

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

**SEMANTIC SEGMENTATION OF HISTORIC DOME SYSTEMS
CONSIDERING PART-WHOLE RELATIONS**



M.Sc. THESIS

Mustafa Cem GÜNEŞ

Department of Informatics

Architectural Design Computing Graduate Programme

JULY 2022

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

**SEMANTIC SEGMENTATION OF HISTORIC DOME SYSTEMS
CONSIDERING PART-WHOLE RELATIONS**



M.Sc. THESIS

**Mustafa Cem GÜNEŞ
523191008**

Department of Informatics

Architectural Design Computing Graduate Programme

Thesis Advisor: Prof. Dr. Mine Özkar Kabakçioğlu

JULY 2022

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

**TARİHİ KUBBE SİSTEMLERİNİN PARÇA BÜTÜN İLİŞKİLERİ
GÖZETİLEREK ANLAMSAL BÖLÜTLENMESİ**

YÜKSEK LİSANS TEZİ

**Mustafa Cem GÜNEŞ
523191008**

Bilişim Anabilim Dalı

Mimari Tasarımda Bilişim Lisansüstü Programı

Tez Danışmanı: Prof. Dr. Mine Özkar Kabakçioğlu

TEMMUZ 2022

Mustafa Cem Güneş, a M.Sc. student of ITU Graduate School student ID 523191008, successfully defended the thesis entitled “SEMANTIC SEGMENTATION OF HISTORIC DOME SYSTEMS CONSIDERING PART-WHOLE RELATIONS”, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Dr. Mine Özkar Kabakçioğlu**
Istanbul Technical University

Jury Members : **Prof. Dr. Gözde Ünal**
Istanbul Technical University

Assoc. Prof. Dr. Mert Nezih Rifaioğlu
Iskenderun Technical University

Date of Submission : 03 June 2022
Date of Defense : 06 July 2022





To my family and friends,



FOREWORD

The thesis study hereby presented was supported by TÜBİTAK (The Scientific and Technological Research Council of Turkey) Project Number: 119K896. The photography and the photogrammetric models used in this thesis were also produced as part of the project.

The computing resources used for the calculations in this thesis were provided by the National Center for High Performance Computing of Turkey (UHeM) under grant number 1010152021.

A very special thanks to Yusuf H. Şahin, Alican Mertan, and their advisor Prof. Dr. Gözde Ünal of the Istanbul Technical University Vision Lab for realizing the preliminary semantic segmentation using the multilayer perceptron and graph-based deep neural networks, and the related labeling, training, and testing. The thesis extends their initial definitions and approaches towards a comparative examination.

I am also thankful to my supervisor Prof. Dr. Mine Özkar for her invaluable patience and advice. I would like to extend my sincere thanks to Assoc. Prof. Dr. Mert Nezi̇h Rifaioğlu for his feedback and inspiring comments.

Thanks should also go to friends in the project research group for their support in collecting images on site and creating photogrammetry models. Berkay Öztürk took photos of the dome of the Sahip Ata Tomb Chamber, Karatay Madrasah South Dome, and created photogrammetry models of the Karatay Madrasah South and North Domes, the dome of the Sahip Ata Tomb Chamber. Sevgi Altun collected images from Karatay Madrasah North Dome. Sena Kayasü and Begüm Hamzaoglu helped with manual measuring and drawings. The data collection process would not be possible without this team. Additionally, I would be remiss not to mention their valuable discussions and feedback about Anatolian domes.

Last but not the least, I would like to wholeheartedly thank my family. I could not have undertaken this journey without their support.

July 2022

Mustafa Cem GÜNEŞ
Architect



TABLE OF CONTENTS

	<u>Page</u>
FOREWORD	ix
TABLE OF CONTENTS	xi
ABBREVIATIONS	xiii
SYMBOLS	xv
LIST OF TABLES	xvii
LIST OF FIGURES	xix
SUMMARY	xxi
ÖZET	xxiii
1. INTRODUCTION	1
2. METHODOLOGY	9
2.1 Modern Documentation Methods in Cultural Heritage.....	9
2.2 Heritage Building Information Modelling	11
2.3 The Dome Systems.....	12
2.4 Photogrammetric Data.....	14
2.5 Synthetic Historic Dome Data Set	21
2.5.1 The dome systems in monumental Anatolian architecture	22
2.5.2 The Level of Detail	22
2.5.3 The parameterization of the parts of the dome systems.....	23
2.5.3.1 The dome.....	23
2.5.3.2 The transition element.....	25
2.5.3.3 The supporting wall	27
2.5.3.4 The supporting arch.....	27
2.5.4 Combining parts of the dome systems	29
2.5.5 User-friendly semi-automatic point cloud exportation	29
2.6 Data representation schemas	30
2.7 Data Processing	33
2.8 Deep Learning	34
2.8.1 Multi-layer perceptron	39
2.8.1.1 PointNet	39
2.8.1.2 PointNet++	40
2.8.2 Graph-based: Dynamic Graph CNN	42
2.8.3 Sequence	44
2.8.3.1 RSNET	45
2.8.3.2 Point2Sequence	47
2.8.4 Transformer-attention based	50
2.8.4.1 Point Transformer	52
2.8.4.2 Point-BERT.....	53
3. EVALUATION	55
3.1 Test Set.....	55
3.2 Metrics.....	55
3.3 Examination Set-Up	57
3.4 Results of the Examinations	58

3.4.1 PointNet++	63
3.4.2 DGCNN.....	71
3.4.3 RSNET	79
3.4.4 Point2Sequence	87
3.4.5 PointTransformer	95
3.4.6 Point-BERT	103
4. CONCLUSION.....	113
REFERENCES	117
CURRICULUM VITAE.....	127



ABBREVIATIONS

10k	: 10000
2k	: 2048
3D	: Three-Dimensional
AB_M	: The Dome of the Abdulaziz Masjid
AK_GUD	: The Dome of the Konya, Akşehir Gdk Minare Masjid
AK_ULU	: The Main Dome of the Konya, Akşehir Great Mosque
AL	: Main Dome of the Konya Alaeddin Mosque
ALS	: Airborne/Aerial Laser Scanning
BERT	: Bidirectional Encoder Representations from Transformers
BiLSTM	: Bi-Directional Long Short-Term Memory
BIM	: Building Information Modeling
CNN	: Convolutional Neural Networks
CPU	: Central Processing Unit
DGCNN	: Dynamic Graph Convolutional Neural Networks
FPS	: Farthest Point Sampling
GB	: Gigabyte
GPU	: Graphics Processing Unit
GRU	: Gated Recurrent Unit
HBIM	: Heritage Building Information Modeling
HDD	: Historic Dome Data Set
HK	: The Main Dome of the Kayseri Hacı Kılıç Mosque
HSV	: Hue, Saturation, Value
IM_M	: The Main Dome of the Konya Ince(Slender) Minaret Madrasa
KM_M	: The Main Dome of the Konya Karatay Madrasa
KM_N	: North Dome of the Konya Karatay Madrasah
KM_S	: South Dome of the Konya Karatay Madrasah
k-NN	: k Nearest Neighbor
LiDAR	: Laser Imaging Detection and Ranging
LSTM	: Long Short-Term Memory
mIoU	: mean Intersection over Union

MLP	: Multi Layer Perceptron
MP	: Megapixel
MPM	: Masked Point Modeling
MRG	: Multi-Resolution Grouping
MSG	: Multi-Scale Grouping
N	: Number
NLP	: Natural Language Processing
oAcc	: Overall Accuracy
P2S	: Point2Sequence
PB	: Point-BERT
PTr	: Point Transformer
RAM	: Random Access Memory
Res	: Resolution
RGB	: Red, Green, Blue
RNN	: Recurrent Neural Network
SA_H	: The Main Dome of the Konya Sahip Ata Hanikah
SA_T	: The Dome of the Tomb Chamber Room, Konya Sahip Ata Islamic Complex
Sec	: Second
TB	: Triangular Band
TLS	: Terrestrial Laser Scanning
UAV	: Unmanned Aerial Vehicle
Val	: Validation
ViT	: Vision Transformer

SYMBOLS

L	: Label
$P_{x,y,z}$	: Position in x, y, and z Directions
$N_{x,y,z}$	: Surface Normal Vector in x, y, and z Directions
r	: Radius
f	: Flatness Ratio
p	: Pointedness Ratio
α	: Angle
n	: Number of Drum Polygon Edges



LIST OF TABLES

	<u>Page</u>
Table 2.1 : Full and partial real-world models.....	15
Table 2.2 : The setting of captured images for photogrammetry. 18 MP and 24 MP stand for 5198x3462 and 6019 and 4023 consecutively. S6000, C600D, and C650D denote Sony Alpha ILCE-6000, Canon 600D, and Canon 650D in order.....	16
Table 2.3 : The number of the points in raw and clean points.....	17
Table 2.4 : Parameters of the supporting arch.....	27
Table 3.1 : Properties of used deep learning algorithms.....	57
Table 3.2 : Preprocessing settings.....	58
Table 3.3 : The results of HDD.....	59
Table 3.4 : Test accuracies of the real-world full models.....	60
Table 3.5 : Test mIoUs of the real-world full models.....	60
Table 3.6 : Test accuracies of the real-world partial models.....	61
Table 3.7 : Test mIoUs of the real-world partial models.....	62
Table 3.8 : Results of PointNet++ using HDD and 2k points.....	64
Table 3.9 : Results of PointNet++ using HDD and 10k points.....	65
Table 3.10 : Results of PointNet++ on full models, 2k points.....	67
Table 3.11 : Results of PointNet++ on full models, 10k points.....	68
Table 3.12 : Results of PointNet++ on partial models, 2k points.....	69
Table 3.13 : Results of PointNet++ on partial models, 10k points.....	70
Table 3.14 : Results of DGCNN on HDD, 2k points.....	72
Table 3.15 : Results of DGCNN on HDD, 10k points.....	73
Table 3.16 : Results of DGCNN on full models, 2k points.....	75
Table 3.17 : Results of DGCNN on full models, 10k points.....	76
Table 3.18 : Results of DGCNN on partial models, 2k points.....	77
Table 3.19 : Results of DGCNN on partial models, 10k points.....	78
Table 3.20 : Results of RSNET on HDD, 2k points.....	80
Table 3.21 : Results of RSNET on HDD, 10k points.....	81
Table 3.22 : Results of RSNET on full models, 2k points.....	83
Table 3.23 : Results of RSNET on full models, 10k points.....	84
Table 3.24 : Results of RSNET on partial models, 2k points.....	85
Table 3.25 : Results of RSNET on partial models, 10k points.....	86
Table 3.26 : Results of P2S on HDD, 2k points.....	88
Table 3.27 : Results of P2S on HDD, 10k points.....	89
Table 3.28 : Results of P2S on full models, 2k points.....	91
Table 3.29 : Results of P2S on full models, 10k points.....	92
Table 3.30 : Results of P2S on partial models, 2k points.....	93
Table 3.31 : Results of P2S on partial models, 10k points.....	94
Table 3.32 : Results of PTr on HDD, 2k points.....	96
Table 3.33 : Results of PTr on HDD, 10k points.....	97
Table 3.34 : Results of PTr on full models, 2k points.....	99
Table 3.35 : Results of PTr on full models, 10k points.....	100

Table 3.36 : Results of PTr on partial models, 2k points.	101
Table 3.37 : Results of PTr on partial models, 10k points.	102
Table 3.38 : Results of PB on HDD, 2k points.	104
Table 3.39 : Results of PB on HDD, 10k points.	105
Table 3.40 : Results of PB on full models, 2k points.	108
Table 3.41 : Results of PB on full models, 10k points.	109
Table 3.42 : Results of PB on partial models, 2k points.	110
Table 3.43 : Results of PB on partial models, 10k points.	111



LIST OF FIGURES

	<u>Page</u>
Figure 1.1 : A regular Scan-to-HBIM process. This research study investigates the semantic segmentation problem. Revised from (Güneş et al., 2021).....	7
Figure 2.1 : Simplified representation of drum.....	14
Figure 2.2 : Photogrammetric model (full models) cleaning and labeling.	18
Figure 2.3 : Photogrammetric model (partial models-1) cleaning and labeling.	19
Figure 2.4 : Photogrammetric model (partial models-2) cleaning and labeling.	20
Figure 2.5 : Label of distribution of real-world point clouds per model.	20
Figure 2.6 : Label of distribution of real-world point clouds per model type.....	21
Figure 2.7 : Full (top), flat/segmental (middle) and pointed domes (bottom). Revised from (Güneş et al., 2021).....	24
Figure 2.8 : Dome with a lantern.	25
Figure 2.9 : The edge number of the polygons and the drum height form the transition elements.....	26
Figure 2.10 : Height and width variables define the wall.	27
Figure 2.11 : Arch types use various parameters.	28
Figure 2.12 : Label distribution of real-world point clouds per model type.	30
Figure 2.13 : The folder structure of the ShapeNet_HDD.....	32
Figure 2.14 : The folder structure of the S3DIS_HDD.....	32
Figure 2.15 : The folder structure of the ShapeNet55-34_HDD.....	32
Figure 2.16 : Data preprocessing.	33
Figure 2.17 : The neural network structure of the PointNet architecture.	40
Figure 2.18 : A hierarchical point set feature learning schema.	41
Figure 2.19 : A Multiresolution concatenation, A hierarchical point set feature learning schema.	41
Figure 2.20 : Edge feature edge convolution.	43
Figure 2.21 : Structure of the dynamic graph convolutional neural network.	43
Figure 2.22 : Flowchart of the RSNET sequence model.	46
Figure 2.23 : Overlapping blocks are used in the training phase.....	47
Figure 2.24 : Blocks used in the test phase.....	47
Figure 2.25 : Flowchart of the P2S model.	48
Figure 2.26 : Local point patches of the model.....	49
Figure 2.27 : Schema of a transformer.....	50
Figure 2.28 : Pre-training and Fine-Tuning steps of the BERT.....	51
Figure 2.29 : (Top) Semantic segmentation and classification structure of the Point Transformer.	53
Figure 2.30 : Sub-structures of the main methods in the Point Transformer Network.	53
Figure 2.31 : The network structure of the Point-BERT.....	54
Figure 3.1 : Confusion matrices of PointNet++ on HDD, (top) 2k, and (bottom) 10k.	63
Figure 3.2 : Confusion matrices of PointNet++ (left) full and (right) partial real-world models in 2k.	66

Figure 3.3 : Confusion matrices of PointNet++ (left) full and (right) partial real-world models in 10k.....	66
Figure 3.4 : Confusion matrices of DGCNN on HDD, 2k (left), and 10k (right).....	71
Figure 3.5 : Confusion matrices of DGCNN (left) full and (right) partial real-world models in 2k.	74
Figure 3.6 : Confusion matrices of DGCNN (left) full and (right) partial real-world models in 10k.	75
Figure 3.7 : Confusion matrices of RSNET on HDD, 2k (top) and 10k (bottom)....	79
Figure 3.8 : Confusion matrices of RSNET (left) full and (right) partial real-world models in 2k.	82
Figure 3.9 : Confusion matrices of RSNET (left) full and (right) partial real-world models in 10k.	82
Figure 3.10 : Confusion matrices of Point2Sequence on HDD, 2k (top) and 10k (bottom).	87
Figure 3.11 : Confusion matrices of P2S (left) full and (right) partial real-world models in 2k.....	90
Figure 3.12 : Confusion matrices of P2S (left) full and (right) partial real-world models in 10k.....	90
Figure 3.13 : Confusion matrices of PTr on HDD, 2k (top), and 10k (bottom).	95
Figure 3.14 : Confusion matrices of PTr (left) full and (right) partial real-world models in 2k.....	98
Figure 3.15 : Confusion matrices of PTr (left) full and (right) partial real-world models in 10k.....	98
Figure 3.16 : Confusion matrices of PB on HDD, 2k (top) and 10k (bottom).....	103
Figure 3.17 : Confusion matrices of PB (left) full and (right) partial real-world models in 2k.....	106
Figure 3.18 : Confusion matrices of PB (left) full and (right) partial real-world models in 10k.....	107

SEMANTIC SEGMENTATION OF HISTORIC DOME SYSTEMS CONSIDERING PART-WHOLE RELATIONS

SUMMARY

This research investigates the usage of different deep learning methods of semantic segmentation in the context of classifying the parts of the historic domes of Anatolian monumental architecture regarding part-whole relations. Documentation of cultural heritage is a detailed process that focuses on recording the state of the items with high accuracy and all the required information. However, manual documentation techniques are error-prone that lead the human-caused inconsistencies in the collected data. Even though photogrammetry and laser scanning-based tools supply three-dimensional data, the workflows that use the collected data just as a reference for human-involved, re-drawing projects such-as ortho-photo drawings are both time-consuming and error-prone. Particularly, in the Heritage-Building Information Modeling context scan-to-HBIM tasks that focus on modeling the architectural elements with different layers of information by preserving unique shapes and deformations, transferring detailed and unique elements requires precise solutions.

The presented study provides a semi-automatic, deep learning-based, as-is modeling workflow that interprets the semantic relations of the structural parts of the domes of historical Anatolian buildings. Deep neural network-based semantic segmentation algorithms can examine the given data and learn the generalized relation of the parts in the scenes. Effective usage of the networks relies on the existence of variants and numerous instances of models in the data set to train. However, the collected real-world data is limited due to the impossibility of capturing the multiple variations of different architectural elements. A synthetic data set, namely, Historic Dome Data Set(HDD) including 1537 models, is created to address this constraint by using the parametric modeling tools referring to the simplified definitions of the four parts of the historical Anatolian domes. The models include only position and normal values without color definition. Four types of deep neural networks are trained using the created data set to examine the results of different data processing approaches.

The part-whole relations are usually well defined in the design, construction, and documentation tasks for humans. There are usually sequential but interrelation connections of the architectural units on different scales. However, the definition of the relations is based on the identified instances of the architectural elements. On the other hand, deep neural networks investigate the contextual, part-whole relations of the different classes to extract semantic information using relatively ambiguous patches and interpret the link between them in various ways. The presented study examines the outcomes of the direct implementation of multi-layer perceptron, graph, sequential, transformer-attention-based semantic segmentation algorithms into the historic dome system documentation process.

To evaluate the semantic segmentation outcomes both visual and numerical results are provided and discussed. In addition to HDD, eleven real-world dome system models

from the Anatolian Seljuk period are collected using photogrammetry tools to assess the proposed methods.

To sum up, the presented research examines different deep learning-based semantic segmentation methods to acquire meaningful and accurate results based on the relations of the architectural elements. Also, a synthetic data set and an adaptive parametric modeling method for dome systems are provided that can be adapted into similar research. The results show that even if the trained networks perform well in the synthetic data set, and the real-world data includes all the parts of the dome systems, the exceptions in the real-world examples, such as sparsity of the points, noise in the collected data, unseen parts, and unlabeled elements confuse the segmentation process. Further studies might implement unlabeled objects, train the deep learning models with partial models, and focus on a unit-based investigation that focuses on creating the different sizes of patches on the point clouds to examine the relations of the part in the point clouds in an architecturally meaningful way.



TARİHİ KUBBE SİSTEMLERİNİN PARÇA BÜTÜN İLİŞKİLERİ GÖZETİLEREK ANLAMSAK BÖLÜTLENMESİ

ÖZET

Bu çalışma Anadolu’da bulunan tarihi yapılarının kubbelerini oluşturan ve destekleyen yapısal elemanların parça bütün ilişkisine göre anlamsal bölütlenmesi bağlamında farklı derin öğrenme yaklaşımlarının etkilerini incelemektedir. Kültürel mirasın belgelenmesi, mimari elemanların yüksek doğruluk oranıyla ve gerekli olan tüm bilgilerle kayıt altına almaya odaklanan detaylı bir süreçtir. Bununla birlikte, geleneksel belgeleme yöntemleri çalışılan geometrilerin düzensizliği, özgünlüğü, kullanılan araçların veya yöntemlerin hata payına izin verir olması ve insan kaynaklı oluşabilecek aksaklıklardan ötürü hataya açıktır. Fotogrametri ve lazer tarama tabanlı araçlar, belgelenen parçaların üç boyutlu temsillerini çok düşük hata payıyla dijital ortama aktarsa da elde edilen verinin dolaylı kullanımları iş akışını aksatmakta ve elde edilen modellerin gerçeğe yakınlığını bozmaktadır. Özellikle Miras Yapı Bilgi Modelleme (MYBM) kapsamında, mimari elemanların farklı bilgi katmanları ile eşsiz şekilleri ve deformasyonları korunacak şekilde modellenmesine odaklanan taramadan-YBM’ye aktarılması çalışmaları yüksek doğruluk oranı gerektirmektedir.

Nokta bulutu olarak taranmış kültürel miras eserlerinin YBM ortamında temsili sırasında elde edilen veriyi farklı şekillerde işlemek mümkündür. İlk olarak, üç boyutlu veriyi belirli düzlemlere yansıtarak perspektiften arındırılmış referans imajlar geleneksel çizim yöntemlerine benzer bir yaklaşım sunmaktadır. Bu metod, elde edilen verinin olduğu haliyle işlenmesi yerine iş akışına kullanıcı kontrolünde ek adımlar eklemektedir. Dolayısıyla, iş akışı insan emeğine dayalı olması sebebiyle hataya açıktır. Ek olarak, doğrudan üç boyutlu modellerin kullanılması adına kullanıcı odaklı etiketleme ve gruplama işlemleri gerçekleştirilmektedir. Ancak, bu yöntem de kullanıcı odaklı iş akışı sebebiyle emek-yoğun bir süreç oluşturmaktadır.

Gerçek dünyadan toplanan verilerin işlenmesi adına yüksek kaliteyle ve istenen bilgiyi koruyacak şekilde süreci standardize etmek adına farklı yaklaşımlar denenmiştir. Geleneksel yöntemlere benzer şekilde elde edilen verinin belirli düzlemlere yansıtılması sayesinde fotoğraflarda olduğu gibi iki boyutlu otomatik sınıflandırma işlemleri uygulanabilmektedir. Ancak bu yaklaşım kaydedilen yapının üç boyutlu ilişkilerini göz ardı etmektedir ve sadece derinliği sınırlı olan düzlemsel alanlarda ya da yüzeylerde anlamlı sonuçlar vermektedir. Diğer bir yandan, süreci anlamlı şekilde otomatize etmek adına farklı makine öğrenmesi algoritmaları üç boyutlu nokta bulutlarının sınıflandırılması amacıyla kullanılmıştır. Özellikle bazı uygulamalar noktaların birden çok geometrik özelliğini işlenebilecek istatistiksel bir bilgi olarak kullanmaktadır ve kaydedilmiş olan nokta bulutlarının kullanışlı şekilde bölütlenmesine imkan tanımaktadır. Yine de süreci yarı otomatik duruma getiren bu yaklaşım genellenebilir olmaktan uzaktır. Belirli bir kültürel miras belgelemesi problemi bağlamında çözümün genellenebilir olması; ayrıştırılması ve tanınması gereken mimari elemanların ilişkilerinin anlamlı şekilde işlenebilir ve deşifre edilebilir olmasıyla ilgilidir. Makine öğrenmesi algoritmalarının daha özelleşmiş bir grubu olan

derin öğrenme algoritmaları insanlardaki sinir ağlarına benzer şekilde birbirine bağlı bütünleşik işlemler yaparak kendilerine sunulan örneklem kapsamındaki anlamsal bilgiyi çözümleyebilmektedir. Nokta bulutlarının bağlamsal ilişkiler göz önünde bulundurularak anlamlı sınıflandırılmaları, kültürel miras eserlerinin farklı mimari elemanlara ait noktaları tanıyarak YBM'ye geçiş sürecinin otomatikleştirilmesi açısından önemli bir potansiyel barındırmaktadır.

Bu araştırma, tarihi yapıların olduğu gibi modellenmesi kapsamında, yarı otomatik bir iş akışı içerisinde, Anadolu coğrafyasında bulunan tarihi kubbelerinin yapısal elemanlarının ilişkilerinin deşifre edilmesi ve farklı örneklerde bu bilginin kullanılması amacıyla derin öğrenme ağlarından faydalanmaktadır. Derin sinir ağları tabanlı anlamsal bölütleme algoritmaları, eğitim için kullanılan verileri inceleme ve modellerdeki parçaların genelleştirilmiş ilişkilerini öğrenme yeteneğine sahiptir. Ağların etkin kullanımı eğitilecek veri kümesinin içeriğinin çeşitliliğine ve örneklerin çokluğuna dayanmaktadır. Ancak, tarihi yapılardan toplanacak örneklerde yeterli varyasyonun bulunmaması, kayıt altına alma işleminin ekonomik açıdan masraflı olması ve uzun süre alması bir kısıt oluşturmaktadır. Bu kısıtı aşmak için Anadolu'daki tarihi örneklerle dayandırılarak, kubbe, geçiş elemanı, kemer ve duvardan oluşan dört sınıf yapı elemanı barındıran 1537 adet kubbe sistemi modeli parametrik modelleme araçları kullanılarak Tarihi Kubbe Veri Kümesi (HDD) kapsamında oluşturulmuştur. Modeller kubbe sistemlerinin zemindeki merkezinden fotoğraf çekilerek oluşturulan nokta bulutlarını referans alacak şekilde oluşturulmuştur. Çalışılan detay düzeyinde, renk ve malzeme bilgisinin parametrik modelleme sırasında gerçekçi ve doğru uygulamasının zorluğundan ötürü yalnızca noktaların uzaydaki pozisyonları ve normal bilgileri temsil edilmiştir.

Oluşturulan veri seti kullanılarak, farklı veri işleme yaklaşımlarının kubbe sistemi elemanlarının anlamsal bölütlemesine etkilerini değerlendirmek amacıyla dört ayrı türden toplamda altı derin öğrenme ağı eğitilmiştir. Parça bütün ilişkileri, belirli elemanların benzer ilişkiler içerisinde düzenli kullanımı, tasarım, yapım ve belgeleme sırasında net şekilde gözlemlenebilir. Özellikle farklı mimari birimlerin genellikle sıralı ya da birbirine bağlı ilişkileri vardır. Bu düzenli ilişkiler mimari elemanların birbiriyle olan bağlarına dayanmaktadır. Örneğin tarihi yapılarda tuğla birimlerinin belirli geometrik düzenlerde kullanılması farklı yüzeylerde hem fonksiyonel hem de estetik sonuçlar oluşturmalarını sağlamıştır. Öte yandan, çoğu derin sinir ağı, anlamlandırılmaya çalışılan sahnelerde nispeten belirsiz sınırlara sahip birim parçalar kullanarak anlamsal ilişkileri çözümler ve geneller. Bu ilişkiler, bağlam farkındalığı olan bölgesel ve genel ilişkiler olarak tanımlanabilir. Yani, küçük ölçekte nasıl bir ortamda bulunulduğu ve genel ilişki kuralları tanımlanırken büyük ölçekte parçaların birbiriyle olan detay ilişkileri incelenir.

Sunulan çalışma çok katmanlı algılayıcı, grafik-tabanlı, ardışık ve transformatör bölütleme algoritmalarının tarihi kubbe sistemi belgeleme sürecinde doğrudan uygulanmasının sonuçlarını incelemektedir. Anlamsal bölütlemenin sonuçlarını değerlendirmek amacıyla hem görsel hem de sayısal sonuçlar sunulmakta ve tartışılmaktadır. Elde edilen sonuçlar arasında karşılaştırma yapabilmek adına birleşimlerin kesişimi, doğruluk, kesinlik, duyarlılık ve F1 skoru kriterleri kullanılmaktadır. Bunlara ek olarak birbiriyle karışan nokta sınıflarının takibini yapabilmek adına karışıklık matrisi kullanılmıştır. Elde edilen görsel sonuçlar sayesinde referans modellerle tahmin sonucu üretilen modeller karşılaştırılarak niteliksel değerlendirmeler gerçekleştirilmektedir.

Bu kapsamda, değerlendirme örneklemini genişletmek amacıyla HDD'ye ek olarak Anadolu Selçuklu dönemi eseri olan on bir gerçek kubbe sistemi modeli, fotogrametri yöntemiyle oluşturulmuştur. Oluşturulan gerçek nokta kümeleri bütün modeller ve parçalı modeller olmak üzere iki alt-küme altında incelenmiştir. Bütün modeller, tanımlanan dört sınıfın da büyük eksik parçalar olmadan temsil edildiği, noktaların göreceli olarak dengeli dağıldığı örneklerden oluşmaktadır. Parçalı modeller ise, tanımlanan dört sınıftan bir kısmını içeren, noktaların uzayda standart olmayan yoğunluklarla yerleştiği, belirli yapısal elemanları temsil eden noktalar bulunsa bile alan ya da fotogrametrik modelleme yöntemlerinin kısıtlarından ötürü parçaların bütününe temsil edilemediği, boş bölgeler içeren nokta bulutlarını temsil etmektedir.

Özetle, sunulan araştırma, mimari elemanların ilişkilerine dayalı olarak anlamlı ve doğru sonuçlar elde edebilmek için farklı derin öğrenme tabanlı anlamsal bölütleme yöntemlerini anıtsal Anadolu mimarisi kubbe sistemleri özelinde incelemektedir. Ek olarak, araştırma kapsamında, benzer problemlere adapte edilebilecek kubbe sistemi veri kümesi oluşturulmuş ve araştırmanın kapsamına göre özelleştirebilir parametrik modelleme yöntemi sunulmuştur. Sentetik veri ile eğitilen ağlar aynı kümenin test bölümünde ve yüksek kalitedeki bütün modellerde oldukça başarılı yakın sonuç verse de gerçek fotogrametrik verilerle yapılan değerlendirmelerde sonuçlar farklılık göstermektedir. Bu farklılıklar temel olarak, gerçek verilerdeki nokta bulutlarının yoğunluğunun tutarsızlığı, farklı sınıflara sahip noktaların dengesiz dağılımı, parçalı modellerin parça bütün ilişkisinin yeterli biçimde deşifre edilememesi, kapı boşluğu, pencere boşluğu gibi detayların sentetik veri kümesinde tanımlanmamış olması, veri kümesinin sadece kare tabana oturan kubbe sistemlerini referans alması, eğitim kümesinin sınırlı çeşitte örnek barındırması, gibi etkenlerden kaynaklanmadır.

