HERITAGE IDENTIFICATION OF MONUMENTS USING DEEP LEARNING TECHNIQUES

Submitted by

Nisha Devi S(21MDB005)

in partial fulfillment for the award of the degree of

MASTER OF SCIENCE
Data Science & Business Analysis

DEPARTMENT OF COMPUTER SCIENCE

RATHINAM COLLEGE OF ARTS AND SCIENCE

(AUTONOMOUS)

COIMBATORE - 641021 (INDIA)

May - 2023

RATHINAM COLLEGE OF ARTS AND SCIENCE
(AUTONOMOUS)
COIMBATORE - 641021

BONAFIDE CERTIFICATE

This is to certify that the Phase-I report entitled **Heritage Identification Of Monuments Using Deep Learning Techniques** submitted by **Nisha Devi S**, for the award of the Degree of Master of Computer Science specialization in **"Data Science & Business Analysis"** is a bonafide record of the work carried out by her under my guidance and supervision at Rathinam College of Arts and Science, Coimbatore

Mrs. V.Kanimozhi M.E.,(Ph.D)	**Mr.Siva Prakash M.Tech., (Ph.D)**
Supervisor	Mentor

Submitted for the University Examination held on 20.04.2023

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

RATHINAM COLLEGE OF ARTS AND SCIENCE
(AUTONOMOUS)
COIMBATORE - 641021

DECLARATION

I am **Nisha Devi S**, hereby declare that this thesis entitled **"Architectural Heritage Images Classification Using Deep Learning with CNN ",** is the record of the original work done by me under the guidance of **Mrs.V.Kanimozhi M.E.,(Ph.D)**, Faculty Rathinam college of arts and science, Coimbatore. To the best of my knowledge this work has not formed the basis for the award of any degree similar award to any candidate in any University.

Signature of the Student:

Nisha Devi S

Place: Coimbatore

Date: 12.04.2023

COUNTERSIGNED

Mrs.V.Kanimozhi M.E.,(Ph.D)
Supervisor

Contents

Acknowledgement iii

List of Figures iv

List of Tables v

Abstract vi

1 Introduction 1

1.1 Digital Documentation 2

1.2 Problem Formulation 6

1.3 Main objectives and contributions 8

2 Background 10

2.1 HBIM MODELS 11

2.2 HBIM and Auxillary Tools 13

2.3 Models and HBIM through Web 15

3 Dataset Description 18

4 Methodology **20**

4.1 Convolution Neural Network 21

4.1.1 Convolution Neural Network in image processing 22

4.1.2 The Input Image 23

4.1.3 The Pre-processing Stage 23

4.1.4 The Face Detection Stage 24

4.1.5 The Feature-Extraction Stage 24

4.1.6 The Classification Stage 24

4.1.7 Training Stage 25

4.1.8 Prediction Stage 25

4.2 Stochastic Gradient Descent 25

4.3 Hyperparameter Optimization 27

4.4 Full Training of AlexNet Network 31

4.5 Full Training of a Residual Network (ResNet) 32

5 Result and Discussion **34**

6 Conclusion **33**

7 References **34**

Acknowledgement

On successful completion for project look back to thank who made in possible. First and foremost, thank **"THE ALMIGHTY"** for this blessing on us without which we could have not successfully our project.I are extremely grateful to **Dr.Madan.A. Sendhil, M.S., Ph.D.,** Chairman, Rathinam Group of Institutions, Coimbatore and **Dr. R.Manickam MCA., M.Phil., Ph.D.,** Secretary, Rathinam Group of Institutions, Coimbatore for giving me opportunity to study in this college.I am extremely grateful to **Dr.R.Muralidharan, M.Sc., M.Phil., M.C.A., Ph.D.,** Principal Rathinam College of Arts and Science(Autonomous), Coimbatore.Extend deep sense of valuation to **Mr.A.Uthiramoorthy, M.C.A., M.Phil., (Ph.D),** Rathinam College of Arts and Science (Autonomous) who has permitted to undergo the project. Unequally I thank **Mr.P.Sivaprakash, M.Tech., (Ph.D).,** Mentor and **Dr.Mohamed Mallick, M.E., Ph.D.,** Project Coordinator, and all the Faculty members of the Department - iNurture Education Solution pvt ltd for their constructive suggestions, advice during the course of study.I convey special thanks, to the supervisor **Mrs.V.Kanimozhi M.E.,(Ph.D)** who offered their inestimable support, guidance, valuable suggestion, motivations, helps given for the completion of the project. I dedicated sincere respect to my parents for their moral motivation in completing the project.

List of Figures

3.1 Overview of the dataset 19

4.1 Alexnet Architecture 31
4.2 Resnet Architecture 33

5.1 Sample of Predicted accuracy based on ResNet-18 32

List of Tables

5.1 Accuracy table 35

Abstract

The categorization of photographs captured during architectural asset assessment is a critical job in the digital recording of cultural resources. Because many photos are often processed, categorization is a time-consuming (and hence error-prone) job that takes a long time. The availability of automated solutions to aid these sorting chores would greatly enhance a key aspect of digital documentation. Furthermore, precise picture categorization allows for better administration and more efficient searches via particular phrases, assisting in researching and understanding the historical object in the issue. The major goal of this paper is to use deep learning (RESNet18) methods for categorizing photographs of architectural history, especially using convolutional neural networks. The usefulness of training these networks from scratch vs. just fine tweaking pre-trained networks is assessed for this purpose. This has been used to categorize items of interest in pictures of architectural heritage structures. Because no datasets appropriate for network training have been found, a new dataset has been constructed and made public. In terms of accuracy, promising results have been produced, and it is believed that implementing these approaches might considerably contribute to the digital recording of architectural history.

Chapter 1

Introduction

Throughout history, mankind has felt the urge to know and represent the outside world. Not only to acknowledge it, but also to modify the territory by constructing civil works and other architectural structures: buildings, monuments, bridges, etc.

In order to do the planning in a rural or urban environment, it is necessary to know its characteristics in different levels. So before the landscape architectural design process or the urban planning we need to take a good look to the nowadays city structure and building features, specially if we are working in old cities or in areas with singular buildings, places where it is necessary to balance the tradition with the new architecture. In this context it is basic to have an accuracy graphic documentation and tools to manage it, in order to make easier the process of planning a new landscape in harmony with the environment. So from the ancient civilizations maps and territory representation have been present.

Since Ancient Turkey, city representations have been found in tablets or directly on some building walls. For example, in Fig. 1 the city of Catal Höyük is represented on a table.

In Ancient Mesopotamia, topographic maps were also created (GA-SUR map 3800-2500 B.C.), the Egyptians invented the groma for topographic surveys, instrument used later by the Romans to elaborate their cartography. In the 16th Century, navigation charts were of great importance for cartography. The map elaborated by Juan de la Cosa in 1500 illustrated the location of the New World.

During the same era, several concepts and laws related to perspective (works by Da Vinci and Durer) and stereoscopy (Fig. 2) were developed. However, there is also proof that Euclide (300 B.C.) had some knowledge about the stereoscopic vision.

