neural networks. The NCS is powered by the low power high performance Movidius Visual Processing Unit (VPU) (Fig. 4.6). The VPU inside the Movidius Neural Compute Stick is the Intel Movidius Myriad VPUs. They are full-fledged system-on-chips (SoC)

designed specifically for on-device computer vision and neural network applications. Myriad VPUs have dedicated architecture for high quality image processing, computer vision, and deep neural networks, making them suitable to drive the demanding mix of vision-centric tasks in modern smart devices. The first Movidius (they are now at the second version) used in this research project came with 12 Streaming Hybrid Architecture Vector Engine Cores that can run computations in parallel (Fig. 4.6).

The benefits of choosing the NCS for running Neural Networks can be resumed as follows:

- real-time on device inference, cloud connectivity not required;
- energy-efficient CNN processing;
- all data and power provided over a single USB type A port
- the possibility of running multiple devices on the same platform to scale performance.

There are three steps involved in running deep learning models on edge devices powered by Intel NCS (Fig. 4.7):

1. Training the model on a GPU-based infrastructure using TensorFlow or Caffe;
2. Optimizing the trained model as a compiled graph to run on Intel Movidius;
3. Loading the graph onto the device for inferencing.

There are several NNs that can be used within the Movidius and they depend also upon the platform used for transferring the network inside the Movidius. In this research Tensorflow has been used and the following ones represented the networks already tested:

- Facenet based on inception-resnet-v1;
- Inception v1, v2, v3,v4;
- Inception ResNet v2;
- Mobilenet V1 1.0;
- TinyYolo v2 via Darkflow transformation;
- VGG 16 (Configuration D);
- SSD Inception v2;
- SSD Mobilenet v1, v2.

In this case TinyYolo v2 has been used for its well assessed higher speed and its better performances in recognizing objects.

In order to make the customized CNN usable with NCS it is necessary to convert them using the Movidius Neural Compute Software Development Kit (NCSDK). Intel NCS can be attached to an Ubuntu 16.04 PC or a Raspberry Pi running Raspbian Stretch OS, as in this case.

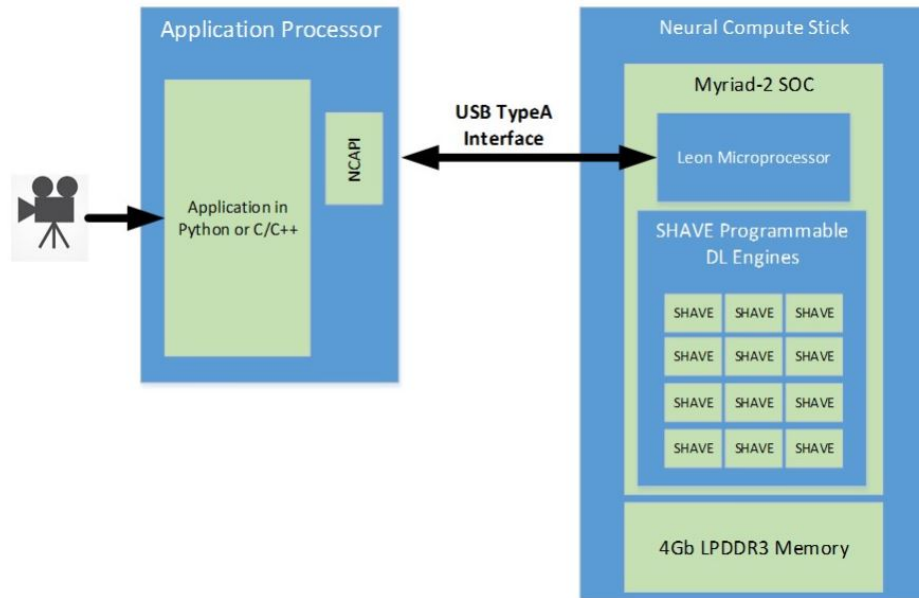


Figure 4.6: Movidius processor[Intel, 2019a].

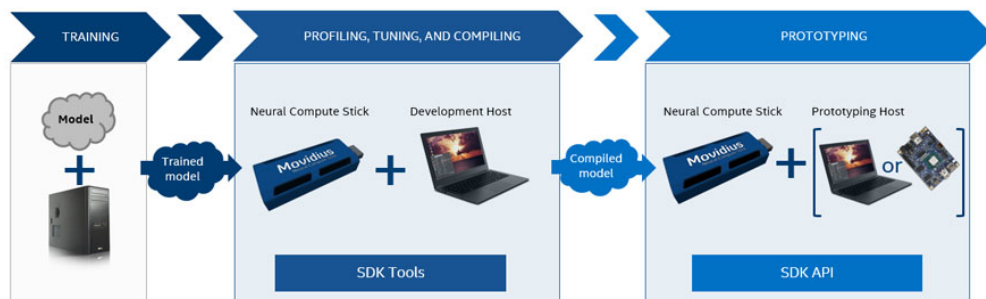


Figure 4.7: Movidius training and use [Intel, 2019b].

4.2.2.5 On-site technician

The technician on-site performs the survey wearing the Hololens. In this first development the technician has to take a picture of the scene including the object he wants to recognize. The picture is taken through the air-tap gesture with the hand held upright, similar to a mouse click or select. This is used in most HoloLens experiences for the equivalent of a "click" on a UI element after targeting it with Gaze.

With future developments of the application this step could be eliminated and becoming automatic.

The technician can perform also another task. In case of failed recognition of the object the operator on-site will have the possibility of manually inserting the building component choosing among a library of possible objects.

4.2.2.6 Real world scene

The real-world scene reported in the architecture is the surrounding environment that the on-site technician is called to interpret. For this reason it has been included into the system architecture, because it is the object of the recognition process itself.

4.3 Data transfer

Developing this architecture a main issue comes to light: the communication between different devices. Among these communication issues the main one was represented by the channel Hololens-Raspberry.

Each frame captured from Hololens camera video stream had to be sent to the Raspberry Pi. Single frames has been preferred to video stream for the following reasons:

- it's easier to associate each photo with its 3D-information obtained at the instant of the shot;
- there is no need for the support of Real Time Protocol required by video streaming that could degradate the performance of the Hololens Device, already in charge of overlapping reality with holograms in real-time;
- neural networks inputs for object recognition are single frames.

Each frame captured by the Hololens goes through the following steps Fig. 4.8:

- i. sending pre- processing;
- ii. sending;

- iii. receiving;
- iv. sending post-processing;
- v. pre-processing for neural networks;
- vi. neural network inference;
- vii. send back NN output to Hololens.

Steps (i) and (iv) can be jpeg-compression, cropping and/or scaling. Steps (v) depends on the NN. Exchanging data between Hololens and Raspberry required the development of a custom socket over a WiFi network between Raspberry (hotspot) and Hololens (client). The socket can be either TCP or UDP. Table 1 ?? shows possible configurations. Data rate is in the worst case, 30 MBps, its measured range is about 50 ± 20 Mbps (uncertainty is due to unpredictable network events in both TCP or UDP socket).

Since the chosen NN YOLOv2 unit expects a float number and received frames are in byte format, a pre-processing step, (v), performs the conversion from byte to float.

Since the purpose is to recognize objects inside pictures, the Raspberry sends back to Hololens (through the socket) bounding boxes defined by x,y,w,h (respectively the x and y coordinates of the bounding box center and the width and height of the box) and the frame id (used by Hololens to retrieve the saved 3d information of the frame to place the hologram in the three-dimensional space). Filtering will take place only in Raspberry to reduce Hololens work.

Table 4.1 Performance tests, each one elaborate 150 photos. (a) Total time for sending raw photos (cropping and scaling timens are neglected, max 10 ms). (b) cropped from 869x504 photo, (c) scaled from 896x504.

Resolution	Size [Kb]		Times [ms]			
	raw	jpeg	encoding	sending	total	total ^(a)
1408x792	3'340	154	94	5	99	111
896x504	1'350	60	46	2	91	45
416x416 ^(b)	519 ^(b)	24	29	0.8	46	17
416x416 ^(c)	519	32	45	1.1	62	17
416x234 ^(c)	292 ^(b)	23	36	0.8	46	10

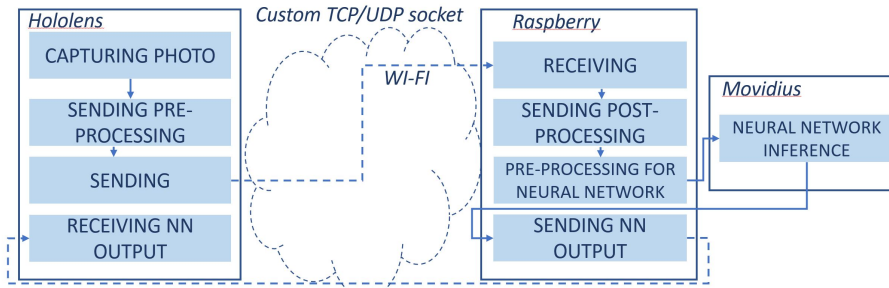


Figure 4.8: Image steps through the recognition process.

4.4 Automatic survey process

In this paragraph the whole survey process will be explained in depth. Before starting describing the process on-site two specifications have to be given. The system proposed in this research starts from a pre-existing geometric 3D building model of the construction object of the survey. This can be derived from a straightforward translation of pre-existing cad drawings or semi-automatic geometric survey techniques as exposed in Chapter 2. The reason of this assumption is that many researches focus on the collection of geometric data while still few studies have been pursued on the collection of functional data especially as far as assets under maintenance are concerned.

The second specification is that the connection between Hololens and Raspberry is not inserted in the process representation in Figure 4.11. This procedure has to be done before starting the survey. After connecting the Hololens to the Raspberry network the socket for the communication has to be set with the connection menu (Fig. 4.2). At this point the technician is ready to start the survey.

According to what explained in the previous paragraph the first step of the process (Fig. 4.11) is setting the 3D building information model of the construction object of the survey. Once inside the building the survey starts from a specific room. The second task is loading the model of the room. This operation can be done in several different ways: the room could be equipped with a tag, the number of the room displayed outside can be recognised or the space can be searched among a pre-defined library. This latter method is the method chosen in this case.

Then after entering the room the model alignment is the following task. This operation is performed in two steps: set position and set rotation. Set position adjusts the position of the room model through the selection in the real scene of an agreed point. The selection of the point means indicating the correct point with the gaze and then air-tapping (like the left click in the computer) inside the Hololens display. The selection process is repeated for the rotation setting of the model. Also in this case there is an agreed point. In order to assess

the correctness of the alignment an automatic procedure of standard deviation calculation provides an alarm in case this procedure has to be repeated.

Once this procedure is done the real recognition of the object starts. First of all the object has to be framed by the technician. Then another air-tap is necessary as it represents the recognition procedure trigger. This movement automatically takes a picture, send it to the Raspberry and makes the picture recognition starts. The Tiny YOLOv2 input is a 416x416x3 tensor, corresponding to the frames to be processed (416x416 is resolution, 3 is number of float for pixel). The network (loaded on Movidius) processes this input and returns an output of 13x13x[(C+5)x5]. The frame is divided in 507 cells of same width and height, for each cell it finds 5 bounded boxes and for each of them it calculates: width, height, horizontal and vertical distance from top left corner; objectiness which is the probability of an object to rely in this bounding box (not a particular type of object, just an object); C probabilities indicating the object category in the bounding box. Then it filters bounding boxes with objectiness lower than a fixed threshold and, for the rest ones, filter those object categories having probability lower than an arbitrary threshold.

This recognition data are then sent back to the Hololens and specifically to the recognition application. It reads the response and then it displays as an hologram the bounding box around the object and the object category (Fig. 4.10). If everything is correct it is possible to go ahead, otherwise the recognition procedure has to be performed again. If everything is correct the application places an hologram of the object inside the space. If the placement is correct it is possible to continue if there are other objects inside the room, otherwise the process is over.

The room alignment has to be done just once for every room. On the other hand the recognition procedure has to be performed for every object whose data has to be collected. Since the YOLO is able to perform multiple recognition for each picture assets, components inside the same photo are recognized simultaneously within just one recognition procedure.