Devam eden çalışmalar, gerçek modellerde karşılaşılan ve yapısal sınıflardan birine ait olmayan elemanları da barındırabilir, herhangi bir sebeple parçalı olarak elde edilen modellerin anlamlandırılabilmesi için eğitim veri kümesine parçalı veriler de eklenebilir. Bir kubbe sisteminin daha geniş bir bağlamda incelenmesi için eyvan ve tonoz geometrileri plan şemalarına uygun şekilde çalışmaya dahil edilebilir. Çalışma kapsamında tercih edilen detay düzeyine istinaden kullanılmayan renk ve doku bilgisinin parametrik modellere, tarihi yapılardaki bilgiye uygun olacak şekilde işlenmesi çalışılabilir. Ek olarak, mimari elemanların birimlerinin büyüklükleri ve çalışılan detay düzeyleri göz önüne alınarak derin öğrenme ağlarında kullanılan örneklem birimleri üzerinden detaylı analizler gerçekleştirilebilir. Bu tez kapsamında gerçekleştirilen analizler kullanılan ağların doğrudan uygulamasını içermektedir ancak gerçek verilerde karşılaşılan problemlerin giderilebilmesi için sunulan modern ağlardan bir tanesi seçilerek detaylı bir ince ayar çalışması yürütülebilir. Bunlara ek olarak, bu çalışma, taramadan Miras Yapı Bilgi Modellemesi'ne geçiş kapsamında anlamsal bölütlemeye odaklansa da sınıflandırılan nokta bulutlarının kümelenmesi ve mimari elemanların YBM ortamında yeniden oluşturulması nihai sonuç ürün hakkında yönlendirici çıktılar sunabilir.



1. INTRODUCTION

Traditional methods of documentation of architectural cultural heritage are usually error-prone and do not benefit from the advantages that modern tools bring. Even though the use of photogrammetry and laser scanning-based data capturing tools provides mostly accurate three-dimensional data, there are some steps to be improved to record different architectural elements as a part of a whole with high accuracy using their semantic relations. This research focuses on the impacts of different types of three-dimensional semantic segmentation networks on historic dome systems regarding the part-whole relations of the structural elements. The study investigates the parts of domes systems in the context of historic Anatolian architecture referencing the buildings constructed between the 13th and 15th centuries. Even though the research addresses a specific geography and period, dome systems used in this thesis could be adapted to benefit from any style and design approach of multiple architectural epochs and communities. The definitions do not target one specific design approach but instead offer a generic and limited solution for multiple and related historic Anatolian domes. Dome systems that are examined in the process and share a simple square floor plan, which defines the dimensions of shapes and sets the boundary of a geometry to be investigated. Also, the proposed procedure can be implemented in different kinds of domes of different eras in the Cultural Heritage context, and investigated areas and elements can be extended.

Parametric modeling methods are used to create simplified synthetic versions of the monumental Anatolian domes that serve as a collection of photogrammetric dome system models. Subsequently, deep neural networks specialized in classifying the points of the models by predicting the structural part. The presented study investigates explicitly the networks that interpret data in 3 different strategies. Even though the data is expressed as low-detailed synthetic models, the performance of the deep learning models is tested in various real-world examples with different types of structural elements. The results are evaluated quantitatively and qualitatively considering the usability of documenting historical buildings in a scan-to-HBIM

process. Finally, contributions and limitations are discussed, and future studies are declared.

Cultural heritage studies include documentation tasks that are the main part of the initial steps. However, conventional methods such as manual drawing and measuring are extremely time-consuming and labor-intensive (Malinverni et al., 2019; Anil et al., 2013). To address these drawbacks, recent studies profits from digital tools that can collect information quickly and with precision. Various studies investigated the usage of photogrammetry and terrestrial laser scanning (TLS).

Photogrammetry is a method that combines a set of images and yields a three-dimensional point cloud that keeps position and color values at acceptable levels. Similar to modeling from one-by-one images, some studies focus on creating models from video capture (Kadi & Anouche, 2020). Similar to modeling from one-by-one taken images, some studies focus on creating models from video captures (Nagakura et al., 2015). Additionally, some studies use unmanned aerial vehicles, using both the video and photo capturing methods in favor of collecting the details from hard-to-reach spots (Sestras et al., 2020). On the other hand, TLS is a method that registers geometric shapes with a very low level of deviation because of the high accuracy of measuring using laser beams. It is also preferred due to the high accuracy and detailed results that it provides in the documentation process of cultural heritage (Andriasyan et al., 2020; Quattrini et al., 2015). Additionally, combining both methods is used to improve the overall quality and enrich the information by adding new layers with additional sensors such as infrared (Khalil & Stravoravdis, 2019). Consequently, while TLS offers precise and dense models in return for expensive and specialization-required processes, photogrammetry provides low-cost, easy-to-produce, and quick results in exchange for lower quality.

Even though point clouds store specific local information per point, they would be meaningful when they retain required information to be used by humans or algorithms in the documentation process. While manually segregating the architectural elements in the collected points is one of the first methods, it creates an error-prone, labor-intensive, inefficient bottleneck in the documentation process (Xiong et al., 2013; Anil et al., 2011). This research addresses automating the segmentation processes by using deep learning tools regarding the required precision of the segmented outputs that will be used to build Historic Building Information Modeling (HBIM) geometries.

Scan-to-Building Information Modeling (Scan-to-BIM) or as-is modeling concentrates on transforming collected point cloud data into meaningful building information models that contain required information embedded in geometries. In the Cultural Heritage Domain, Scan-to-Heritage BIM (Scan-to-HBIM) represents an end-to-end workflow starting from point clouds to semantic architectural element representations. Various studies focus on documenting the cultural heritage buildings with distinct workflows corresponding to their focus. Nevertheless, in general, the process starts with data collection and continues with segmenting point clouds regarding the definitions of the architectural elements, creating the HBIM objects. The presented study focuses on the segmentation step vital for accurately modeling the parts of the historical buildings.

There are three types of approaches to segment the required data. The initial approach is using the images directly to identify the elements on them. The second method is using a point cloud to represent the architectural elements but projecting them to ease the segmentation process skipping the complexities of the third dimension brought in. Last and the most realistic solution is interpreting the properties of the points in the point clouds in three dimensions. To extract required semantic information from the images, Pavoni et al. (2020) and Oses et al. (2014) investigated the segmentation problem of the different types of masonry walls. They used deep neural networks to segment colored wall images accurately. Similarly, Loverdos et al. (2021) identified the masonry elements and the mortars in between them to semantically run a structural analysis on the acquired information and analyze the possible deformations in different situations. Even though the studies show that segmentation results provide the expected solutions, this approach is unsuitable for a three-dimensional workflow where the representation of the elements in different directions is needed. Also, it is limited to color values that might be misleading in extreme cases. Thus, since it is particularly efficient for analyzing plain surfaces such as walls but not in scenes with curvilinear objects with depth and depth.

Point clouds provide additional information such as the neighborhood in three-dimensional space, and normals of the values in addition to color values. Yet in some cases, a combination of images and point cloud properties gives accurate results. Since the data processing includes the characteristic of both of the representations this method is called 2.5D to emphasize the correspondence between three-dimensional

and two-dimensional data. For example, Valero et al. (2018) segmented stone blocks and mortars using the point clouds as raw input but projecting the point clouds onto a planar surface to a well-defined skeleton-like network of mortars. Embedding depth values into the points provided additional information on image-like analysis. This method addresses a specific type of problem in the cultural heritage domain. However, analyzing curvilinear shapes with different details or spherical geometries might be error-prone and constraining. Therefore, segmentation of the elements of the dome system may not be optimal due to complex three-dimensional characteristics.

Lastly, as a consequence of advantages brought by improvements in hardware and algorithms, some studies concentrated on directly segmenting the historical elements of the cultural heritage using point clouds. To illustrate, Kadi and Anouche (2020) used point clouds as a basis of parametric models of different parts of the historic elements. Even though they represented the recorded elements in the BIM environment, their method relies on a large amount of manual process and is far from being automated. Yet, they create a grammar-like library that can be used in future studies using a python script to generate models corresponding to the collected models. Tiberti and Milani (2020) dealt with complex masonry wall analysis problems using three-dimensional point clouds. Apart from the previous masonry wall cases, this research considers the behavior of the materials in three dimensions and investigates the multi-leaf walls regarding the shapes of the units. Khalil and Stravoravdis (2019) fused both point clouds with infrared data to evaluate the thermal performance of the building. Their study shows that multiple sensors might be used simultaneously to gather additional information to be used in the documentation and analysis steps. However, this research lacks automation of the segmentation of the elements and parametric representation of the elements.

Regarding the automation of the segmentation process, Teruggi et al. researched (2020) the hierarchical segmentation procedures using the multiresolution approach. Using Random Forest, a machine learning algorithm serves to interpret geometrical features of the points; hence, they acquire semantic information on different scales. Their ontological investigation is valuable for dealing with multiple levels of details in one model. Yet, semantic understanding of the method can be improved to validate the robustness of the method. By defining objects in the BIM environment at different levels, the workflow would offer an end-to-end process. Similarly, Croce et al. (2021)

inspected dividing different architectural elements of historical buildings using machine learning tools and reconstructing them in BIM environments by extracting required parameters. They use algorithms that use additional geometric features such as eigenvalues, planarity, and linearity as input. Also, the study manages to handle labeling seventeen architectural classes. Barontini et al. (2021) document different elements of a historic building regarding their damage states and shapes as BIM objects. This particular use case is vital in the cultural heritage domain for integrating recorded data, the state of the structure, and the geometric model.

Lately, some studies have focused on processing the raw point clouds with deep learning tools and extracting semantic information regarding the expected properties. Some studies (Pierdicca et al., 2020) label the points considering the semantic relations of the architectural elements using deep neural networks. Examining three-dimensional data and learning from examples is promising because most of the documented architectural elements are meaningful in a context. The parts are usually nested into each other, and to disseminate the points with semantic information, an examination of the general and local context is required. Since deep neural networks used for point cloud semantic segmentation interpret the local and global features in addition to color location and surface normal values, deep neural networks can learn how to identify the class points of the architectural models. To generalize the seven examples, deep neural networks require extensive data. Nonetheless, documenting the buildings which have similar characteristics is laborious and unaffordable. Despite the technical limitations, the study shows that it is possible to adapt deep-learning networks to documentation tasks and classify the parts of the buildings.

Moreover, some research focuses on the problem of limited data in the cultural heritage domain. For instance, Pierdicca et al. (2019) model different types of columns with various details using the Blender platform and python programming language. They designed an algorithm that can generate a number of synthetic column models. Yet the usage of the interface and adaptability of the workflow to other cultural heritage problems might be challenging. Morbidoni et al. (2020) used individual models acquired from online resources to expand the coverage of the training dataset. Consequently, to overcome the limited amount of data problems in the cultural heritage context, synthesizing relevant models and implementing them into the semantic segmentation offers significant potential. Regarding the documentation of

the historic dome systems of Anatolia, this method is preferred due to the shortage of data in that specific context.

Furthermore, parametrization and reconstruction of BIM objects are vital steps in as-is modeling. Andriasyan et al. (2020) examined the parametric reconstruction of the architectural elements that were collected as points clouds using different methods and platforms. Similarly, representation of the mesh geometries in BIM environments is another essential point in the workflow due to challenges in representing geometric models with their details (Yang et al., 2019). In a standard scan-to-HBIM process, the objects need to be represented in standardized formats containing all the acquired information, especially extracting the dimensions automatically, representing deformations, and keeping data to the original scan as much as possible are essential concerns. Thus, to model and represent the required data, we need to segment the point clouds as precisely and accurately as possible.

Representations of architectural elements have multiple circumstances regarding the scale of the project and the inspection problem. Similarly, the elements are not always modeled with an exact detail level in as-is models. Instead, we might require different types of information for different purposes. Several studies examined the reconstruction and documentation of architectural objects with different specifications in the BIM environment (Banfi, 2020; Brumana et al., 2019). In order to standardize the workflow, BimForum (2021) defined the Level-of-Development that determines the geometric quality and embedded information in the models. The higher level of details is represented starting from LOD100 and goes down to LOD500 for simplified approximate geometrical representations with optional embedded information.

This research discusses the semantic segmentation of the historic dome systems regarding the diverse methods that evaluate raw point clouds data. This research uses a synthetic data set of monumental Anatolian domes to train deep neural networks. The work advances in extracting semantic information and preserving the part-whole relations of the elements. Principally in design and construction processes, architectural objects get nested with each other in a predefined order. However, even though the relationship is linked, it is not always sequential, and different elements in different stages create bases for the elements connected to it. Considering the correlation between the architectural units and the way of data depiction, the preferred methods impact the outcome. Thus, this research will discuss the consequences of

interpreting raw point cloud data with the Historic Dome Data Set (HDD) of the historic Anatolian architecture case.

HDD is a synthetic parametric dome data set that includes four elements called a dome, transition element, wall, and arch under a category called ‘dome system’ that represents fundamental structural elements of the dome and its supporting element. The data set involves 1536 point cloud models that are created, defining variations of the elements, defining parameters, and combining them respecting the real-world examples. Even though the exported data set is narrowed down to the style of monumental Anatolian architecture, it is a tool that any researcher can contribute by tweaking some points of the workflow and implementing it in their unique generation, documentation, or segmentation problems. The HDD synthesizer algorithm is not only capable of generating monumental Anatolian domes but also adaptable to different types of styles such as Persian, Byzantine, Roman, and Seljuk.

As mentioned earlier in as-is recording and scan-to-BIM workflow, geometries are represented in a specific detail level or a multi-level of details. This research focuses on segmenting parts of the dome systems by not relying on their color information and surface details that can be defined by material characteristics. Instead, it evaluates the positions and normals of the points. BIMForum (2021) defined the grade of the implemented information on geometries by the notion of Level-of-Detail. Considering the detail level of the represented dome systems, the examination gets shaped in levels of detail 200 (LOD 200).

To sum up, in the context of documentation of cultural heritage, this thesis examines the results of different deep learning approaches to semantic segmentation of point clouds using photogrammetric models. This research offers an end-to-end scan-to-HBIM process but focuses on semantic segmentation. The study exemplifies the reconstruction of a sample model to prove that the investigated methods provide valuable results that will be used in further steps (Figure 1.1).

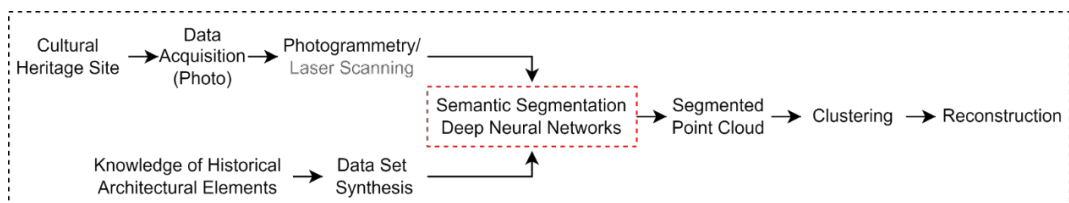


Figure 1.1 : A regular Scan-to-HBIM process. This research study investigates the semantic segmentation problem. Revised from (Güneş et al., 2021).



2. METHODOLOGY

In a proposed scan to BIM workflow, this work has several sub-steps and attention points that define the research process. First, the documentation problem in the cultural heritage context will be examined. Subsequently, the usage of photogrammetry and photogrammetric data in historic dome systems of the historical Anatolian architecture context is defined by referring to the simplified definitions made for Anatolian Seljuk and early Ottoman periods. Next, data synthesizing for creating a data set is explained. Then, semantic segmentation approaches of the deep neural networks are discussed regarding the relationship between the data processing method and architectural elements. The methods in each step are discussed in detail, and the acquired results of the methods are evaluated in the following evaluation section.

2.1 Modern Documentation Methods in Cultural Heritage

Documenting existing buildings and artifacts and their parts has always been an important task that requires high precision. The traditional workflow includes manual measuring using the measuring instruments, draft sketches, photographs, and computer-aided drawings. It consists of an un-coordinated, error-prone, discontinuous transfer of knowledge in multiple steps through different mediums.

Recently, advancements in technology made the processes more accurate and organized, by using new data acquisition methods such as photogrammetry and laser scanning. Both data acquisition methods record the real-world as-is data as accurately as possible. While photogrammetry provides more user-friendly, economic, and quick results but sacrifices the resolution of the result, laser scanning registers the scale with reliable and very high accuracy despite the requirement of expensive equipment and specialized users.

Regarding the cost-effective and simple workflow of the process, photogrammetry is being largely used for capturing objects at different scales. The process starts with taking photos with the same setting to merge them easily later in the process. After

acquiring enough photos of the objects with different angles and frames photogrammetry software merges them to export a three-dimensional model as mesh or point cloud.

In the photogrammetric data acquisition process, in the first step, photos can be collected with a professional camera or any device that can use the same settings consistently with high resolution. So the opportunity of using personal cameras, including modern smartphones, gives users freedom in the recording stage. Also, digital model making has several options concerning the priorities of the user, such as rapidness, a fee of the usage, and being error-prone. There are a couple of commercial products such as Metashape Agisoft (2022), Capturing Reality RealityCapture (2022), Autodesk ReCap Pro (2022), 3DFlow 3DF Zephyr (2022), and open-source photogrammetry software COLMAP (Schönberger & Frahm, 2016), AliceVision Meshroom (Griwodz et al., 2021) that return valuable results. Inherently, paid software presents a more user-friendly and automated user interface for a limited period or limited usage scenarios. Conversely, free and open-source are flexible and permit limitless tries with excessive parameters. On the other hand, a laser scanning workflow usually not only requires professional settings and tools that collect data but also a professional user who has prior knowledge of using the tools. In both methods, the software outputs textured mesh or point clouds with positions of points, color values, and normal values if required.

In the cultural heritage field, taking photos or scanning objects using a tripod and moving inside or around the architectural objects is a common practice. Terrestrial Laser Scanners (TLS) are used for collecting information while ensuring that the recording device is steady (Quattrini et al., 2015). However, some elements such as domes, ceilings, pediments, minarets, and ornaments are usually higher than the eye-level, so to capture the details of all the elements, unmanned aerial vehicles (UAV) are being used. To illustrate, Murtiyoso et al. (2016) examined documentation of a historical facade while acquiring all the required information. Primarily, they focused on capturing the details from close range. Similarly, Murtiyoso et al. (2017) investigate the photogrammetric modeling using close-range photos acquired from historical building exteriors. These studies emphasize the potential of using the UAVs in the documentation process for accessing all visible parts of the building and creating a model with a minimum amount of data loss. Moreover, Croce et al. (2019) exemplified

the usage of UAVs in a large-scaled area, including architectural elements on different scales in the scan-to-HBIM process. To sum up, these studies show a significant potential of the photogrammetric data acquisition using the UAVs to enrich the quality and amount of the data by close-range recording from different angles.

In addition to using photogrammetric data acquisition using UAVs, some research uses Aerial/Airborne Laser Scanning (ALS) for capturing information directly from laser scanning devices. Kedzierski & Fryskowska (2014) investigated a case of combining TLS and ALS to enhance the overall quality and capture the models without missing details of the Urban modal. Correspondingly, this method is promising in gathering a data set with minimum detail loss and a holistic meaningful model.

Moreover, instead of using singular images acquired from the cameras, some research extracts photogrammetric models directly from videos. Especially when capturing larger areas that do not require superior precision and do not get impacted by weak lighting conditions, this approach provides quick and high-quality outcomes. For instance, Xu et al. (2016) integrated the results of TLS and a photogrammetric model acquired from a video captured by a UAV. Capturing terrestrial information by laser scanning and video capturing using aerial devices benefits from the simplicity of sequential image extraction from the videos and provides data to generate three-dimensional models that can be used individually or combined with other data acquisition methods.

2.2 Heritage Building Information Modelling

Alongside the different technical approaches to collecting data from a cultural heritage site, a common practice is storing and using the recorded data in a way that would be meaningful for professionals who analyze and use it. Building Information Modeling (BIM) is being used especially, to keep the collected data in a common format that different disciplines can use, preserving essential information embedded in geometric models on various scales. In BIM, each model is represented with predefined properties, such as physical attributes, dimensions, material, and any other information required in documentation or modeling. Even though there are several different software environments for BIM, all of the software represents models in the same structure and convention which is called Industry Foundation Classes (IFC) (Ochmann et al., 2019). This standard provides a comprehensive representation and flow of data

in different platforms and keeps the information validated. Nonetheless, in most cases, IFC uses generalized descriptions of the architectural design objects with a couple of parameters (Quattrini et al., 2015). This situation might be challenging for representing architectural elements with irregular shapes, characteristics, and deformation.

Cultural Heritage tasks require unique libraries for the architectural element they are dealing with because in common practice existing libraries aim to represent new well-defined objects with smooth surfaces placed on a regular site. However, architectural elements in historic sites are usually deformed due to aging, structural load, and environmental impacts (Barontini et al., 2021; Valero et al., 2019). This information is vital because it is used in structural analysis, preservation, damage analysis, or any task related to captured data. Heritage Building Information Modeling addresses these requirements for keeping documented digital data as close to real-world data as possible, taking all the unique shapes and additional information about the shapes into account. This assignment requires exceptional definitions of the distinct elements (Barazzetti, 2016).

Documentation of the process has several steps, such as acquiring data, processing the data, and getting the HBIM model that includes all the required information. The end-to-end process is named Scan-to-HBIM. Whether labor-intensive or fully automated, Scan-to-HBIM includes three-dimensional data capturing, extracting the meaningful information from the collected model, and reconstructing the final BIM objects with all the segmented architectural elements, including the optional supplementary details (Santagati et al., 2018).

The presented study addresses a specific step of the Scan-to-HBIM process by explicitly investigating the machine-learning of an architectural knowledge of dome systems from synthetic data sets and predicting the labels of collected raw dome-system point clouds using deep learning-based semantic segmentation tools. The aim is to inspect different deep learning methods that process the data to automatically label the points used in the reconstruction of the parts of the dome system.

2.3 The Dome Systems

Starting from the old ages, domes are used to create spacious spaces while structurally distributing the loads to elements under them. Historical domes usually have spherical

or elliptic shapes depending on the use of the material and the design decisions. For example, the dome of the Pantheon which is made of concrete has a semi-circle shape with an approximate radius of 22 m that fits on a circular base (Masi et al., 2018; Aliberti & Alonso-Rodríguez, 2017). Similarly, the great heritage of the Byzantine era, Hagia Sophia has a semi-dome with a radius between 16.69 and 16.81 depending on the chosen direction calculated (Bianchini, 2020). With a half dome and pendentives, the dome ensures both structural and architectural prerequisites.

Also, Persian Architects designed remarkable domes in different eras starting from the pre-Islamic era to today. In particular, they created phenomenal samples of semi-elliptical domes. To illustrate, semi-elliptic domes of the Zoroastrian temple and palace of Ardeshir with 6.65 m of radius stand on squinch transition elements (Ashkan & Ahmad, 2009) are remaining as a heritage of the Sassanid era. Additionally, Persian culture has several different kinds of dome forms regarding the shape of the inner dome, the number of the shell, and the connections of the shells (Valibeig et al., 2017). Furthermore, in the Seljuk Empire era, various structures with hemispherical and semi-elliptical domes were created with a diverse range of transition elements (Pashai et al., 2019). Moradi (2020) specifically analyzes the transition zones of the Seljuk empire and represents their evolution in different periods and geographies. Thus, regardless of the period or location, historical dome geometries have both structural and aesthetic properties that carry the load of the structure and transfer it to the ground using the sub-elements.

Restoration and documentation projects recorded in detail Anatolian period and early Ottoman period structures and the features of the dome systems were inspected detailly. The domes of the stated era and region are investigated in three main divisions: domes, transition elements, and structural elements (Güneş et al., 2021). In the schema of the investigated dome systems, the room starts with a square floor plan and the parts of the domes system transform it into a polygon that supports the round foundation of the dome (Şimşek, 2010). The circular foundation of the dome might be shaped in semi-spherical, flat, and pointed forms. Four fundamental types can be used to express transition elements: planar triangle, triangular band, pendentive, squinch, triangular band, and planar triangle (Turan & Yaldız, 2018). A drum element over the transition elements heightens the dome, optionally. In the scope of this research, vertically erected parts are based on the polygonal end of the main transition element

defined in a transition elements class (Figure 2.1). Yet it should be noted that even though functionally it provides secondary support for the dome, not all the drum parts follow the same design principles. The instances that follow the curvature of the dome without having a polynomial base except the pendentive cases are accepted as a part of the dome. A wall or an arch can provide structural support to the dome system. Multiple alternatives of arches include semi-circular, segmental/flat, pointed, single-centered tangents, and double-centered tangents (Güneş et al., 2021). The dataset presented in this research labels all the geometries of the domes systems as a part of 4 main classes: the dome, the transition element, the wall, and the arch that involves all the described sub-categories.

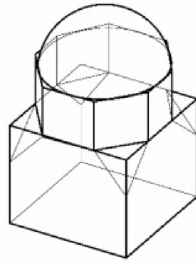


Figure 2.1 : Simplified representation of drum (Turan & Yıldız, 2018, 176).

Dome geometries are investigated as a part of a problem of cultural heritage documentation in different studies. Capone & Lanzara (2019a) investigated the representation of the Renaissance domes in the HBIM environment by creating a library based on procedural modeling. Lanzara & Capone (2020) combined both procedural modeling and documentation approaches in a sense of automatic and valid recording of specific dome types that are based on specific shapes and patterns. Capone & Lanzara (2019b) exemplified the methods of reconstruction of the dome geometries using the acquired point clouds and predefined parametric dome libraries.

2.4 Photogrammetric Data

To evaluate the achievement of the results of deep learning-based semantic segmentation algorithms that are trained with synthetic low-detailed models, a set of real-world data is used. To acquire the required points clouds 60 to 160 photos are taken in a specific setting that is required for getting clear, dense, and accurate results. In the photography capturing sessions Canon 600D, Canon 650D, and Sony Alpha ILCE-6000 are used for the main raw image shooting tasks.

All the images that are used in the photogrammetric modeling are collected within studies of the TÜBİTAK (The Scientific and Technological Research Council of Turkey) research project (119K896) and by a team of its researchers. Berkay Öztürk collected the images of Konya, Sahip Ata Tomb Chamber, and Karatay Madrasah south dome cases while I took photos of the rest of the dome cases which are Konya, Sahip Ata Hanikah; Konya, Karatay Madrasah main dome; Konya, Ince Minareli Madrasah; Kayseri, Hacı Kılıç Mosque; Konya, Karatay Madrasah north dome; Konya, Alaaddin Keykubat Mosque; Konya, Abdulaziz Masjid; Konya, Akşehir, Gündük Minareli Mosque; and Konya, Akşehir, Great Mosque.

Real-world cases are split into two according to the quality of the images, and the representation of different elements in a scene. The dome systems with enough high-quality images that include all the parts of a dome are classified as “full” and the cases with limited photos and narrow view-angle with low-quality lighting conditions are stated as “partial”. While full cases provide full models with all the elements and ideal scenes, partial cases return partial models that are challenging to process after the photogrammetric modeling step. Two groups of target models are listed in Table 2.1. Due to varying conditions, each set of images of the domes system was collected with specific settings (Table 2.2).

All the samples are created assuming that the user would collect data under a dome and capture a wide range of images. Therefore, photogrammetric point cloud models are sampled from a surface representation of the part of the dome system. While some studies take the thickness of the architectural elements into account (Milani & Cecchi, 2013) this research dismisses this information.

Table 2.1 : Full and partial real-world models.

Full Models	Partial Models
Konya, Ince Minareli Madrasah, Main – IM_M	Konya, Alaaddin Keykubat Mosque - AL
Konya, Karatay Madrasah, Main - KM_M	Konya, Sahip Ata, Tomb Chamber – SA_T
Konya, Sahip Ata, Hanikah - SA_H	Konya, Karatay Madrasah, South - KM_S
Kayseri, Hacı Kılıç Mosque- HK	Konya, Abdulaziz Masjid – AB_M
	Konya, Akşehir, Gündük Minareli Mosque – AK_GUD
	Konya, Karatay Madrasah, North - KM_N
	Konya, Akşehir, Great Mosque- AK_ULU

Table 2.2 : The setting of captured images for photogrammetry. 18 MP and 24 MP stand for 5198x3462 and 6019 and 4023 consecutively. S6000, C600D, and C650D denote Sony Alpha ILCE-6000, Canon 600D, and Canon 650D in order.

Dome System	Camera	N.	Res (MP)	Raw	Tripod	Shutter (sec)	F Dist. (mm)	Focal (F/mm)	ISO
SA_H	S-6000	136	24	Yes	Yes	0.62	20	8	125
KM_M	C600D	120	18	Yes	Yes	8	18	8	100
IM_M	C600D	123	18	Yes	Yes	1.6	18	5	100
HK	C600D	162	18	Yes	Yes	8	18	7.1	200
AK_ULU	C600D	68	18	Yes	Yes	4	18	5	400
KM_S	C650D	109	18	Yes	Yes	8	18	8	100
SA_T	C650D	85	18	Yes	Yes	10	18	8	100
AB_M	C600D	72	18	Yes	Yes	2.5	18	5	200
AK_GUD	C600D	63	18	Yes	Yes	4	18	7.1	100
KM_N	C600D	75	18	Yes	Yes	6	18	7.1	100
AL	C600D	160	18	Yes	Yes	0.8	18	2	200

Subsequently, the collected images are processed and exported as point clouds. In this research point clouds of the real-world examples are created using the 2021.01 version of the Meshroom software (Griwodz et al., 2021), and Metashape Agisoft (2022). The photogrammetric models of Konya, Sahip Ata Tomb Chamber (SA_T), Konya, Karatay Madrasah South (KM_S), and North (KM_N) dome cases were prepared by fellow researcher Berkay Öztürk.

Subsequently, photogrammetric models should be processed before using them in examinations to make data computable and standardize the solutions. Initially, photogrammetry software returns point clouds with millions of points. To be able to select the region of interest that includes only the points of dome systems the elements that are not included in the segmentation process are cropped out. Next, the point clouds are simplified by lessening the number of points while preserving a sufficient amount of points for plausible labeling. This step gives optimal points for using the data in the labeling step. Table 2.3 shows the simplification in the number of points in the models after cleaning. Then, parts of the dome systems are labeled by splitting data into chunks regarding parts of a dome system in a scene. Subsequently, split data batches are imported into Rhinoceros, Grasshopper environment and subsampled fitting each model into ten thousand points considering the ratio of the number of

diverse parts to standardize the results. The labeled points are numerically represented in a new layer of information using a number from zero to three. For visual surveys, the dome, transition elements, arch, and wall classes are represented with blue, green, red, and yellow in order (Figures 2.2, 2.3, and 2.4). In the final data set, the number of dome points is dominant compared to the points in other classes (Figures 2.5 and 2.6).

Considering that photogrammetry software requires both a powerful central processing unit (CPU) with high multithreading capabilities and well-built graphics processing units (GPU), the images are processed in (1) High-Performance Computing (HPC) servers of Istanbul Technical University with 40 threaded CPU, 375GB of RAM, and four of 32 GB Nvidia V100 GPUs, (2) Google Cloud Platform with 8 threaded CPU 8GB of RAM and, one 16GB Nvidia T4 GPU. Computing resources of HPC servers were provided by the National Center for High Performance Computing of Turkey (UHeM) under grant number 1010152021 for modeling IM_M, HK, SA_H, and KM_M cases.

Table 2.3 : The number of the points in raw and clean points.