In general, the references indicate that Niepce invented the photography in 1816 and Laussedat used the camera lucida in 1860 for cartographic and architectural surveys (Fig. 3). One of the first applications of the image was the catalogues of monuments and their surroundings. Meydenbauer remains the most representative author of them. Thanks to him and his work many monuments could be reconstructed after the World Wars. Currently, the existing technologies enable the obtention of geometry and information for proficient catalogues.

Data acquisition to represent any element undergoes constant evolution as it incorporates new tools and needs. In the following epigraphs a review of different techniques used from three centuries ago to the present, advantages and drawbacks of each one will counterpoise.

1.1 Digital Documentation

Before planning any intervention in an asset of heritage interest, it would be desirable to have the most complete documentation possible and preferably in digital format to facilitate its management and sharing. Such documentation would correspond to the current state of the asset but should ideally continue in later phases to assist in monitoring and maintenance tasks. Obtaining this documentation is not easy but it is necessary to help to preserve and disseminate the tangible cultural heritage. Generally, we can say that digital documentation comprises two main sections: (a) the task itself, of measuring and taking data and systematic images and the subsequent storage; and (b) the classification, interpretation and management of the available information (both obtained in the previous process as well as the existing one) [1].

The documentation and preservation of the heritage is an activity on the increase for several reasons: first, the administrations dedicate more resources to these issues because of the sociocultural value and its economic impact in the surroundings of the considered good; secondly, the magnitude of threats is high (natural degradation, attacks, wars, natural disasters, air pollution, climate change, vandalism and neglect); and finally the available technical resources are increasingly advanced and much more accessible. In particular, improvements in the speed and accuracy of image acquisition devices, multispectral sensors and many other data collection systems, as well as the availability of very advanced software tools, have led this trend [2]. There is in fact an international organization: "CIPA Heritage Documentation" [3] (founded in 1968 by the International Council on Monuments and Sites (ICOMOS) [4] and the International Society for Photogrammetry and Remote Sensing: (ISPRS) [5]) in charge of transferring new measurement and visualization technologies to the field of documentation and heritage conservation. All these technologies can be used for many purposes of interest in heritage conservation, such as historical interpretation, the study of the evolution of the asset, planning interventions, monitoring and supervision of the state, comparisons of different phases, simulation of its degradation, detection of pathologies and impairments, computer assisted restoration, the application of virtual and augmented reality techniques, digital catalogues, integration in Geographic Information Systems (GIS) and Building Information Modeling (BIM) environments, dissemination and many more [6,7,8]. These new technologies, therefore, can be a powerful tool to improve the classical standard of heritage measurement and documentation and create a new methodology. However, care should be taken with their use, as these technologies must be studied and properly adapted in order to be fully effective and useful. Proof of this is that, despite all these potential applications and the constant pressure from international heritage organizations, a standardized approach to digital documentation in the field of cultural heritage has not yet been achieved.

In any case, it is always desirable for the methodology (and the corresponding documentation technologies used) to offer several important qualities: accuracy, access to small spaces or spaces difficult to access, adaptation to different typologies of the architectural heritage, low cost, preferably contactless and fast. Since all of these properties are not usually found in a single technique, most documentation projects related to large and complex sites integrate and combine multiple sensors and techniques to achieve more accurate and complete results [2]. Consequently, digital heritage documentation requires the integration of different types of information: 3D models, photographs, thermographs, multispectral images and historical documents, among others. Obviously, the documentation of cultural heritage must take into account not only the raw data itself but also the corresponding metadata and paradata, which are fundamental aspects to consider [9,10].

In the practice of patrimonial conservation, the professionals involved usually accumulate large amounts of information, including photographs, drawings and field notes for their analysis, studies and work to be done. Specifically, the use of all kinds of images is one of the most common sources of documentation, even for 3D modeling of elements. There is no doubt that the amount of images that are handled in some heritage documentation project is enormous. The improvement, low cost and portability of cameras, and especially those integrated in mobile phones, have propitiated this. Of course, photographs taken by professionals are usually well-catalogued and should be the basis of the corresponding documentation but many photos taken by non-professionals are easily accessible on the web and can constitute a valuable complementary source of information. Its interpretation and classification is a complex and tedious task, as much for the variety of elements to interpret as for the huge amount that is necessary to handle in some cases. It is common to have hundreds and even thousands of photographs of each building (including images from historical archives) and, in many cases, the same information has been registered at least twice, because generally there is no mechanism to indicate that the information already exists or where it can be found. If all these images are not classified correctly, they are not useful (they cannot be indexed and therefore the search is difficult). It takes a lot of time and effort to locate information that is known or assumed to exist, but is inaccessible because it has not been stored and catalogued correctly. Estimated in terms of cost, this effort can be quite significant. Needless to say, it is much higher when the information cannot be found and must be regenerated [1]. However, the semantic categorization of these images, based as much on the high level (general meaning of the scene) as the low level (individual details), has still received little attention from the scientific community [11]. Therefore, the development of tools to facilitate their classification automatically would be highly desirable.

1.2 Problem Formulation

The Italian territory is full of buildings with historical and cultural value, which require more or less invasive transformation. For this reason, methods and tools to store, share, and manage information on their past, present, and future status are necessary. This implies continuous updating between the methods of massive data acquisition, more and more precise in metric accuracy, and digital models, ever more complete for information quality [1,2]. The presence on the territory of a very high number of existing buildings, many of them of high historical and cultural value, which require more or less incisive transformation interventions, has favored the extension of a European directive of 2014 (EUPPD 2014/24/EU). It promotes a new approach concerning the entire building process (design, representation, construction, management, and maintenance) and invites the use of BIM not only for new construction interventions but also for restoration, adaptation, or maintenance. In this context, it is necessary to keep in mind the link between these operations and knowledge and documentation of the history and current-state of the artifacts. They are closely connected with the activities of acquisition of historical heritage data. The integration of survey data, now increasingly complete, heterogeneous, and shareable, and the HBIM systems, allows for a lot of reality-based information to be brought in. This information (metric, geometric, morphological, material, chromatic) expressed through digital models allows for improvement of the knowledge of the building and offers control using the acquired data in the development of subsequent projects.

The construction industry uses BIM for its decentralized planning and control of interventions, but it also influenced the complex management of architectural built heritage. Heritage BIM pursues the modeling of architectural elements according to their constructive and historical-artistic characteristics [3,4,5].

The HBIM processes allow, through digital platforms and integration with survey data, for the investigation of new possibilities of managing the cultural heritage data, from the general to the detailed scale, associating quantitative and qualitative characteristics. The first concern physical parameters, metric, geometric, morphological, and spatial information; the latter, instead, constitute all those contingent or permanent properties linked to the formal aspects of the analyzed structures. However, the possible interactions between the HBIM and the survey data are still in progress because the purposes for which BIM systems are born and used change when they are used for built heritage. Then, because of the growing up of professional figures involved in the management of the objects in question; not only designers, engineers, installers but also historians, restorers, figures in charge of the protection of cultural heritage. Furthermore, the processes of data systematization and organization for models of cultural heritage follow paths defined each time with reference to the different needs, to the prefixed objective, and the characteristics of the objects analyzed. In fact, it appears that a procedure to be followed has not yet been consolidated.