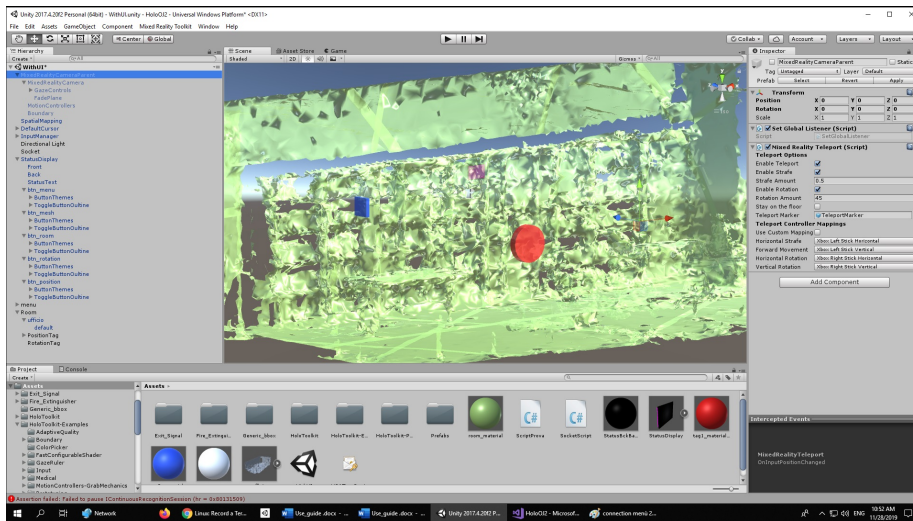


Figure 4.9: Position (the blue square) and rotation (the red circle) points.



Figure 4.10: Bounding box and object category at the end of the the recognition procedure.

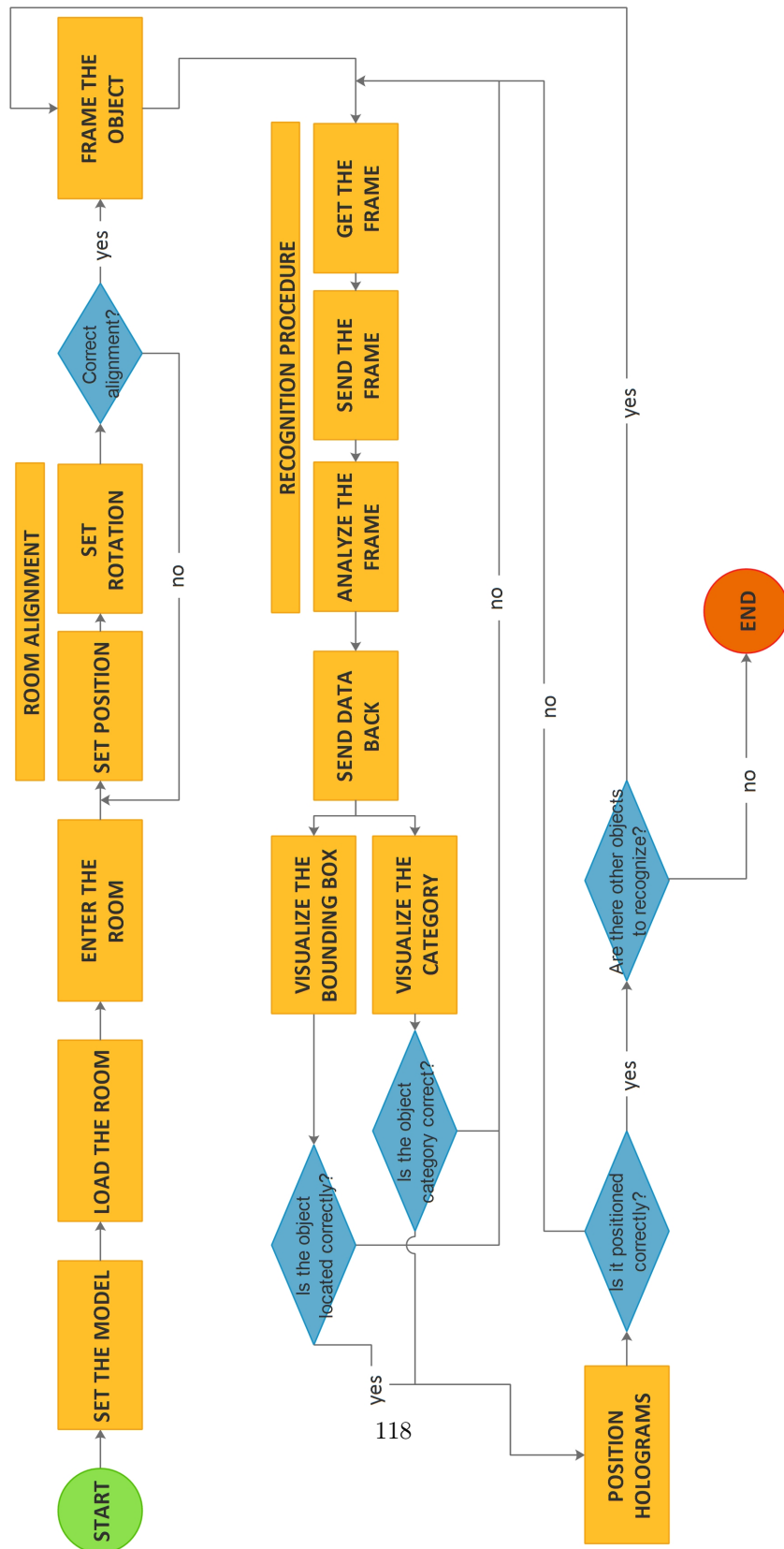


Figure 4.11: Recognition process steps.

4.5 Conclusion

This chapter illustrates the components of the system proposed. They are divided into MR environment components, referring to digital components, and real environment parts, mainly hardware, like the Hololens or the embedded system composed by the Raspberry and the Movidius. The choice of composing the system like this derives from technical features of the objects, expressed also in Chapter 3 and resumed here, and the role they play is explained in depth.

After explaining system components the data transfer solution has been detailed. The choice of working only with the network provided by the Raspberry came from the will of avoiding the use of local Wi-Fi for data transmission. Even if the majority of big constructions, such as universities, airports, stations, are already equipped with diffuse Wi-Fi connections one of the strength of the system proposed relies in its autonomy. Furthermore this connection guarantees an easier way to send and receive data between the MR tool and the embedded system.

Finally the on-site process has been exposed, from entering the building till data collection is performed. The following ones are the automatic steps:

- loading of the room model
- the whole recognition procedure, from sending the snapshot to the Raspberry, transferring it to the Movidius and analysing the frame;
- sending back to the Hololens the information;
- visualizing both the bounding box and the object category;
- positioning the hologram.

On the other hand the steps that the technician has to perform manually are:

- setting the model;
- setting position and rotation for room alignment;
- taking a snapshot of the scene where the object that has to be recognized is.

The object recognition procedure is all automatic, starting just with an air-tap that activate the capture of the image, its delivery to the embedded system and the NN exploitation. Also the display of the recognition response is all automatic leaving to the technician the task of confirming or not the results.

PROOF OF CONCEPT

5.1 Introduction

After the development of the system, as explained in Chapter 4, a proof of concept for testing the use of the whole system has been produced. In this chapter this feasibility test will be detailed.

It started from the customization of Neural Network, Tiny YOLOv2 in this case. For training the network the process started with images collection. Then several training sessions have been pursued in order to investigate the right number of pictures for an efficient object recognition. Finally customized networks have been validated in order to verify if their performances were compliant to the mAP obtained by the training.

Those tests have been done both on a desktop and in real environment. The real environment chosen have been one of the buildings of Politechnic University of Marche.

Finally a test using the whole system have been performed at DICEA Department.

5.2 Neural Network training

The realization of the first proof of concept of the whole system started from the training of Neural Networks.

The following paragraphs will describe three moments of NN training: the dataset creation, the training sessions and the validation procedure. The collection of pictures for the dataset have been pursued following the methods explained in paragraph 3.5.2.

As far as the training sessions are concerned several experiments have been done working with different types of datasets.

Finally the validation procedure has been performed as already explained in 3.5.4.

5.2.1 Datasets creation

The correct dataset to train the network should have specific features and it has to include at least one image for every existing type of object. This is due to the fact that the customized network has to be able to recognize the component no matter its external appearance, which could vary according to the object taken into consideration. Differences in geometric features and external aspect have to be included, as well as all the plausible points of view of the object. As expressed by [Radovic et al., 2017] the point of view of the object deeply affects an efficient object recognition.

The number of pictures for obtaining a customized network depends upon the number of objects or object types, plus the features of the component itself. For instance it is easier to detect a fire extinguisher, which is a red cylinder on a white wall, than electronic sockets.

In order to enlarge the dataset it has been made up of both original pictures and graphically re-edited photos. Training have been done with both datasets containing only original pictures and datasets with re-edited photos. The pictures coming from real scenarios have been taken with smartphones cameras. Dimensions can be mixed in datasets since the network chosen re-sizes pictures before performing recognition.

In this research two objects have been introduced into the datasets:

- Object 1: fire extinguisher;
- Object 2: emergency signal.

Among the pictures referring to emergency signals there was a distinction between different types of item (Fig. 5.1):

- Emergency sign;
- Emergency sign door;
- Emergency sign man;

and the total number of original images for every type of emergency sign is reported in Table 5.2. The total number of original pictures have been divided into 8 datasets as expressed in Table 5.1:



Figure 5.1: From left to right: emergency sign door, emergency sign man, emergency sign images categories.

These collections of images have been combined in different datasets for different training sessions. Starting from the dataset containing all the images of the fire extinguisher, in order to investigate network training performances, this was subsequently broken down into multiples of 100 (Table 5.4). In addition to this it has been created 4 datasets with both original images and augmented ones. In Table 5.3 it is possible to see the proportions between original and re-edited photos for two datasets referring to emergency signs and the two datasets for fire extinguishers. Finally one last dataset (COMBINED DATASET) has been formed by the combination of Dataset 500 of fire extinguishers and Dataset 7 and 8 about emergency signs. This last dataset have been used for training the network for the recognition of both objects categories at the same time.

Table 5.1 Dataset composed by original images.

Object	Dataset	Number and origin of images
FIRE EXTINGUISHER (total 1000 images)	Dataset 1	175 original images from UnivPM and Flickr
	Dataset 2	118 original images from University of Edinburgh
	Dataset 3	286 images from Google
	Dataset 4	127 images from Google
	Dataset 5	270 images from Google
	Dataset 6	24 images from Instagram
EMERGENCY SIGNS (total 581 images)	Dataset 7	45 images from UnivPM
	Dataset 8	536 images from Google

Table 5.2 Emergency signs categories original images.

EMERGENCY SIGNS IMAGES		
Emergency sign	Emergency sign door	Emergency sign man
53 images	76 images	452 images

Next paragraph will depict all trainings performed with different combination of the aforementioned datasets.

5.2.2 Training sessions and results

Several training sessions have been performed using the aforementioned datasets. All these trainings have been done exploiting pre-trained tiny YOLOv2. The chosen network came from a training with CoCo dataset and thus pre-trained weights have been used. In all cases the customized parameters of the .cfg file were: batch=64, this means we will be using 64 images for every training step; subdivision=8, the batch will be divided by 8; classes=1, the number of categories we want to detect; filters=30, from formula in Paragraph 3.5.3; learning rate=0.001, advised by the developer of YOLO in order to avoid false minimum point.

Training n.2 was performed using a not pre-trained tiny YOLOv2 and this means that no pretrained weights have been used. In this case the network starts with random weights that will be improved during the training session itself.

Finally for emergency signs training the filter parameter was changed since it depends from the number of classes. For this reason it was set equal to 40. For the training with the combined dataset (fire extinguishers + emergency signs) it was 45 (4 different classes).

Table 5.3 Dataset with both original and re-edited pictures.

Object	Dataset	Original and re-edited images
FIRE EXTINGUISHERS	Dataset 350	350 images, 175 original pictures plus 175 coming from a just rotation augmentation
	Dataset 4000	4000 images, only 175 original
EMERGENCY SIGNS	Dataset 9	1373 images, 539 original and 834 from re-editing
	Dataset 10	310 images, 155 original and 155 from augmentation

Table 5.4 Fire extinguisher training datasets from combination of original images dataset.

Training Dataset	Dataset combination
Dataset 1000	all images of fire extinguishers
Dataset 100	images taken from Dataset 1
Dataset 200	images taken from Dataset 1, Dataset 6 and 1 image from Dataset 2
Dataset 300	images taken from Dataset 1, Dataset 6 and 101 images from Dataset 2
Dataset 400	images taken from Dataset 1, Dataset 6, Dataset 2 and 83 images of Dataset 3
Dataset 500	images taken from Dataset 1, Dataset 6, Dataset 2 and 183 images of Dataset 3
Dataset 600	images taken from Dataset 1, Dataset 6, Dataset 2 and 283 images of Dataset 3
Dataset 700	images taken from Dataset 1, Dataset 6, Dataset 2, Dataset 3 and 97 images of Dataset 4
Dataset 800	images taken from Dataset 1, Dataset 6, Dataset 2, Dataset 3, Dataset 4 and 70 images of Dataset 5
Dataset 900	images taken from Dataset 1, Dataset 6, Dataset 2, Dataset 3, Dataset 4 and 170 images of Dataset 5

The following one is the list of all the training session for fire extinguishers only with the results in terms of mAP reached:

- TRAINING 1 → 1000 original pictures taking the total of images about fire extinguishers. Maximum mAP reached 89%.
- TRAINING 2 → 1000 original pictures taking the total of images about fire extinguishers. Maximum mAP reached 90,80%.
- TRAINING 3 → Dataset 100 original pictures taken among the photos of the first dataset. Maximum mAP reached 89,11%.
- TRAINING 4 → Dataset 200 original pictures taken partially from Dataset 1 plus Dataset 6 plus 1 picture of Dataset 2. Maximum mAP reached 92,86%.