Phase	IM_M	KM_M	SA_H	HK	AK_ULU	KM_S
Raw	25.338.362	88.142.357	35.000.000	82.650.672	2.093.501	9.986.189
Clean	577.321	22.685	295.777	50.000	16.916	51.410
	SA_T	AB_M	AK_GUD	KM_N	AL	
Raw	2.783.266	5.000.000	16.515.640	13.097.578	30.771.427	
Clean	709.533	2.738.034	30.129	52.907	36.488	

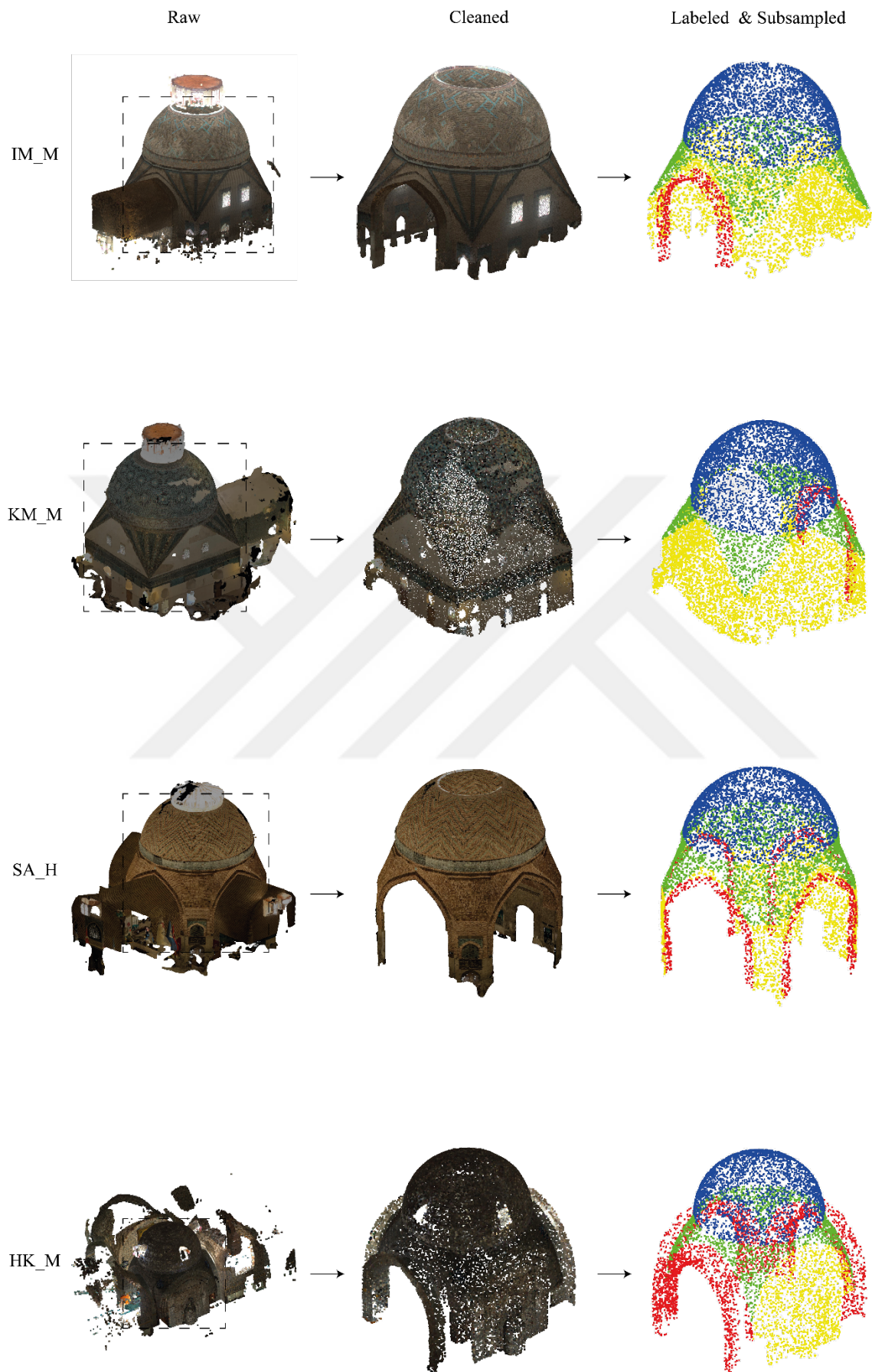


Figure 2.2 : Photogrammetric model (full models) cleaning and labeling.

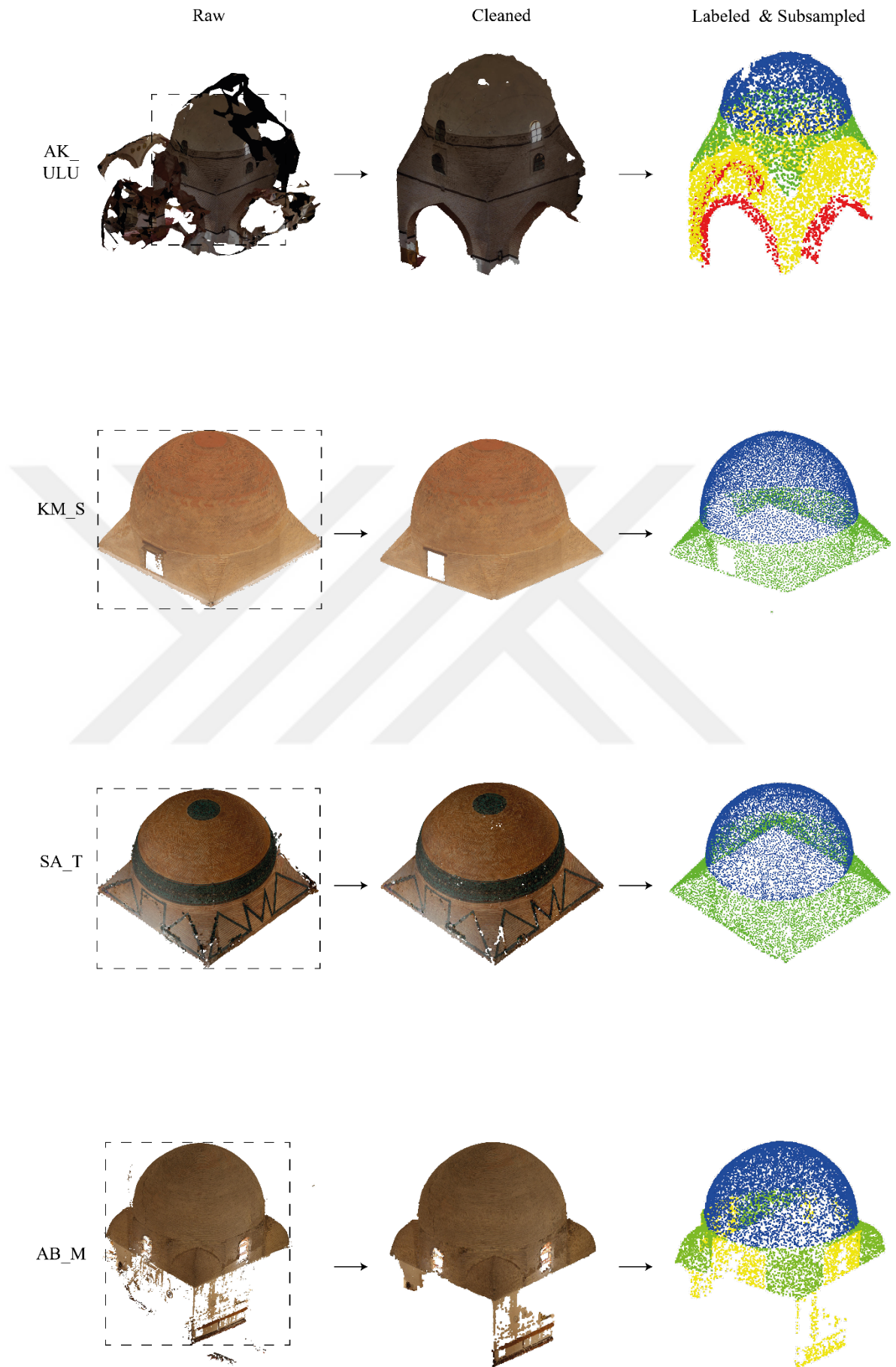


Figure 2.3 : Photogrammetric model (partial models-1) cleaning and labeling.

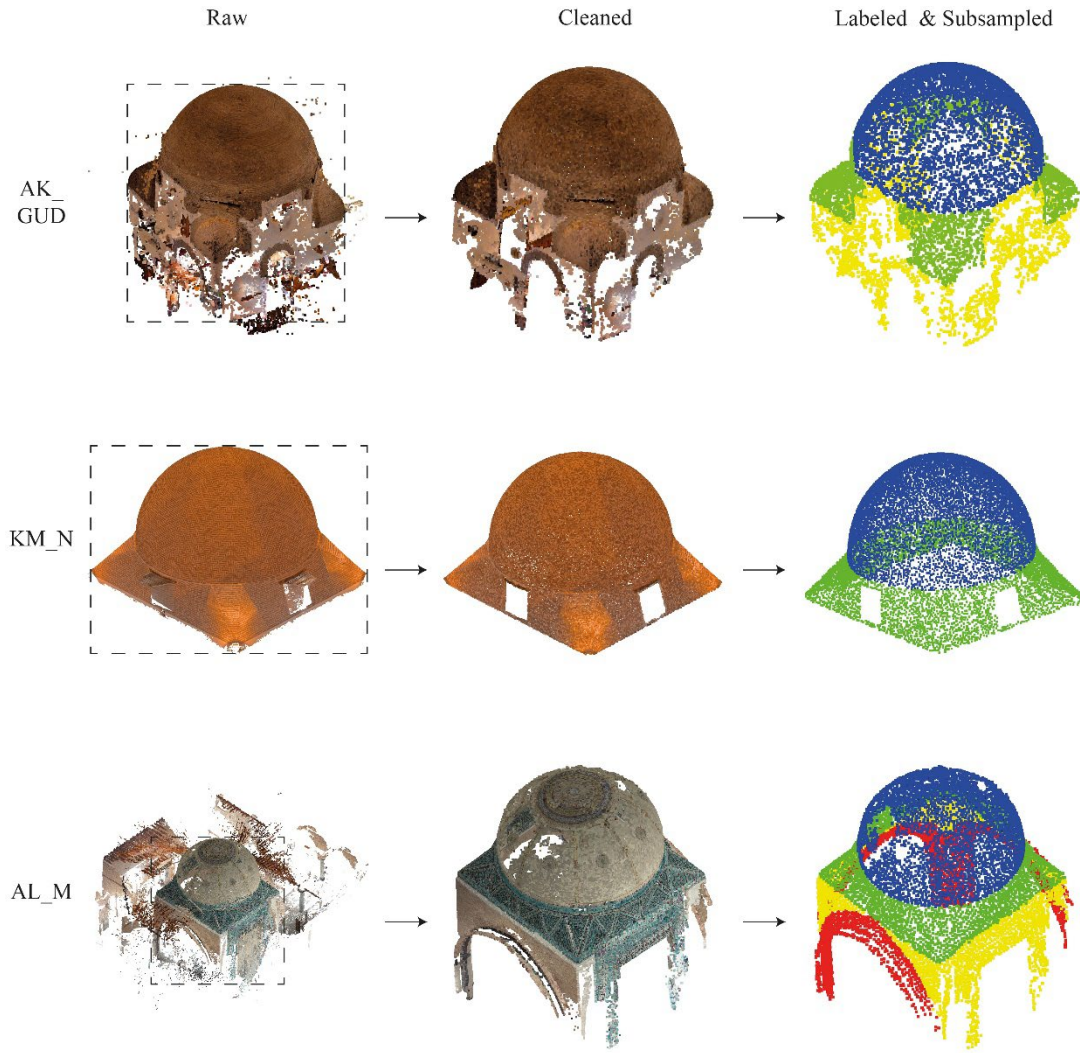


Figure 2.4 : Photogrammetric model (partial models-2) cleaning and labeling.

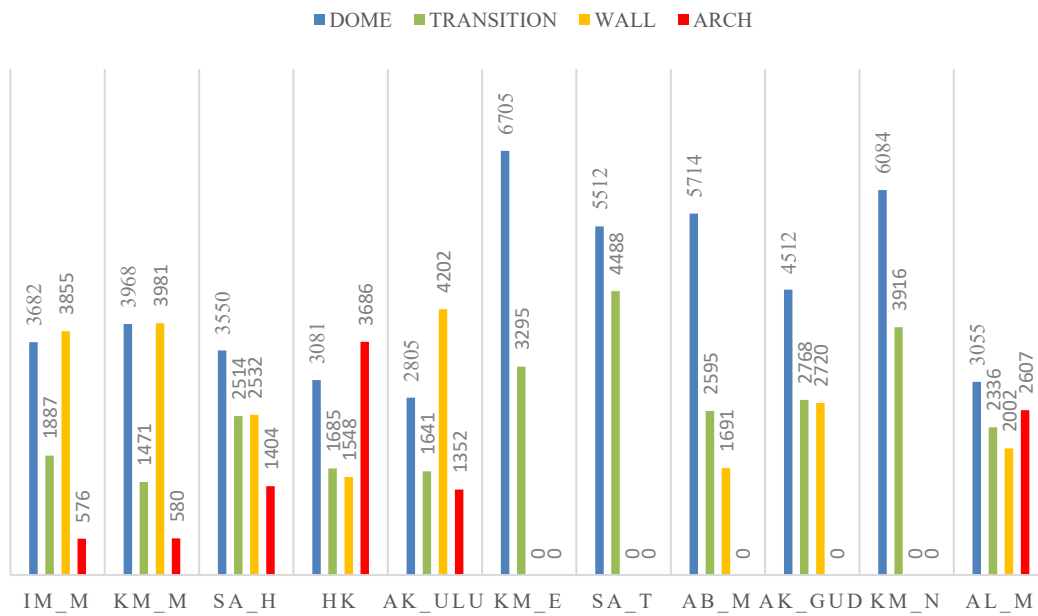


Figure 2.5 : Label of distribution of real-world point clouds per model.

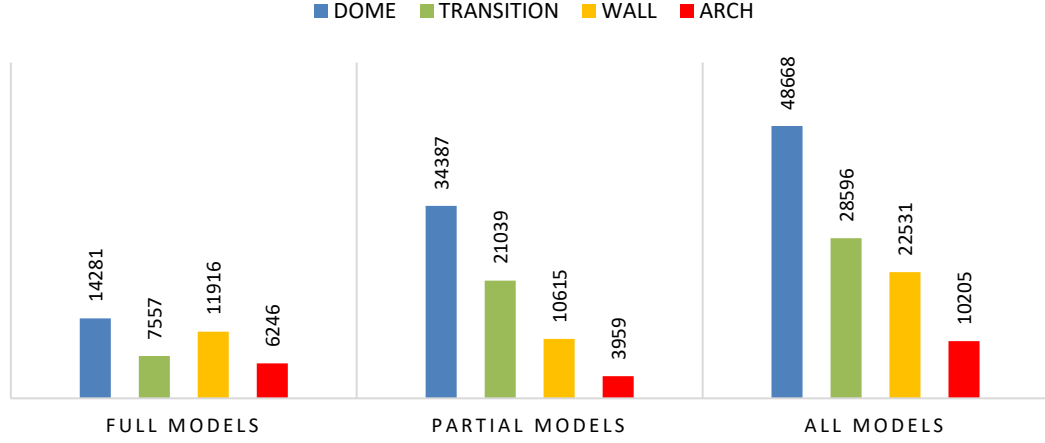


Figure 2.6 : Label of distribution of real-world point clouds per model type.

2.5 Synthetic Historic Dome Data Set

Historic Dome Data Set (HDD) is a collection of labeled point cloud models of dome systems synthetically generated using parametric definitions. An investigation includes data generation and evaluation steps presented and published as a part of CAADFutures 2021 (Güneş et al., 2021). The text of section 2.5 summarizes related content from the paper co-authored by a team of architects and computer engineers working on the semantic segmentation problem of historical dome models using a synthetic data set. Visuals used in this section are either directly implemented from that paper or adapted with limited modifications, all credit to the author of the thesis. HDD, my main contribution to the paper, defines a parametric generative algorithm respecting the simplified inspections of monumental Anatolian domes. HDD uses parametric modeling methods that target to generate sufficient and accurate dome models that can be used in deep learning algorithms for training semantic segmentation algorithms.

For assessing the success of the implementation of HDD in the semantic segmentation process, fellow researchers Yusuf H. Şahin and Alican Mertan completed the initial deep learning training and tests under the supervision of Prof. Dr. Gözde Ünal. Alican Mertan generated labeled point clouds from the labeled mesh geometries that were acquired from the dome system synthesizer algorithm. Both Alican Mertan and Yusuf H. Şahin conducted the training of deep neural networks, labeling real-world models, and testing in both real-world and synthetic data sets. They used PointNet (Qi et al., 2017a), DGCNN (Wang et al., 2019), and ODFNet (Sahin et al., 2022) as deep learning

algorithms and reviewed the results using a confusion matrix, accuracy, and mIoU as evaluation metrics. These are further explained in Section 2.8.1 and 2.8.2. The version of HDD used in the paper lacked definitions of a lantern opening on top of a dome and blind arch. Moreover, the networks are tested with a blind arch of Sahip Ata Hanikah and its surroundings and the main dome of the Ince Minareli Madrasah. The thesis uses this prior HDD definition and deep learning methodologies as the base and uses them with a revised set of data to adapt the workflow into the other data interpretation approaches that are explained in the following sections.

2.5.1 The dome systems in monumental Anatolian architecture

Historical Anatolian domes from 13-15th centuries are recorded and analyzed in multiple studies that focus on transition elements and dome geometries of the buildings. Okçuoğlu (1995) investigates the transition elements of Anatolian Seljuk praying rooms while providing images and simple floor plans supported by key dimensions of the structures. Also, restoration projects focusing on the Anatolian Seljuk town prayer rooms (Baş, 2008) expose dimensions, proportions, and the relations of the elements of different dome systems. Additionally, an inspection of praying rooms in the 13th century of Anatolian Seljuk architecture (Turan, 2018) represented dimensions in both sections and plans while providing a schematic representation of the part of the dome systems. These simple dome system representations without thickness are used as references to define basic geometries. Moreover, transition elements of the early Ottoman period (Şimşek, 2010; Özcan, 2008) provide essential details of the Anatolian dome systems.

The Historic Dome Data Set is created respecting the guidance of the limited resources aforementioned. Yet the detail level or representation of the existing information can be improved by professional documenters or additional resources for more accurate and specific definitions. The current state of the HDD expresses the simple representations of different variations and combinations of the parts of the historical Anatolian dome systems that use square floor plans.

2.5.2 The Level of Detail

Objects can be represented in multiple levels of detail (LOD). Specifically, in the architecture engineering construction (AEC) field, models and drawings at different scales bring different types of information related to particular tasks. According to

BIMForum (2021) preserving all the geometrical and non-geometrical information, and colors on a detailed model are named LOD 500 while expressing only conceptual representation with volume, position, and direction excluding color and other additional details defines as LOD 100. The parametric model in the research is defined in LOD 200 involving approximate geometrical features and labels.

2.5.3 The parameterization of the parts of the dome systems

In the data set, some of the main types of architectural elements of the mentioned classes are represented. For example, muqarnas ornaments are excluded as transition elements due to their complex and unique characteristics. The elements such as windows, doors, and lanterns are discarded because of the indiscriminate definitions of placing them in the structural dome system. Thus there is an intentional gap between the real-world data and the synthetic data set.

The HDD synthesizer has probabilistic attributes to define the dimensions of the architectural parts. Considering the overfitting problem in deep neural network training which prevents learning in the training process we need a large number of diversified samples to train. For this reason, even if the created model does not directly fit into the real-world examples the algorithm adds to or subtracts small values from some of the dimensions such as “height, width, thickness; flatness ratio of arches”, and the “flatness ratio of the dome”, “height of the drum” if it exists. In the same way, the algorithm combines elements randomly. Instead of generating all of the possible combinations the HDD synthesizer targets to generate unseen combinations with different parameters. To illustrate in the process variations of “with drum or direct” transition, “lantern or closed” dome, pointed, semi-sphere or flat dome; type of arches combined randomly. Yet, since the instance of some examples such as blind arches is rarely seen in real-world examples, in HDD the possibility of sampling these elements is reduced intentionally.

2.5.3.1 The dome

The main shape of the dome is defined in three ways: semispherical, flat, and pointed domes (Figure 2.7). The cutting plane of a sphere determines the radius and the height of the dome for flat and semispherical cases. The focal point of the sphere and the radius are defined by the flatness ratio (f) while the pointedness ratio (p) parameterizes the sharpness of the dome shape (equation 2.1). Additionally, respecting the examples

such as Ince Minareli Madrasah, Sahip Ata Hanikah, and Karatay Madrasah with lantern gaps on the top of the dome, the lantern option is adapted in the algorithm. Lantern cuts the top of the dome geometry by preserving approximately 2/3 of the dome curve that defines the dome boundary of the sphere-like shape (Figure 2.8).

	$r_{base} (cm)$	f	$r_{sphere} (cm)$	
Full Dome	$w_{wall}/2$	$f = 0$	r_{base}	
Flat Dome	$w_{wall}/2$	$0 < f < 1$	$\frac{r_{base} (1 + f^2)}{2f}$	(2.1)
		p		
Pointed Dome	$w_{wall}/2$	$0.5 < p < 1$	$2 \cdot r_{base} \cdot p$	

The figure displays three types of dome geometries: Full Dome, Flat Dome, and Pointed Dome. Each type is illustrated with a 3D perspective view, a cross-section, and a plan view. The 3D views show the dome's base and its relationship to the supporting structure. The cross-sections show the dome's profile, with labels for the base radius (r_{base}) and the sphere radius (r_{sphere}). The plan views show the dome's footprint, with labels for the base radius (r_{base}) and the parameter p .

Figure 2.7 : Full (top), flat/segmental (middle) and pointed domes (bottom). Revised from (Güneş et al., 2021).

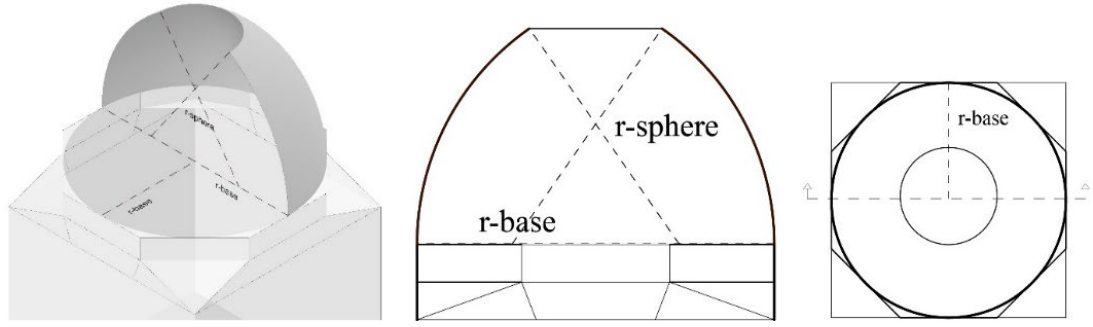


Figure 2.8 : Dome with a lantern.

2.5.3.2 The transition element

The square-shaped floor plan transformed into circular dome geometries by the distinctive elements in monumental Anatolian architecture. This research investigates the transition elements aforementioned in section 2.3.

The dome systems involve drum geometries along with the transition elements. A drum provides secondary transition and is geometrically distinguishable from a dome geometry. A drum geometry follows the polygonal base defined by transition elements and the exceptional circular base created by the pendentive transition zone in the context of this thesis. Additionally, a circular drum geometry is accepted as a part of a dome geometry because they share similar geometric features such as curvature and the model that this research proposes does not use any color value to define or distinguish a dome zone that shares similar properties with a drum. Okçuoğlu (1995, p.18) distinguished domes into two: octagonal and circular; and stated that circular domes are commonly used in Great Seljuk, Anatolian Seljuk, and Iranian buildings for strengthening a base of a dome. Özcan (2008, p. 69) reviewed the use of drums in early Ottoman and stated that a dome of a building might have thicker base walls to provide structural stability and visual transition which resembles a drum. These parts can be seen as both an independent drum and a part of structurally optimized dome geometry. Also, Turan and Yaldız (2018) state that drum geometry is an additional architectural element that connects a dome and elements below it. Additionally, it should be noted that some variations including drums might be contradicting a specific period of design and the HDD definition does not include constraints for offering generic solutions. For targeting a particular architectural style, the HDD definition can be updated and combinations and number of polygons can be limited in further studies. In the data set, simple squinch, complex squinch, corner triangle (planar triangle), Turkish triangle

(planar triangle), pendentive, planar triangular band, simple prismatic triangular band, and the complex prismatic triangular band are modeled as a part of a general transition element class. The form of the transition zone is defined by the parameters such as the height of the transition part, the edge number of the drum polygon, and the width of the wall (Figure 2.9). The common parameters are defined in equation 2.2.

$$\alpha = \frac{360}{n}, \quad n \in \{8, 16, 32, 48\}, \quad 0 \leq \text{height}_{\text{drum}} \leq 150, \quad \text{height}_{\text{drum}} \in \mathbb{R} \quad (2.2)$$

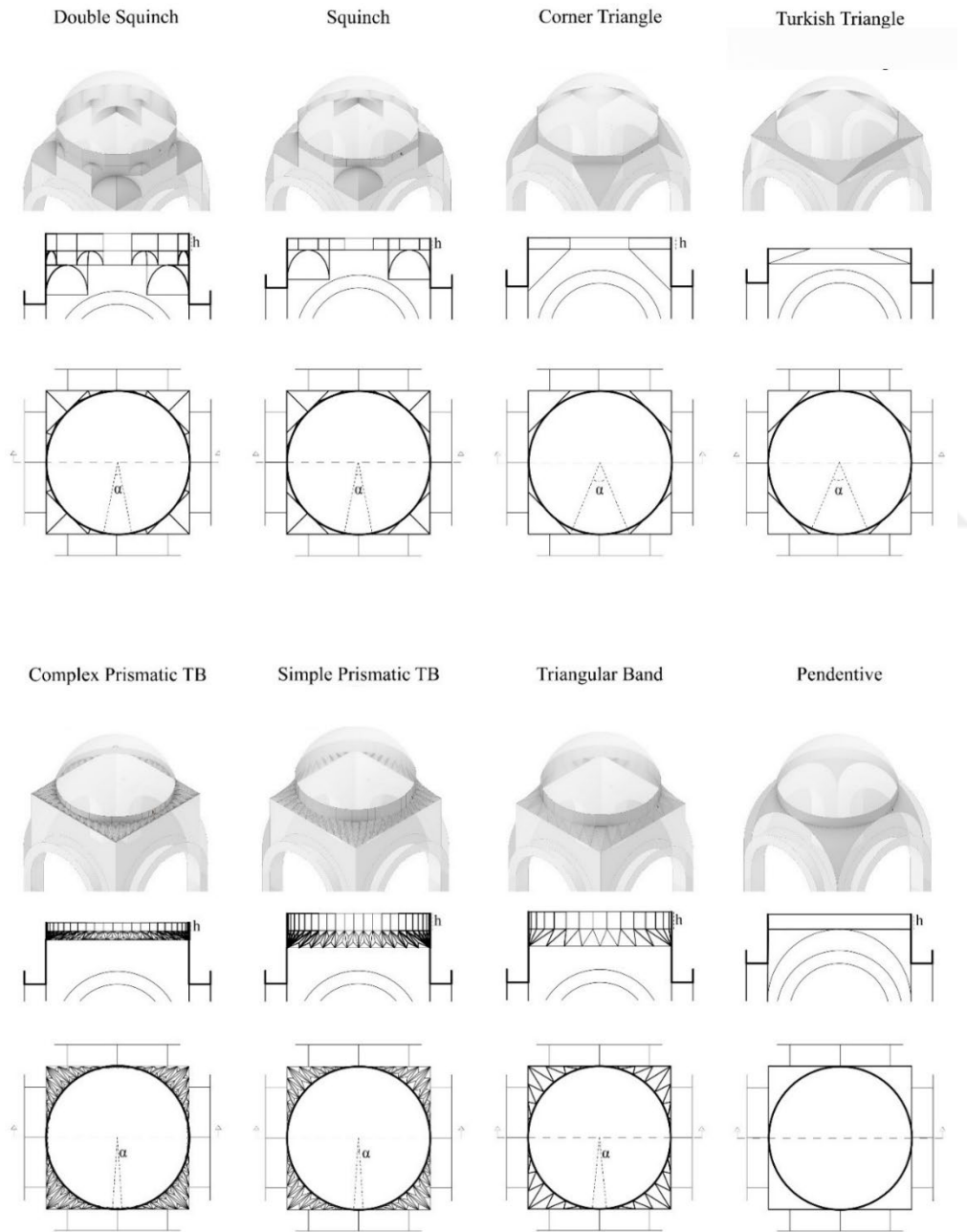


Figure 2.9 : The edge number of the polygons and the drum height form the transition elements (Güneş et al., 2021).

2.5.3.3 The supporting wall

Plane walls, arches, or combinations of both elements can be placed on each wall face (Güneş et al., 2021). The arrangement of the supporting elements defined by height and width values generates several variations (Figure 2.10).

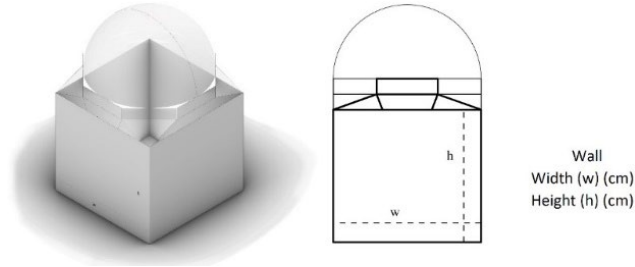


Figure 2.10 : Height and width variables define the wall (Güneş et al., 2021).

2.5.3.4 The supporting arch

Arches have a semi-circular, drop, flat, single-centered tangent, or double-centered tangent variations. Figure 2.11 and Table 2.4 show the created arch samples using the length between the wall edge and the columns, dept, thickness, radius, and height properties (equation 2.3).

Table 2.4 : Parameters of the supporting arch.