The present research consists of three phases. The first phase, cognitive, consists of a survey of the state of the art in the framework of the construction of models for the Cultural heritage, the criteria followed, and the systemization of data in HBIM processes. The second phase, analytical, is aimed at identifying case studies in the field of existing cultural heritage; based on their typological representativeness and their characteristics, the problems with which to verify application procedures are identified. In the third phase, the final phase, critical considerations deriving from the previous phases and potential elements of innovation flow together.

1.3 Main objectives and contributions

The definition of a methodology for the switch from a numerical model to a parametric model according to BIM process is still in progress and is studied in different researches. A growing need to store and share information and models of architectural heritage has prompted scholars to test different ways to approach the HBIM. They still show some limitations, but not so much in the integration between different kinds of data. The main difficulty is the semi-automatic operation that allows transit from the numerical model—the point cloud—to the parametric model. This step consists of the construction of more or less articulated architectural elements, through the reconstruction of semantic identity, and ensuring correspondence with the real object and with its metric-dimensional aspects. Actually, point clouds record the geometric, chromatic, and material characteristics. To have information about the topology and semantic features of the architectural objects is necessary to produce other types of models, such as geometrical, architectural, or the parametric ones. It is a reverse engineering operation, in which the reading and segmentation of the point cloud, after recognizing characteristic regions, is the first step to identify surface boundaries that facilitate the modeling process. These activities are semi-automatic or completely automatic, through the progress achieved by the systems supported by the BIM processes.

Chapter 2

Background

The exam of progress achieved in academic research and in work experiences shows that a big step has been taken towards the automation of the process of point-cloud modeling. The algorithms and software's plug-ins are easily applicable for the segmentation and automatic modeling of point clouds that describe flat surfaces or primitive geometries. However, they generate incorrect results when trying to represent geometries of complex and irregular historical buildings. Three-dimensional modeling of any artifact implies an organized and orderly composition of digital elements but, when applied to HBIM, must go beyond the typical workflow. The ambitious purpose of the best match between the real object and the virtual model makes it necessary to structure different phases to define and optimize the workflow. The heterogeneity of the built heritage means that the definition of structured protocols is useful to represent the characteristics of the case study, because the BIM process was not created to study built heritage. To apply the BIM methodology to architectural heritage is possible thanks to continuous technological advances, but also considering its theoretical implications to propose new implementation. The scholars involved in the study of

HBIM have developed different methods and used technologies for three-dimensional modeling of existing historic buildings and the use of parametric components. A careful analysis of existing literature, but also of research still in progress shows how the BIM approach for architectural heritage can be set up by three different approaches. The first category includes those studies that adopted only commercial platforms for BIM processes to create models of existing architecture creating libraries of parametric objects. The second category concerns researches, which combine BIM systems with auxiliary tools or plug-ins, including open source software or commercial data storage and management (i.e., GIS). In the third category are researches that combine HBIM with web applications.

2.1 HBIM MODELS

The researches belonging to the first category aim at construction of HBIM models with the creation of the related libraries. Although the software used is often different (Autodesk Revit, Graphisoft Archicad, etc.), the methodology of survey data elaboration is the same. The sections on numerical models allow for the optimization of parametric modeling operations. However, most of the cases analyzed do not state what the level of automation of the process was, which makes it difficult to understand how to implement the management processes. In addition, the creation of HBIM libraries start from 3D modeling of parametric objects based on point clouds. It integrates the primitive geometries provided by the platform used with the documentation related to the chronology of the life of the artifacts. Two problems arise from the analysis of different cases: The integration between the different formats, resolved through the GDL (Geometric Description Language) script language included in the software, and the modeling of irregular shapes, once identifying their profile on the numerical model imported in the BIM platforms. López et al. (2017) developed this process [6] to model the Romanesque church of Santa María la Real de Mave, Palencia, Spain. The creation of the dedicated library has provided, in the first instance, the collection of information on the built space and its semantic structure, then the processing and organization of the data obtained. The sections of the numerical model, the aid of a grid that considers the characteristics of the object, the rules and the construction schemes of the architectural period to which the building belongs, have allowed the creation of simple and homogeneous surfaces. For the more complex ones, the profiles were first represented on a plane (2D), then transformed into a solid element (3D).

Del Giudice and Osello (2013) [7,8] model different architectural elements directly on the numerical model. The strength of these studies does not lie in the creation of libraries, not developed, but in the approach to temporal calculation, effective for the organization and control of projects for the management of the heritage. Biagini et al. (2016) [9] used a similar workflow for the connection between clouds of points and tools for modeling in the BIM software. They underline how the fundamental problems are (1) the identification and separation of the components to be modeled according to their type, hierarchy, and material, and (2) the lack of flexible tools effective in modeling historic buildings.

Ma et al. (2015), Cheng et al. (2015), and Adami et al. (2016) [10,11,12] describe how 3D modeling of architectural components with peculiar characteristics can help scholars to learn about each real element by improving the maintenance, management, and restoration processes of the entire building. Examination of these works shows that implementing modeling processes, particularly if using a semi-automatic approach, takes a long time. This still constitutes a weakness in the structure of a methodology, above all because software has not yet been optimized for the automatic conversion of point clouds into BIM components.

2.2 HBIM and Auxillary Tools

Some HBIM applications in the field of architectural heritage use auxiliary tools, which influence the modeling process. Dore et al. (2015) [13] base the digital modeling of the four courts in Dublin on historical bibliographic documentation and the analysis of the current state. The reading of specific aspects is useful to compose a library of parametric objects thanks to the GDL 3D ruled plug-in. The elements of the library are constructed according to two different methods: The first, used for regular components, is based on archival documents for understanding shapes and geometries; the second, set up for complex or irregular objects, includes editing operations of the numerical model by defining regions and section planes at different levels.

The study by Nieto et al. (2016) [14] proposes, instead, an innovative process for the cataloging information on artifacts of rather high complexity (i.e., archaeological architecture. The data analysis starts with the definition of grids on the surfaces analyzed for the study of the changes that the different elements have undergone over time. Other authors, such as Oreni et al. (2016, 2017) [15] and Barazzetti et al. (2015) [16] promote the structure of HBIM libraries for structural analysis, which analyze, in detail, elements of which it is necessary to know the geometry and, above all, the variations in the repetition of similar components. The research presented by Quattrini et al. (2015) [17] is interesting because it considers the numerical model as a source of information in its complexity. Therefore, it is not sectioned or fragmented, and the modeling of components takes place directly on the raw data. It guarantees quality and precision in the modeling of regular geometries, built using the Autodesk Revit parametric element libraries, and the complex ones, created through B-Rep operations of. The research group use an additional open source plug-in, Protégé, to integrate parametric data with each modeled element.