- TRAINING 5 → Dataset 300 original pictures taken from Dataset 1 plus Dataset 6 plus 101 picture of Dataset 2. Maximum mAP reached 98,91%.
- TRAINING 6 → Dataset 400 original pictures taken from Dataset 1 plus Dataset 6 plus Dataset 2 plus 83 pictures from Dataset 3. Maximum mAP reached 99,30%.
- TRAINING 7 → Dataset 500 original pictures pictures taken from Dataset 1 plus Dataset 6 plus Dataset 2 plus 183 images from Dataset 3. Maximum map reached 73,43%.
- TRAINING 8 → Dataset 600 original pictures pictures taken from Dataset 1 plus Dataset 6 plus Dataset 2 plus 283 images from Dataset 3. Maximum map reached 82,73%.
- TRAINING 9 → Dataset 700 original pictures taken from Dataset 1 plus Dataset 6 plus Dataset 2 plus Dataset 3 plus 97 images from Dataset 4. Maximum map reached 82,73%.
- TRAINING 10 → Dataset 800 original pictures taken from Dataset 1 plus Dataset 6 plus Dataset 2 plus Dataset 3 plus Dataset 4 plus 70 images from Dataset 5. Maximum mAP reached 76,68%.
- TRAINING 11 → Dataset 900 original pictures taken from Dataset 1 plus Dataset 6 plus Dataset 2 plus Dataset 3 plus Dataset 4 plus 170 images Dataset 5. Maximum mAP reached 78,13%.
- TRAINING 12 → Dataset 350, 175 original pictures taken from Dataset 1 plus 175 photos coming from a simple augmentation of the first dataset (just rotation with different degrees). Maximum mAP reached 94,95%.
- TRAINING 13 → training with 4000 pictures only 175 original coming from Dataset 1. Maximum mAP reached 16,43%.

This is instead the list of all the training session for emergency signs only with the results in terms of MaP reached:

- TRAINING 14 → training with Dataset 9, 1373 pictures, 539 original and 834 from re-edited photos. Maximum mAP reached 86,40%.
- TRAINING 15 → training with Dataset 7 and 8, 581 original photos. Maximum mAP reached 80,98%.
- TRAINING 16 → training with Dataset 10, 310 original photos, 155 original and 155 from augmentation. Maximum mAP reached 84,46%.

Finally there is the training of the combined dataset:

- TRAINING 17 → COMBINED DATASET, union of Dataset 500, Dataset 7 and Dataset 8. Maximum map reached 71,98%.

After training the customized networks have to be validated so as to verify their performances.

5.2.3 Training sessions validation

The validation has been executed according to the procedure described in Paragraph 3.5.4. As far as trainings from 1 to 11 are concerned the validation dataset used has been always the same. It contained 100 original pictures. The choice of using the same dataset for all these validations led to a direct comparison among performances.

Other trainings used the validation rule advised by YOLO developers. It establishes the use of 86% of the whole dataset for training and 20% for validation. Another parameter to take into consideration was the level of confidence threshold for the validation. It has been set equal to 60% for all the validation processes. For the validation process it has been taken into consideration Precision, Recall and F1 (Section 3.5.4), calculated from the number of true positive, false positive and false negative obtained within the total of images. Figure 5.2 reported an example of a validation sheet of a network.

In order to perform the validation it was necessary to select one weight file. The weight files were saved every 1000 iterations. For this reason the weight file selected for the validation has always been the closest to the maximum mean average precision obtained. All the mAP values, total number of iterations and weight file chosen can be checked in Table 5.8.

Table 5.8 reported also all the validation parameters obtained that take into consideration the total number of instances in the validation dataset. Since some pictures could show more than one object the number is reported in Table 5.8.

Figure 5.3 shows the maximum mAP values obtained for every training session. On the other hand Figure 5.4 displays the values of precision, recall and F1 calculated through the validation process. Following these calculations it is possible to express some observations:

- it can be seen that all the F1 values are acceptable since they are higher than 80%;
- an high percentage of image augmentation deeply worsen the performances of the network;
- moderate percentage of image augmentation are still acceptable.

Also the kind of augmentation affects the dataset and the real-world performance. It is in fact pointless performing an augmentation that produces images

Chapter 5

graphically far from what is the typical image of the components object of recognition. For instance it would be unlikely to see a fire extinguisher upside down, so when setting rotation maximum degrees it would be reasonable to stop between -15° and $+15^\circ$.





	A	B	C	D	E	F	G	H	I	J	K	L
1	image	boundingbox	confidence	TP/FP/FN								
2	MNMG_Dataset_SIGN_FE\obj\03_3617254\351L											
3			100.00%: TP						VALIDATION THRESHOLD 80%		TRUE POSITIVE	378
4	GNL_FE\obj\TITLE\ED4568_everyday_4_Sig_The_worl		72.00%: TP						TOTAL NUMBER OF FE 217		FALSE POSITIVE	74
5	WTRAINING_Dataset_SIGN_FE\obj\TH_SA_FLAB										FALSE NEGATIVE	45
6			100.00%: TP								PRECISION	0.8276
7	uLFE_SIGNTRAINING_Dataset_SIGN_FE\obj\T1		99.00%: TP								RECALL	0.8533
8	7_innasalety-05-kg-dop-fire-extinguisher.jpg		72.00%: TP								F1	0.8404
9												

Figure 5.2: Validation sheet reporting the bounding box, true positive, false positive, false negative and calculating performance indexes.

Moreover from these validations it is evident that the higher the number of pictures the better the performances is not true, with or without augmentation. Following these validations that have been carried on computers, real-world test have been performed so as to verify customized network performances with changing boundary conditions.

Table 5.5 Training validations.

TRAINING 1			
Total number of FE = 277		TRUE POSITIVE	271
		FALSE POSITIVE	6
		FALSE NEGATIVE	6
no. of iterations	1754		
mAP max	89.00%	PRECISION	0.97834
validation weights	last	RECALL	0.97834
mAP weights	88.74%	F1	0.97834
TRAINING 2			
Total number of FE = 277		TRUE POSITIVE	266
		FALSE POSITIVE	8
		FALSE NEGATIVE	11
no. of iterations	1614		
mAP max	90.80%	PRECISION	0.9708
validation weights	13000	RECALL	0.96029
mAP weights	90.60%	F1	0.96552
TRAINING 3			
Total number of FE = 277		TRUE POSITIVE	273
		FALSE POSITIVE	32
		FALSE NEGATIVE	4
no. of iterations	1614		
mAP max	89.11%	PRECISION	0.89508
validation weights	421100	RECALL	0.98556
mAP weights	84.76%	F1	0.93814
TRAINING 6			
Total number of FE = 277		TRUE POSITIVE	266
		FALSE POSITIVE	8
		FALSE NEGATIVE	11
no. of iterations	8558		
mAP max	99.30%	PRECISION	0.9708
validation weights	419000	RECALL	0.96029
mAP weights	94.97%	F1	0.96552

Table 5.6 Training validations.

TRAINING 7			
Total number of FE = 277		TRUE POSITIVE	267
		FALSE POSITIVE	8
		FALSE NEGATIVE	10
no. of iterations	10989		
mAP max	73.43%	PRECISION	0.97091
validation weights	422000	RECALL	0.9639
mAP weights	73.13%	F1	0.96739
TRAINING 14			
Total number of ES = 432		TRUE POSITIVE	406
		FALSE POSITIVE	63
		FALSE NEGATIVE	26
no. of iterations	491000		
mAP max	86.40%	PRECISION	0.86567
validation weights	491000	RECALL	0.93981
mAP weights	86.40%	F1	0.90122
TRAINING 15			
Total number of ES = 158		TRUE POSITIVE	145
		FALSE POSITIVE	26
		FALSE NEGATIVE	13
no. of iterations	441000		
mAP max	80.98%	PRECISION	0.84795
validation weights	441000	RECALL	0.91772
mAP weights	80.98%	F1	0.88146
TRAINING 16			
Total number of ES = 224		TRUE POSITIVE	200
		FALSE POSITIVE	44
		FALSE NEGATIVE	24
no. of iterations	482000		
mAP max	84.87%	PRECISION	0.81967
validation weights	457000	RECALL	0.89286
mAP weights	84.46%	F1	0.8547

Table 5.7 Training validations.

TRAINING 17			
Total number of OBJECTS = 421		TRUE POSITIVE	376
		FALSE POSITIVE	76
		FALSE NEGATIVE	45
no. of iterations	434000		
mAP max	71.98%	PRECISION	0.83186
validation weights	434000	RECALL	0.89311
mAP weights	70.76%	F1	0.8614

Table 5.8 Training validations.

TRAINING	DATASET	mAP	PRECISION	RECALL	F1
TRAINING 1	DATASET 1000	89.00%	97.83%	97.83%	97.83%
TRAINING 2	DATASET 1000	90.80%	97.08%	96.03%	96.55%
TRAINING 3	DATASET 100	89.11%	89.51%	98.56%	93.81%
TRAINING 4	DATASET 200	92.86%	/	/	/
TRAINING 5	DATASET 300	98.91%	/	/	/
TRAINING 6	DATASET 400	99.30%	97.08%	96.03%	96.55%
TRAINING 7	DATASET 500	73.43%	97.09%	96.39%	96.74%
TRAINING 8	DATASET 600	82.73%	/	/	/
TRAINING 9	DATASET 700	82.73%	/	/	/
TRAINING 10	DATASET 800	76.68%	/	/	/
TRAINING 11	DATASET 900	78.13%	/	/	/
TRAINING 12	DATASET 350	94.95%	/	/	/
TRAINING 13	DATASET 4000	16.43%	/	/	/
TRAINING 14	DATASET 9	86.40%	86.57%	93.98%	90.12%
TRAINING 15	DATASET 7 AND 8	80.98%	84.80%	91.77%	88.15%
TRAINING 16	DATASET 10	84.46%	81.97%	89.29%	85.47%
TRAINING 17	COMBINED DATASET	70.76%	83.19%	89.31%	86.14%

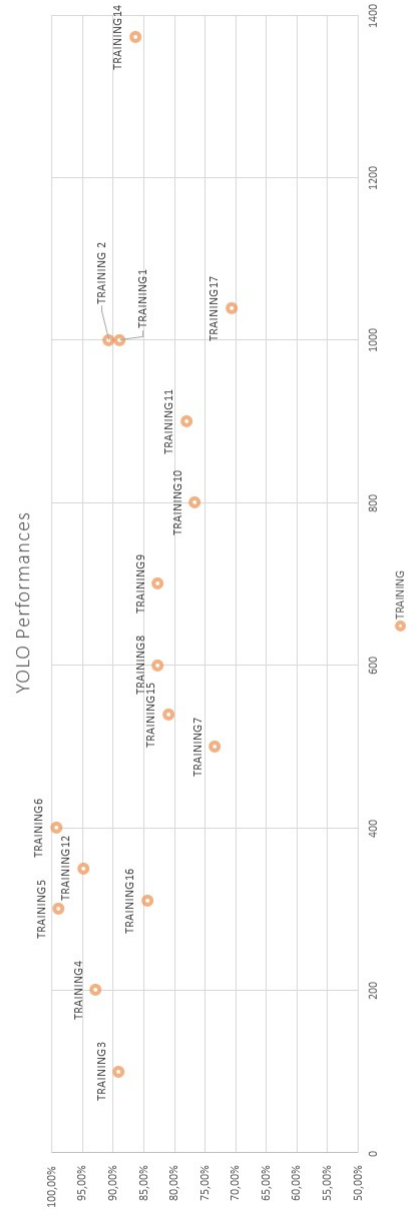


Figure 5.3: Maximum mAP for every network training (except Training 13). X axes is the number of images in the dataset used for the training.