Parameter	Parameter ²
d = depth of the arch (cm)	W = inner width (cm)
o = offset from the wall (cm)	r = radius of a circle
t = thickness of the arch (cm)	c = center of an arch (cm)
h_1 = height of the linear part (cm)	α = angle
h_2 = height of the curved part (cm)	f = flatness ratio
H = total height of arch (cm)	

$$50 \leq d \leq 150, d \in \mathbb{R}$$

$$0 \leq o, o \in \mathbb{R}$$

$$20 \leq t \leq 70, t \in \mathbb{R}$$

$$50 \leq h_1, h_1 \in \mathbb{R}$$

$$0 \leq h_2, h_2 \in \mathbb{R}$$

$$H = h_1 + h_2 + t, H \in \mathbb{R}$$

$$(2.3)$$

$$W = W_{wall} - 2(o + t), W_{wall} \in \mathbb{R}$$

$$r \in \mathbb{R}$$

$$c_{x,y,z} \in \mathbb{R}$$

$$0 < \alpha < 90, \alpha \in \mathbb{R}$$

$$f \in \mathbb{R}$$

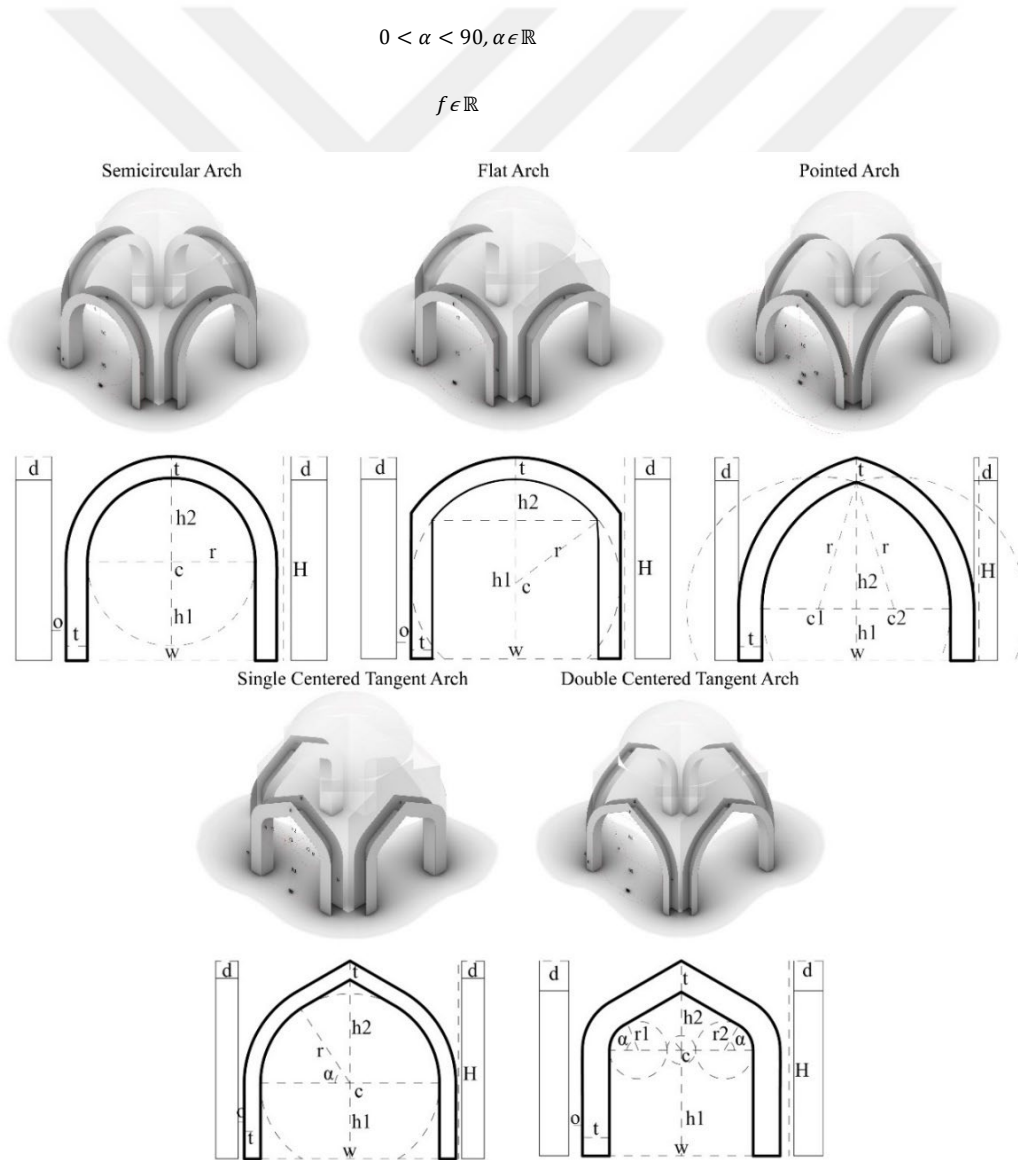


Figure 2.11 : Arch types use various parameters (Güneş et al., 2021).

2.5.4 Combining parts of the dome systems

The shapes of the dome models are generated individually as NURBS geometries (Güneş et al.,2021). Then the algorithm splits the walls using the transition element geometries and then excludes the unnecessary parts. The following function checks the existence of an arch for each wall. In the cases with arches, the wall is split and the arch geometry is fitted inside of it. For the blind arch cases, the method places a wall section through the exterior part of the arches. Next, depending on the presence of a drum it is combined with the transition elements, and the bottom level of the dome geometry is set as the top level of the vertical drum geometry. The dome model generator synthesizes all the elements by sharing the same base rectangular shape and unique parameters for their specifications.

2.5.5 User-friendly semi-automatic point cloud exportation

All four parts of the dome system are defined by interconnected parameters as a part of parametric or procedural modeling. Firstly, they are created as NURBS models as surfaces. Then, the algorithm samples 10.000 points in total per model. To label, the part red color value is added to the coordinate's information. The script has the capability of exporting mesh models that can be used by different algorithms or subsequent sampling steps. However, since it is possible to export information of points that can directly be processed; the script exports models as point clouds in the format "X position" "Y position" "Z Position" and "Red value" in ".txt" format. Volvox plug-in in Grasshopper is used for manipulating the data and for the exportation steps.

The data set synthesizer uses visual programming functions in Grasshopper that are integrated into Rhinoceros as well as the C# and Python scripts to automate the process as much as possible. While C# methods are responsible for a number defined number of collections that defines each choice of the element in the exportation batch. In the creation of this data set, each time the script ran 192 models were created. Moreover, another node of a C# script is set for optional mesh exportation of the created models, though in the use case of this research direct exportation of point clouds satisfy demands. In addition to the generation and exportation of geometries, the algorithm saves screenshots of the models in a specific angle with a parallel view to follow the synthesizing process in case any problem happens in the workflow and visually

comprehend the overall results. In the exported data set ratio of the wall label is the dominant class (Figure 2.12).

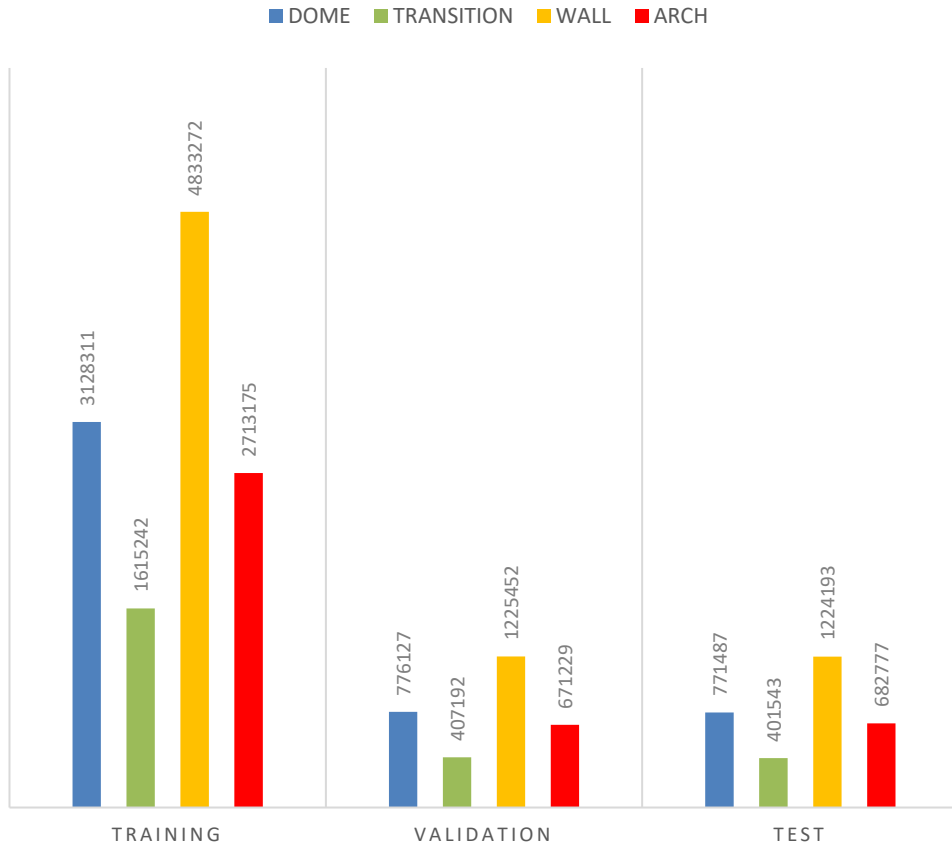


Figure 2.12 : Label distribution of real-world point clouds per model type.

2.6 Data representation schemas

Photogrammetric or LiDAR collected data can be represented in many different kinds of data types and structures. For example, the order of the data columns, keeping column header, separating characters of the columns, decimal length, and any other detail can be different in different types of data acquisition methods. Even though this diversity in point cloud representation does not cause any conflict in the design and documentation process, deep learning algorithms require standard types of data in a specific format.

To quantitatively evaluate the deep neural networks, some benchmarks with extensive datasets are used. Especially in semantic segmentation tasks of point clouds, ShapeNet (Chang et al., 2015), PartNet (Mo et al., 2019), and S3DIS (Armeni et al., 2016), are some of the most common benchmarks that include non-temporal models.

ShapeNet possesses several sub-collections serving different purposes. Instead of including different types of objects in a model, it involves the main classes of the models as well as the labels of the parts. While ShapeNet offers one hierarchical level of data structure in part-whole relations PartNet intends to augment the grade of the detail by expressing all the sub-labels of the parts in relation to parent units. Since these datasets provide isolated models instead of combining different types of objects in complex scenes, they are called part segmentation. Technically, the computation process is almost the same as the semantic segmentation process but there is a notional difference.

S3DIS is an indoor scenes dataset that includes architectural elements and furniture as main labels. Considering that a data cluster is created to be used in identifying different architectural objects in complex scenes it is defined as a large-scale indoor space dataset.

These are several datasets that are used on different scales and for different purposes and each dataset has its way of data representation. But this study follows the data structure and representation methods of ShapeNet (Figure 2.13) and S3DIS (Figure 2.14) for the ease of implementation into deep learning models. The dome systems in the HDD ontologically are represented as a part of a whole; each element is linked and shapes each other. On the other hand, elements like a dome, arch, wall, and transition elements are still distinguishable by themselves. Thus this study accepts HDD as a part of both part segmentation and semantic segmentation regarding paying sufficient attention to contextual relations of the architectural elements.

The ShapeNet representation of the Historic Dome Data Set specifically refers to ShapeNetCoreV0. Before exporting the point clouds the coordinate points are normalized between -1 to 1 on each axis. This process can be thought of as fitting the model inside a unit sphere to keep values computable for deep learning algorithms.

Additionally, the Point-BERT network uses a dataset named “ShapeNet55” for pre-training and fine-tuning the bidirectional encoder representation networks (Figure 2.15). Apart from the previous data representation structures, this dataset includes positions of the points and labels of the models without preserving the classes of the parts. Thus the historic dome data set was also prepared following the same data structure excluding the labels of the points.

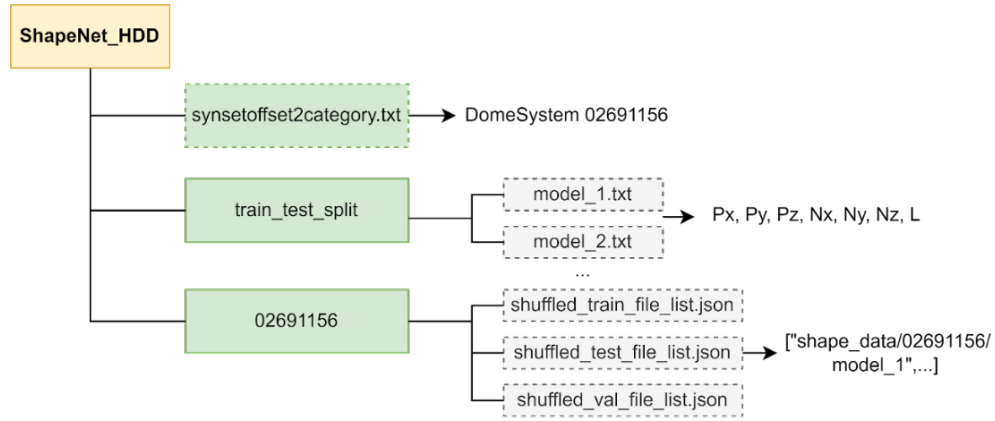


Figure 2.13 : The folder structure of the ShapeNet_HDD

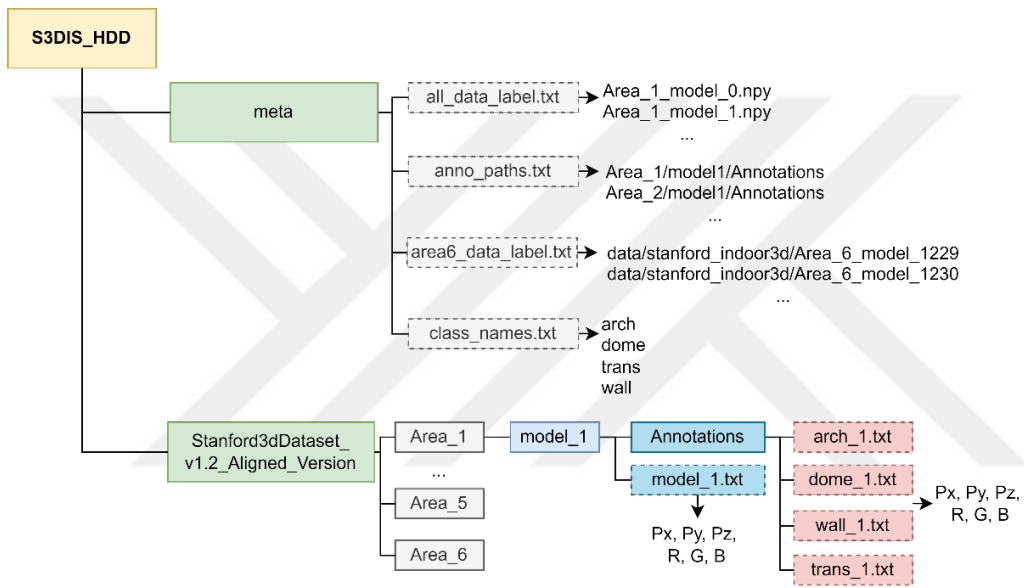


Figure 2.14 : The folder structure of the S3DIS_HDD

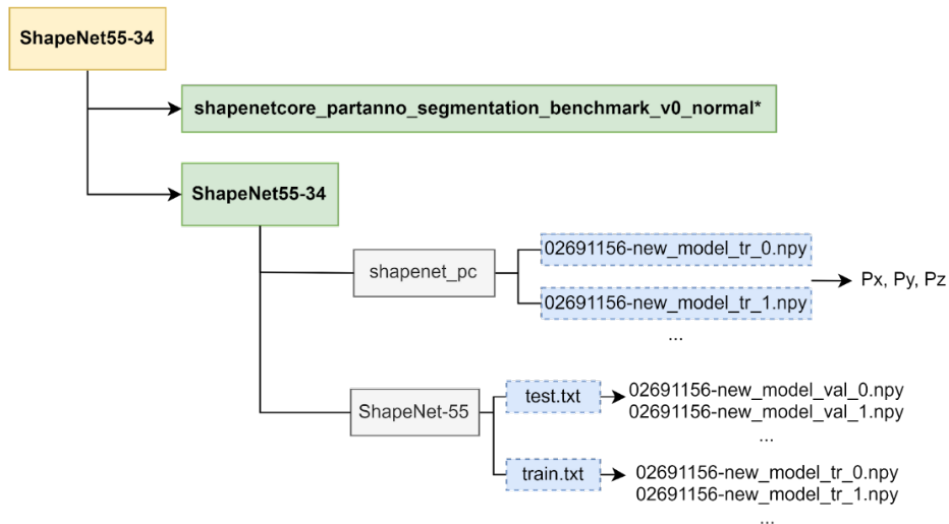


Figure 2.15 : The folder structure of the ShapeNet55-34_HDD

2.7 Data Processing

Collected photogrammetric data is usually unstructured, unordered, unscaled, and scattered. For simple visualization tasks there it does cause any problem but keeping deep learning computations incalculable scale point cloud data needs to be standardized numerically. Firstly, we centralize our models in a couple of steps. Initially, the algorithm finds maximum and minimum values of points on the x, y, and z-axis. Then the code subtracts the average values that are calculated by using the extreme points in each axis from all the points to fit the model center in the $\langle 0,0,0 \rangle$ coordinate. Even though negative and positive values in each axis are distributed evenly, in most cases, deep learning algorithms require coordinate values of the point cloud models to be fitted in between the -1 and 1 to prevent vanishing and exploding gradient values. This method is also known as normalization which subtracts the minimum value in each axis from points and then divides them by the difference between maximum and minimum values. On the other hand, some networks use shifted data that use 0 as the minimum value without mapping the interval. The diagram shows the pre-processing options and steps (Figure 2.16). These modifications are applied regardless of the data preparation schema. To make the workflow simple and robust, the Pandas and SciPy libraries are used to define and modify the data sets.

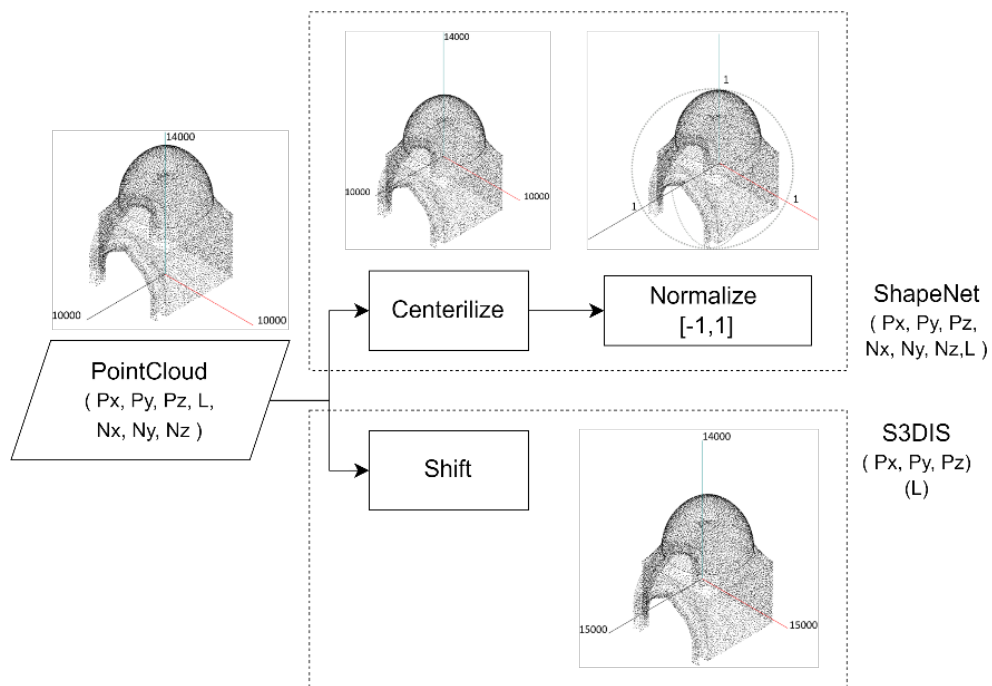


Figure 2.16 : Data preprocessing.

2.8 Deep Learning

Machine learning algorithms are widely used in different fields including architectural design and documentation tasks in cultural heritage. Both two-dimensional and three-dimensional algorithms provide a wide range of solutions in terms of documentation and extracting meaningful information from the geometric data that is collected. This section starts with a review of two-dimensional computer vision networks. Then, it continues advancements in three-dimensional vision methods. Next, it discusses implementations of both of the methods in a cultural heritage context. Lastly, the impacts of the three different approaches in the three-dimensional point cloud semantic segmentation process are detailly explained considering the segmentation problem of historic domes.

Image recognition and detection of the particular elements in the visuals have been a challenging problem for decades. Initially, LeNet (LeCun et al., 1998) proposed the concept of a deep neural network that recognizes low-resolution number images using the convolutional neural networks (CNN) with 7 Layers. The network identifies the characteristics of the handwritten number images by extracting their features of them. However, there has not been a great leap for a long time in terms of usage of image recognition based on deep neural networks due to the limited power of computer hardware, lack the large datasets, and naive algorithms that are not successful at generalizing the learned information.

Remarkably, two-dimensional computer vision algorithms are extremely popular after the successful implementations of deeper networks with immense data sets. Starting from the AlexNet (Krizhevsky et al., 2012) the image classification, semantic segmentation, and detection algorithms got advanced immensely due to the availability of powerful hardware, crowded datasets, and efficient and well-designed algorithms. The usage of convolutional neural networks that can extract meaningful information from the regions of images; augmenting the data by using simple transformations such as rotation, translation mirroring; dropping out some calculations on the fly, and using overlapping pooling layers to generalize the predictions revolutionized the deep learning algorithms. AlexNet acquired 84.6% in the ImageNet classification dataset (Deng et al., 2009) using 5 convolutional layers and 3 fully connected layers. Subsequently, VGG network achieved 92.7% accuracy in the ImageNet by

interpreting data with smaller-sized receptive fields within more layers (Simonyan & Zisserman, 2014). Then, GoogLeNet improved the results on ImageNet to 93.3% by using the consecutive inception blocks that apply multiple convolutions in different sizes a pooling and layer, and concatenate the results to extract more features in each layer (Szegedy et al, 2015). Instead of choosing one convolution setting this network provides multiple computations and uses all of them together. Subsequently, ResNet (He et al., 2016) achieved 96.4% thanks to a deep structure with 152 layers and skip connections that prevent losing meaning in the process. Consequently, these advancements in the usage of convolutional neural networks, availability of large datasets, and powerful computer units allowed a great leap in classification tasks using deep neural networks. However, these solutions are created for images that have a well-defined grid-like structure of pixels, hence, direct implementation of the convolutional networks into three-dimensional point cloud data is challenging due to complex relations and uneven distributions of the points.

Extracting additional information in a given image such as an area or pixels of a certain class is another demanding task in computer vision. Semantic segmentation algorithms that identify the classes of the pixel of item-wise and object detection networks that define a boundary for elements in the images and predict their label evolved and got used for many different purposes (Mo et al., 2022). To illustrate, these methods have contributed to many tasks such as medical semantic segmentation (Asgari et al., 2021), documentation in the cultural heritage field (Pellis et al., 2022), and remote sensing (Wang et al., 2020).

Classifying the parts of the images in a rectangular area helps to identify the different and multiple objects in an image. For example, one of the pioneer region-based object detection studies, R-CNN (Girshick et al., 2014) defines possible regions of interest (RoI) to be classified in further steps. Then it warps images for processing them in CNN layers to predict the label of the selected rectangular area. It is a simple idea of extracting the sub-items in the image context. However, the algorithm is remarkably slow and storage-dependent which limits the usage cases. The following research, Fast R-CNN (Girshick, 2015), addresses these problems by initially using convolution methods to extract the features and then identifying the ROIs. This approach permits the re-usage of the extracted features from the first convolutional layer by the various region proposals. Subsequently, Faster R-CNN (Ren et al., 2015) focused on sharing

more values in region proposal and label identification process to make the process even faster while offering the same kind of outputs (Huang et al., 2017). Thereafter several algorithms improved the quality of the process in the following research considering the best area proposal with high prediction accuracy using the least time and computational power (Zhao et al., 2019). In the documentation context the approximate area extraction approach would be meaningful in detecting the existence of particular shapes such as stones, brick, or regardless of the precision identification of the cracks and damages would be highly beneficial in the documentation process. Yet, working on two-dimensional data is limiting for merging the extracted information with models precisely.

Identifying the objects in the images and segmenting them pixel-by-pixel using the semantic information learned from the provided data set is another challenging task. Fully Convolutional Networks (Long et al., 2015) uses image classification algorithms as the backbone of the network and by combining the relevant information acquired in the earlier steps that can be used to define the label of the pixels. Similarly, U-Net (Ronneberger et al., 2015) encodes and decodes the relevant information to identify the classes of the images using the well-defined structure that concatenates the semantic information gathered through the process. Subsequently, Segnet (Badrinarayanan et al., 2017) uses a similar deep learning architecture that remembers the positions of the utilized pixels in the calculation layers and integrates this knowledge in the decoder section that upsamples the extracted knowledge and labels the data in fine-grained detail. Additionally, Pyramid Scene Parsing Network (Zhao et al., 2017) is particularly targeted at acquiring accurate contextual knowledge by using different sizes of convolutional layers after feature mapping to keep segmentation processes aware of the global relations in the scene. Moreover, DeepLabV3+ (Chen et al., 2018) aims to improve the overall segmentation quality by using both encoder-decoder architecture and the concept of spatial pyramid pooling while keeping global and local details. Benefitting the simplification and identification capabilities of the convolutional neural networks, semantic segmentation classifies the elements in the images pixel-wise by extracting the local information within a global context of the images. Learning the relations of the objects in the images is vital because semantic information can be meaningful only in a context hence processing local areas of the images while remembering the overall features of the images helps to extract

expressive materials from the images. Nonetheless, the usage of semantic segmentation in architectural cultural heritage documentation would be limited unless the documented object is planar because most of the items designed and used are related to each other in three-dimensional space intentionally.

In addition to semantically labeling the pixels, instance segmentation algorithms such as Mask-RCNN (He et al., 2017) provide outputs that identify each unique element in the given image individually. Extracting information about distinct objects even in the same class would improve the documentation processes in terms of evaluating the state of each element in the visuals. Because in especially historical buildings there are lots of elements that are repetitive. However, lacking the depth information would create a complex situation where the complex structures are documented.

The main advantage of 2D image processing is the ease of standardization regarding the matrix-like pixel structure of the visuals. This convenience helps convolutional neural networks to return robust and predictable results. Even though the analysis of two-dimensional images brought too many qualities to the different workflows, they lack the geometric representations and the features of the shapes in the three-dimensional space. Hence it is challenging to extract detailed architectural information when the collected data lacks depth information.

On the other hand, processing three-dimensional data is considered a bit complex and challenging due to lacking standardized and organized structure of the data (Zhang et al., 2019). While in two-dimensional image processing workflows pixels are the main units to be processed in three-dimensional algorithms, points that keep coordinates together with any other additional information such as color values and surface normal values (Wang et al., 2019). However, varying distances between the points and fluctuating density of the points in the three-dimensional space cause irregularities in the numerical representations and calculations (Elberník & Vosselman, 2011). Hence there is not a direct transfer of usage of deep neural networks designed with convolutional neural networks that are used to process images (Guo et al., 2020).

Unlike pixels in the photos, each point of a point cloud is connected to its neighbors in different forms. Hence regarding the number of neighbors, information of a point is calculated in compliance with its surroundings. Also, in semantic segmentation tasks, in addition to color information colors and the surface normals of the points can be

computed but they are unessential (Che et al., 2019). While in some cases only coordinates of the points would be enough to extract the features of the different architectural elements and identify the shapes, in the particular tasks of identifications of an item which has diverse colors with high contrast or the shapes looking in different directions as a part of a design color and surface normals would help to reveal the semantic relations of the points.

Similarly, many tasks in the cultural heritage and remote sensing domain concatenate definitive information such as planarity, linearity, surface variation, sphericity, verticality, anisotropy, omnivariance, and eigentropy that represent geometric qualities numerically (Blomley et al., 2014). To illustrate Weinmann et al. (2013) extracted three-dimensional features of the points to use them in the classification procedure. Teruggi et al. (2020) used extracted geometrical features in a random forest segmentation algorithm to distinguish the elements in the documentation of the elements of the historical Pomposa Abbey and Milan Cathedral. Moreover, in some cases, different color spaces are used to augment the robustness of the models. To illustrate, red, green, and blue (RGB) color space is commonly used in machine learning and image processing tasks (Griffiths & Boehm, 2019). However, hue, saturation, and values (HSV) might lead to more accurate results due to their resemblance to human understanding of colors (Pierdicca et al., 2019). Therefore, the depth of the networks and the concatenated information might differ. In the same way, data processing approaches of the deep networks are distinct.

Machine learning algorithms are used in several domains as well as in architecture and cultural heritage (Croce et al., 2021) thanks to the advantages they bring. As a specific form of machine learning, deep neural networks are also getting implemented in various workflows. It became excessively common, especially after the successful implementation of image classification with deeper networks, extensive datasets, and hardware with strong parallel computation capabilities.

Consequently, this research investigates different types of deep learning techniques that impact the recognition of architectural elements in specific contexts. In architecture and construction, the elements in different scales combine both in sequential order and contextual relations. For example, in the domes of the Anatolian Seljuk context, the transition elements are needed to provide the transformation from a rectangular base plan to circular dome geometry. The location of the transition

element and its relation with the other elements such as the wall, arch, and dome are well-defined in most conditions. Relying on the design principles each element has various shapes however since their architectural and structural usages are the same semantically, they are considered in the same class regardless of the differences in the shape. Additionally, the definition of the architectural elements is made to acquire the sub-elements of the dome systems that have different structural functions. Further studies can define their unique problems with different levels of detail while keeping the workflow the same.

The target of the presented study is to examine the results of different deep learning methods regarding the way of processing the input data. The networks that are used with Historic Dome Data Set are chosen intentionally for representing the deep neural networks that use multi-layer perceptron, graph, and sequential computation approaches. Multi-layer perceptron-based studies extract meaningful information for the points by using simple networks. Graph-based algorithms that investigate the neighborhood properties of the points. Sequence models that work similarly to natural language processing (NLP) algorithms that interpret the features of the points considering the consecutive links of the points and the group of points are detailly discussed in the following section.

2.8.1 Multi-layer perceptron

2.8.1.1 PointNet

Utilization of three-dimensional data for computer vision purposes such as classification and semantic segmentation became prevalent starting with the stimulating study, PointNet (Qi et al., 2017a). Point clouds require permutation invariance that makes any sequence of the collected data return the same results (Guo et al., 2020).

PointNet architecture provides a solution that is permutationally invariant by using a symmetrical method. Using the shared multi-layer perceptrons it learns the geometrical features of the points. Initially, it acquires a global feature for the given model and by concatenating global feature calculations with the results acquired in the middle of the process the network starts predicting the label of each point instead of guessing the class of the whole scene (Figure 2.17). PointNet has a simple structure that learns the generic features of the objects in models and segments them using the

global features. Nonetheless, it lacks the capability of learning local qualities with high resolution (Guo et al., 2020).