Fregonese et al. (2015) and Rechichi et al. (2016) [18,19] use open source software (3DReshaper, BIM3DGS), which are used for the processing of survey data and for the construction of parametric models, or in the integration with GIS applications (SIGEC and SICaR), such as with studies by Baik et al. (2015, 2017) [20]. In this case, the combination of the highly-detailed modeling based on the segmentation of the survey data in main parts (general portions) and secondary parts (detailed elements), allow for describing Islamic architecture, and the Autodesk InfraWorks GIS system is difficult because of the integration of information at the territorial scale with those on the architectural one.

The disadvantage of the approaches analyzed in this category concerns the integration between parametric architectural elements modeled on surveying data. The main limitations concern the auxiliary software, which, although useful and decisive in some ways, could cause the loss of information when exporting data, invalidating a deep knowledge of the building.

2.3 Models and HBIM through Web

The evolution of ICT allows for the use and access of heterogeneous information thanks to technologies that can understand different languages and put them into communication. The technological advance involved, also, the field of BIM and HBIM and the creation of web-oriented interfaces that collect data within a single information model. The experience of Quattrini et al. (2017) [21] is meaningful as the group developed a methodology within a particularly complex context. The research, conducted on the Church of Santa Maria in Portonovo, succeeds in a certain way in summarizing all the problems identified to date in the field of HBIM. The work proposes a real solution to the request for (almost) total interoperability between BIM models, rich in information organized in a hierarchical way through ontologies, and their query in the context of the semantic web. The process followed is interesting for its approach at different levels of depth; secondly for the way in which a semantically-structured 3D model is shared in a commonly used environment, that of the browser. The user browses the data through queries and thus accesses 3D/2D models or parts of them, digital worksheets, and multimedia content such as pdf, video, images, or web links. The methodology followed demonstrates that it is possible to switch from the parametric representation of the HBIM to the management of the 3D web objects. This operation allows for better understanding of the single elements through thematic information on the architectural organism and allows the description of the semantic contents, connecting them with a thematic databases (construction technologies, abacus elements, etc).

Actually, the use of BIM processes for the enhancement and management of heritage influences three different aspects: knowledge, modeling, and validity of data. A fundamental difference is the role of knowledge in HBIM with respect to that required in the design process. In this case, the knowledge of the built architecture match the semantic modeling. It is a consequence of the survey data processing, and if it is set with respect to archive documents, it allows for the understanding of information through its correct interpretative context, thus it can be shared by optimizing the programming and execution of subsequent operations. Parametric and informative modeling of historical heritage is difficult, both in terms of geometric transposition of the continuity of the real world and of its qualitative description. These difficulties are also associated with the intrinsic rigidity of the parametric modeling workflow and the construction of libraries of digital objects that clash with the variability and uniqueness of the built environment, especially when it has ancient origins, or the result of the stratification of different interventions, or is in a poor state of preservation.

Modeling within HBIM processes involves an important discretization operation that still faces the impossibility of using automated systems to unravel these features. The various BIM platforms allow different types of checks of the built model to highlight any collisions between interfering elements or the compliance with reference regulations. Reference is made to the clash detection functions, in the second case to those of mode checking allowed by the Autodesk Revit software, which follow in the field of design (for example, about the fire resistance of the materials used). The extension of BIM processes to built heritage has highlighted two other types of controls to validate the built models. They are the metric and geometric adherence between the numerical model and the parametric model, and secondly to the semantic decomposition of the model. On this aspect, recent academic studies have proposed the introduction of the Level of Reliability (LOR) as an indicator of the reliability of an information model, or of the digital objects that make it up [22]. Reconsidering these aspects opens up possible development scenarios to implement the consolidated methods of integrated survey and intervention in the Cultural heritage using information systems, with a view to guaranteeing an increasingly controlled structure of data that influences the scientific nature of the whole process.

Chapter 3

Dataset Description

Architectural Heritage Elements Dataset (AHE) is an image dataset for developing deep learning algorithms and specific techniques in the classification of architectural heritage images. This dataset consists of 10235 images classified in 10 categories: Altar: 829 images; Apse: 514 images; Bell tower: 1059 images; Column: 1919 images; Dome (inner): 616 images; Dome (outer): 1177 images; Flying buttress: 407 images; Gargoyle (and Chimera): 1571 images; Stained glass: 1033 images; Vault: 1110 images. It is inspired by the CIFAR-10 dataset but with the objective in mind of developing tools that facilitate the tasks of classifying images in the field of cultural heritage documentation. Most of the images have been obtained from Flickr and Wikimedia Commons (all of them under creative commons license).

Figure 3.1: Overview of the dataset

Chapter 4

Methodology

Deep learning is actually a subset of machine learning. It technically is machine learning and functions in the same way but it has different capabilities. The main difference between deep and machine learning is, machine learning models become well progressively but the model still needs some guidance.A branch of machine learning called deep learning appeared. The popularity of machine learning and the development of the computing capacity of computers enabled this new technology. Deep learning as a concept is very similar to machine learning but uses different algorithms. While machine learning works with regression algorithms or decision trees, deep learning uses neural networks that function very similarly to the biological neural connections of our brain.

If a machine learning model returns an inaccurate prediction then the programmer needs to fix that problem explicitly but in the case of deep learning, the model does it by him.The first advantage of deep learning over machine learning is the needlessness of the so-called feature extraction. Automatic car driving system is a good example of deep learning.Deep learning that enable the network to learn from unsupervised data and solve complex problems. Deep Learning approaches such as Convolutional Neural Network, Auto Encoder, Deep Belief Network, Recurrent Neural Network, Generative Adversal Network and Deep Reinforcement Learning are the algorithms used in Deep Learning.In our project we are using Convolutional Neural Networks.

4.1 Convolution Neural Network

One of the most popular deep neural networks is Convolutional Neural Networks. It is a class of deep neural networks, most commonly applied to analyze visual imagery and specializes in processing data that has a grid-like topology, such as an image. A digital image is a binary representation of visual data.Convolutional neural networks are composed of multiple layers of artificial neurons.The main advantage of CNN compared to its predecessors is that it automatically detects the important features without any human supervision.It can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.The main advantage of CNN compared to its predecessors is that it automatically detects the important features without any human supervision. CNN utilizes spatial correlations which exist with the input data. Each concurrent layer of the neural network connects some input neurons.

4.1.1 Convolution Neural Network in image processing

A convolutional neural network is a type of artificial neural network used in image recognition and processing that is specifically designed to process pixel data.CNNs are fully connected feed forward neural networks. CNNs are very effective in reducing the number of parameters without losing on the quality of models. Images have high dimensionality (as each pixel is considered as a feature) which suits the above described abilities of CNNs.It as a machine learning algorithm that can take in an input image, assign importance weights and biases to various objects in the image, and then can differentiate one from the another.It works by extraction features from the image. Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

We use Convolutional Neural Network and Deep Learning based yolo for Real Time Detection and Recognition of Human Faces, which is simple face detection and recognition system is proposed in this paper which has the capability to recognize human faces in single as well as multiple face images in a database in real time with masks on or off the face. Pre-processing of the proposed frame work includes noise removal and hole filling in colour images. After pre-processing, face detection is performed by using CNNs architecture. Architecture layers of CNN are created using Keras Library in Python. Detected faces are augmented to make computation fast. By using Principal Analysis Component (PCA) features are extracted from the augmented image. For feature selection, we use Sobel Edge Detector.