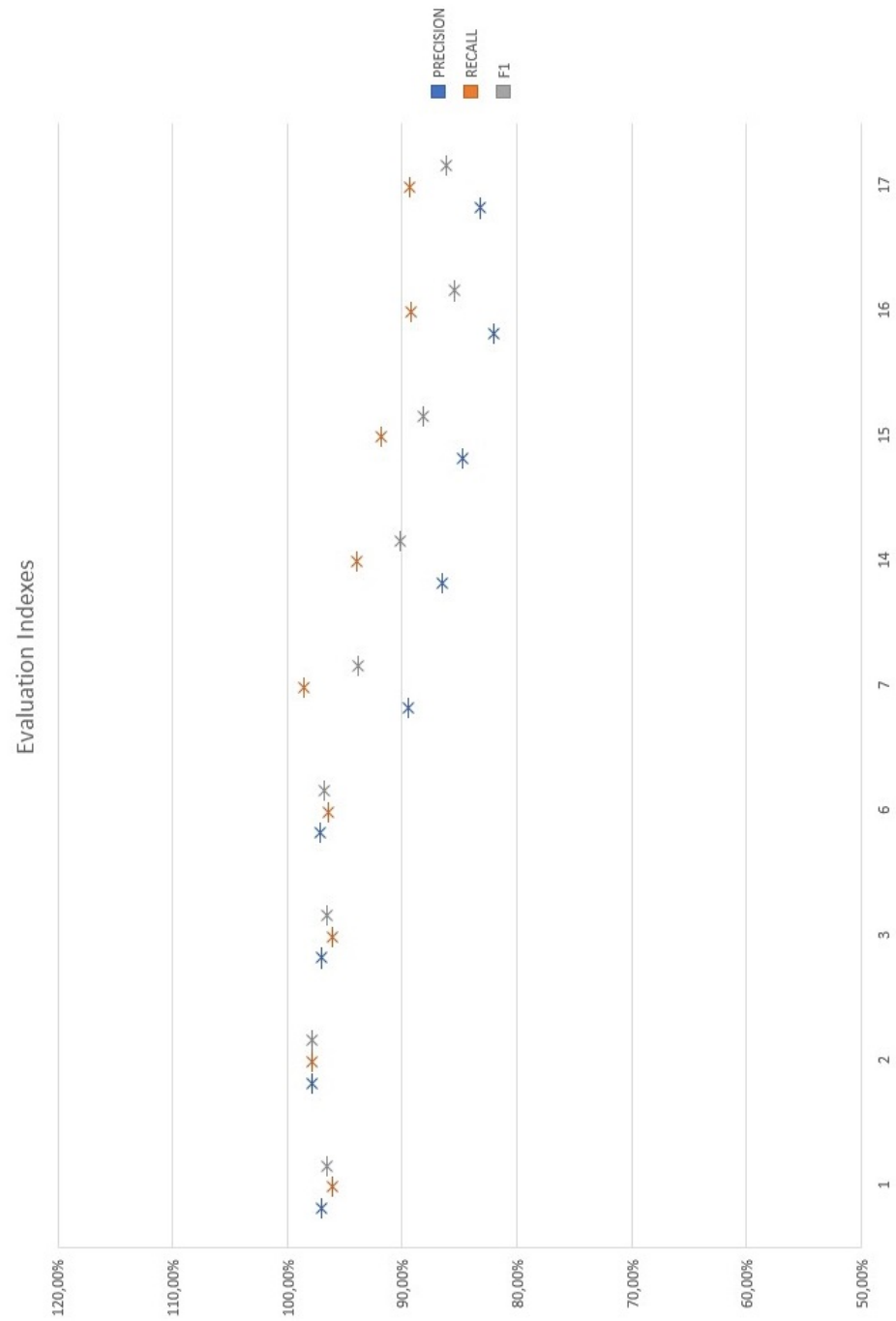


Figure 5.4: Precision, Recall and F1 values coming from networks validation. Training number on the x axes.

5.3 Testing the Neural Network performances

With the aim of testing the network in real world conditions it has been performed a test inside the Politechnic University of Marche premises. The building used for the test is denominated Blocco Aule Sud (BAS) (Fig. 5.5). It is a structure with a longitudinal development (Fig. 5.6) and three of its perimeter walls are curtain walls. This means that light conditions can be very adverse especially when using mixed reality device like the HoloLens for taking pictures of the object that has to be recognized. Additionally testing good performances with changing light conditions is one of the main worries about systems that work with images.

Another interesting thing that has to be taken into consideration with the proposed system is the point of view of the object since it could affect a good performance in recognition.

The test has been done with all the embedded system as can be seen in Figure 5.7.

The network has been always able to recognize the object although with different level of confidence (Tab. 5.9 and Tab. 5.10). Less than 10% of the object have obtained a value of level of confidence lower than 60%.

In 9 cases the recognition did not worked at the first attempt. This was due not to light condition but to the chosen point of view. When the white label usually on top of fire extinguisher was not visible the network struggled in recognize the object at the first attempt.

Photos of the point of view that worked for the recognition are reported at the end of this chapter.



Figure 5.5: BAS building.



Figure 5.6: First floor corridor showing the longitudinal development.

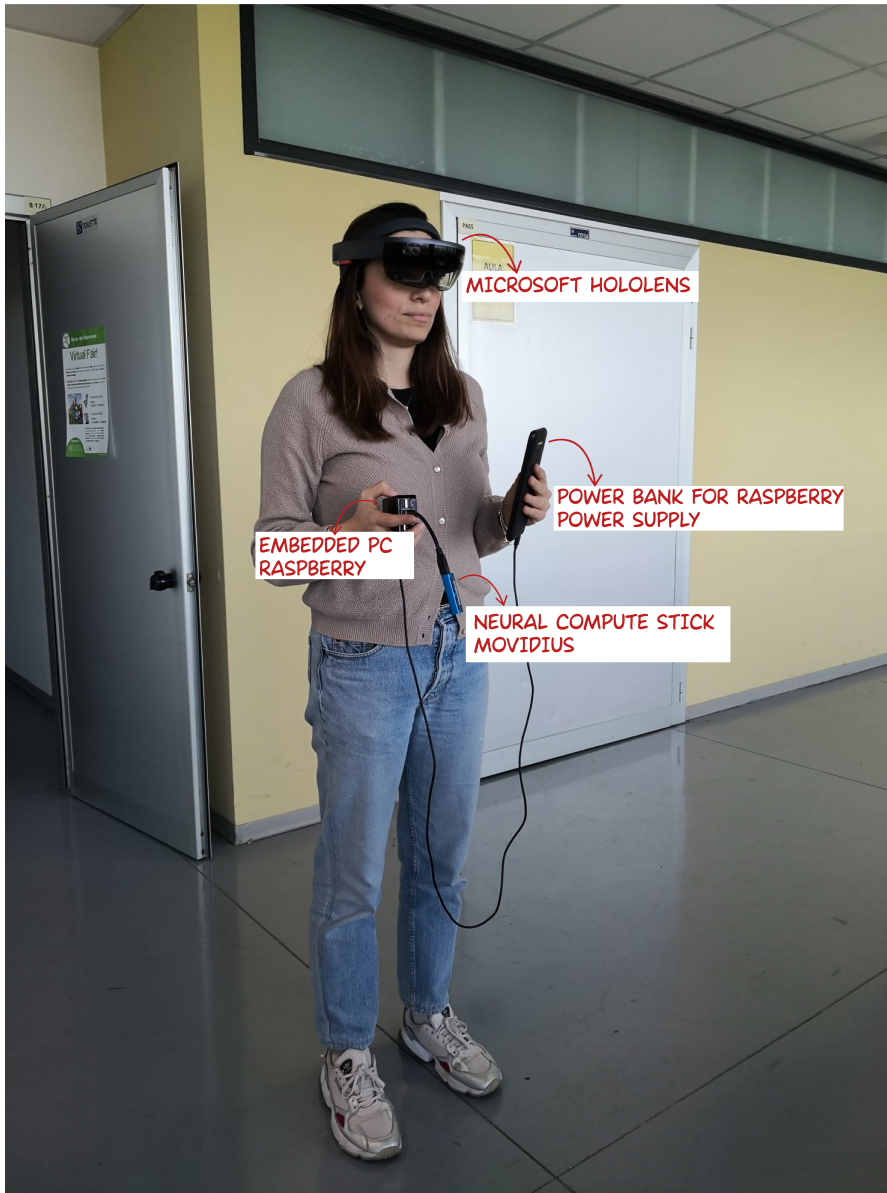


Figure 5.7: Embedded system including Movidius, Raspberry and HoloLens.

Table 5.9 Neural Network performances test results (Ground and First floors).

Floor	Number of FE	Level of Confidence
GROUND FLOOR	FE 1	88%
	FE 2	53%
	FE 3	99%
	FE 4	98%
	FE 5	53%
	FE 6	99%
	FE 7	93%
	FE 8	89%
	FE 9	98%
	FE 10	98%
	FE 11	96%
	FE 12	99%
	FE 13	92%
	FE 14	92%
	FE 15	99%
FIRST FLOOR	FE 1	95%
	FE 2	99%
	FE 3	97%
	FE 4	96%
	FE 5	99%
	FE 6	99%
	FE 7	93%
	FE 8	82%
	FE 9	99%
	FE 10	89%
	FE 11	96%

Table 5.10 Neural Network performances test results (Basement floor).

Floor	Number of FE	Level of Confidence
BASEMENT FLOOR	FE 1	99%
	FE 2	61%
	FE 3	68%
	FE 4	82%
	FE 5	55%
	FE 6	98%

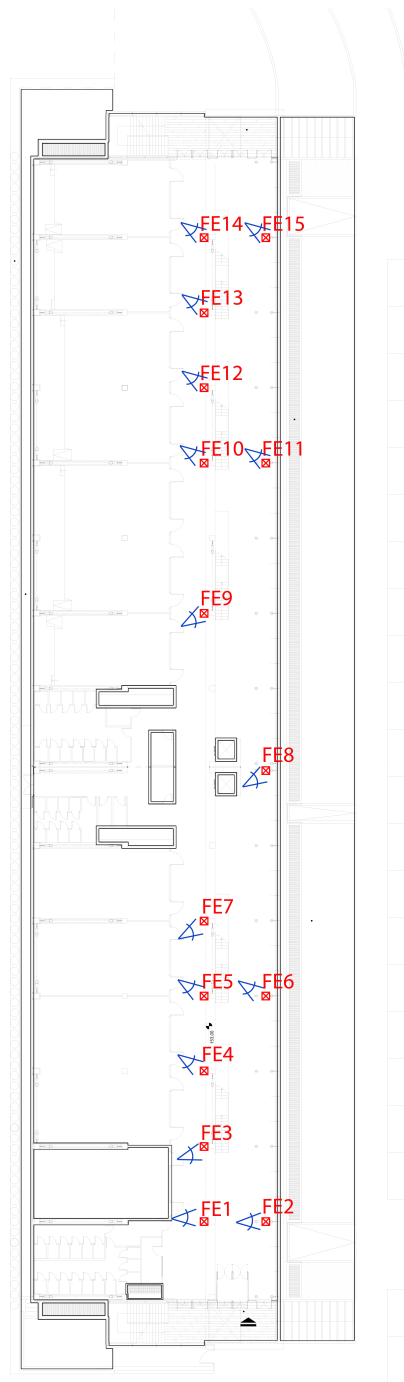


Figure 5.8: Fire extinguishers at ground floor (in red) plus pictures points of view (in blue).

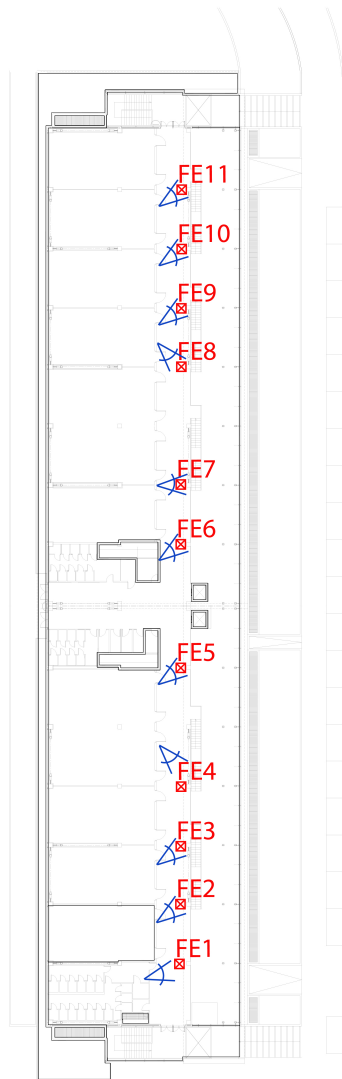


Figure 5.9: Fire extinguishers at first floor (in red) plus pictures points of view (in blue).