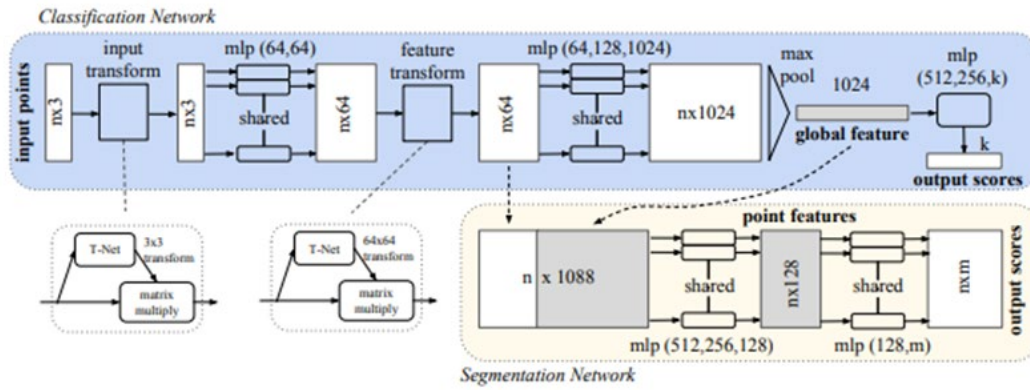


Figure 2.17 : The neural network structure of the PointNet architecture (Qi et al., 2017a, p.3).

2.8.1.2 PointNet++

Consequently, PointNet++ (Qi et al., 2017b) concentrated on filling the gaps in former PointNet architecture by proposing a hierarchical learning schema that obtains more local details in addition to a global context.

The process commences with set abstraction methods which acquire the local features. Initially, the method gets local sets of points, samples the centroids of the acquired sets then finds the neighbor points of the centroid. Following use a simple version of a PointNet in the local groups. To sample the points, the network uses Iterative Farthest Point Sampling (FPS) method that creates uniformly distributed point sets. The grouping method finds the K number of neighboring points around the center points of the local sets. Yet instead of finding all fixed K numbers of points around the given points the algorithm uses a ball query that limits the searching volume for getting more accurate local features. A ball query can be simply thought of as a sphere that separates the search area and excluded area. This limitation is meaningful in a sense of processing point clouds with sparse and non-standard density. To illustrate, in capturing steps some parts of the model might be missing and there might be unintentional gaps in the model. Additionally, it overcomes the merge of distant point sets that are part of different elements in the point cloud. Subsequently in the PointNet layer, local features are calculated using the relative coordinates, regarding the centroids of the subsets. The segmentation section of the method regenerates the fine-grained model commencing from the coarse results of the set abstraction functions by

interpolations and skip connections that concatenate the extracted features with larger size point sets. Finally, it predicts a label for each class (Figure 2.18).

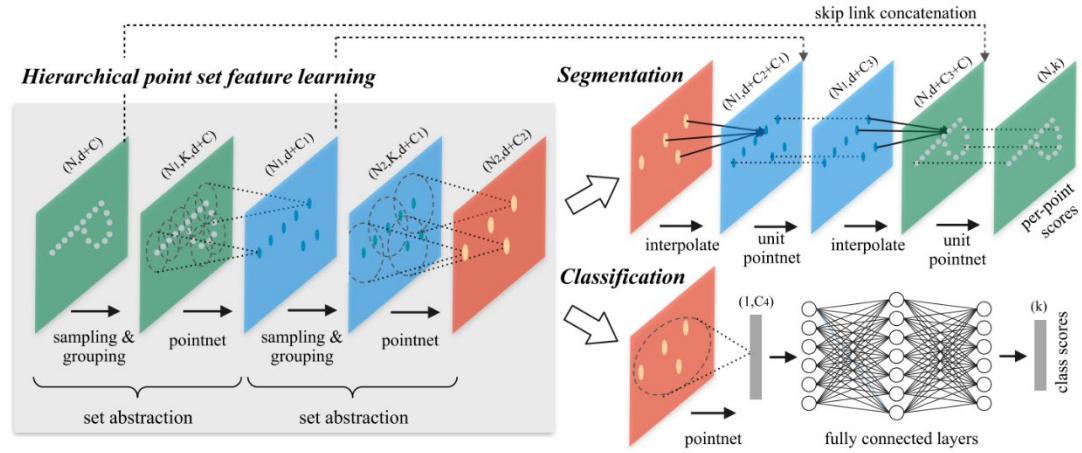


Figure 2.18 : A hierarchical point set feature learning schema (Qi et al., 2017b, p.3).

In the algorithm K (the number of the neighboring points) and the radius of the searching sphere are the hyperparameters that the user can fine-tune according to the problem and the provided dataset.

The varying density of the point clouds might impact the quality of the results. To overcome the problems that non-uniform data create, the algorithm proposes Multi-scale grouping (MSG) and multi-resolution grouping (MRG) methods. Since MRG is faster and more accurate the algorithm uses it (Figure 2.19). The network uses random input dropout to augment the data and generalize to predictions.

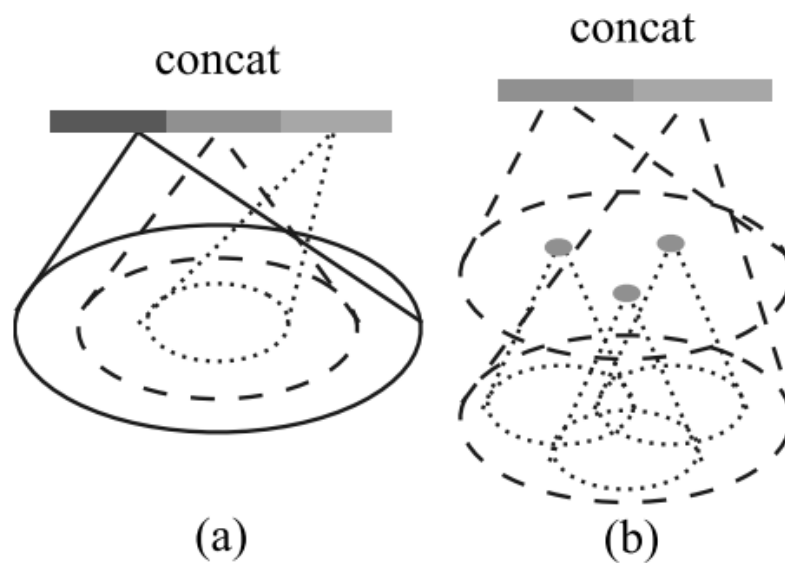


Figure 2.19 : A Multiresolution concatenation, A hierarchical point set feature learning schema (Qi et al., 2017b, p.4).

As a result, the investigated MLP based algorithms simplify the geometries by picking uniformly distributed points and their neighbors. Despite the fact that it learns global properties it is naive to learn the relations of the local regions. Thanks to the max pooling feature in the PointNet architecture it manages to process unstructured data (Duan et al., 2019). This research is examining two versions of PointNet++ architecture with models having 2048 points and 10000 points. The results of the models are discussed in the evaluation section.

2.8.2 Graph-based: Dynamic Graph CNN

Regardless of the domain of the model the point clouds contain interconnected sub-parts that are meaningful in a context. Even though learning the local and global features of the sub-point sets provides vital knowledge for predicting the labels, the relations of the points and the point sets also define the semantics of the models. Especially in the context of cultural heritage the parts of the buildings and the architectural elements are meaningful in a context where all the design decisions affect the related elements. Thus, in addition to examining the local and global features of the points, interpreting the graph-like relations of the points would improve the quality of the process in terms of acquiring the best solutions that take semantics into account. Graph-based semantic segmentation algorithms determine the features of the points by interpreting the relations of the points with their neighbors. This approach defines local and global context by evaluating the correlations of the points using the edge values between the points.

Dynamic Graph Convolutional Neural Networks (DGCNN) (Wang et al., 2019) assess the data by dynamically aggregating in the point sets. This deep learning architecture uses a method called EdgeConv that extracts meaningful local geometric features. The algorithm creates a graph for each point using the EdgeConv methods. EdgeConv uses the points as vertices and creates edges by connecting the vertices. For each vertex, the method finds the k number of nearest neighbors (k -NN) including itself (Figure 2.20). Then using the created graph, it extracts an edge feature. The Edge feature keeps the features for both local and global positions while preserving a permutation invariant structure of the data by pooling the values at the end of the EdgeConv method. Consequently, it shows that both graph and convolution ideas are implemented into the network to learn the semantic relations of the points better. Even if the related

points are not next to each other, the concept of EdgeConv is capable of identifying the link between them. Yet it loses the ability when the distance gets larger (Lee et al., 2021).

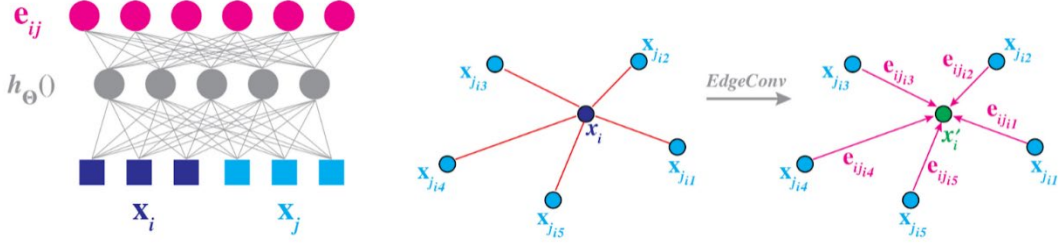


Figure 2.20 : Edge feature edge convolution (Wang et al., 2019, p.3).

The algorithm updates the edge values for each layer while running. DGCNN does not apply any sampling or shape reduction method so all the points are being processed starting from beginning to end (Figure 2.21). However, calculating the edge features on the fly and preserving the initial shape makes the calculations computationally expensive (Bello et al., 2020).

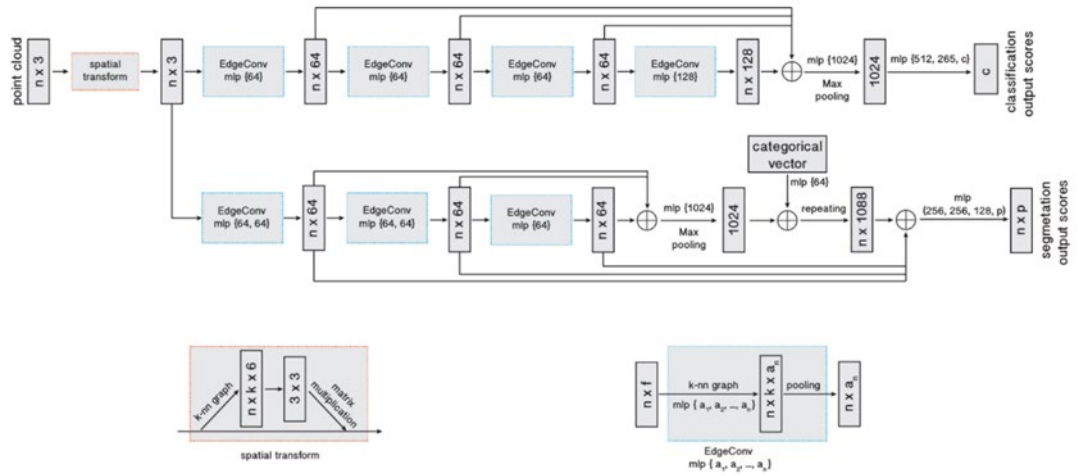


Figure 2.21 : Structure of the dynamic graph convolutional neural network (Wang et al., 2019, p.4)

To sum up, representing the relations of the points as graphs provides both local and global semantic information while learning the relations of the local patches. Particularly, in a cultural heritage documentation context where the parts of the buildings are intentionally designed and constructed in relation to each other using such an approach would help to unrevealing the semantics of the buildings. Many studies use DGCNN directly in the standard and historical as-is modeling workflow (Ma et al., 2020). Moreover, some studies adapt the method by manipulating the

structure of the deep learning architecture (Pierdicca et al., 2020; Morbidoni et al., 2020). Our computer engineer colleagues, Yusuf Sahin, Alican Mertan, and Gözde Ünal completed the forerunner examinations using DGCNN trained with a primitive version of HDD that has fewer samples that lack definitions of a blind arch and opening variations of dome lantern (Güneş et al., 2021). The initial examination only focused on a section of the Ince Minareli Madrasah Main dome system (IM_M) and a specific portion of the Sahip Ata Hanikah (SA_H) case that includes a blind arch and its surroundings. Even though the definition and variations in the synthetic data set, and real-world examples are updated in the process the final exploration is completed benefitting from the first results and workflow. This thesis examines two versions of DGCNN architecture with models having 2048 points and 10000 points. The results of the models are discussed in the evaluation section.

2.8.3 Sequence

Even though some studies are formed around the concept of permutation invariance, sequential models intentionally break this attribute to recognize parts of geometrically consecutive models. These types of networks work with the same principles as NLP algorithms that are also known as recurrent neural networks (RNN) (Elman, 1990). RNN algorithms decode the structure of the sequential units that are consecutively linked to each other in a context. To illustrate NLP tasks, the networks investigate texts word by word. Predicting the words in a sentence depends on prior expressions and the correlation between the words and sentences. The sequential deep learning models learn consecutive grammatical and structural relations of the word in a given specific language or translate from one language to another (Cho et al, 2014a). Relying on the grammatical structures of the languages it is possible to encode and decode the interrelated units in a context of part-whole relations considering the hierarchical structure (Chomsky, 1956). With a similar approach, Stiny and Gips (1971) investigated ruled-based structured visual grammars using the shapes as parts in a context. Apart from the type of the units and the language, these studies investigate rule-based connections of the parts.

Approaches in architectural design and construction depend on the proper combinations of the units on different scales prioritizing both the functions and aesthetics. Hence, capturing similar patterns between the elements is achievable in the

buildings that are built with resembling style. Particularly, historical masonry buildings exemplify the usage of the units like brick and stone to create architectural elements such as walls, arches, vaults, and domes following certain three-dimensional patterns. In different levels of detail, we see that the sub-elements are used as parts such as using walls, arches, vaults, and other related elements to define a building envelope. The patterns are also observable in the documentation process as well as design and construction. The presented research particularly investigates the application of the sequential deep neural networks that interprets the units of the models to segment the elements in the historical dome systems context. Yet, since each deep learning algorithm is adapted into using a particular type of sequence units such as a limited number of point groups or sliced three-dimensional area there is a gap between the architectural units and the units used in the algorithms.

In point cloud segmentation tasks, considering the ontological relations of the elements and pieces, sequential interpretation of data can give advantageous results in terms of semantically segmenting the parts of a whole. Although splitting models into small-scale batches using the constant volumes instead of using the approximate volumes of the different architectural elements, decoding the interrelations of the acquired slices would be beneficial in a way of identifying the irregularly split architectural elements.

Learning from ordered sets decodes the relations of the portions in a context. However, traditional RNN networks tend to suffer from vanishing gradient problems, in other words, lose track of information if the given set is too long. Long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and gated recurrent units (GRU) (Cho et al., 2014b) mostly suppress this problem by using more complex structures that can memorize longer series. Moreover, instead of investigating the sets only in one direction bi-directional networks investigated the correlation of the units considering both relations forward and backward (Shuster & Paliwal, 1997). This method might provide supplemental information in documentation tasks because the relationship between the elements always can be considered two-directional.

2.8.3.1 RSNET

Extracting meaningful information from three-dimensional point clouds to classify the points is examined in several studies. Initially, Recurrent Slice Networks (Huang et al., 2018) splits the three-dimensional space into rectangular prisms to get sequential

units to be processed by RNN algorithms. The slicing occurs in x, y, and z dimensions in the space (Figure 2.22). The series starts from the closest section to the 0 points and get larger. Dimensions of the slices affect the quality acquired local features. For capturing the preferred resolution, the network slices the point cloud using the real-world dimensions. In the Network global features are learned through slice pooling and unpooling layers (Figure 2.22). The examinations show that the gated recurrent unit is a base RNN method. Also, the extensive test reflects that in a room/building scale using 2 cm as a slicing resolution performed the best. Also, 1 meter as a sliding block size provided the best possible outputs. In the experiment with historical dome systems, the granted parameters are used to detect the fine-grained shape transitions between the different parts. Even though the algorithm achieves learning the relations of the batch of the models since is based on RNN architecture it can keep a limited amount of data to be processed.

However, since we cannot identify the dimensions or the point labels before segmentation and standardization of the process, hence three-dimensional slices are used to split models into rectangular prism volumes and investigate the sequential relations of the blocks.

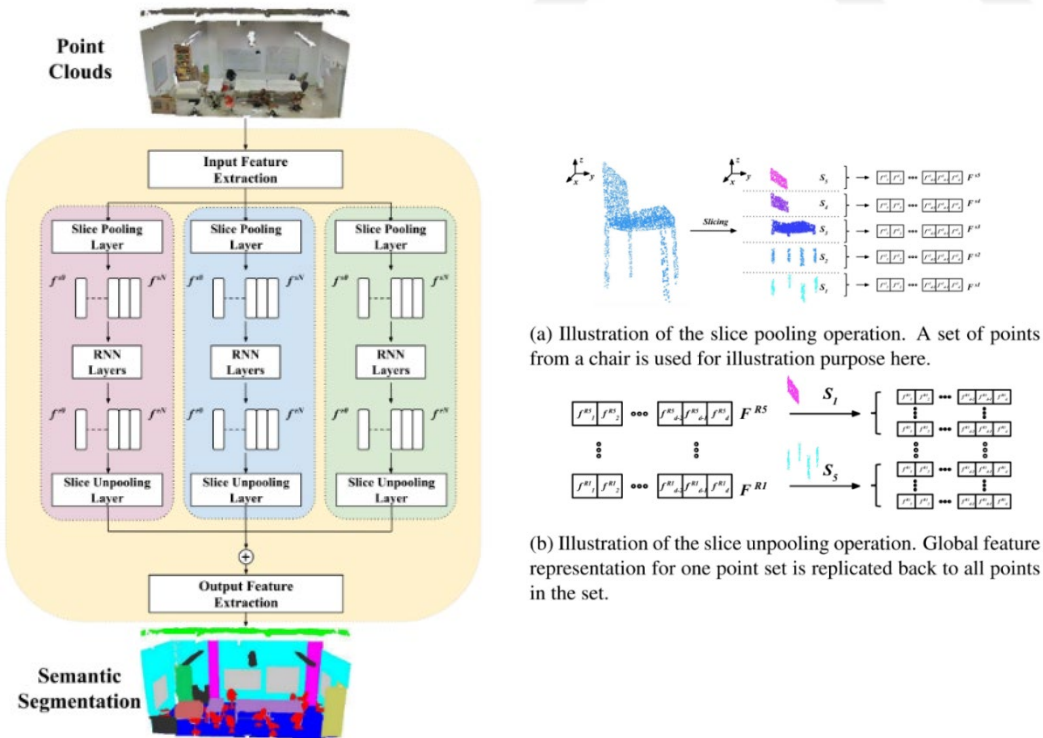


Figure 2.22 : Flowchart of the RSNET sequence model (Huang et al., 2018).

The consecutive data processing starts from a corner of a grid and ends in the opposite corner. The training batches use overlapping blocks with a 1-1m base and 0.5m stride for not to miss the details at the edges of the blocks (Figure 2.23). On the other hand, the test batch uses the same block dimension with the 1m non-overlapping strides (Figure 2.24).

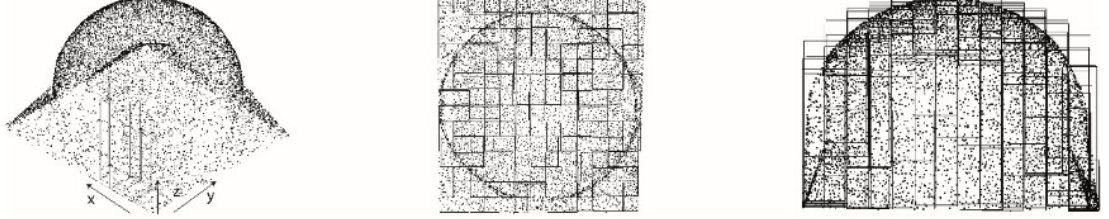


Figure 2.23 : Overlapping blocks are used in the training phase.



Figure 2.24 : Blocks used in the test phase.

2.8.3.2 Point2Sequence

Secondly, Point2Sequence (P2S) (Liu et al., 2019) uses attention-based sequence architecture for semantic segmentation of point clouds. The algorithm conceals the high-resolution knowledge in the local regions using multi-scale fields with attention mechanisms.

P2S network has 5 steps: multi-scale area establishment, area feature extraction, encoder-decoder feature aggregation, and local region feature aggregation. The multi-scale area establishment step includes sampling, searching, and grouping layers. M number of points are picked as the centroids of the local areas. The farthest point sampling method is applied to acquire the point sets. K_t number of points are being sought using the k -NN method with limited distance in space in local regions and grouped to form multi-scale areas. In the following step, feature extraction, a multi layer perceptron is applied to acquire the features of each multi-scale region. In this

process relative coordinate system is used to reference the local sets. Relative local coordinates provide a permutationally invariant solution. Then, to acquire high-resolution context-based knowledge from the local sets, an attention-based encoder-decoder is applied. Attention-based algorithms assign weight to the parts of the input regarding the impacts on the result. So, in the learning and predicting process the deep learning model gets the required directives without even scanning all the clusters of inputs (Bahdanau et al., 2014). In other words, the model learns the contextual influence of input units. The significance of different areas is emphasized by the attention set up. The attention-based sequence to sequence architecture accumulates the consecutive features of multi-scale regions of all the local regions. In the following step, a global feature vector is gathered by a simple MLP network. Finally, in the segmentation part of the network, extracted global features are used to identify the label of each unique point using by interpolating the features twice (Figure 2.25). Additionally, multiple MLPs used the improve the performance.

The original study investigates the hyperparameters of the P2S network. The network performs the best when the number of sampled points(M) is 384. The amount of the sampled point affects the diffusion of the local point sets (Figure 2.26). This is one of the main parameters to define the units of the whole. Additionally, the extensive examinations resulted that the LSTM structure providing the top accuracy. For the dome system examination, all the parameters are implemented as it is described in the original study. To conclude the network combines both attention and sequential model approaches to learn from the given models. Although the network accomplishes the semantic segmentation task with rational results, it is still fragile against the vanishing gradient problem that recurrent neural networks suffer from.

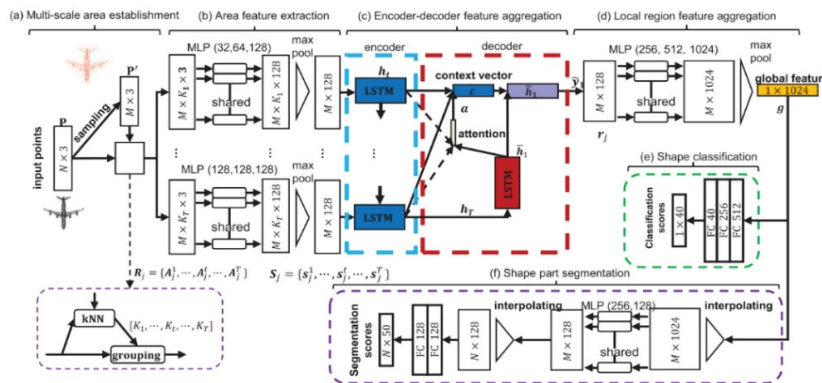


Figure 2.25 : Flowchart of the P2S model (Liu et al., 2019).

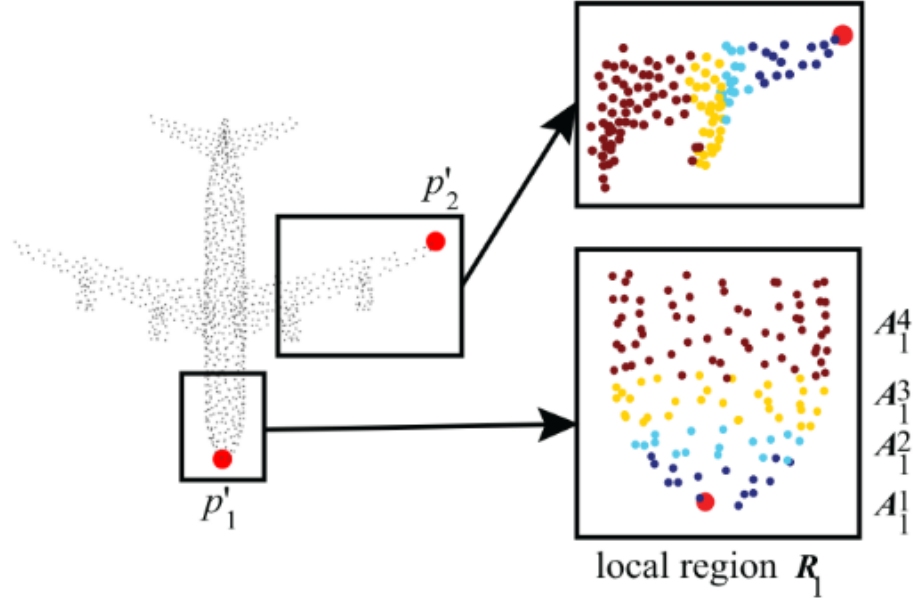


Figure 2.26 : Local point patches of the model (Liu et al., 2019).

As with the other deep learning approaches sequential models are used in remote sensing and architectural cultural heritage documentation tasks in several different contexts. Grilli et al. (2019) applied a Bi-LSTM-based custom semantic segmentation algorithm to a historical temple and a renaissance building to recognize the architectural elements individually. Even though the algorithm accomplishes identifying most of the points, other machine learning algorithms perform better numerically. Similarly, Atik and Duran (2021) applied both LSTM and BiLSTM-based sequence models to classify the points in an aerial photogrammetry context using the geometrical features of the points. The study emphasized using both color and geometric data with BiLSTM performed the best in different input settings. Additionally, Özdemir and Remondino (2019) segmented the point cloud of an urban scale model captured by an airborne laser scanning tool. They use a custom BiLSTM algorithm that uses the geometric features of the points. Their method returns the segmented outputs while being fragile to varying densities of the point clouds. All three examples employ geometric features that are extracted by the users. Also, the methods lack an understanding of context-awareness. Despite the fact that presented real-world applications performed well with simple and “shallow” deep neural networks with BiLSTM structures, well-designed algorithms that focus on learning local, global, and sequential relations of the models might improve the quality of the results. Exceptionally, in recognition of the Anatolian dome system context, more precise methods that work with minimal features would provide useful results.

2.8.4 Transformer-attention based

RNN-based sequential models recognize local-global features in semantic segmentation tasks by interpreting the data in sequence, yet they suffer from vanishing gradient problems and limitations of consecutive computing architecture design.

Transformers address these limitations by using an attention-based architecture that keeps all the contextual qualities without losing the sequential information in extended sets thanks to the positional encoding and multi-head attention modules in the algorithms (Vaswani et al., 2017). The design of the method is permutation invariant but positional embedding keeps the positions of the units to make the sequence computable in parallel while keeping the contextual standpoints (Figure 2.27).

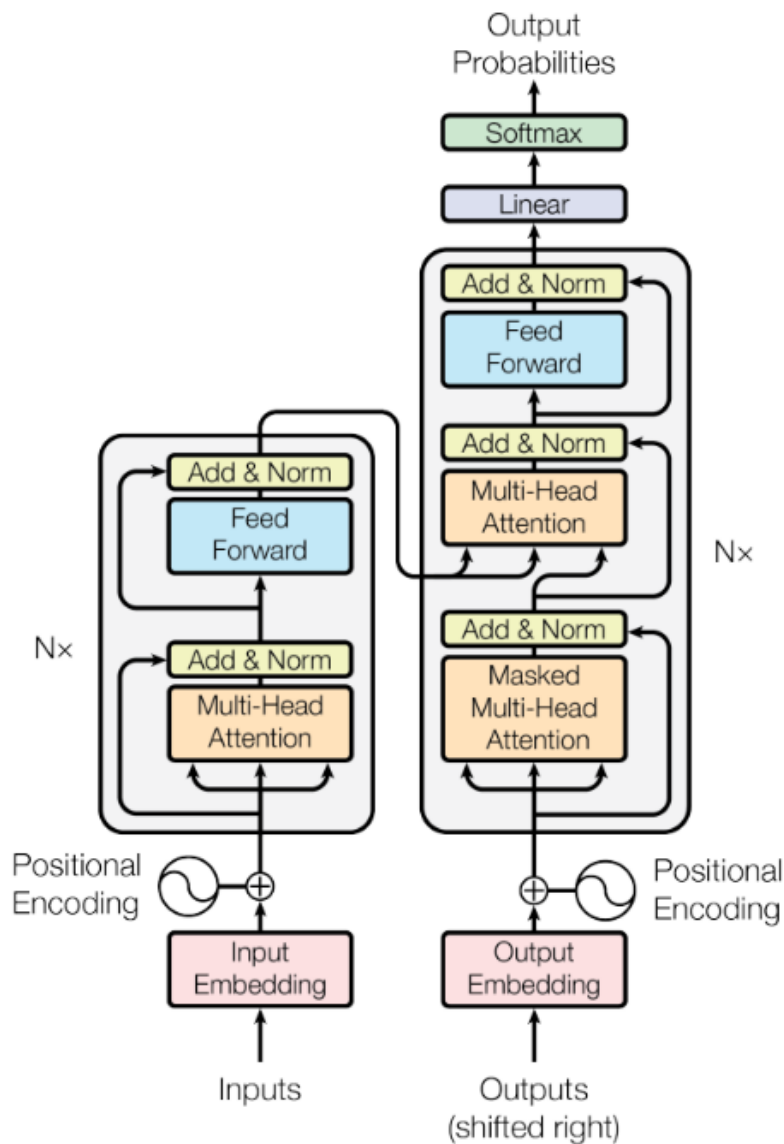


Figure 2.27 : Schema of a transformer (Vaswani et al., 2017).

NLP and machine translation tasks benefit from the contextual and sequential interpretation of the data. Additionally, Vision Transformers (ViT) (Dosovitskiy et al., 2021) adapted the transformer architecture into two-dimensional computer vision tasks. ViT network simply slices the images into grid cells and flattens them into a linear set of data that can be defined as consecutive elements. This application exemplifies the effective implementation of transformer networks into computer vision problems. Different from the RNN-based models all the computations can be done parallelly which makes the computation process relatively efficient and fast.

Transformer-based algorithms provide contextual understanding while processing the consecutive data. Yet, the encoding of the features is usually one-directional. Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) analyses the provided data in a bi-directional structure. Moreover, it adds two additional steps in the workflow; pre-training and finetuning. Pre-training step unsupervised learning phase that randomly masks some of the inputs in the sequence and tries to fulfill the gaps in interpreting the contextual information. Additionally, it predicts the following sentence. Different from the previous transformer mechanisms the tasks take the information in both directions into account. These two sub-tasks help the network to understand the context and the language. Fine-tuning step prepares the deep neural network for the specific assignments. After the two extra beginning steps (Figure 2.28), the network is trained in a regular way to serve a specific purpose. BERT carries out that knowledge further by encoding the relations bidirectionally while keeping the sequential and effective features in the network.