4.1.2 The Input Image

Real-time input images are used in this proposed system. Face of person in input images must be fully or partially covered as they have masks on it. The system requires a reasonable number of pixels and an acceptable amount of brightness for processing. Based on experimental evidence, it is supposed to perform well indoors as well as outdoors i.e. passport offices, hospitals, hotels, police stations and schools etc.

4.1.3 The Pre-processing Stage

Input image dataset must be loaded as Python data structures for pre-processing to overturn the noise disturbances, enhance some relevant features, and for further analysis of the trained model. Input image needs to be pre-processed before face detection and matching techniques are applied. Thus pre-processing comprises noise removal, eye and mask detection, and hole filling techniques. Noise removal and hole filling help eliminate false detection of face/ faces. After the pre-processing, the face image is cropped and re-localised. Histogram Normalisation is done to improve the quality of the pre- processed image.

4.1.4 The Face Detection Stage

We perform face detection usingHAAR Cascade algorithm.This system consists of the value of all black pixels in greyscale images was accumulated. They then deducted from the total number of white boxes. Finally, the outcome is compared to the given threshold, and if the criterion is met, the function considers it a hit.In general, for each computation in Haar-feature, each single pixel in the feature areas can need to be obtained, and this step can be avoided by using integral images in which the value of each pixel is equal to the number of grey values above and left in the image.

4.1.5 The Feature-Extraction Stage

Feature Extraction improves model accuracy by extracting features from pre-processed face images and translating them to a lower dimension without sacrificing image characteristics. This stage allows for the classification of human faces.

4.1.6 The Classification Stage

Principal Component Analysis(PCA) is used to classify faces after an image recognition model has been trained to identify face images. Identifying variations in human faces is not always apparent, but PCA comes into the picture and proves to be the ideal procedure for dealing with the problem of face recognition. PCA does not operate classifying face images based on geometrical attributes, but rather checks which all factors would influence the faces in an image. PCA was widely used in the field of pattern recognition for classification problems.PCA demonstrates its strength in terms of data reduction and perception.

4.1.7 Training Stage

The method is based on the notion that it learns from pre- processed face images and utilizes CNN model to construct a framework to classify images based on which group it belongs to. This qualified model is saved and used in the prediction section later. In CNN model, the stages of feature extraction are done by PCA and feature selection done by Sobel Edge Detector and thus it improves classification efficiency andaccuracy of the training model.

4.1.8 Prediction Stage

In this stage, the saved model automatically detects theofthefface maskimagecaptured by the webcam or camera. The saved model and the pre-processed images are loaded for predicting the person behind the mask. CNN offers high accuracy over face detection, classification and recognition produces precise and exactresults.CNN model follows a sequential model along with Keras Library in Python for prediction of human faces.

4.2 Stochastic Gradient Descent

Although any method can be used to train the optimization convolutional networks, one of the most common is the stochastic gradient descent using the mini-batch of samples.

The gradient calculation requires the error of the last layer to previous layers to be backpropagated. The backpropagation of errors [44,45] is a method of calculating gradients that can be used in the method of stochastic gradient descent to train neural networks grouped in layers. This is really a simple implementation of the chain rule of derivatives, speeding the calculations of all required partial derivatives. As mentioned, once a pattern has been applied to the input of the network as a stimulus, this propagates from the first layer through the upper layers of the network, to generate an output. The output signal is compared to the desired output and an error signal for each of the outputs (error vector) is calculated.

The error outputs are propagated backwards from the output layer to all neurons in the hidden layer contributing directly to the output. This process is repeated layer by layer, until all neurons in the network have received an error signal describing their relative contribution to the total error. There are many optimizations of this method, such as Momentum, Adagrad, RMSProp, Adam, Nesterov, Adadelta, etc. In our case, we use the first-mentioned, incorporating the term known as momentum (which can be understood as the average of the previous gradients), which reduces oscillations that cause local minima, thus accelerating convergence. There are other optimizations, such as weight decay (regularization term), which penalizes changes in the weights and prevents them from being too large.

The importance of this process is that, as the network is trained, the neurons in the intermediate layers organize themselves in such a way that the different neurons learn to recognize different characteristics of the total input space. After training, when these neurons are presented with an arbitrary input pattern that contains noise or is incomplete, the neurons in the hidden layers of the network will respond with an active output if the new input contains a pattern that resembles the feature that the Individual neurons have learned to recognize during their training.

4.3 Hyperparameter Optimization

In both cases (full training or fine-tuning), the training of these networks requires the adjustment of certain variables called hyperparameters (momentum, weight decay, learning rate, etc.), specifically in the context of algorithms based on stochastic gradient descent (which are the most common). To optimize this setting, it is interesting to consult [48]. The hyperparameters that are usually considered in the first place are: the initial learning rate, its decay value and the intensity of regularization, but there are many others that can also be important, such as the momentum, the decay of the weights, the number of iterations, etc.

Regarding the hyperparameters themselves, we can say the following. Learning rate: This is one of the most important, if not critical, hyperparameters, as it determines the amplitude of the jump to be made by the optimization technique in each iteration. If the rate is very low it will take a long time to reach convergence and if it is very high it could fluctuate around the minimum or even diverge. The asymptotic convergence rates of SGD are independent of sample size. Therefore, the best way to determine the correct learning rates is to perform experiments using a small but representative sample of the training set. When the algorithm works well with that small set of data, the same learning rates can be maintained and trained with the complete dataset [49]. Another possible option is to use dynamic learning rates (which are reduced when converging to the solution). This dynamic must be predefined and must therefore be adapted to the specific characteristics of each dataset. Momentum: As the parameters approach a local optimum, improvements can slow down, taking a long time to finally reach the minimum. Introducing a term that "boosts" the optimization technique can help to further improve model parameters towards the end of the optimization process. This term, called momentum, will consider how the parameters were changing in recent iterations, and will use that information to keep moving in the same direction. Specifically, the momentum term increases for dimensions whose gradients are pointing in the same directions and reduces updates for dimensions whose gradients change direction. As a result, faster convergence is achieved and oscillation is reduced. Size of the mini-batch: In our case, we use the stochastic gradient descent method with a random subset (mini-batch) of the training data at each iteration. If the size of the mini-batch is too small, convergence will be slow and it is also not possible to take advantage of some type of highly efficient operations (intelligent matrices). If the size is too large, the speed advantages offered by this method are reduced, as larger subsets of training data are used. In any case, its impact mainly affects the training time and hardly affects the results obtained. A value of 32 may be a good initial approximation. Weight decay: This value is an additional term in the weight update rule that causes the weights to drop exponentially to zero and determines the importance of this type of regularization in the gradient calculation. Generally, the more examples of training you have, the weaker this term will be and the more parameters you have to adjust (very deep nets, large filters, etc.), the higher this term should be.