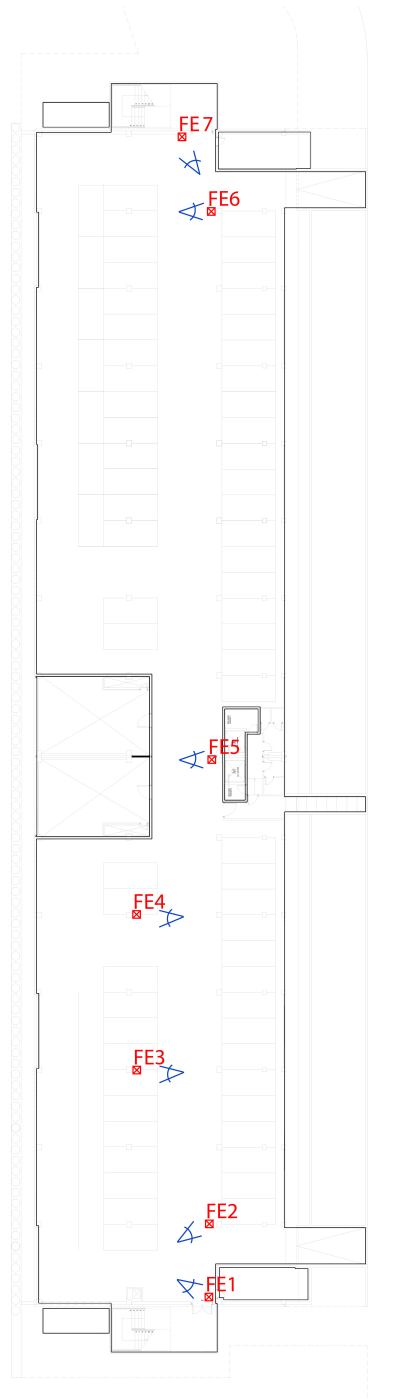


Figure 5.10: Fire extinguishers at basement floor (in red) plus pictures points of view (in blue).

5.4 Testing of the system

A final feasibility test has been performed regarding the whole system. This test took place inside the DICEA Department, Politechnic University of Marche. This test run again through the steps explained in section 4.4. The process started with the connection to the embedded system for performing the recognition procedure (Fig: 5.11). The following step was loading the BIM model and select the right room. After these steps the room alignment was performed. This task is pursued manually and it is composed by two steps: set position (Fig. 5.12) and set rotation (Fig. 5.13) of the room model. In the figures about those steps it is possible to see the setting points, the blue square for positioning and the red circle for rotation. Those points are located manually by the technician inside the room. Then an automatic procedure gives back an alarm if the alignment between the BIM model and the mesh produced by the HoloLens goes over a fixed threshold. The recognition procedure is completely automatic and it starts with the air-tap gesture inside the display of the HoloLens. This operation takes a snapshot, then it is automatically sent to the embedded system (Raspberry-Movidius), analyzed and data are sent back to the HoloLens. The visualization of the bounding box around the object and the confidence score is automatic as well as the insertion of the hologram inside the scene. In Fig. 5.14 it is possible to see the hologram positioned inside the room and in Fig. 5.15 the technician view without the BIM model displayed. In this case the object was recognized at the first attempt with a confidence score of 89.70%.



Figure 5.11: Application interface for HoloLens connection to embedded system.



Figure 5.12: Set position task.



Figure 5.13: Set rotation task.



Figure 5.14: Positioning of the hologram inside the scene.



Figure 5.15: Real-World scene referring to Fig 5.14

5.5 Discussion

The proof of concept of the system developed has highlighted some drawbacks:

- the creation of the dataset is an expensive procedure. With this research it has been estimated that 500 pictures are necessary for a single object category. In case of complex building with hundreds of different asset components necessary images numbers can become massive and therefore difficult to manage.
- training of combined network able to recognize more than one category object showed that when it is the same network that recognize more than one object the performance decrease in comparison with single object category network trained with the same number of images. This suggest that it could be useful to work with network in series (every network recognize one object) instead of multi-category single network. One future development could investigate the performances of these two different approaches.
- Pictures of some specific objects can be difficult to retrieve both in real world taking pictures and with web scraping.
- The validation of the network require long time especially for real world test. The on-site neural network performance validation must be done at the end of every training and this involves high costs.

In order to improve the system proposed further steps can be performed:

- the network could be enriched with other objects categories;
- the room alignment can be performed by means of an automatic procedure;
- further operations on the hologram of the recognized object could be added to the application.

The first further step should be the introduction of other objects categories. According to the use of the whole system there could be the need for more categories also referring to different assets or a more detailed identification of object types. Furthermore the best process for the achievement of this aim has to be investigated. The identification of the fire extinguisher type, for instance, could be performed with the recognition of the entire object or of its components (e.g. the identification of the manometer means that the type is a CO₂ fire extinguisher). Another aspect to take into consideration is the NN performance. Adding a second object lightly changed the performance; increasing the number of objects could deeply affect the correct response of the network. In this case the use of several NN in parallel could be decisive.

As far as the second improvement is concerned, the mesh generated by the Hololens and the one generated from a BIM model could be automatically compared for room alignment. In order to generate a mesh from the BIM model of the building, first attempts have been done using the software Unity.

Finally the application could be enriched with further operation in order to complete the inventory with the information that can not be retrieved automatically. It would be useful adding the possibility of inserting data not collected automatically by the network such as the last revision for the fire extinguisher. The same system could be also thought and then used for the periodic check to asset conditions.

As regards verification of the system on-site the first goal is performing the test on a huge number of objects in real-world conditions. This would lead to a more precise definition of performances of the system with different light conditions and various type of objects (fire extinguishers can slightly change aspects).

Lastly more tests have to be done for the correct localization inside rooms, and secondly inside the building. Despite the great potentialities of the Hololens, the drift suffered by the device still represents a problem in complex buildings. Large spaces or repetitive layouts deeply affect the performance of this device that could be supported by added sensor for enhancing its potentialities [Garon et al., 2017].

5.6 Conclusion

In this chapter the real working of the system is exposed.

First of all the network training started from images collection. Several different datasets have been created with both original and re-edited pictures so as to test different methods for network training. According to the different features in the various datasets 17 trainings have been performed.

Following the training the new customized network have been validated using precision, recall and F1 as evaluation parameters. Validation results demonstrates that best training performances do not depend upon the higher number of pictures. At the end of these tests 500 has resulted the number of pictures sufficient for obtaining good performances with customized networks.

Subsequently a real-world test has been pursued in order to test the network with the embedded system and different light conditions. Also in this case the network performed well with all the object recognized and less that 10% with a level of confidence lower that 60%.

Finally the whole component recognition process has been tested in real-environment showing the feasibility of this system usage.



Figure 5.16: Fire extinguishers point of view during BAS test, basement floor. First row, left to right FE1, FE2, FE3. Second row FE4, FE5, FE6. Third row FE7.



Figure 5.17: Fire extinguishers point of view during BAS test, ground floor. First row, left to right FE1, FE2, FE3. Second row FE4, FE5, FE6. Third row FE7.

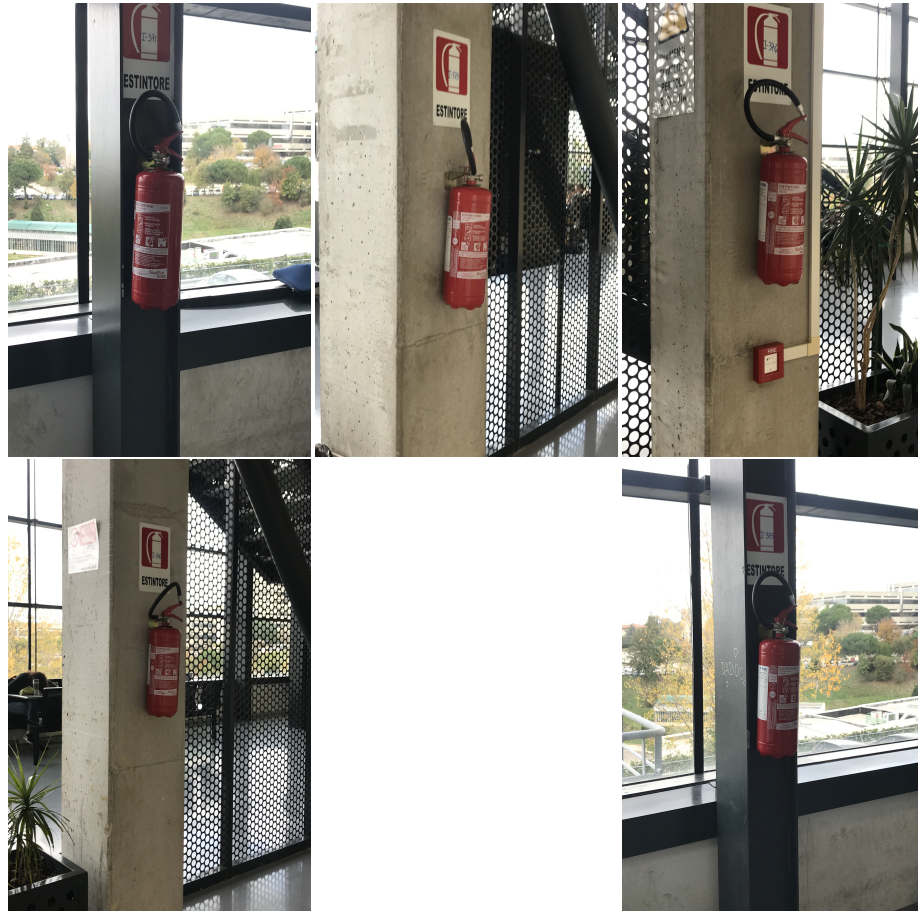


Figure 5.18: Fire extinguishers point of view during BAS test, ground floor. First row, left to right FE11, FE12, FE13. Second row FE14, FE15.

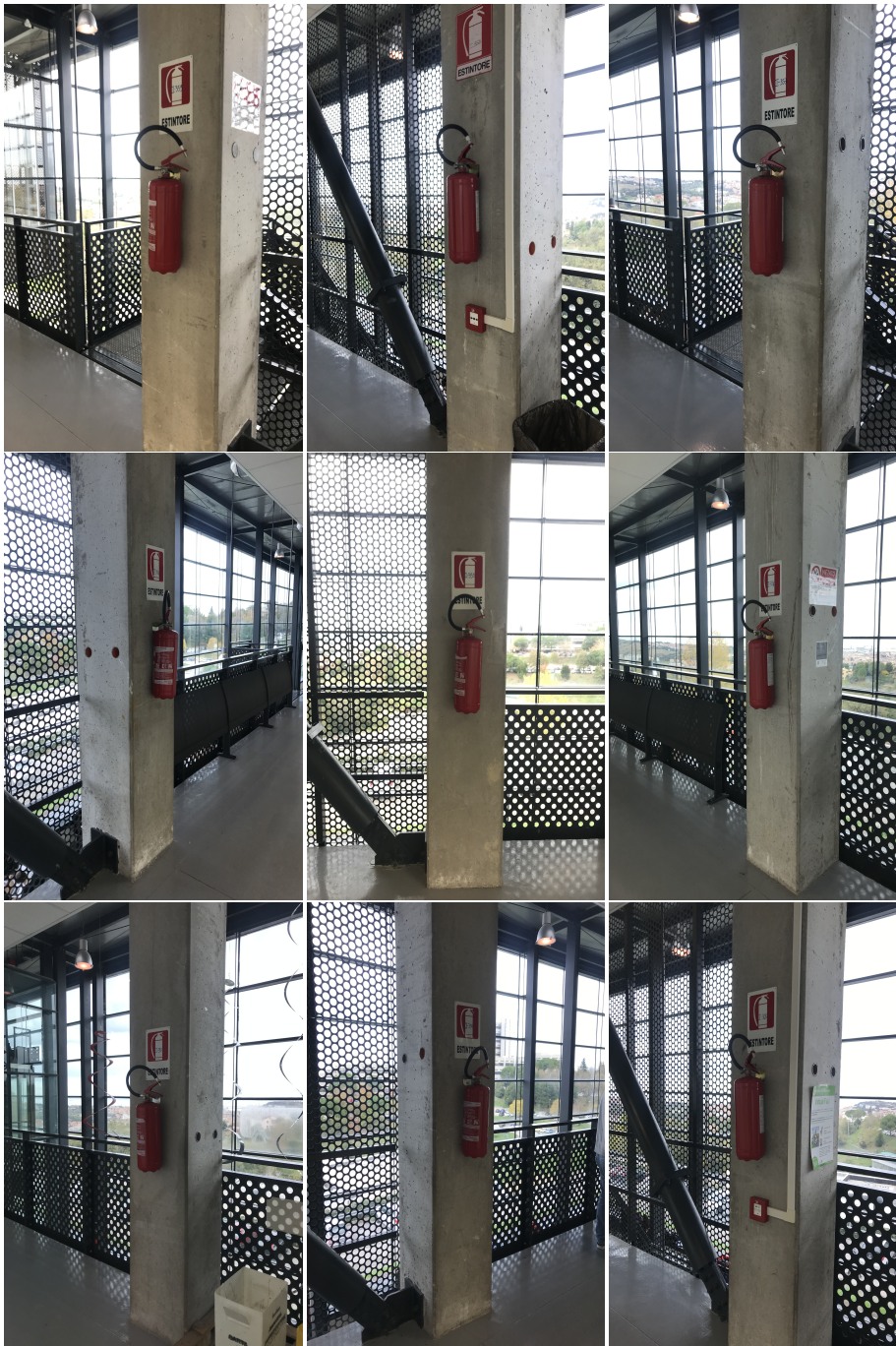


Figure 5.19: Fire extinguishers point of view during BAS test, first floor. First row, left to right FE1, FE2, FE3. Second row FE4, FE5, FE6. Third row FE7, FE8, FE9.