Number of iterations: One way to know the number of iterations to perform (without reaching overfitting) is to extract a subset of samples from the training set (note that the test set has previously been removed from the complete dataset) and to use it in an auxiliary way during training. This subset is called the validation set. The role of the validation set is to evaluate the network error after each epoch (or after every certain number of epochs) and determine when it begins to increase. Since the validation set is left out during training, the error committed on it is a good indication of the network error over the entire test set. Consequently, the training will be stopped when this validation error increases and the values of the weights of the previous epoch will be retained. This stopping criterion is called early-stopping. Early-stopping is a simple way to avoid overfitting, i.e., even if the other hyperparameters cause overfitting, early-stopping will greatly reduce overfitting damage that would otherwise occur. It also means that it hides the excessive effect of other hyperparameters, possibly hindering the analysis that one might want to do when trying to figure out the effect of individual hyperparameters.

In addition to the criteria discussed for each hyperparameter, certain general details must be taken into account. Implementation: Larger neural networks often require a lot of training time, so tuning the hyperparameters can be very time-consuming. One option is to design a system that generates random hyperparameters (within reasonable ranges) and performs training, evaluating the performance achieved and storing model control points (along with their corresponding statistics). Subsequently, these control points can be inspected and analyzed to outline the appropriate hyperparameter optimization strategies.

Use cross validation or not: In most cases, if the validation set is large enough, cross-validation is not required. Search intervals for hyperparameters: It is advisable to search for hyperparameters using a logarithmic scale; at least for the learning rate and for the strength of regularization, as they have multiplicative effects on the training dynamics. Random search or search by grid: Randomized trials are more efficient for hyperparameter optimization than grid-based assays. In addition, this is also generally easier to implement. Border values: A hyperparameter can sometimes be searched at an inappropriate interval. Therefore, it is important to check that the adjusted hyperparameter is not at one end of that range, since the optimum value of the hyperparameter might be is outside our search range.

Initialization of the parameters: This operation can be deceptively important. In general, we can say that bias terms can often be initialized to 0 without problems. The weight matrices are more problematic, for example, if all values are initialized to 0, the activation function may generate null gradients; if all weights were equal, the hidden units would produce the same gradients and behave the same (thus wasting parameters). A possible solution is to initialize all elements of the weight matrix following a zero-centered Gaussian distribution with a standard deviation of 0.01.

The initial learning rate is often the most important hyperparameter and therefore its correct adjustment should be ensured. Its value is usually less than 1 and greater than 106. Usually, 0.01 is used as a typical value, but this logically depends on each case. Following this methodology, the hyperparameters used in the different trainings shown in the next section have been selected.

4.4 Full Training of AlexNet Network

For the first of the tests carried out, the AlexNet network was chosen, which is a well-known network and widely used in this type of task. It is recognized as the one that led to the resurgence of these techniques. This network was developed by Krizhevsky et al. [16], is a deep CNN architecture and was the winning model in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC-2012) [50]. In this challenge, the models try to classify the images into 1000 different categories (generic such as "volcano", "obelisk" or "lemur"). In contrast to earlier CNN models, AlexNet consists of five convolutional layers, of which the first, second, and fifth are followed by pooling layers, and three fully connected layers (a total of approximately 60 million parameters).

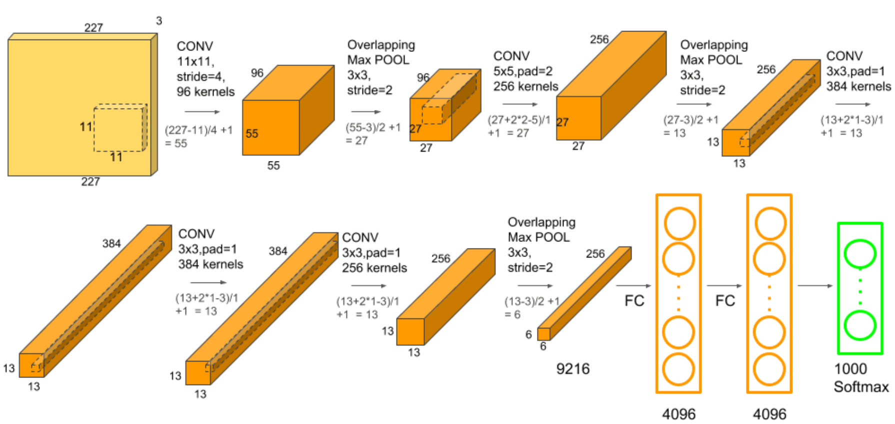

Figure 4.1: Alexnet Architecture

The success of AlexNet is attributed to certain practical solutions, such as Rectified

Linear Units (ReLU), data augmentation and dropout. The ReLU, which is simply a half-wave rectifier function such that f(x) = max (x; 0), can significantly accelerate the training phase; Data augmentation is an effective way to reduce over-fitting when training a large CNN, generating more training images by trimming small patches and horizontally flipping those patches; while the dropout technique, which reduces the co-adaptations of neurons by randomly establishing the zero value at the exit of some hidden neurons, is used in fully connected layers to reduce overfitting. In short, the success of AlexNet popularized the application of large CNNs in the tasks of visual recognition, so it has become a classic architecture within the CNNs.

4.5 Full Training of a Residual Network (ResNet)

We also decided to use the original residual network developed by He et al., of Microsoft [15], which has led to a growing adoption of this specific type of network due to its good results. The depth of the networks has a decisive influence on their learning, but adjusting this parameter optimally is a very difficult task. In theory, when the number of layers in a network increases, its performance should also improve. However, in practice, this is not true for two main reasons: the vanishing gradient (many neurons become ineffective/useless during the training of such deep networks); and the optimization of parameters is highly complex (by increasing the number of layers, it increases the number of parameters to adjust, which makes training these networks very difficult, leading to higher errors than in the case of shallower networks).

The residual networks seek to increase the network's depth without such problems

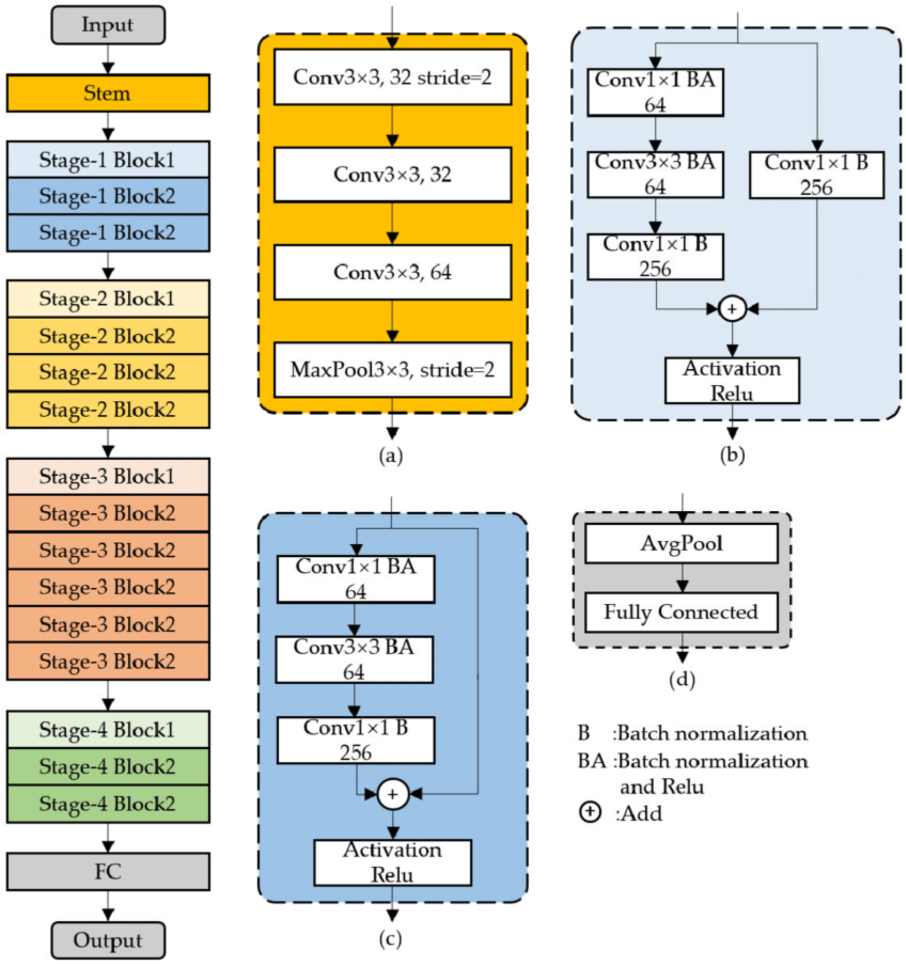

Figure 4.2: Resnet Architecture

affecting the results. The central idea of residual networks is based on the introduction of an identity function between layers. In conventional networks, there is a nonlinear function y = H (x) between layers (underlying mapping), as shown on the left of Figure 5. In residual networks, we have a new nonlinear function y = F (x) + id (x) = F (x) + x, here F (x) is the residual (on the right of Figure 5). This modification (called shortcut connections) allows important information to be carried from the previous layer to the next layers. Doing this avoids the problem of the vanishing gradient.

Chapter 5

Result and Discussion

In this section the experimental setting is introduced first, to establish the basic idea of the work

Architectural Heritage Elements Dataset (AHEDataset), which generated in three versions: Originally the dataset was published in two versions, first one contains images of different sizes and the second was scaled into 128×128 pixels, as well as the small dataset was created with a small size of 64×64 and 32×32. The third version was selected as a subset of each class which consists of 500 images that scaled into 224 × 224 pixels to be compatible with the pre-trained CNN.The dataset was partitioned randomly into 70% for training and 30% for validation. 2- The pre-trained Convolutional Neural Network (GoogLNet, ResNet 18 and Alexnet) were used during the course of this research to classify the Architectural Heritage Elements Dataset. All the CNN that used were modified to be suitable for those kinds of images. The modification starts from the input layer to be able for multi-channel images, convolutional layer to select size of filters proper the size of the entire images, fully connected layer was edited to be suitable the number of categories in the used dataset, finally the output classification layer was edit based on number of classes of fully connected layer. Our experimental study using AHEDataset original version was trained. Dataset are trained fully by three pre-trained Convolutional Neural Networks such as alexnet ,resnet18 and google net. Table shows a summary of the results based on a different test performed. Consider the dataset with 64×64 image size, the resnet-18 achieved the highest accuracy of 93.4 at the same learning rate and iteration in comparison to other techniques. In another case, the highest accuracy achieved is 91 based on alexnet under the same conditions for images with a size of 128×128.

Pretrained CNN algorithm	Image size	Epoch	Validation iteration	Training iteration	Learning rate	Accuracy
Resnet 18	64*64	20	98	588	0.0003	93.4
Alexnet	64*64	20	98	588	0.0003	91.0
Googlenet	64*64	20	98	588	0.0003	87.4

Table 5.1: Accuracy table

Figure show samples of a testing image with Predicted accuracy for each sample of an original dataset based on ResNet-18 respectively.

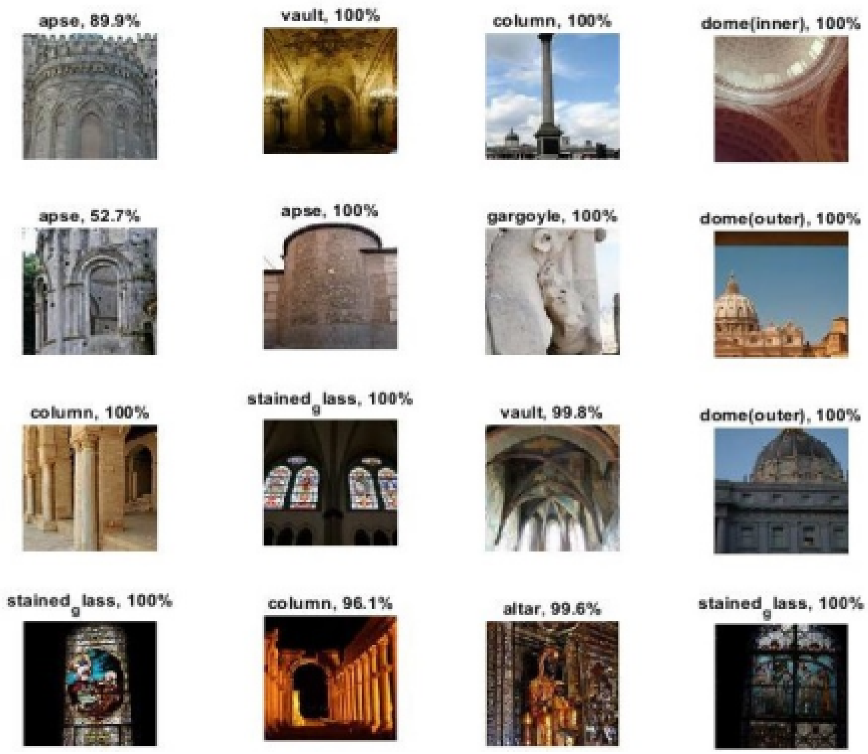

Figure 5.1: Sample of Predicted accuracy based on ResNet-18

Chapter 6

Conclusion

This work presents a model for architecture heritage image classification using deep learning with a convolutional neural network. In this model, CNN uses a pre-trained structure (GoogLNet , ResNet-18 and Alexnet). The classification results which are achieved by Resnet 18 overperformed the other techniques in comparison with other CNN on dataset. However, ResNet-18 produced better classification results compared with other pre-trained CNN with the highest accuracy 93.4

Chapter 7

References

1. Letellier, R.; Schmid, W.; LeBlanc, F. Recording, Documentation, and Information Management for the Conservation of Heritage Places: Guiding Principles; Routledge: London, UK; New York, NY, USA, 2007.