Figure 5.20: Fire extinguishers point of view during BAS test, first floor. First row, left to right FE10, FE11.

CONCLUSIONS

6.1 Conclusions

This research work started from the critical issue of the lack of a surveying corresponding to actual situation for existing buildings. This is particularly urgent for huge buildings stocks owners, such as Public Administrations. Furthermore relying on updated functional models of buildings is of primary importance in complex constructions (e.g. stations, airports, hospitals).

Moreover information accessibility is seen by the majority of the professional as the most urgent issue to solve for achieving a higher efficiency in the sector. Information related to existing building is often not available or not updated, while data coherent to the current situations are essential for planning operations.

Current researches still focus mainly on the geometric aspects of construction and little attention is given to building assets components which are the ones more often subjected to maintenance operations.

The relationships between geometrical and functional data are instead essential especially for emergency management.

Even new approaches such as the BIM one struggle to emerge because their potentialities rely on the presence of structured data. Developing as-built or as-is building model is an expensive task and even if some automation is starting to improve processes costs have not been reduced significantly since long post-processing phases are still required for data interpretation.

The proposed system wants to exploit the man-machine intelligence collaboration in order to avoid the long post-processing phase of data collected. The man contribution is the expert knowledge while the machine is used for its speed and reliability. All the operations are performed on-site so as to avoid the post-processing phase and giving the technician the opportunity of checking data collected directly.

The system development started with investigation on new technology that could work as an interface between technician and information. Mixed Reality with its possibility to overlap holograms to the real world and allowing interaction with digital data and real environment resulted the best choice.

Then Neural Networks applications on object recognition have been studied. YOLO came out to be the most performing for object recognition in real-time. The developed system is composed by several components that can be divided

into Mixed and real Environment. The object recognition application have been built for Microsoft Hololens. This application exploits YOLO neural network for object detection. With the aim of having a system that worked on-site the recognition is performed in an embedded system which includes a Raspberry and a Neural Compute Stick Movidius.

At the end of the development validation tests have been carried on in order to verify networks performances.

Finally the performed feasibility test was able to show the possibility of using this system on-site.

6.2 Research Contributions

The innovation proposed by this research lays in the combination of different new technologies for an on-site survey of building components. The result is the exploitation of man-machine parallel working and the profitable combination of mixed reality and NN.

The use of YOLO Neural Networks lead to the following results:

- a dataset for fire protection system components has been created and it can be used not only for facility management purposes and shared with the community;
- this study systematized the information for YOLO neural network customization;
- the right number of images for YOLO customization has been investigated and instructions on this provided to who may be interested in training his own neural network.

Furthermore a method for connecting Hololens and Raspberry has been developed using the latter as a computing device for YOLO running. The Hololens object recognition application allows to give semantic to building components recognized and to generate functional data.

6.3 Suggestions for future RD

Future developments of the system could be represented by:

- adding object categories to the dataset and training more customized networks;
- verify the performances of the use of single object neural network in series or one multi-object neural network;
- testing the capabilities of new MR device, in particular Hololens2 that is supposed to be equipped with an AI processor on board;

- developing the possibility inside the application of modifying objects and manually add information;
- automatize the technician localization inside the building.

One further comparison that can be done is the improvement of the system performance using artificial intelligence services in cloud that widen the computing power.

BIBLIOGRAPHY

- [Abiodun et al., 2018] Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A. E., and Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11):e00938.
- [Adams and Hannigan, 2018] Adams, S. S. and Hannigan, F. P. (2018). Advances in Human Factors in Energy: Oil, Gas, Nuclear and Electric Power Industries. 599:69–77.
- [Akcamete and Akinci, 2010] Akcamete, A. and Akinci, B. (2010). Potential utilization of building information models for planning maintenance activities.
- [Al-masni et al., 2018] Al-masni, M. A., Al-antari, M. A., Park, J. M., Gi, G., Kim, T. Y., Rivera, P., Valarezo, E., Choi, M. T., Han, S. M., and Kim, T. S. (2018). Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system. *Computer Methods and Programs in Biomedicine*, 157:85–94.
- [AlexeyAB, 2019] AlexeyAB (2019). <https://github.com/AlexeyAB/darknet>.
- [Ali, 2019] Ali, M. (2019). Artificial neural network based screening of cervical cancer using a hierarchical modular neural network architecture (HMNNA) and novel benchmark uterine cervix cancer database. *Neural Computing and Applications*, 31(7):2979–2993.
- [Ammari and Hammad, 2014] Ammari, K. E. and Hammad, A. (2014). 2014 - Collaborative BIM-based Markerless Augmented Reality Framework for Facilities Maintenance - Ammari.pdf. pages 657–664.
- [Augmentor, 2019] Augmentor, G. (2019). Images augmentor. <https://github.com/mdbloice/Augmentor>.
- [Becerik-Gerber et al., 2011] Becerik-Gerber, B., Jazizadeh, F., Li, N., and Calis, G. (2011). Application areas and data requirements for bim-enabled facilities management. *Journal of construction engineering and management*, 138(3):431–442.
- [Becker et al., 2018] Becker, R., Falk, V., Hoenen, S., Loges, S., Stumm, S., Blankenbach, J., Brell-Cokcan, S., Hildebrandt, L., and Vallée, D. (2018). Bim – towards the entire lifecycle. *International Journal of Sustainable Development and Planning*, 13(1):84–95.

- [Berg and Vance, 2017] Berg, L. P. and Vance, J. M. (2017). Industry use of virtual reality in product design and manufacturing: a survey. *Virtual reality*, 21(1):1–17.
- [Bigne et al., 2016] Bigne, E., Llinares, C., and Torrecilla, C. (2016). Elapsed time on first buying triggers brand choices within a category: A virtual reality-based study. *Journal of Business Research*, 69(4):1423–1427.
- [Bloch and Sacks, 2018] Bloch, T. and Sacks, R. (2018). Comparing machine learning and rule-based inferencing for semantic enrichment of BIM models. *Automation in Construction*, 91(July 2017):256–272.
- [Bonandrini et al., 2005] Bonandrini, S., Cruz, C., and Nicolle, C. (2005). Building Lifecycle Management, International Conference on Product Lifecycle Management. *Plm*, 2005(January 2005):461–471.
- [Bonetti et al., 2018] Bonetti, F., Warnaby, G., and Quinn, L. (2018). Augmented reality and virtual reality in physical and online retailing: A review, synthesis and research agenda. In *Augmented reality and virtual reality*, pages 119–132. Springer.
- [Braun et al., 2019] Braun, A., Jahr, K., and Borrmann, A. (2019). Formwork detection in UAV pictures of construction sites. *eWork and eBusiness in Architecture, Engineering and Construction*, pages 265–271.
- [Brilakis et al., 2010] Brilakis, I., Lourakis, M., Sacks, R., Savarese, S., Christodoulou, S., Teizer, J., and Makhmalbaf, A. (2010). Toward automated generation of parametric BIMs based on hybrid video and laser scanning data. *Advanced Engineering Informatics*, 24(4):456–465.
- [Camera di Commercio, 2012] Camera di Commercio, R. (2012). *Il mercato pubblico dei servizi FM*.
- [Canalys, 2017] Canalys (2017). Media alert: Virtual reality headset shipments top 1 million for the first time.
- [Carrozzino and Bergamasco, 2010] Carrozzino, M. and Bergamasco, M. (2010). Beyond virtual museums: Experiencing immersive virtual reality in real museums. *Journal of Cultural Heritage*, 11(4):452–458.
- [CCSInsight, 2017] CCSInsight (2017). Clear potential for virtual reality headsets after a slow start.
- [Cesarotti et al., 2014] Cesarotti, V., Benedetti, M., Dibisceglia, F., Di Fausto, D., Inrona, V., La Bella, G., Martinelli, N., Ricci, M., Spada, C., and Varani, M. (2014). Bim-based approach to building operating management: a strategic lever to achieve efficiency, risk-shifting, innovation and sustainability. In *Proc. Conference: XVIII International Research Society for Public Management (IRSPM) Conference, at Ottawa, Canada*.

- [Chalhoub and Ayer, 2018] Chalhoub, J. and Ayer, S. K. (2018). Using Mixed Reality for electrical construction design communication. *Automation in Construction*, 86(May 2017):1–10.
- [Chatterjee, 2016] Chatterjee, H. S. (2016). Various types of convolutional neural network. <https://towardsdatascience.com/various-types-of-convolutional-neural-network-8b00c9a08a1b>.
- [Chen et al., 2011] Chen, Y. C., Chi, H. L., Hung, W. H., and Kang, S. C. (2011). Use of tangible and augmented reality models in engineering graphics courses. *Journal of Professional Issues in Engineering Education and Practice*, 137(4):267–276.
- [Chiabrando et al., 2016] Chiabrando, F., Sammartano, G., and Spanò, A. (2016). Historical buildings models and their handling via 3d survey: From points clouds to user-oriented hbim. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 41(September):633–640.
- [Codinhoto et al., 2013] Codinhoto, R., Kiviniemi, A., Sergio Kemmer, and Cecilia Gravina da Rocha (2013). BIM-FM Implementation: An Exploratory Investigation. 2(June):1–15.
- [Collins et al., 2017] Collins, J., Regenbrecht, H., and Langlotz, T. (2017). Visual coherence in mixed reality: A systematic enquiry. *Presence: Teleoperators and Virtual Environments*, 26(1):16–41.
- [Corneli et al., 2019] Corneli, A., Naticchia, B., Cabonari, A., and Bosché, F. (2019). Augmented Reality and Deep Learning towards the Management of Secondary Building Assets. *Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC)*, (Isarc).
- [Cosenza et al., 2018] Cosenza, E., Salzano, A., Menna, C., Asprone, D., and Serra, M. a. (2018). Digitalizzazione del danno sismico di edifici su piattaforma BIM attraverso tecniche di intelligenza artificiale. *Ingenio*, 2:1–17.
- [Das, 2017] Das, S. (2017). Cnn architectures: Lenet, alexnet, vgg, googlenet, resnet and more... <https://medium.com/analytics-vidhya/cnn-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>.
- [Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-j., Li, K., and Fei-fei, L. (2009). ImageNet : A Large-Scale Hierarchical Image Database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2–9.
- [Devetakovic and Radojevic, 2007] Devetakovic, M. and Radojevic, M. (2007). Facility management: a paradigm for expanding the scope of architectural practice. *International Journal of Architectural Research: ArchNet-IJAR*, 1(3):127–139.

- [Díaz-Vilariño et al., 2015] Díaz-Vilariño, L., González-Jorge, H., Martínez-Sánchez, J., and Lorenzo, H. (2015). Automatic LiDAR-based lighting inventory in buildings. *Measurement: Journal of the International Measurement Confederation*, 73:544–550.
- [Dieck et al., 2016] Dieck, T. M. C., Jung, T., and Han, D.-I. (2016). Mapping requirements for the wearable smart glasses augmented reality museum application. *Journal of Hospitality and Tourism Technology*, 7(3):230–253.
- [Ding and Drogemuller, 2009] Ding, L. and Drogemuller, R. (2009). Towards sustainable facilities management. In *Technology, Design and Process Innovation in the Built Environment*, pages 399–418. Spon Press.
- [Dir. 24, 2014] Dir. 24 (2014). Directive 2014/24/EU of the European Parliament and of the Council. *Official Journal of the European Union*.
- [D.M. 560, 2017] D.M. 560 (2017). Decreto Ministro MIT n. 560 del 1.12.2017.pdf.
- [Donath and Thurow, 2007] Donath, D. and Thurow, T. (2007). Integrated architectural surveying and planning Methods and tools for recording and adjusting building survey data. 16:19–27.
- [Dünser et al., 2006] Dünser, A., Steinbügl, K., Kaufmann, H., and Glück, J. (2006). Virtual and augmented reality as spatial ability training tools. *ACM International Conference Proceeding Series*, 158:125–132.
- [East, 2007] East, W. (2007). BIM for Construction Handover. *Journal of Building Information Modeling*.
- [Eastman et al., 2011] Eastman, C., Teicholz, P., Sacks, R., and Liston, K. (2011). *BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors*. John Wiley & Sons.
- [Eh Phon et al., 2014] Eh Phon, D. N., Ali, M. B., and Halim, N. D. A. (2014). Collaborative augmented reality in education: A review. *Proceedings - 2014 International Conference on Teaching and Learning in Computing and Engineering, LATICE 2014*, pages 78–83.
- [Everingham et al., 2015] Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136.
- [Fan et al., 2014] Fan, S.-L., Skibniewski, M. J., and Hung, T. W. (2014). Effects of building information modeling during construction. , 17(2):157–166.
- [Fang et al., 2018] Fang, Q., Li, H., Luo, X., Ding, L., Luo, H., and Rose, T. M. (2018). Automation in Construction Detecting non-hardhat-use by a deep learning method from far- field surveillance videos. *Automation in Construction*, 85(May 2017):1–9.