2. Remondino, F. Heritage Recording and 3D Modeling with Photogrammetry. Remote Sens. 2011, 3, 1104– 1138.

3. CIPA Heritage Documentation. Available online: http://cipa.icomos.org/ (accessed on 25 September 2017).

4. ICOMOS, International Council on Monuments Sites. Available online: http://www.icomos.org/ (accessed on 25 September 2017).

5. ISPRS, International Society of Photogrammetry and Remote Sensing. Available online: http://www.isprs.org/ (accessed on 25 September 2017).

6. Beck, L. Digital Documentation in the Conservation of Cultural Heritage: Finding the Practical in best Practice. Int. Arch. Photogramm. Remote Sens. Spat. Inf.

Sci. 2013, XL-5/W2, 85–90.

7. Hassani, F.; Moser, M.; Rampold, R.; Wu, C. Documentation of cultural heritage; techniques, potentials, and constraints. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 2015, XL-5/W7, 207–214.

8. López, F.J.; Lerones, P.M.; Llamas, J.; Gómez-García-Bermejo, J.; Zalama, E. A framework for using point cloud data of heritage buildings towards geometry modeling in a BIM context: A case study on Santa Maria la Real de Mave Church. Int. J. Archit. Heritage 2017, 11, doi:10.1080/15583058.2017.1325541.

9. Apollonio, F.I.; Giovannini, E.C. A paradata documentation methodology for the Uncertainty Visualization in digital reconstruction of CH artifacts. SCIRES-IT 2015, 5, 1–24.

10. Di Giulio, R.; Maietti, F.; Piaia, E.; Medici, M.; Ferrari, F.; Turillazzi, B. Integrated Data Capturing Requirements for 3d Semantic Modelling of Cultural Heritage: The INCEPTION Protocol. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 2017, XLII-2/W3, 251–257.

11. Oses, N.; Dornaika, F.; Moujahid, A. Image-based delineation and classification of built heritage masonry. Remote Sens. 2014, 6, 1863–1889.

12. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 2016, 38, 142–158.

13. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.

14. Cireşan, D.; Meier, U.; Schmidhuber, J. Multi-column deep neural networks for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. arXiv 2015, arXiv:1512.03385.

15. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.

16. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv 2014, arXiv:1409.1556.

17. Szegedy, C.; Toshev, A.; Erhan, D. Deep neural networks for object detection. In Proceedings of the 26th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 2553–2561.

18. Mnih, V.; Hinton, G. Learning to Label Aerial Images from Noisy Data. In Proceedings of the 29th International Conference on Machine Learning (ICML-12), Edinburgh, UK, 27 June–3 July 2012; pp. 567– 574. 20. Gao, F.; Huang, T.;

Wang, J.; Sun, J.; Hussain, A.; Yang, E. Dual-Branch Deep Convolution Neural Network for Polarimetric SAR Image Classification. Appl. Sci. 2017, 7, 447, doi:10.3390/app7050447.

19. . Tajbakhsh, N.; Shin, J.; Gurudu, S.; Hurst, R.; Kendall, C.; Gotway, M.; Liang, J. Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning? IEEE Trans. Med. Imaging 2016, 35, 1299–1312.

20. Gao, Y.; Lee, H.J. Local Tiled Deep Networks for Recognition of Vehicle Make and Model. Sensors 2016, 16, 226, doi:10.3390/s16020226.

21. Li, C.; Min, X.; Sun, S.; Lin, W.; Tang, Z. DeepGait: A Learning Deep Convolutional Representation for View-Invariant Gait Recognition Using Joint Bayesian. Appl. Sci. 2017, 7, 210, doi:10.3390/app7030210.

22. Pedraza, A.; Bueno, G.; Deniz, O.; Cristóbal, G.; Blanco, S.; Borrego-Ramos, M. Automated Diatom Classification (Part B): A Deep Learning Approach. Appl. Sci. 2017, 7, 460, doi:10.3390/app7050460.

23. Liu, L.; Wang, H.; Wu, C. A machine learning method for the large-scale evaluation of urban visual environment. arXiv 2016, arXiv:1608.03396.

24. Sa, I.; Ge, Z.; Dayoub, F.; Upcroft, B.; Perez, T.; McCool, C. DeepFruits: A Fruit Detection System Using Deep Neural Networks. Sensors 2016, 16, 1222, doi:10.3390/s16081222.

25. Chu, W.-T.; Tsai, M.-H. Visual pattern discovery for architecture image classification and product image search. In Proceedings of the 2nd ACM International Conference on Multimedia Retrieval, Hong Kong, China, 5–8 June 2012.

26. Goel, A.; Juneja, M.; Jawahar, C.V. Are buildings only instances?: Exploration in architectural style categories. In Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing, Mumbai, India, 16–19 December 2012.

27. Mathias, M.; Martinovic, A.; Weissenberg, J.; Haegler, S.; Van Gool, L. Automatic Architectural Style Recognition. In Proceedings of the 4th ISPRS International Workshop 3D-ARCH 2011, Trento, Italy, 2–4 March 2011; Volume XXXVIII-5/W16, pp. 171–176. 30. Shalunts, G.; Haxhimusa, Y.; Sablatni, R. Architectural Style Classification of Building Facade Windows. In Advances in Visual Computing 6939; Springer: Las Vegas, NV, USA, 2011; pp. 280–289.

28. Zhang, L.; Song, M.; Liu, X.; Sun, L.; Chen, C.; Bu, J. Recognizing architecture styles by hierarchical sparse coding of blocklets. Inf. Sci. 2014, 254, 141–154.

29. Xu, Z.; Tao, D.; Zhang, Y.; Wu, J.; Tsoi, A.C. Architectural Style Classification Using Multinomial Latent Logistic Regression. In Computer Vision—ECCV 2014; Springer: Cham, Switzerland, 2014; Volume 8689, pp. 600–615.

30. Llamas, J.; Lerones, P.; Zalama, E.; Gómez García -Bermejo, J. Applying Deep Learning Techniques to Cultural Heritage Images within the INCEPTION Project.

In EuroMed 2016: Digital Heritage. Progress in Cultural Heritage: Documentation, Preservation, and Protection. Part I, Nicosia, Cyprus, 31 October–5 November 2016; Springer: Cham, Switzerland, 2016; Volume 10059, pp. 25–32.

31. Lu, Y. Food Image Recognition by Using Convolutional Neural Networks (CNNs). arXiv 2016, arXiv:1612.00983.

32. Yanai, K.; Kawano, Y. Food image recognition using deep convolutional network with pre-training and fine-tuning. In Proceedings of the IEEE International Conference on Multimedia Expo Workshops (ICMEW), Turin, Italy, 29 June–3 July 2015; pp. 1–6. 36. Datahub. Available online: https://datahub.io (accessed on 25 September 2017).