- [Feiner et al., 1997] Feiner, S., MacIntyre, B., and Webster, A. (1997). A Touring Machine: Prototyping 3D Hobbit Augmented Reality Systems for Exploring the Urban Environment. *Personal Technologies*, pages 208–217.
- [Feng and Lin, 2017] Feng and Lin (2017). Smoothing Process of Developing the Construction MEP BIM Model - A Case Study of the Fire-Fighting System. (Isarc).
- [Flavián et al., 2019] Flavián, C., Ibáñez-Sánchez, S., and Orús, C. (2019). The impact of virtual, augmented and mixed reality technologies on the customer experience. *Journal of Business Research*, 100(January 2018):547–560.
- [FMLink, 2018] FMLink (2018). Reducing the total cost of ownership through a lifecycle approach. <https://fmlink.com/articles/reducing-the-total-cost-of-ownership-through-a-lifecycle-approach/>.
- [Fonnet et al., 2017] Fonnet, A., Alves, N., Sousa, N., Guevara, M., and Magalhães, L. (2017). Heritage BIM integration with mixed reality for building preventive maintenance. *EPCGI 2017 - 24th Encontro Portugues de Computacao Grafica e Interacao*, 2017-Janua:1–7.
- [Freeman et al., 2017] Freeman, D., Reeve, S., Robinson, A., Ehlers, A., Clark, D., Spanlang, B., and Slater, M. (2017). Virtual reality in the assessment, understanding, and treatment of mental health disorders. *Psychological medicine*, 47(14):2393–2400.
- [Gallaher M. and Gilday, 2004] Gallaher M., O'Connor A., D. J. and Gilday, L. (2004). Cost Analysis of Inadequate Interoperability in the U . S . Capital Facilities Industry.
- [Garon et al., 2017] Garon, M., Boulet, P. O., Doironz, J. P., Beaulieu, L., and Lalonde, J. F. (2017). Real-Time High Resolution 3D Data on the HoloLens. *Adjunct Proceedings of the 2016 IEEE International Symposium on Mixed and Augmented Reality, ISMAR-Adjunct 2016*, pages 189–191.
- [Ghosh and Schwartzbard, 1999] Ghosh, A. K. and Schwartzbard, A. (1999). A study in using neural networks for anomaly and misuse detection. In *USENIX security symposium*, volume 99, page 12.
- [Github, 2019] Github (2019). Darknet: yolov3workflow. https://github.com/reigngt09/yolov3workflow/tree/master/1_WebImage_Scraping.
- [Griffin et al., 2017] Griffin, T., Giberson, J., Lee, S. H. M., Guttentag, D., Kandaurova, M., Sergueeva, K., and Dimanche, F. (2017). Virtual reality and implications for destination marketing.
- [Hamledari et al., 2017] Hamledari, H., McCabe, B., and Davari, S. (2017). Automation in Construction Automated computer vision-based detection of components of under-construction indoor partitions. *Automation in Construction*, 74:78–94.

- [Haeuss, 2017] Haeuss, P. (2017). 8 major challenges the Australian VR industry is facing right now. <http://patriciahaeuss.com/8-major-challenges-the-australian-vr-industry-is-facing/>.
- [Hichri et al., 2013] Hichri, N., Stefani, C., De Luca, L., Veron, P., and Hamon, G. (2013). From point cloud to BIM: a survey of existing approaches. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-5/W2:343–348.
- [Hoffman and Novak, 1996] Hoffman, D. L. and Novak, T. P. (1996). Marketing in hypermedia computer-mediated environments: Conceptual foundations. *Journal of marketing*, 60(3):50–68.
- [Honkamaa et al., 2007] Honkamaa, P., Siltanen, S., Jäppinen, J., Woodward, C., and Korkalo, O. (2007). Interactive outdoor mobile augmentation using markerless tracking and GPS. *Proceedings of the Virtual Reality International Conference VRIC Laval France*, pages 285–288.
- [Hubel and Wiesel, 1962] Hubel, D. H. and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *The Journal of physiology*, 160(1):106–154.
- [Huval et al., 2015] Huval, B., Wang, T., Tandon, S., Kiske, J., Song, W., Pazhayampallil, J., Andriluka, M., Rajpurkar, P., Migimatsu, T., Cheng-Yue, R., Mujica, F., Coates, A., and Ng, A. Y. (2015). An Empirical Evaluation of Deep Learning on Highway Driving. pages 1–7.
- [Ihde, 1990] Ihde, D. (1990). *Technology and the lifeworld: From garden to earth*. Number 560. Indiana University Press.
- [Intel, 2019a] Intel (2019a). Intel® Movidius™ neural compute stick. <https://movidius.github.io/ncsdk/ncs.html>.
- [Intel, 2019b] Intel (2019b). Introduction to neural compute stick Movidius. <https://movidius.github.io/ncsdk/index.html>.
- [Irizarry et al., 2014] Irizarry, J., Gheisari, M., Williams, G., and Roper, K. (2014). Ambient intelligence environments for accessing building information: A healthcare facility management scenario. *Facilities*, 32(3):120–138.
- [Issa and R.A., 2014] Issa, I. M. and R.A., R. (2014). Enhancing Spatial and Temporal Cognitive Ability in Construction Education Through Augmented Reality and Artificial Visualizations. *Computing in civil and building engineering*, pages 955–1865.
- [Jansen and Zhang, 2007] Jansen, K. and Zhang, H. (2007). Scheduling malleable tasks. *Handbook of Approximation Algorithms and Metaheuristics*, pages 45–1–45–16.

- [Jeong et al., 2018] Jeong, H. J., Park, K. S., and Ha, Y. G. (2018). Image Preprocessing for Efficient Training of YOLO Deep Learning Networks. *Proceedings - 2018 IEEE International Conference on Big Data and Smart Computing, BigComp 2018*, pages 635–637.
- [Juang et al., 2013] Juang, J. R., Hung, W. H., and Kang, S. C. (2013). Sim-Crane 3D+: A crane simulator with kinesthetic and stereoscopic vision. *Advanced Engineering Informatics*, 27(4):506–518.
- [Keady, 2013] Keady, R. A. (2013). Financial impact and analysis of equipment inventories. *Facilities Engineering Journal*.
- [Kelly et al., 2013] Kelly, G., Serginson, M., Lockley, S., Dawood, N., and Kassem, M. (2013). BIM for Facility Management: a review and a case study investigating the value and challenges. *13th International Conference Applications of Virtual Reality*, (October):30–31.
- [Kerrebroeck et al., 2017] Kerrebroeck, H. V., Brengman, M., and Willems, K. (2017). Escaping the crowd: An experimental study on the impact of a virtual reality experience in a shopping mall. *Computers in Human Behavior*, 77:437–450.
- [Kim et al., 2019] Kim, D., Liu, M., Lee, S., and Kamat, V. R. (2019). Automation in Construction Remote proximity monitoring between mobile construction resources using camera-mounted UAVs. *Automation in Construction*, 99(April 2018):168–182.
- [Kim et al., 2017] Kim, K., Kim, H., and Kim, H. (2017). Image-based construction hazard avoidance system using augmented reality in wearable device. *Automation in Construction*, 83(April):390–403.
- [Kolar et al., 2018] Kolar, Z., Chen, H., and Luo, X. (2018). Automation in Construction Transfer learning and deep convolutional neural networks for safety guardrail detection in 2D images. *Automation in Construction*, 89(May 2017):58–70.
- [Kononenko, 2001] Kononenko, I. (2001). Machine learning for medical diagnosis: History, state of the art and perspective. *Artificial Intelligence in Medicine*, 23(1):89–109.
- [Kopsida and Brilakis, 2016] Kopsida, M. and Brilakis, I. (2016). BIM registration methods for mobile augmented reality-based inspection. *eWork and eBusiness in Architecture, Engineering and Construction - Proceedings of the 11th European Conference on Product and Process Modelling, ECPPM 2016*, (September):201–208.
- [La Delfa et al., 2016] La Delfa, G. C., Monteleone, S., Catania, V., De Paz, J. F., and Bajo, J. (2016). Performance analysis of visual markers for in-door navigation systems. *Frontiers of Information Technology and Electronic Engineering*, 17(8):730–740.

- [Lamio et al., 2019] Lamio, F., Farinha, R., Laasonen, M., and Huttunen, H. (2019). Classification of Building Information Model (BIM) Structures with Deep Learning. *Proceedings - European Workshop on Visual Information Processing, EUVIP*, 2018-November.
- [Laptev and Gupta, 2016] Laptev, I. and Gupta, A. (2016). Hollywood in Homes : Crowdsourcing Data. 1:510–526.
- [Lecoutre et al., 2017] Lecoutre, A., Negrevergne, B., Yger, F., Noh, Y.-K., and Zhang, M.-L. (2017). Recognizing Art Style Automatically in painting with deep learning. *Proceedings of Machine Learning Research*, 77(2016):327–342.
- [LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- [Lee et al., 2012] Lee, S.-K., An, H.-K., and Yu, J.-H. (2012). An extension of the technology acceptance model for bim-based fm. In *Construction Research Congress 2012: Construction Challenges in a Flat World*, pages 602–611.
- [Lee et al., 1999] Lee, W., Stolfo, S. J., and Mok, K. W. (1999). A data mining framework for building intrusion detection models. In *Proceedings of the 1999 IEEE Symposium on Security and Privacy (Cat. No. 99CB36344)*, pages 120–132. IEEE.
- [Li et al., 2018] Li, G., Song, Z., and Fu, Q. (2018). A New Method of Image Detection for Small Datasets under the Framework of YOLO Network. *Proceedings of 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference, IAEAC 2018*, (October 2018):1031–1035.
- [Lin et al., 2018] Lin, J.-H. T., Wu, D.-Y., and Tao, C.-C. (2018). So scary, yet so fun: The role of self-efficacy in enjoyment of a virtual reality horror game. *New Media & Society*, 20(9):3223–3242.
- [Lin et al., 2014] Lin, T.-y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P. (2014). Microsoft COCO: Common Objects in Context. pages 1–15.
- [Lin and Su, 2013] Lin, Y.-c. and Su, Y.-c. (2013). Developing Mobile- and BIM-Based Integrated Visual Facility Maintenance Management System. 2013.
- [Liu et al., 2016] Liu, R., Asce, A. M., Issa, R. R. A., and Asce, F. (2016). Survey : Common Knowledge in BIM for Facility Maintenance. 30(2010).
- [Liu et al., 2017] Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., and Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing*, 234(December 2016):11–26.

- [Love et al., 2014] Love, P. E. D., Matthews, J., Simpson, I., Hill, A., and Olatunji, O. A. (2014). Automation in Construction A benefit realization management building information modeling framework for asset owners. *Automation in Construction*, 37:1–10.
- [Lu et al., 2018] Lu, Q., Lee, S., and Chen, L. (2018). Image-driven fuzzy-based system to construct as-is IFC BIM objects. *Automation in Construction*, 92(March):68–87.
- [Ma and Sacks, 2016] Ma, L. and Sacks, R. (2016). A cloud-based BIM platform for information collaboration. *ISARC 2016 - 33rd International Symposium on Automation and Robotics in Construction*, pages 581–589.
- [Mcarthur, 2015] McArthur, J. J. (2015). A building information management (BIM) framework and supporting case study for existing building operations , maintenance and sustainability. 118:1104–1111.
- [McCulloch and Pitts, 1988] McCulloch, W. S. and Pitts, W. (1988). Neuro-computing: Foundations of research. pages 15–27.
- [McKinsey Global Institute, 2017] McKinsey Global Institute (2017). Reinventing Construction: A Route To Higher Productivity. *McKinsey & Company*, (February):20.
- [Meißner et al., 2017] Meißner, M., Pfeiffer, J., Pfeiffer, T., and Oppewal, H. (2017). Combining virtual reality and mobile eye tracking to provide a naturalistic experimental environment for shopper research. *Journal of Business Research*.
- [Microsoft, 2018a] Microsoft (2018a). Holograms. <https://docs.microsoft.com/en-us/windows/mixed-reality/hologram>.
- [Microsoft, 2018b] Microsoft (2018b). Start designing and prototyping. <https://docs.microsoft.com/it-it/windows/mixed-reality/design>.
- [Microsoft, 2018c] Microsoft (2018c). Unity development overview. <https://docs.microsoft.com/en-us/hololens/hololens1-hardware>.
- [Microsoft, 2018d] Microsoft (2018d). Unity development overview. <https://docs.microsoft.com/it-it/windows/mixed-reality/unity-development-overview>.
- [Microsoft, 2018e] Microsoft (2018e). What is mixed reality? <https://docs.microsoft.com/en-us/windows/mixed-reality/mixed-reality>.
- [Milgram and Kishimo, 1994] Milgram, P. and Kishimo, F. (1994). A taxonomy of mixed reality. *IEICE Transactions on Information and Systems*, 77(12):1321–1329.

- [Mill et al., 2013] Mill, T., Alt, A., and Liias, R. (2013). Combined 3D building surveying techniques - terrestrial laser scanning (TLS) and total station surveying for BIM data management purposes. *Journal of Civil Engineering and Management*, 19(Supplement 1):23–32.
- [Molchanov et al., 2017] Molchanov, V. V., Vishnyakov, B. V., Vizilter, Y. V., Vishnyakova, O. V., and Knyaz, V. A. (2017). Pedestrian detection in video surveillance using fully convolutional YOLO neural network. *Automated Visual Inspection and Machine Vision II*, 10334:103340Q.
- [Montserrat et al., 2017] Montserrat, D. M., Lin, Q., Allebach, J., and Delp, E. J. (2017). Training object detection and recognition CNN models using data augmentation. *IS and T International Symposium on Electronic Imaging Science and Technology*, pages 27–36.
- [Moore, 2018] Moore, S. (2018). *Deep learning for computer vision*.
- [Morrison, 2018] Morrison, H. M. H. P. J. E. J. (2018). History of machine learning. <https://www.doc.ic.ac.uk/~jce317/history-machine-learning.html#top>.
- [Motawa and Almarshad, 2013] Motawa, I. and Almarshad, A. (2013). A knowledge-based BIM system for building maintenance. *Automation in Construction*, 29:173–182.
- [Muhanna, 2015] Muhanna, M. A. (2015). Virtual reality and the cave: Taxonomy, interaction challenges and research directions. *Journal of King Saud University-Computer and Information Sciences*, 27(3):344–361.
- [Nakagawa et al., 2009] Nakagawa, M., Kondo, T., Kudo, T., Takao, S., and Ueno, J. (2009). Three-dimensional medical image recognition of cancer of the liver by the revised radial basis function (RBF) neural network algorithm. *Proceedings of the 14th International Symposium on Artificial Life and Robotics, AROB 14th'09*, pages 385–388.
- [Naticchia et al., 2019] Naticchia, B., Corneli, A., Carbonari, A., and Bosché, F. (2019). Augmented reality application supporting on-site secondary building assets management. (July).
- [Nazionale, 2010] Nazionale, O. (2010). *Il boom del facility management in Italia nel primo decennio del XXI secolo*.
- [Oesau et al., 2014] Oesau, S., Lafarge, F., and Alliez, P. (2014). Indoor scene reconstruction using feature sensitive primitive extraction and graph-cut. *IS-PRS Journal of Photogrammetry and Remote Sensing*, 90:68–82.
- [Ordonez et al., 2011] Ordonez, V., Kulkarni, G., and Berg, T. L. (2011). Im2text: Describing images using 1 million captioned photographs. In Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 24*, pages 1143–1151. Curran Associates, Inc.

- [Park et al., 2013] Park, C. S., Lee, D. Y., Kwon, O. S., and Wang, X. (2013). A framework for proactive construction defect management using BIM, augmented reality and ontology-based data collection template. *Automation in Construction*, 33:61–71.
- [Pärn et al., 2017] Pärn, E. A., Edwards, D. J., and Sing, M. C. P. (2017). Automation in Construction The building information modelling trajectory in facilities management : A review. *Automation in Construction*, 75:45–55.
- [Phan and Choo, 2010] Phan, V. T. and Choo, S. Y. (2010). Augmented Reality-Based Education and Fire Protection for Traditional Korean Buildings. *International Journal of Architectural Computing*, 8(1):75–91.
- [Pishdad-Bozorgi et al., 2018] Pishdad-Bozorgi, P., Gao, X., Eastman, C., and Self, A. P. (2018). Planning and developing facility management-enabled building information model (FM-enabled BIM). *Automation in Construction*, 87(February 2017):22–38.
- [Poirier et al., 2015] Poirier, E. A., Staub-french, S., and Forgues, D. (2015). Automation in Construction Measuring the impact of BIM on labor productivity in a small specialty contracting enterprise through action-research. *Automation in Construction*, 58:74–84.
- [Quintana et al., 2017] Quintana, B., Prieto, S. A., Adan, A., and Bosché, F. (2017). Scan-To-BIM for Small Building Components. 20321(July):29–36.
- [Quintana et al., 2018] Quintana, B., Prieto, S. A., Adán, A., and Bosché, F. (2018). Automation in Construction Door detection in 3D coloured point clouds of indoor environments. *Automation in Construction*, 85(October 2016):146–166.
- [Radovic et al., 2017] Radovic, M., Adarkwa, O., and Wang, Q. (2017). Object recognition in aerial images using convolutional neural networks. *Journal of Imaging*, 3(2).
- [Redmon, 2016] Redmon, J. (2013–2016). Darknet: Open source neural networks in c. <http://pjreddie.com/darknet/>.
- [Redmon, 2018] Redmon, J. (2013–2018). <https://github.com/pjreddie/darknet>.
- [Redmon and Farhadi, 2017] Redmon, J. and Farhadi, A. (2017). YOLO9000: Better, faster, stronger. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, volume 2017-Janua, pages 6517–6525.
- [Redmon and Farhadi, 2018] Redmon, J. and Farhadi, A. (2018). YOLOv3: An Incremental Improvement.

- [Redmon et al., 2016] Redmon, J., Girshick, R., Farhadi, A., and Dataset, A. (2016). You Only Look Once : Unified , Real-Time Object Detection.
- [RICS, 2009] RICS (2009). Building maintenance: strategy, planning and procurement, Royal Institution of Chartered Surveyors.
- [Riexinger et al., 2018] Riexinger, G., Kluth, A., Olbrich, M., Braun, J. D., and Bauernhansl, T. (2018). Mixed Reality for On-Site Self-Instruction and Self-Inspection with Building Information Models. *Procedia CIRP*, 72:1124–1129.
- [Rodriguez-gonzalvez et al., 2014] Rodriguez-gonzalvez, P., Gonzalez-aguilera, D., Lopez-jimenez, G., and Picon-cabrera, I. (2014). Automation in Construction Image-based modeling of built environment from an unmanned aerial system. *Automation in Construction*, 48:44–52.
- [Roper and Payant, 2014] Roper, K. and Payant, R. (2014). *The facility management handbook*. Amacom.
- [Saha, 2018] Saha, S. (2018). A comprehensive guide to convolutional neural networks — the eli5 way. <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>.
- [Scherer and Katranuschkov, 2017] Scherer, R. J. and Katranuschkov, P. (2017). Bimification: How to create bim for retrofitting. In *Proceedings of the Joint Conference on Computing in Construction (JC3)*, Heraklion, Greece.
- [Shanbari et al., 2016] Shanbari, H., Blinn, N., and Issa, R. R. (2016). Using augmented reality video in enhancing masonry and roof component comprehension for construction management students. *Engineering, Construction and Architectural Management*, 23(6):765–781.
- [Shen et al., 2010] Shen, W., Hao, Q., Mak, H., Neelankavil, J., Xie, H., Dickinson, J., Thomas, R., Pardasani, A., and Xue, H. (2010). Systems integration and collaboration in architecture, engineering, construction, and facilities management: A review. *Advanced Engineering Informatics*, 24(2):196 – 207. Enabling Technologies for Collaborative Design.
- [Shinde et al., 2018] Shinde, S., Kothari, A., and Gupta, V. (2018). YOLO based Human Action Recognition and Localization. *Procedia Computer Science*, 133(2018):831–838.
- [Shirazi and Ashuri, 2018] Shirazi, A. and Ashuri, B. (2018). Past, Present, and Future of BIM-Enabled Facilities Operation and Maintenance. *Proceeding of Construction Research Congress 2018*, 2010(1):461–471.
- [Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. pages 1–14.
- [Slater, 2003] Slater, M. (2003). A note on presence terminology. *Presence connect*, 3(3):1–5.

- [Smailagic and Siewiorek, 2004] Smailagic, A. and Siewiorek, D. P. (2004). Wearable computing. *Mobile Computing Handbook*, pages 3–23.
- [Succar, 2009] Succar, B. (2009). Building information modelling framework: A research and delivery foundation for industry stakeholders. *Automation in Construction*, 18(3):357–375.
- [Szegedy et al., 2015] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June-2015:1–9.
- [Tao et al., 2018] Tao, J., Wang, H., Zhang, X., Li, X., and Yang, H. (2018). An object detection system based on YOLO in traffic scene. *Proceedings of 2017 6th International Conference on Computer Science and Network Technology, ICCSNT 2017*, 2018-Janua:315–319.
- [task Group, 2012] task Group, P. C. (2012). Government Construction Strategy Final Report to Government by the Procurement / Lean Client Task Group July 2012 Report of the Procurement / Lean Client Task Group. (July).
- [Thornson et al., 2009] Thornson, C. A., Goldiez, B. F., and Le, H. (2009). Predicting presence: Constructing the tendency toward presence inventory. *International Journal of Human-Computer Studies*, 67(1):62–78.
- [Tijtgat, 2017] Tijtgat, N. (2017). Customizing yolo. <https://timebutt.github.io/static/how-to-train-yolov2-to-detect-custom-objects/>.
- [Tussyadiah et al., 2018] Tussyadiah, I. P., Jung, T. H., and tom Dieck, M. C. (2018). Embodiment of wearable augmented reality technology in tourism experiences. *Journal of Travel research*, 57(5):597–611.
- [Valero et al., 2018] Valero, E., Bosché, F., and Forster, A. (2018). Automation in Construction Automatic segmentation of 3D point clouds of rubble masonry walls , and its application to building surveying , repair and maintenance. *Automation in Construction*, 96(August):29–39.
- [Volk et al., 2014] Volk, R., Stengel, J., and Schultmann, F. (2014). Building Information Modeling (BIM) for existing buildings - Literature review and future needs. *Automation in Construction*, 38:109–127.
- [VoTT, 2019] VoTT, G. (2019). Labelling tool. <https://github.com/microsoft/VoTT>.
- [Wang et al., 2019a] Wang, H., Jiang, C., Bao, K., and Xu, C. (2019a). Recognition and Clinical Diagnosis of Cervical Cancer Cells Based on our Improved Lightweight Deep Network for Pathological Image. *Journal of Medical Systems*, 43(9).

- [Wang et al., 2019b] Wang, Q., Bi, S., Sun, M., Wang, Y., Wang, D., and Yang, S. (2019b). Deep learning approach to peripheral leukocyte recognition. *PLOS ONE*, 14(6):1–18.
- [Wang and Dunston, 2007] Wang, X. and Dunston, P. S. (2007). Design, strategies, and issues towards an Augmented Reality-based construction training platform. *Electronic Journal of Information Technology in Construction*, 12(June 2006):363–380.
- [Wang et al., 2017] Wang, X., Dunston, P. S., and Skibniewski, M. (2017). Mixed Reality Technology Applications in Construction Equipment Operator Training. *Proceedings of the 21st International Symposium on Automation and Robotics in Construction*.
- [Watson, 2011] Watson, A. (2011). Digital buildings - Challenges and opportunities. *Advanced Engineering Informatics*, 25(4):573–581.
- [Xiong et al., 2013] Xiong, X., Adan, A., Akinci, B., and Huber, D. (2013). Automatic creation of semantically rich 3D building models from laser scanner data. *Automation in Construction*, 31:325–337.
- [Xue et al., 2018] Xue, F., Lu, W., and Chen, K. (2018). Automatic Generation of Semantically Rich As-Built Building Information Models Using 2D Images: A Derivative-Free Optimization Approach. *Computer-Aided Civil and Infrastructure Engineering*, 33(11):926–942.
- [Yang and Ergan, 2017] Yang, X. and Ergan, S. (2017). BIM for FM: Information Requirements to Support HVAC-Related Corrective Maintenance. *Journal of Architectural Engineering*, 23(4):04017023.
- [Zhao et al., 2015] Zhao, Z. K., Wang, L., and Xu, N. (2015). Deep belief network based 3D models classification in building information modeling. *International Journal of Online Engineering*, 11(5):57–63.
- [Zhou et al., 2017] Zhou, Y., Luo, H., and Yang, Y. (2017). Implementation of augmented reality for segment displacement inspection during tunneling construction. *Automation in Construction*, 82:112–121.