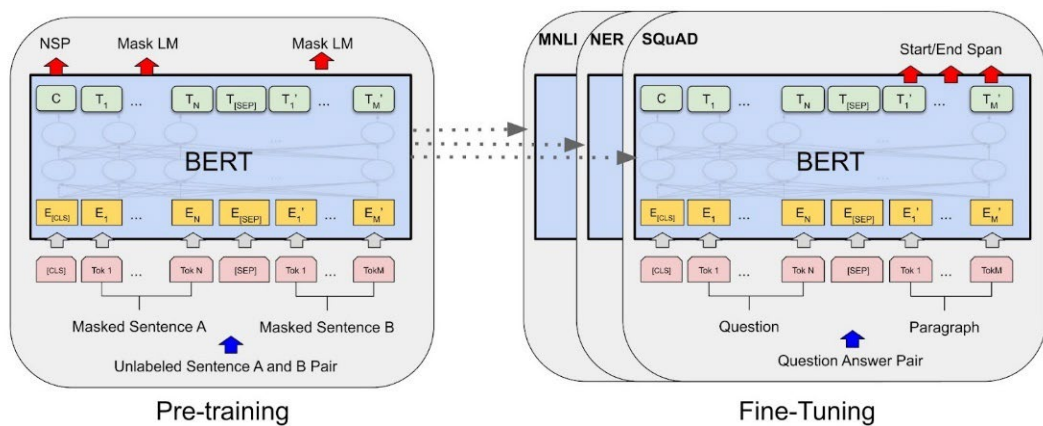


Figure 2.28 : Pre-training and Fine-Tuning steps of the BERT (Devlin et al., 2019).

2.8.4.1 Point Transformer

Advancements in NLP and two-dimensional computer vision networks affected revolutionized the three-dimensional computer vision assignments as well due to the necessity of analysis of complex shape relations in specific contexts. Point Transformer (Zhao et al., 2021), a self-attention-based network implements the transformer concept into point cloud analysis tasks. The algorithm uses local neighborhood (k-NN) to get the sub-group in the initial point cloud and employ the self-attention mechanism in the local sets. Local relations of the input are utilized thanks to position encoding in the self-attention. While standard transformer architectures are structured manually point to transformer architecture three-dimensional positions as possible inputs for position encoding.

The network uses a residual transformer cluster that combines self-attention steps and linear projections to lower the dimension number and speed up the calculations. The block processes the feature vectors related to points in the space. The deep learning architecture applies the block as a fundamental feature accumulation method. The transition down blocks decreases the amount of the data to be processed. In this step farthest point sampling is used to lessen the number of points then, k-NN is applied to regroup the points. Then a max-pooling is applied at the end of the block. The transition-up modules follow the U-NET-like encoder-decoder deep learning architecture that has symmetric skip connections between the encoded and decoded features (Ronneberger et al., 2015). Essentially this block interprets the features for the fine-grained state of the point cloud by decoding and combining the determined features. Finally, a multilayer perceptron is implemented for measuring the last outputs (Figures 2.29 and 2.30).

The Point Transformer network is extensively examined in part segmentation task to identify useful parameters to be used in the process in ShapeNetPart Benchmark (Chang et al., 2015; Yi et al., 2016). Relative position encoding uses the local patches with the neighborhood number 16 provided the optimal resolution in the results. Also, the HDD trained on Point Transformer using the supplied hyperparameters in the semantic segmentation process. The Point Transformer network delivers permutation invariant methods that analyze the three-dimensional data without losing the learned sequential features in the process. The potential of decoding the complex three-dimensional relations in the point clouds is promising in a sense of investigating the

relations of the architectural elements in the documentation of cultural heritage context. Yet ambiguity of the division of the patches compared to the strict boundaries of the architectural elements might be limiting or misleading in some cases.

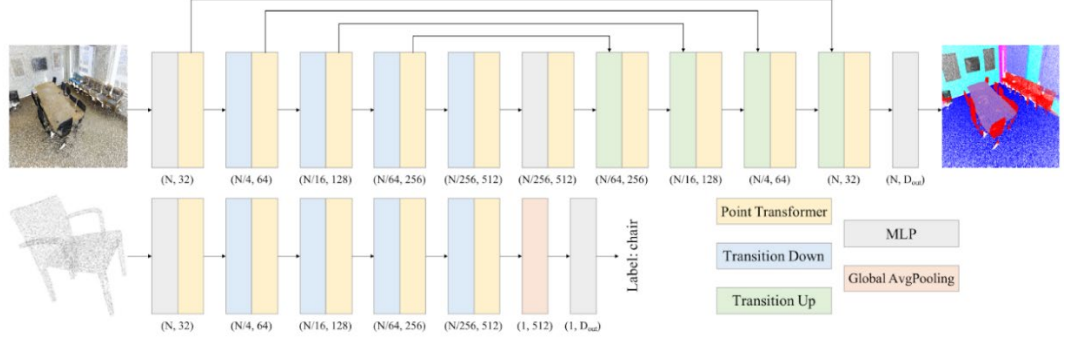


Figure 2.29 : (Top) Semantic segmentation and classification structure of the Point Transformer (Zhao et al., 2021).

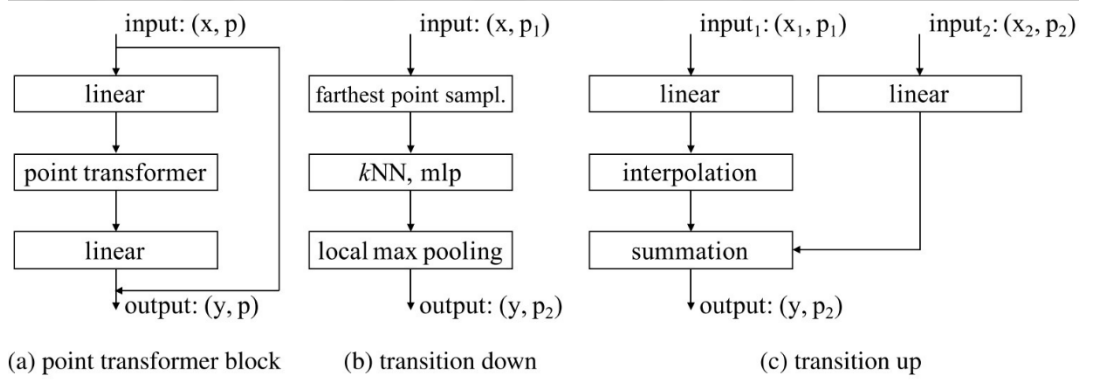


Figure 2.30 : Sub-structures of the main methods in the Point Transformer Network (Zhao et al., 2021).

2.8.4.2 Point-BERT

Benefitting from the progression in NLP, Yu et al. (2021) implemented the idea of Bidirectional Encoding Representation from Transformers into the point cloud understanding tasks. The Point-BERT networks follow the same main three steps of the BERT algorithm: pretraining, fine-tuning, and training. In detail, even though treating a single point in a point cloud as a unit of a whole like a word of a sentence is problematic due to the difficulty of acquiring semantic understandings, using local patches of the whole is meaningful in a sense of extracting sequential qualities of the data in the given context. To acquire representative sub-sets of the clouds farthest point sampling (FPS) is applied and the center point of the patches is used to find a limited number of neighboring (k-NN) points around. For the detected points, relative

positions regarding the center points of the patches are used. The obtained point patches might be used as input in the training yet the algorithms apply mini-PointNet architecture in order to reflect the smaller sets into point embeddings.

In the following step, the point embeddings are used by the point tokenizer to transform them into individual point tokenizers. A decoder accepts the point tokens and re-builds the splitter sub-sets. DGCNN network is implemented into the algorithm to learn neighborhood relations between the tokens. In the end, the sub-sets are rebuilt. Moreover, the algorithm executes Masked Point Modeling (MPM) for pre-train Transformers. MPM hides a part of the given point cloud and gets to know rebuilding the missing point tokens. This process aims to help to rationalize the connections of different geometries within the discrete groups of the point cloud (Figure 2.31). This part of the network is based on self-supervised learning that can produce guidance signals without needing supplemental directives. Thus, this network intends to benefit from parallel processing, a significantly low inductive bias which means generalizing the learned information, and a vast range of characteristics due to the transformer structure it has.

The semantic segmentation application of Point-BERT on the Historical Dome Dataset is used respecting the provided hyperparameters in the original study tested on ShapeNet (Chang et al., 2015). The Masked Point Modeling depth of the model is set to 12 while the feature dimension is 384 with 6 heads. In the sampling phase, the model is split into 128 parts.

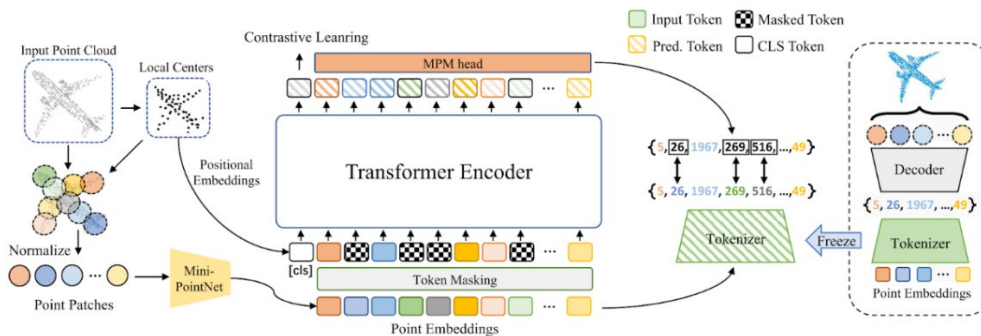


Figure 2.31 : The network structure of the Point-BERT (Yu et al., 2021).

Even though networks use varied methods to evaluate the relations of the points local and global-wise each method profits from the advancements of the previous studies. For example, Point-BERT profits from both the PointNet and DGCNN which process data with Multilayer Layer Perceptrons and graph-based algorithms in order.

3. EVALUATION

3.1 Test Set

To evaluate the semantic segmentation results, this research splits the process into two chunks. In the first step, the deep learning models are evaluated on a test division of the Historic Dome Data Set. Next, the results of the real-world data are examined. Since the Historic Dome Data Set is created focusing on identifying and reconstructing the structural elements of the historic Anatolian domes the results of the acquired models is also significant in terms of reflecting the practical use. Networks are tested on 11 different cases in total.

The prepared data is split into two parts, full models and partial models (Table 2.1). Full models include all the architectural elements with minimal photogrammetric data loss. However, some of the models suffer from weak lighting conditions, limited angles for capturing proper images, and material reflections. The models with a limited number of elements and low-quality point clouds due to these limitations are described as partial models.

3.2 Metrics

The deep learning applications and the results are evaluated using different methods. The common practice is using test accuracy values for comparing the percentages of the true predictions. The classification tasks directly compare the predicted and ground truth classes for the given data. In two-dimensional data, there is only one label for each image. Similarly, point clouds have only one label for the given data for the classification of the given models.

In segmentation tasks, the units are labeled instead of the whole images or models, so individual pixels of the images and the points of the point clouds are used to compare the predicted and control labels. In addition to accuracy percentages, segmentation methods use intersection over union (IoU) calculations to measure the overlapping

parts with the predicted and previously labeled data. It is a simple division of the area of overlap by area of union. The method determines the correctness of the identified units.

Evaluation of the segmented data has additional performance metrics that reflect the success of the process. F1, recall and precision calculations that are based on the identity of the predicted units regarding the predicted and ground truth classes. To define the state of the predicted units four terms are used: “true positive”, “false positive”, “true negative”, and “false negative”. While true and false represent the correctness prediction, positive and negative symbolize being a part of the investigated class or not. True positives express the points of a specific class that are predicted correctly. False-positive shows the number of points that are predicted to be in a class but the prediction is incorrect. True negative indicates the points that are not part of a class and labeled accordingly. Lastly, false denotes the points that are identified as not in the class while actually, it is a part of the investigated class. Using defined “positive” values, “precision” shows the ratio of “true positive” data in all “positive” clusters including both true and false. Also, recall value represents a ratio of “true positive” data in a “true positive” and a “false negative” set which means the percentage of identified items in a ground truth set. Finally, an “F1 Score” calculated a harmonic mean of the precision and the recall to assess segmentation performance that expresses both prediction-based and ground truth base evaluations (Powers, 2011). The stated evaluations could be calculated per model or for the whole test set. The presented study uses both of the results in the assessment context.

Deep learning methods are probabilistic solutions that return very similar but varying solutions in tests. Hence, acquiring only one solution would be misleading. To overcome this constraint each model is tested 20 epochs in testing steps. Then top-5 of the captured results are reserved for calculating the final results according to the accuracy values. Mean accuracies and mean intersection over union values are evaluated using them.

Some of the points in the predicted point cloud models are mislabeled. While visual results illustrate the interchange between the different labels, a “confusion matrix” is used to show numerical values for the representation success rate. Confusion matrices are provided for each test set in the HDD while it is supplied per model in real-world examples.

The qualitative evaluation process is supported by the images of predicted and ground truth points clouds that show the labels of the models using the different colors labels. Dome, transition element, arch, and wall classes are represented in blue, green, red, and yellow colors in order. Both general and local visuals present the results of the segmented deep learning models. In evaluations step two approach is adapted into the process. Initially, the dataset is tested on the test section of the Historic Dome Data Set that is based on the same definition as the training set. Additionally, a data set was created with the real-world photogrammetric models of the monumental Anatolian dome systems. Point clouds are subsampled into 10000 and 2048 points to standardize the benchmarks. The assessment step starts by discussing the synthetic data set under the guidance of both numeric and visual outputs. Since each deep learning method is intentionally included in the process to examine the data processing approach, the test set of the synthetic data is used as a benchmark that benefits the qualities that the neural networks contributed to the identification process. Subsequently, the trained models are experimented on the real-world data to investigate the usage cases of the models in different cases.

3.3 Examination Set-Up

The aforementioned deep learning methods are trained in different settings due to different requirements and the constraints of computational power. The models are grouped into four regarding the approach they use to process the data. The data representation type mentioned for each network to all the deep learning used in the training session is documented in Table 3.2. The applied preprocessing methods are explicitly noted (Table 3.2).

Table 3.1 : Properties of used deep learning algorithms.

Network	Frameworks	Type	Data Type	Pre-train	Fine-tuning
PointNet++	Pytorch (1.9) + PyG	MLP	ShapeNet	Not Req	Not Req
DGCNN	Pytorch (1.9) + PyG	Graph	ShapeNet	Not Req	Not Req
P2S	Tensorflow 1.8	Seq	ShapeNet	Not Req	Not Req
RSNET	Pytorch 1.0	Seq	S3DIS	Not Req	Not Req
PTr	Pytorch (1.9) + PyG	Trans	ShapeNet	Not Req	Not Req.
Point-BERT	PyTorch 1.8.1	Trans	ShapeNet	ShapeNet55	Req.

Table 3.2 : Preprocessing settings.

Network	Point Number	Data Preprocessing		Data Augmentation			Normal	Color
		Norm.	Centr.	Rotate	Translate	Jitter		
PointNet++	10000 / 2048	Yes	Yes	Z-Axis Rand-40	0.01	No	Yes	No
DGCNN	10000 / 2048	Yes	Yes	Z-Axis Rand-40	0.01	No	Yes	No
P2S	10000/ 2048	Yes	Yes	No	No	No	Yes	No
RSNET	~10000/ ~2048	No	No	No	No	No	No	No
PTr	10000 / 2048	Yes	Yes	Z-Axis Rand-40	0.01	No	Yes	No
Point-BERT	10000/ 2048	Yes	Yes	No	No	No	No	No

3.4 Results of the Examinations

The aforementioned deep learning algorithms are trained on the training set of HDD. Consequently, tested with the test split of the HDD for evaluating the performance of the networks (Table 3.3). In the deep learning tasks, RSNET used an approximate number of elements instead of a certain number of points due to the design of the network, the other examinations were completed with 10000 and 2048 points. All networks achieved a certain level of performance. Specifically, RSNET and Point-BERT led the evaluation metrics.

The trained networks were tested with real data to assess the real-work performance of both networks and the dataset (Tables 3.4, 3.5, 3.6, and 3.7). The results show that the investigated deep neural networks tend to perform similar to the synthetic data set where the provided data is clean, not fragmented, and include all the classes in the scene. However, the networks were challenged to understand the context of the scene

when some labels of the models were not in the scene or the existing points of a class is shattered due to the low quality of the photogrammetric models.

The increase in the points impacted the quality of the results in a positive way in the synthetic dataset. However, some networks performed worse than the 2k in the 10k test in real-world examinations. There might be several reasons for this limitation. Using fewer points might be normalizing the positional noises of the point clouds, and uncertain surface normals or the density of the model might be more equalized in the samples with the 2k points. To overcome the limitation dataset might be enriched with the edge cases or each deep neural network can be fine-tuned separately. The following sections will discuss the results of the result per each neural network.

Table 3.3 : The results of HDD.

Network	Point N.	Epoch	Batch Size	Train/Val/Test	oAcc. (%)	mIoU	f1	recall	precision
PointNet++	2048	300	36/36	0.6/0.2/0.2	95.42	91.00	0.96	0.96	0.96
	10000	300	12/12	0.6/0.2/0.2	96.67	92.36	0.97	0.97	0.97
DGCNN	2048	300	36/36	0.6/0.2/0.2	96.15	90.72	0.97	0.97	0.97
	10000	300	10/10	0.6/0.2/0.2	97.17	94.89	0.98	0.98	0.98
RSNET	~2048	18	384	0.6/0.2/0.2	96.87	93.93	0.97	0.97	0.97
	~10000	10	256	0.6/0.2/0.2	98.21	96.49	0.98	0.98	0.98
P2S	2048	300	32	0.6/0.2/0.2	96.48	91.85	0.96	0.97	0.96
	10000	288	32	0.6/0.2/0.2	96.25	91.24	0.96	0.96	0.96
PTr	2048	300	24/24	0.6/0.2/0.2	95.98	92.34	0.96	0.96	0.96
	10000	300	12/12	0.6/0.2/0.2	97.36	94.87	0.97	0.97	0.97
Point-BERT	2048	300	4	0.6/0.2/0.2	97.74	94.59	0.98	0.98	0.98
	10000	300	16	0.6/0.2/0.2	97.93	95.09	0.98	0.98	0.98

Table 3.4 : Test accuracies of the real-world full models.

Network	Point N.	IM_M	SA_H	KM_M	HK	oAcc
PointNet++	2048	88.59	66.53	89.14	88.31	83.13
	10000	83.51	56.56	73.48	74.75	72.21
DGCNN	2048	87.85	72.86	87.26	89.85	83.88
	10000	85.46	75.51	85.96	84.90	82.75
RSNET	~2048	59.70	80.50	75.03	50.75	67.77
	~10000	70.20	81.82	79.27	56.71	72.85
P2S	2048	86.42	74.30	78.05	86.35	81.15
	10000	91.70	66.21	80.85	84.53	80.70
PTr	2048	86.54	70.53	84.13	80.09	80.65
	10000	82.35	55.70	83.58	81.85	75.52
Point-BERT	2048	85.84	71.92	77.34	92.77	81.97
	10000	60.54	64.88	65.00	90.46	70.21

Table 3.5 : Test mIoUs of the real-world full models.

Network	Point N.	IM_M	SA_H	KM_M	HK	mIoU
PointNet++	2048	79.53	49.86	80.41	79.07	71.11
	10000	71.69	39.43	58.09	59.69	56.51
DGCNN	2048	78.95	57.31	77.41	81.50	68.23
	10000	74.62	60.66	75.38	73.76	63.16
RSNET	2048	42.64	67.37	60.04	34.00	52.01
	10000	54.09	69.23	65.60	50.05	59.24
P2S	2048	76.10	59.14	64.05	76.06	68.28
	10000	84.69	49.49	67.89	73.22	67.65
PTr	2048	76.28	54.50	72.62	66.82	67.59
	10000	70.06	38.60	71.81	69.29	60.67
Point-BERT	2048	75.19	56.15	63.05	86.52	69.44
	10000	43.41	48.01	48.13	82.58	54.10

Table 3.6 : Test accuracies of the real-world partial models.

Network	Point N.	AK_ULU	KM_S	SA_T	AB_M	AK_GUD	KM_N	AL	oAcc
PointNet++	2048	77.50	87.51	74.31	89.48	91.95	87.39	91.87	84.91
	10000	83.60	38.62	74.14	83.79	90.23	60.71	79.79	72.60
DGCNN	2048	78.93	38.63	77.40	56.28	74.78	52.73	88.69	66.55
	10000	77.70	29.15	53.29	52.00	69.35	31.21	84.65	56.49
RSNET	2048	69.76	03.41	12.38	05.66	07.01	20.80	89.59	40.06
	10000	73.11	07.56	16.72	08.86	11.16	25.51	89.86	43.07
P2S	2048	79.78	36.85	04.21	60.03	72.73	01.96	92.05	49.14
	10000	77.39	15.92	54.61	62.46	79.67	31.42	87.27	57.54
PTr	2048	80.60	66.77	60.00	65.99	85.90	82.58	86.41	73.18
	10000	76.86	93.44	78.97	83.80	81.78	73.16	85.16	81.31
Point-BERT	2048	80.37	38.81	60.74	82.56	90.38	47.80	95.84	70.93
	10000	82.56	20.74	61.16	61.74	79.36	18.54	90.95	59.29

Table 3.7 : Test mIoUs of the real-world partial models.

Network	Point N.	AK_ULU	KM_S	SA_T	AB_M	AK_GUD	KM_N	AL	mIoU
PointNet++	2048	63.29	77.82	59.17	80.98	85.10	77.79	84.90	73.79
	10000	71.83	58.91	23.93	72.11	82.20	43.59	66.37	56.69
DGCNN	2048	65.21	23.94	63.13	39.17	59.70	35.83	79.69	60.01
	10000	63.54	17.06	36.32	35.14	53.09	18.49	73.38	43.73
RSNET	2048	53.71	01.73	06.60	02.91	03.63	11.60	81.15	25.05
	10000	57.61	03.93	09.21	04.63	05.91	12.62	81.60	27.44
P2S	2048	66.38	22.60	02.17	42.91	57.40	01.01	85.31	32.57
	10000	63.15	08.65	37.60	45.45	66.24	18.65	77.43	40.49
PTr	2048	67.55	51.82	42.93	49.33	75.29	70.50	76.07	57.53
	10000	62.42	87.70	65.26	72.11	69.18	57.68	74.20	68.50
Point-BERT	2048	67.18	24.08	43.61	70.31	82.45	31.40	92.30	54.96
	10000	70.29	11.57	44.05	44.65	65.78	10.21	83.40	42.14

3.4.1 PointNet++

PointNet++ has a non-complicated deep learning architecture that simplifies the geometries and internalizes both local and global features. Since the network learns the relations of sampled regions on different scales it extracts the global regions with high accuracy, yet apprehension of the local relation is limited.

In the synthetic data set test, PointNet++ confuses only the boundaries of the different parts. The confusion matrix in Figure 3.1 shows that transition element points are confused as wall and dome points while other parts are identified with high accuracy in HDD 2k and 10k tests. Particularly, the transition areas where the bottom boundary of the shapes is not planar causes problem in the identification of the ideal boundaries in the prediction step (Tables 3.8 and 3.9). Additionally, the network is weak in detecting bling arch geometries. The results on HDD show that even though the 10k version numerically performed better, the 2k test returned visually plausible outcomes.

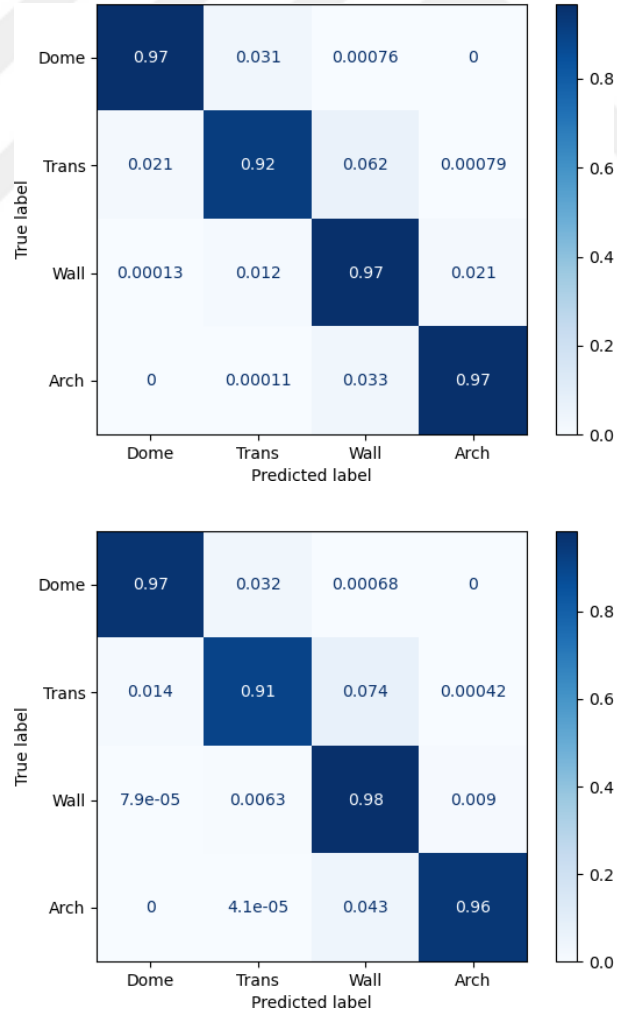


Figure 3.1 : Confusion matrices of PointNet++ on HDD, (top) 2k, and (bottom) 10k.

Table 3.8 : Results of PointNet++ using HDD and 2k points.

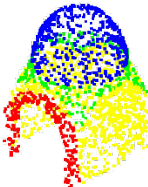
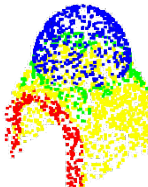
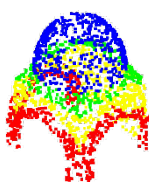
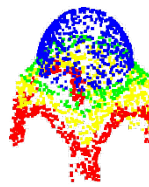
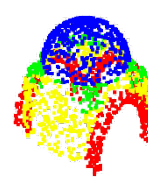
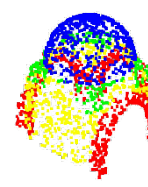
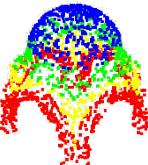
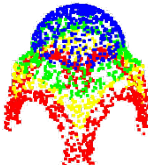
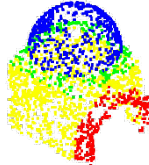
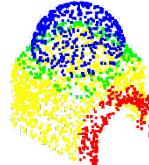
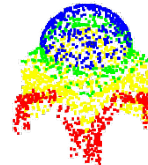
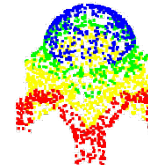
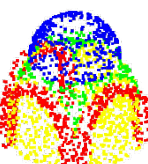
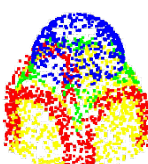
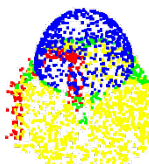
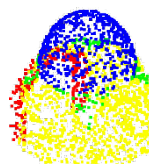
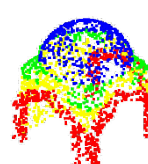
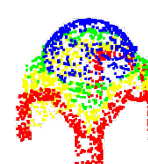
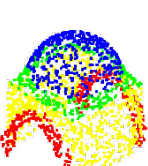
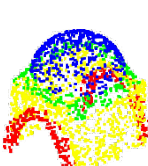
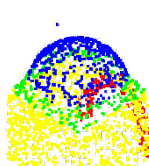
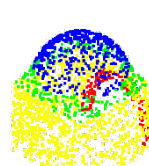
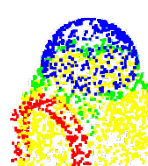
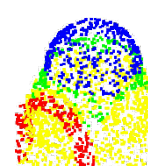
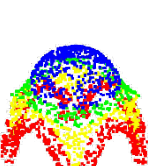
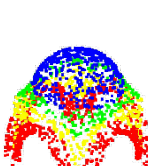
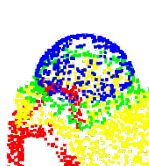
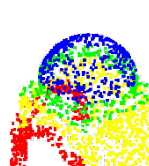
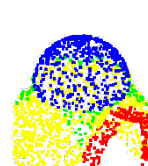
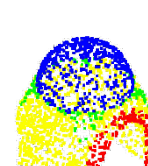
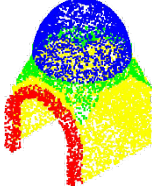
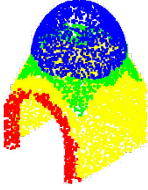
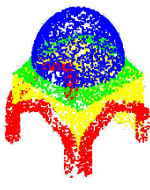
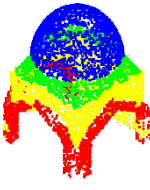
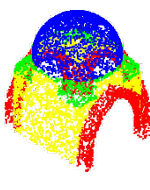
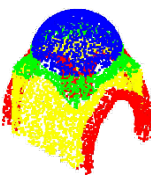
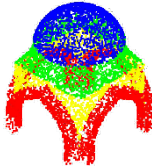
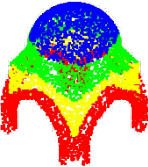
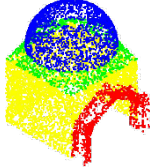
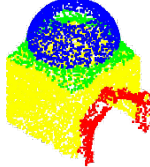
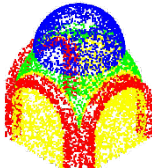
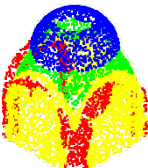
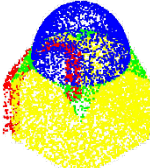
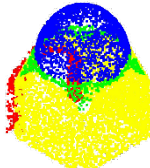
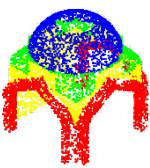
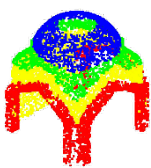
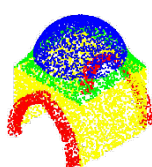
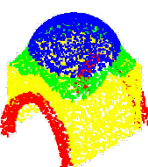
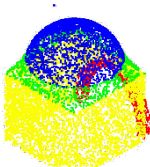
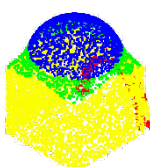
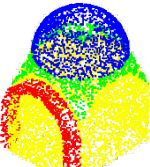
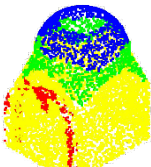
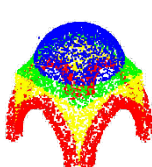
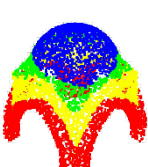
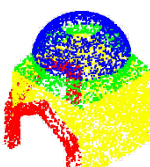
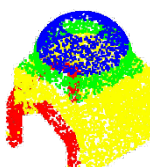
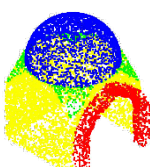
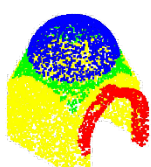
Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

Table 3.9 : Results of PointNet++ using HDD and 10k points.

Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

The PointNet++ network performed relatively well in real-world cases as well, compared to the other networks. The network particularly failed to understand the SA_H case in 10k where arches are extremely narrow and the walls are positioned at the corners instead of the edges of the square floor plan. Additionally, the HK case has some problems due to the irregularly distributed points in the mihrab area embedded

in the wall. On the other hand, particularly, in 2k examples, the deep learning model achieved high quality in terms of identifying approximate regions for the parts of the dome systems even in partial models. Figure 3.2 shows that in both full and partial 2k real-world examination sets dome points are identified with a high ratio while transition elements are mostly confused with domes. The wall points are identified better in full cases thanks to the more regular existence of wall points. Since partial models have a limited number of regularly distributed arch points, PointNet++ identified them with less error rate than the full models. The results demonstrate that PointNet++ learns contextual information with reasonable errors in detecting boundaries of the elements (Tables 3.10, 3.11, 3.12, and 3.13). The drawbacks in definitions of HDD such as representing only ideal scenes, and optimal distribution of the labels in the point cloud cause problems in the segmentation of partial real-world models.

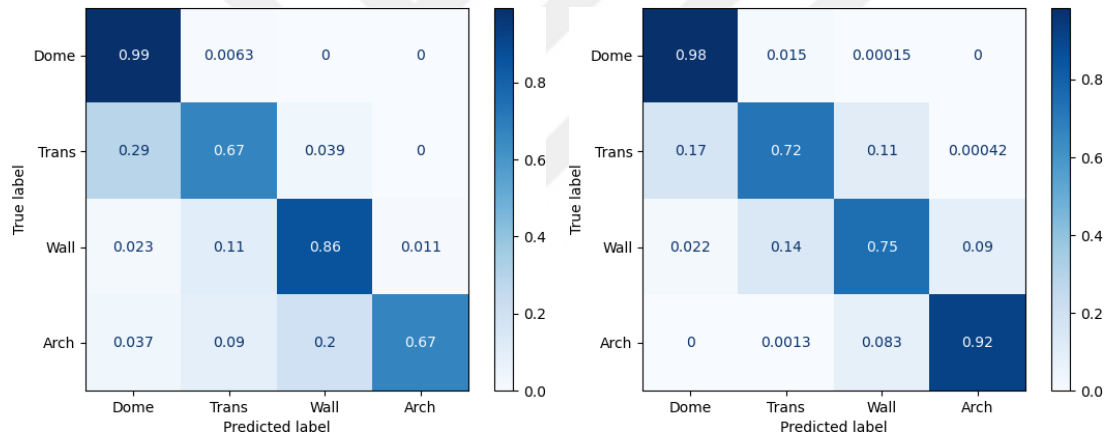


Figure 3.2 : Confusion matrices of PointNet++ (left) full and (right) partial real-world models in 2k.

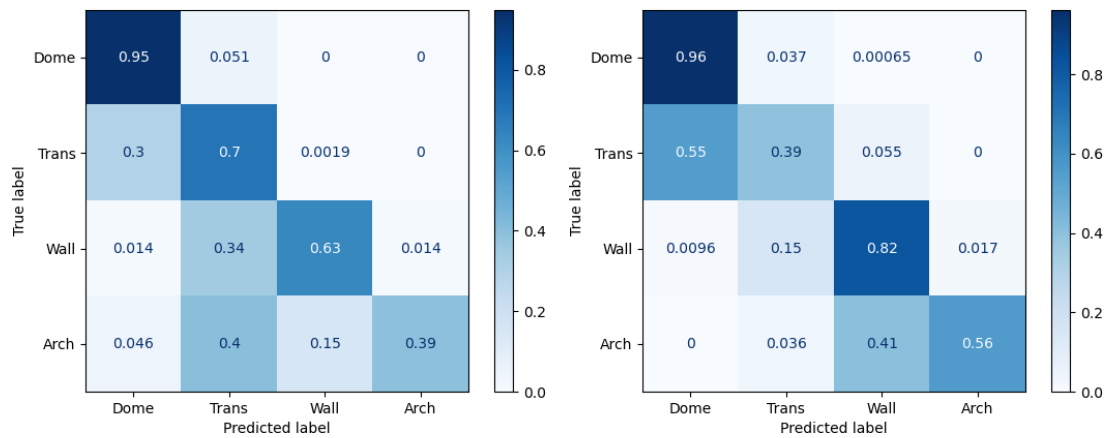


Figure 3.3 : Confusion matrices of PointNet++ (left) full and (right) partial real-world models in 10k.

Table 3.10 : Results of PointNet++ on full models, 2k points.

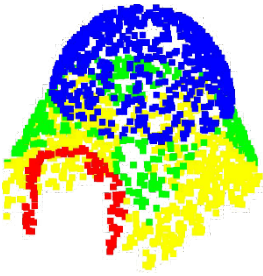
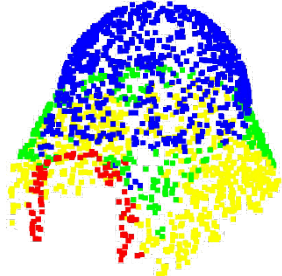
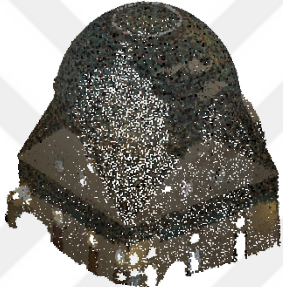
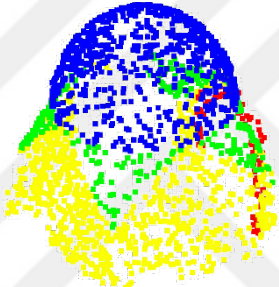
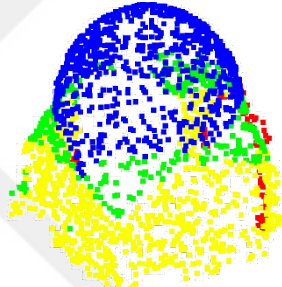
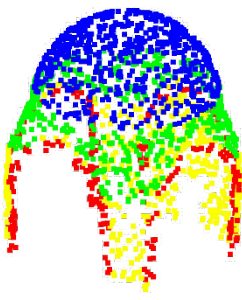
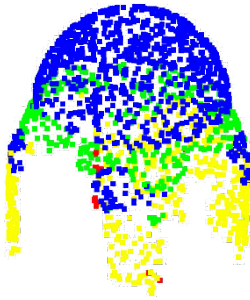
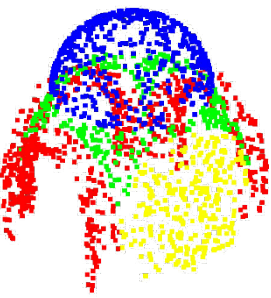
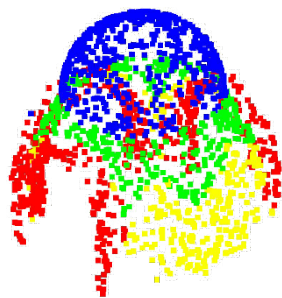
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.11 : Results of PointNet++ on full models, 10k points.

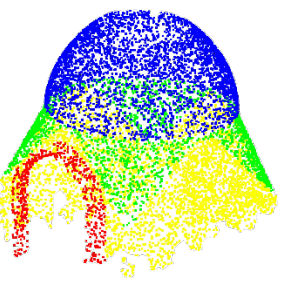
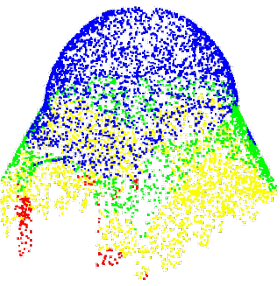
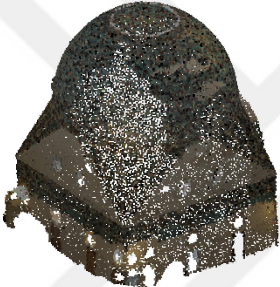
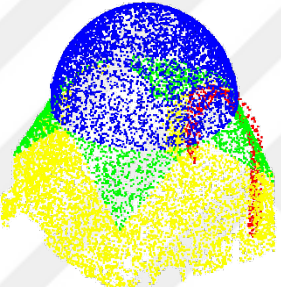
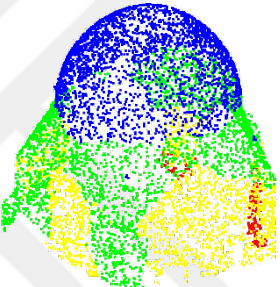
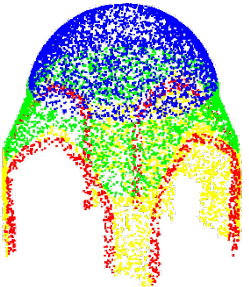
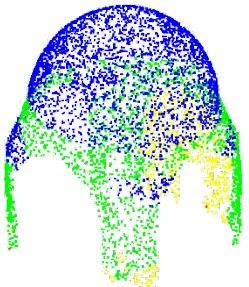
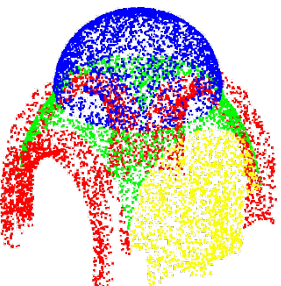
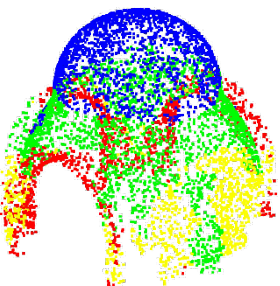
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.12 : Results of PointNet++ on partial models, 2k points.

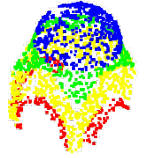
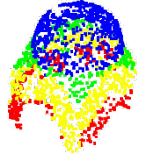
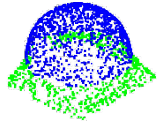
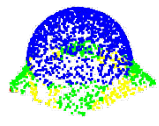
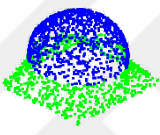
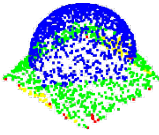
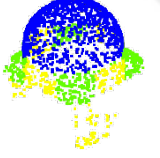
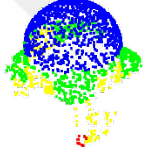
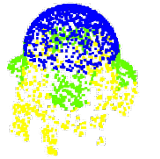
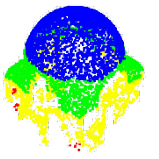
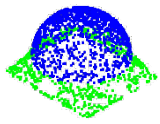
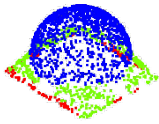
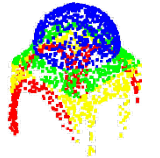
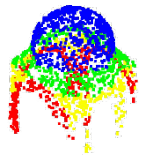
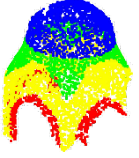
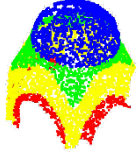
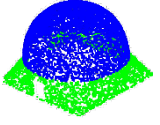
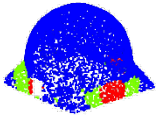
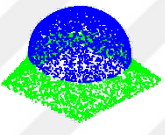
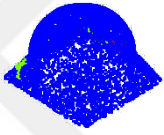
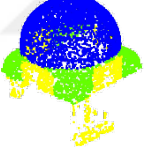
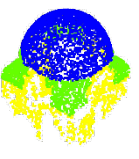
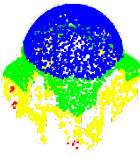
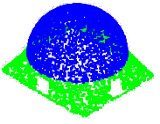
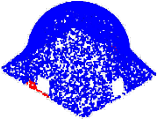
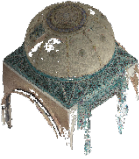
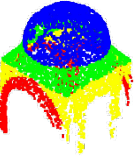
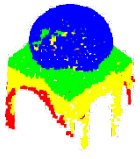
Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

Table 3.13 : Results of PointNet++ on partial models, 10k points.

Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

3.4.2 DGCNN

DGCNN is advanced in examining the local neighborhood relations of the points. In the synthetic data set test, it confused only the boundaries of the different elements. The graph-based network performed in similar qualities in both 2k and 10k test sets of the synthetic data set. Confusion matrices in Figure 3.4 shows in HDD 2k and 10k tests only transition elements are confused with the neighboring parts while other architectural elements are identified with a low level of error. The misprediction of the transition element causes the wrong identification of the points as dome and wall due to the complex relations of these three elements. Additionally, the limited surface area and the point number of the transition zone make the segmentation process error-prone. Particularly, in the test batch of the Historic Dome Data Set, the acquired results reflect that the global relations are successfully identified and there are not any extreme results of segmentation (Tables 3.14 and 3.15). Yet, the high accuracy of the segmentation relies on the existence and fully represented simplified parts. It should be noted that synthetic set shares the same definition of architectural parts with different combinations, ratios, and dimensions.

DGCNN resulted positively in the data batches where the model has all of the four elements in real-world examples. Particularly, in the SA_H case, it performed relatively well and identified the approximate boundaries of the dome and the transition elements. However, the arch could not be identified in the SA_H model. The deep learning model that is tested on full models failed to identify, where the depth of arches is too narrow and the wall geometry has a bumpy surface due to embedded sections in the wall geometry such as mihrab.

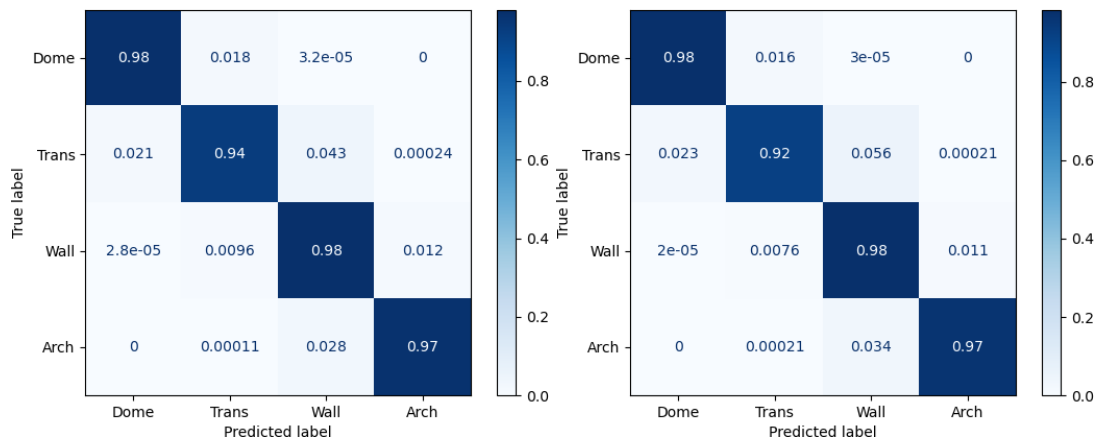


Figure 3.4 : Confusion matrices of DGCNN on HDD, 2k (left), and 10k (right).

Table 3.14 : Results of DGCNN on HDD, 2k points.

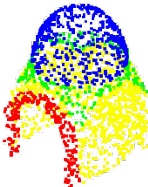
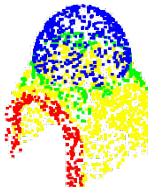
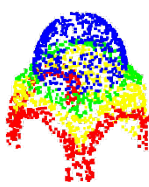
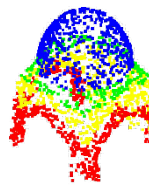
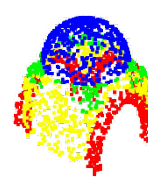
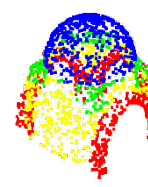
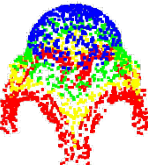
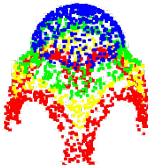
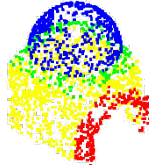
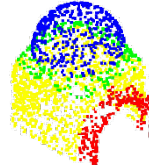
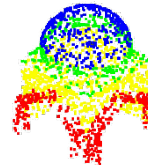
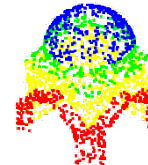
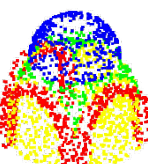
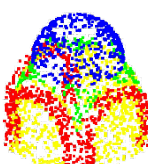
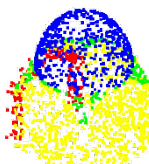
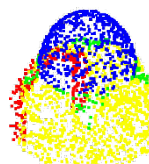
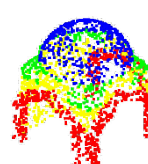
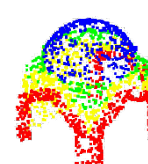
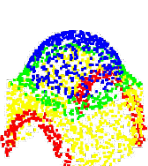
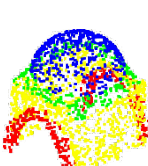
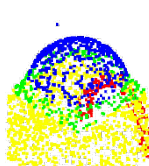
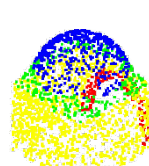
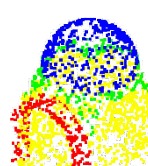
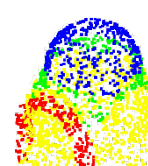
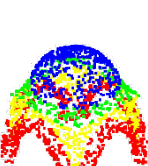
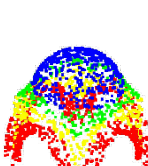
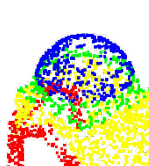
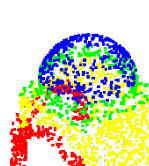
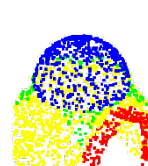
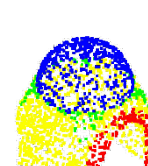
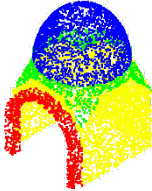
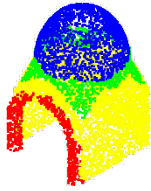
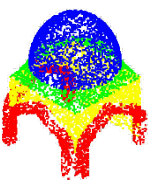
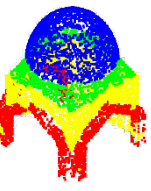
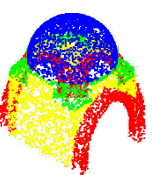
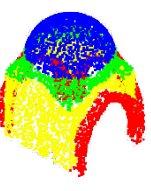
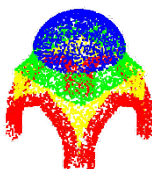
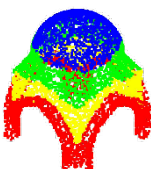
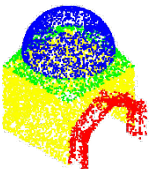
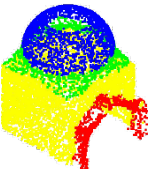
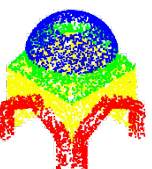
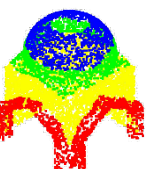
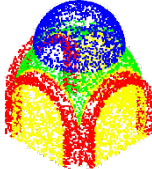
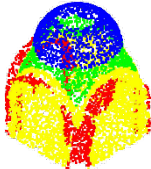
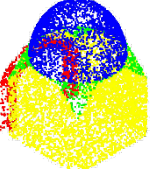
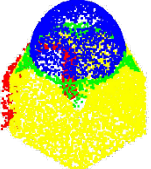
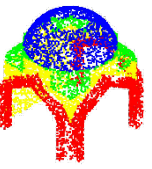
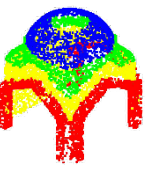
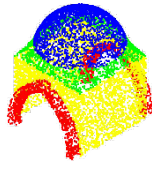
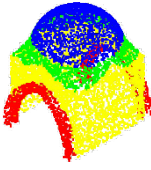
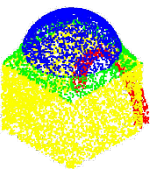
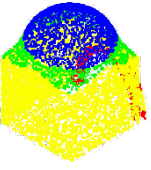
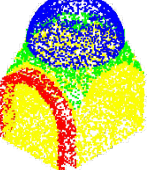
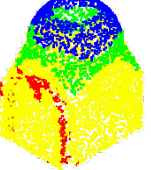
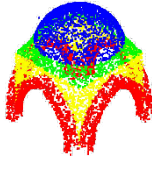
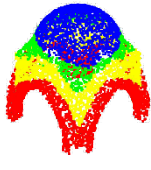
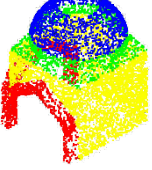
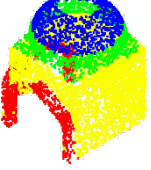
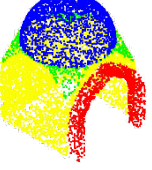
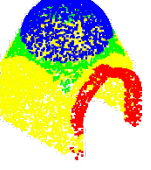
Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

Table 3.15 : Results of DGCNN on HDD, 10k points.

Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

The network failed to understand partial model where the dome systems have two classes instead of four (Tables 3.16, 3.17, 3.18, and 3.19). Particularly, the DGCNN failed to identify the boundaries of the dome geometry of partial models by contrast with the synthetic data set and full data batch. On the other hand, it failed to decode the relations of the points in more crowded cases that are represented as 10k models. The additional points preserve the information of noise and fine-grained details that are not included in the training model so the network fails to generalize learned relations using the graph networks. Specifically, the examples that include only transition elements and dome failed to comprehend the limits of the parts. Confusion matrices in Figure 3.5 show that in 2k partial real-world tests, a high ratio of dome points are identified as transition elements. Also, DGCNN tends to identify transition elements as arches in 2k partial tests. Particularly in partial 2k, full models arch points are identified as wall and transition elements. Figure 3.6 shows correlated ratios of confusion in 10k with 2k examples with a fewer ratio of successful labeling in most of the classes. These results show us full models are suitable for identifying dome geometry with high accuracy while in partial models dome shapes could not be identified clearly. This result is related to the definition of the HDD because the synthetic data set only includes ideal full geometries. And all the contextual knowledge defined in ideal cases. So, this idealization of the dome systems causes a problem in DGCNN when the point cloud model is split and does not include all the elements of dome systems. Even though the definition of simple geometries in a low level of detail used in training represents the parts of a dome system it fails to improve quality in 10k cases.

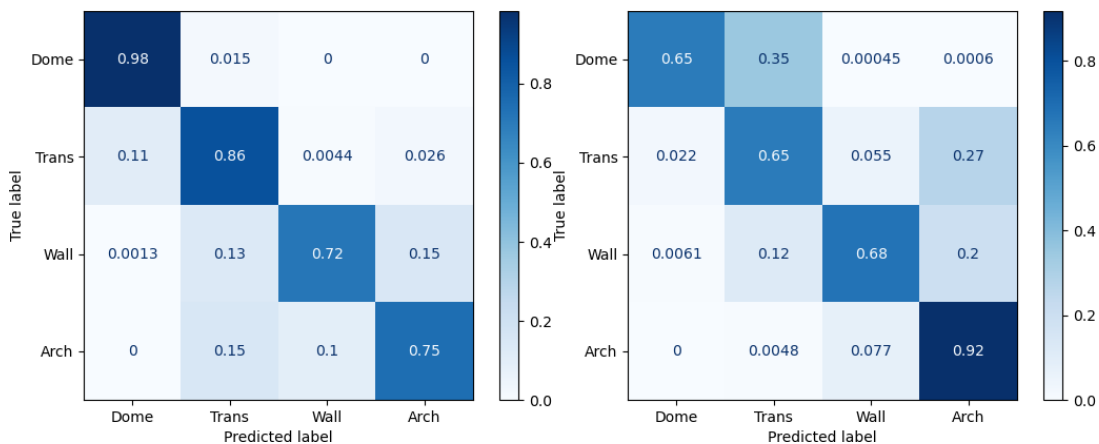


Figure 3.5 : Confusion matrices of DGCNN (left) full and (right) partial real-world models in 2k.

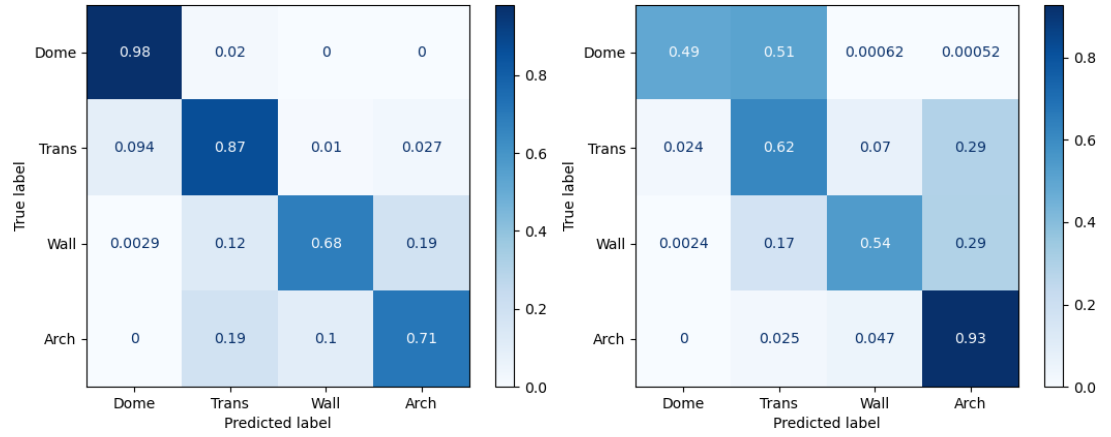


Figure 3.6 : Confusion matrices of DGCNN (left) full and (right) partial real-world models in 10k.

Table 3.16 : Results of DGCNN on full models, 2k points.

Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.17 : Results of DGCNN on full models, 10k points.

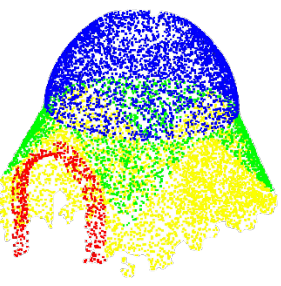
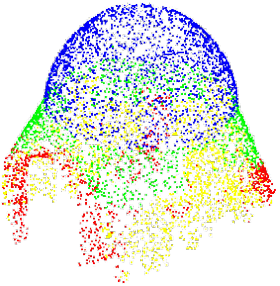
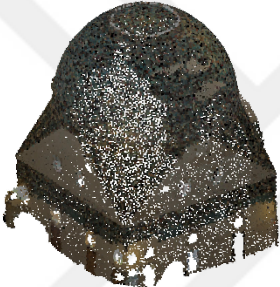
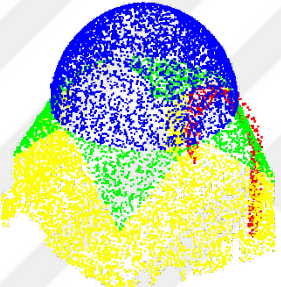
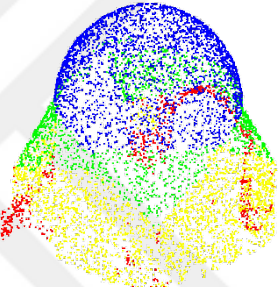
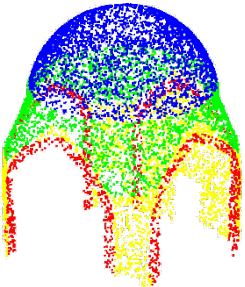
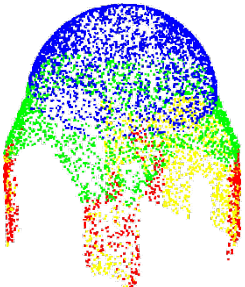
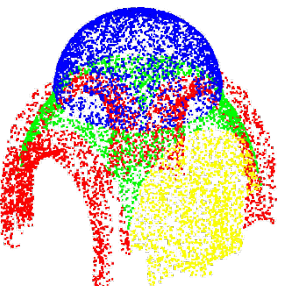
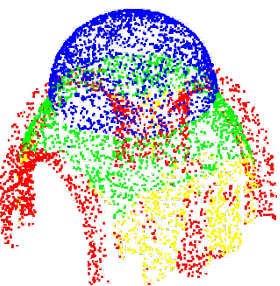
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.18 : Results of DGCNN on partial models, 2k points.

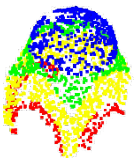
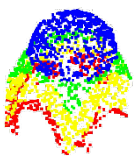
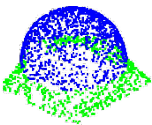
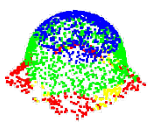
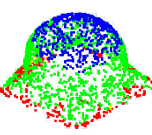
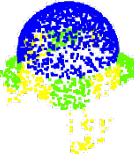
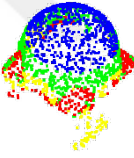
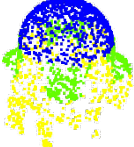
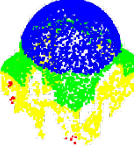
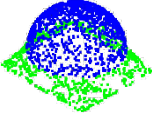
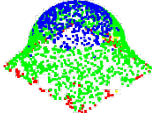
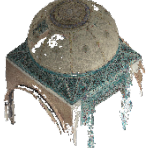
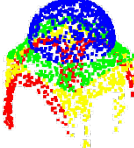
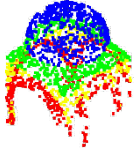
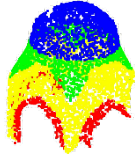
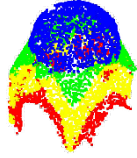
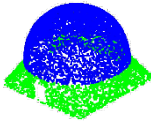
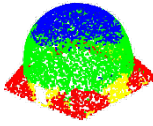
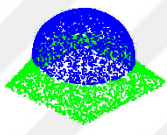
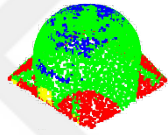
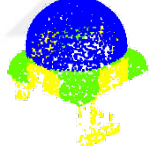
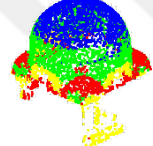
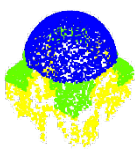
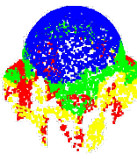
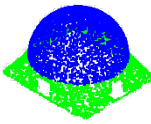
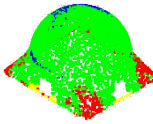
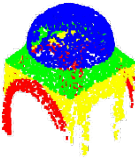
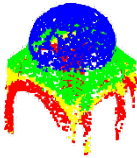
Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

Table 3.19 : Results of DGCNN on partial models, 10k points.

Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

3.4.3 RSNET

RSNET performed well in the synthetic data set. The three-dimensional sequential data interpretation improves the quality where the definition is similar to the training data set. Limited differences occurred only in the boundaries of the different parts of the dome system (Tables 3.20 and 3.21). Moreover, both the 2k and 10k samples resulted in similar qualities. The results show that the deep learning model had some of the most problems in identifying the boundaries of the wall geometries. Figure 3.7 represent that RSNET identifies all the classes with minimal confusion ratio except a relatively low accuracy in transition element points of the HDD test set.

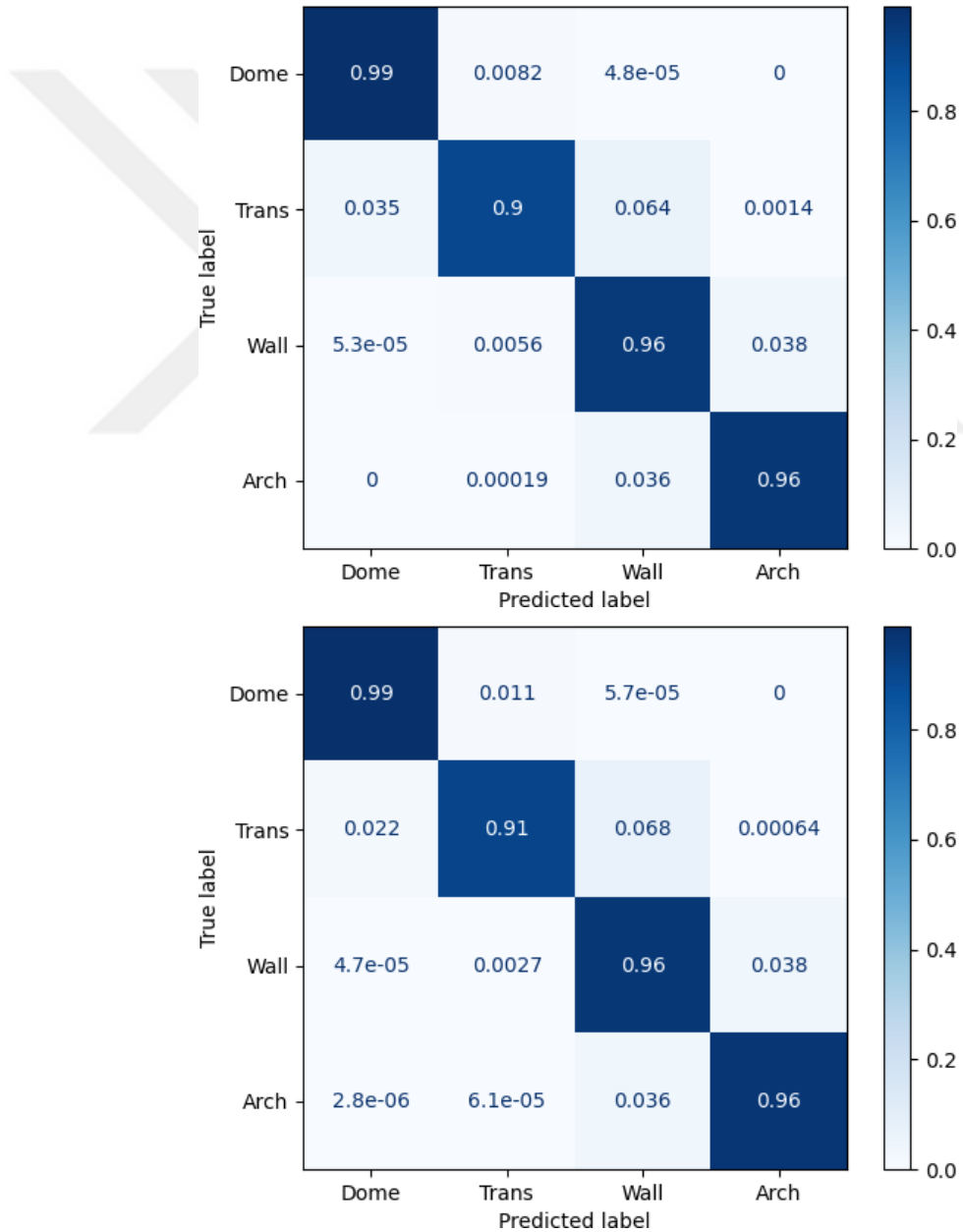


Figure 3.7 : Confusion matrices of RSNET on HDD, 2k (top) and 10k (bottom).

Table 3.20 : Results of RSNET on HDD, 2k points.

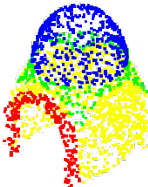
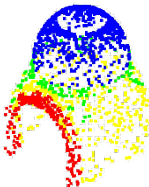
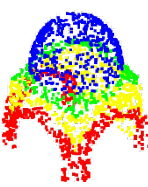
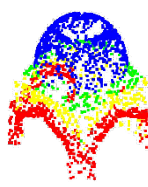
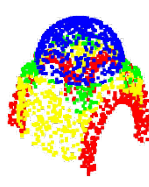
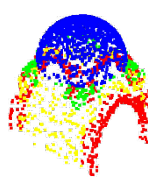
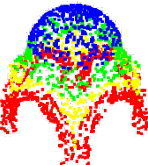
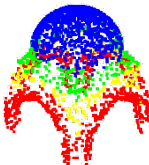
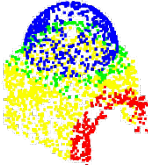
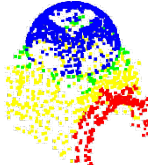
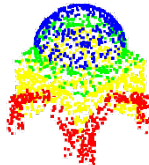
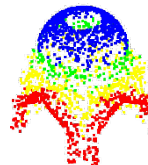
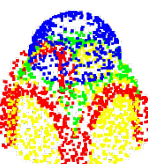
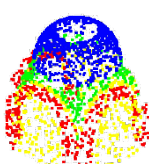
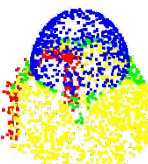
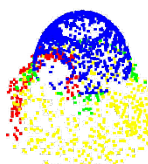
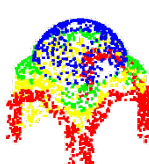
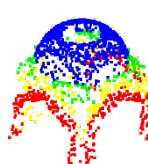
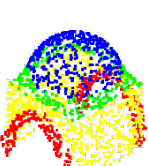
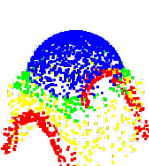
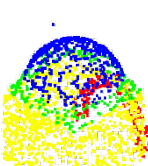
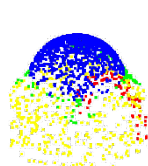
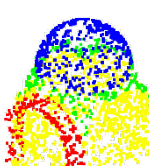
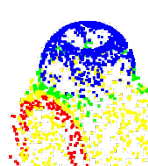
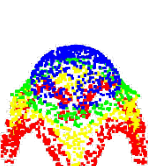
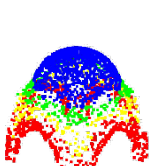
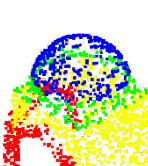
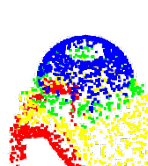
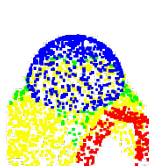
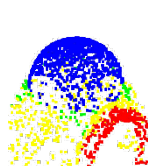
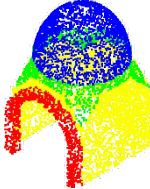
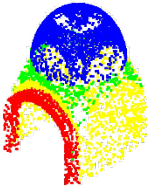
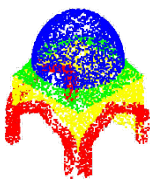
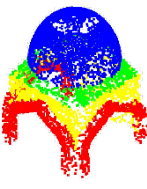
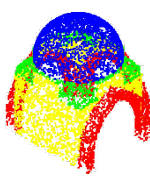
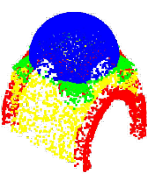
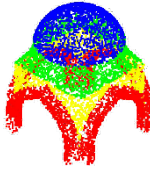
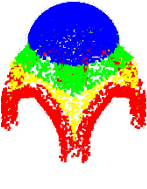
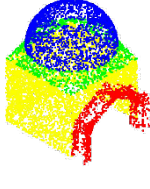
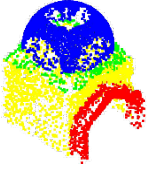
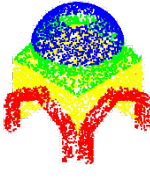
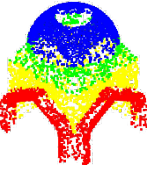
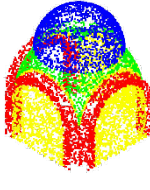
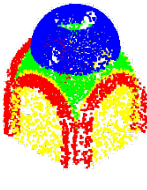
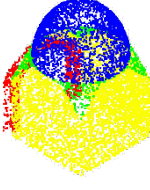
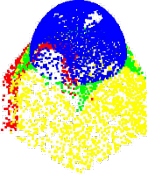
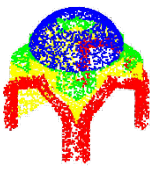
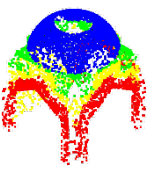
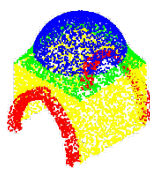
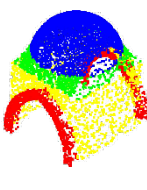
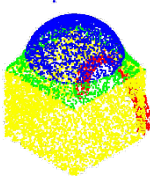
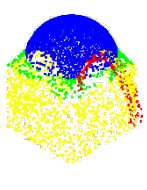
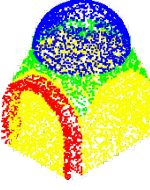
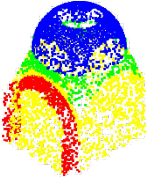
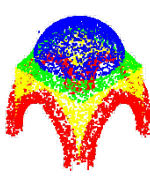
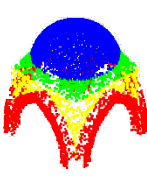
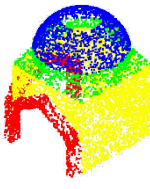
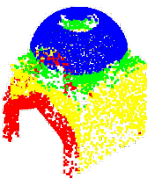
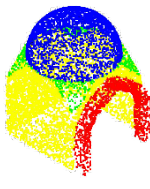
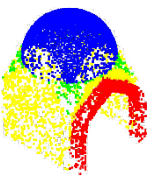
Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

Table 3.21 : Results of RSNET on HDD, 10k points.

Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

The real-world examination with RSNET shows that the deep neural network shows the signs of the problem of identifying only similar models to the training data set and fails to generalize the learn properties from the synthetic data set. Confusion matrices

in Figures 3.8 and 3.9 show that RSNET fails to identify the labels of the points in partial sets except for the dome and arch class. The results of full models are more accurate in general, however, there is significant confusion in classifying wall points. The results of the full models show that the RSNET network is capable of detecting the positions of the parts however the accuracy level is not acceptable in most cases (Tables 3.22 and 3.23). Partial data examinations resulted in negative results, the network could not generalize the learned data and predict when some elements or some parts of the elements do not exist (Tables 3.24 and 3.25). Even though the approach of consecutive interpretation of point batches in the models is promising in a sense of continuous relation-based identification of architectural elements, the RSNET network has some limitations that negatively impact the results. Enriching the training data set with varied and accurate representations of challenging cases might address these limitations.

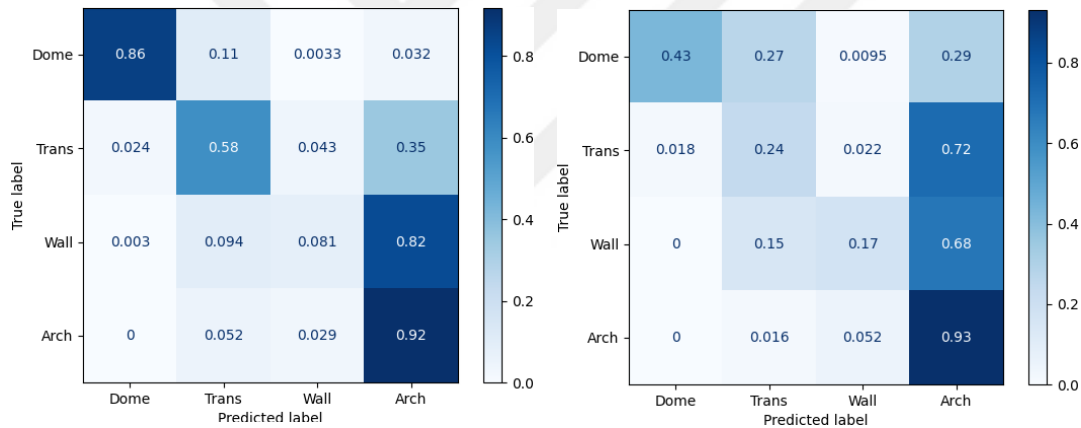


Figure 3.8 : Confusion matrices of RSNET (left) full and (right) partial real-world models in 2k.

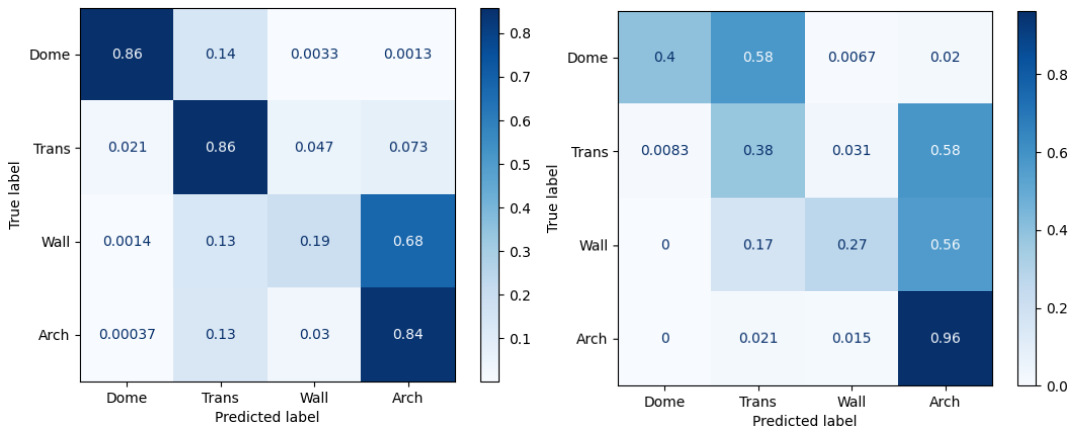


Figure 3.9 : Confusion matrices of RSNET (left) full and (right) partial real-world models in 10k.

Table 3.22 : Results of RSNET on full models, 2k points.

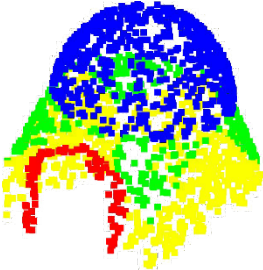
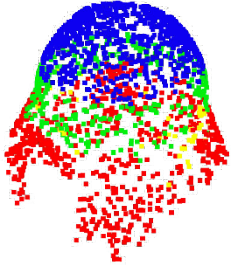
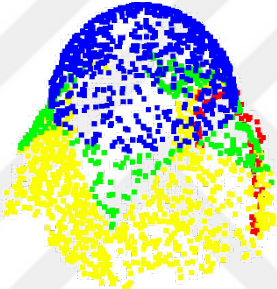
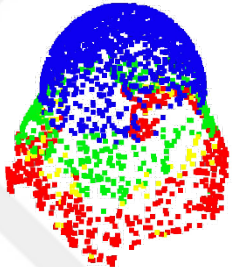
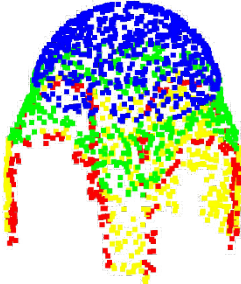
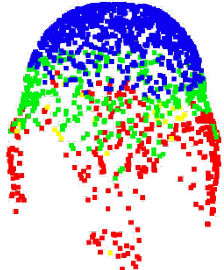
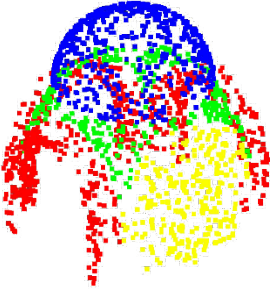
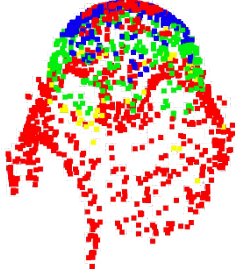
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.23 : Results of RSNET on full models, 10k points.

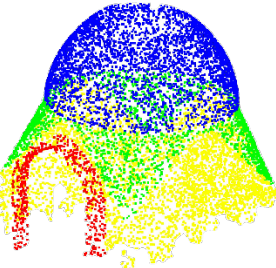
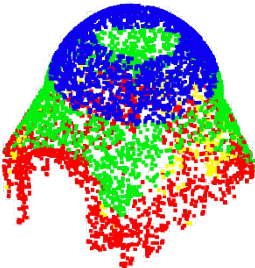
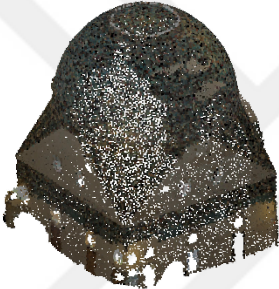
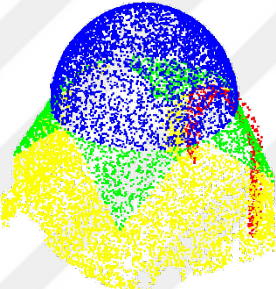
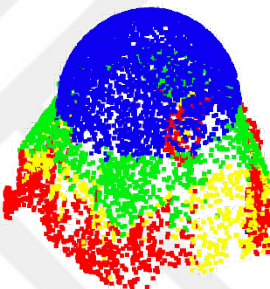
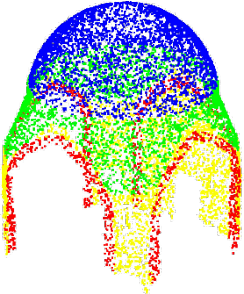
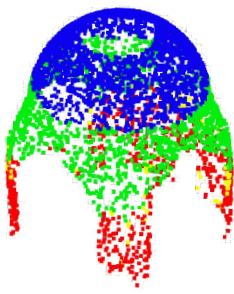
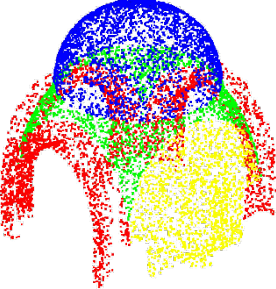
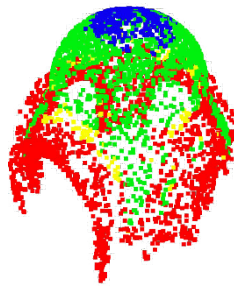
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.24 : Results of RSNET on partial models, 2k points.

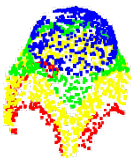
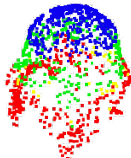
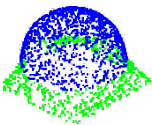
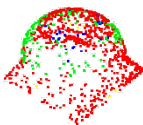
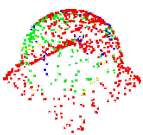
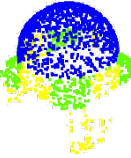
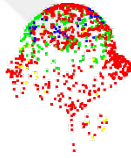
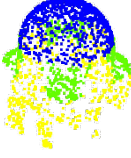
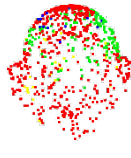
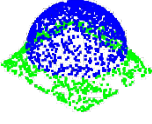
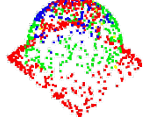
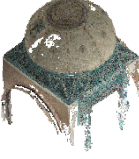
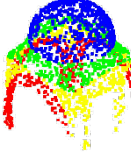
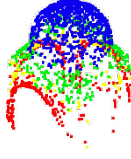
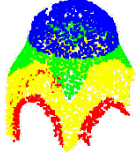
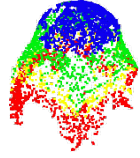
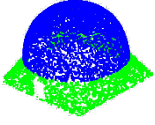
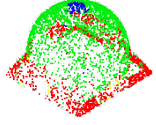
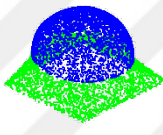
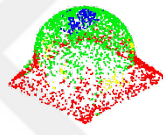
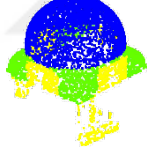
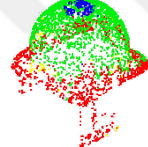
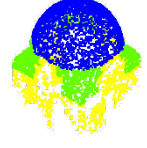
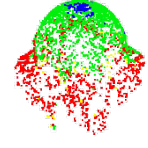
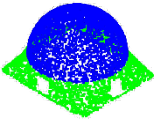
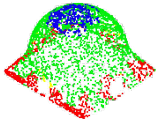
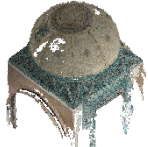
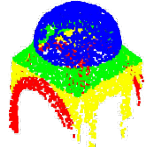
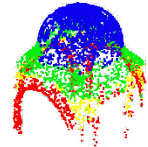
Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

Table 3.25 : Results of RSNET on partial models, 10k points.

Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

3.4.4 Point2Sequence

The RNN-based network that is supported with an attention mechanism performed on average with the synthetic data. Both the examinations with 2k and 10k resulted in similar qualities and the deep neural network performed well both visually and numerically. Confusion matrices in Figure 3.10 show that P2S identified all the classes with minimal error except the confusion in the transition elements. The sequential interpretation of the data batches in returned fine-grained local resolutions and meaningful global relations of the parts of the domes systems (Tables 3.26 and 3.27).

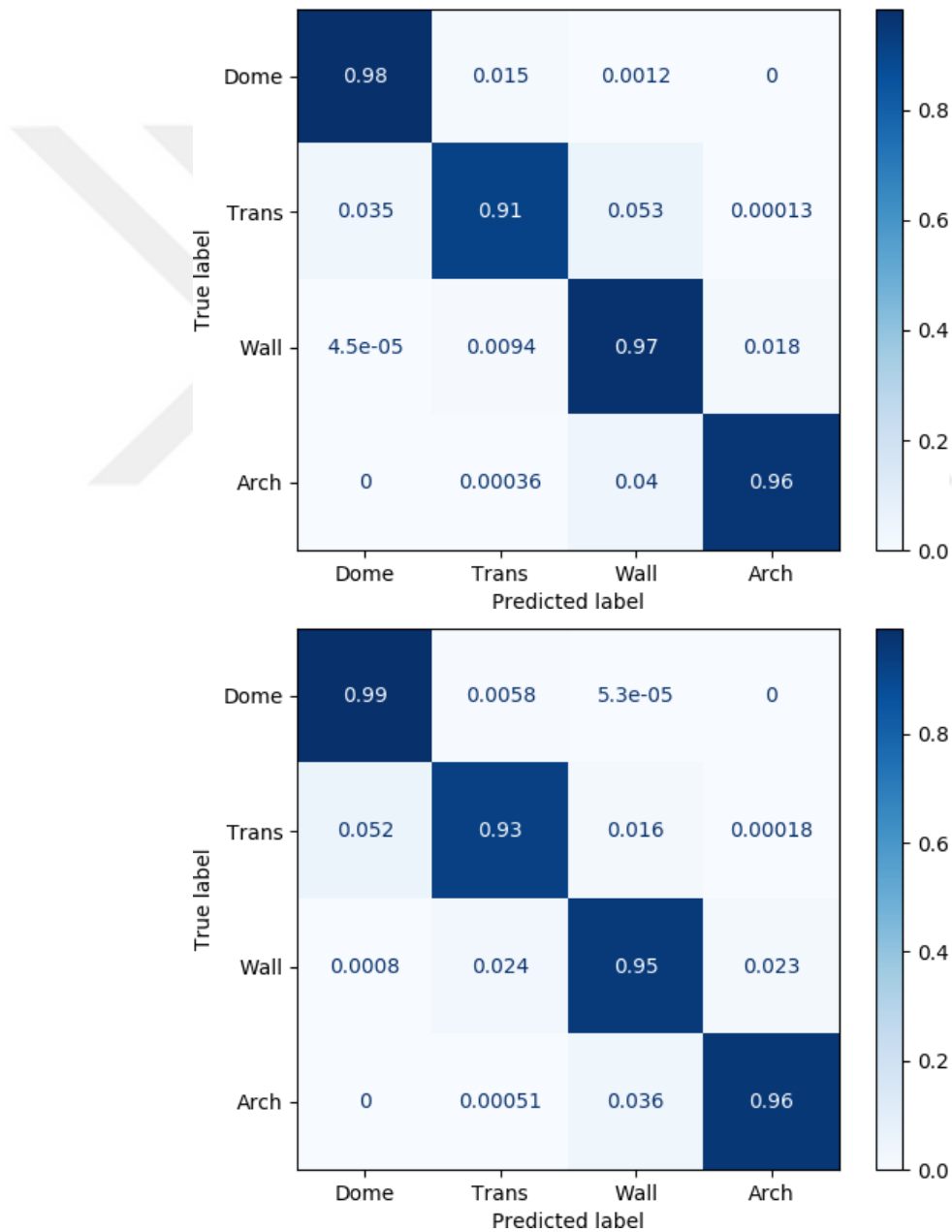


Figure 3.10 : Confusion matrices of Point2Sequence on HDD, 2k (top) and 10k (bottom).

Table 3.26 : Results of P2S on HDD, 2k points.

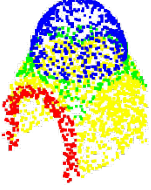
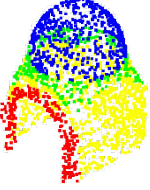
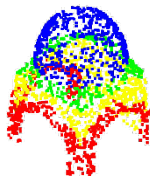
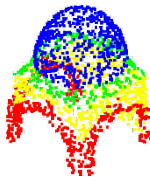
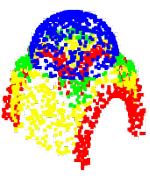
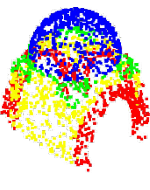
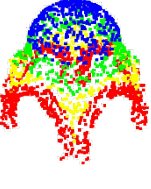
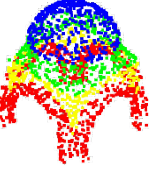
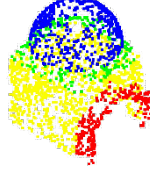
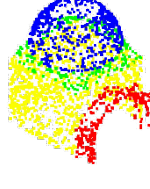
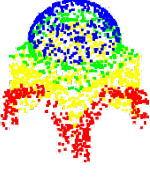
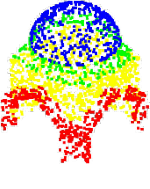
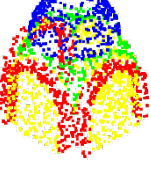
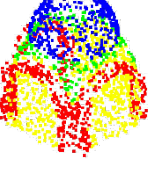
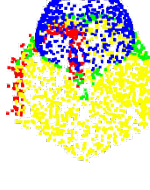
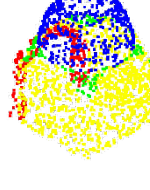
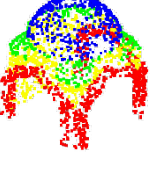
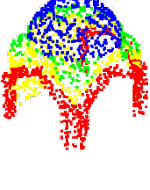
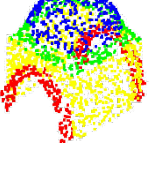
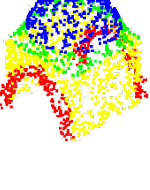
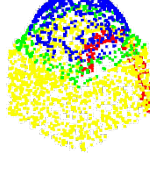
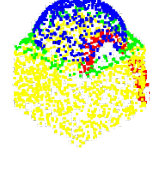
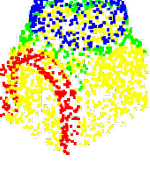
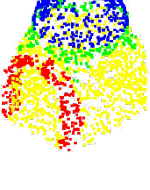
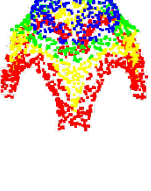
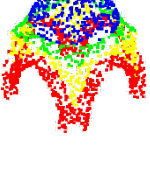
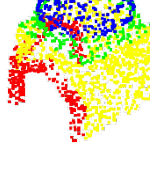
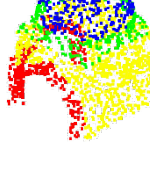
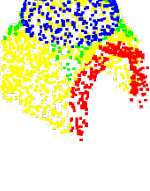
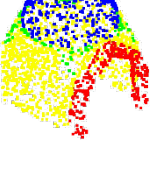
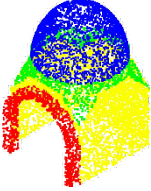
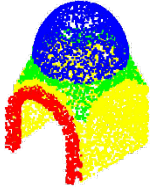
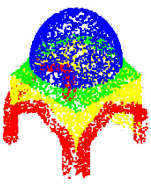
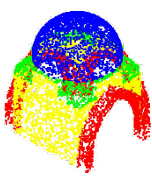
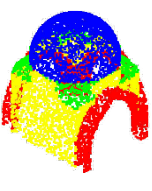
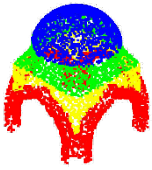
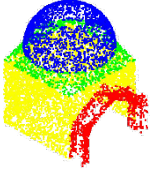
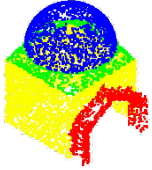
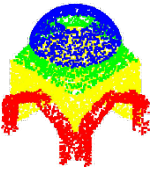
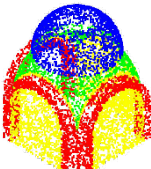
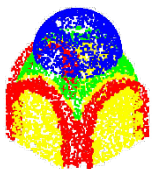
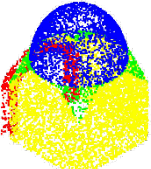
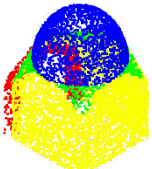
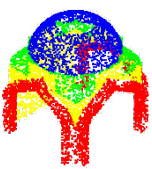
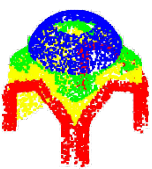
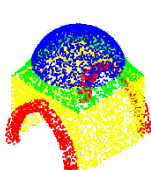
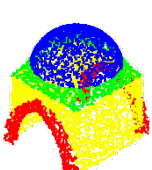
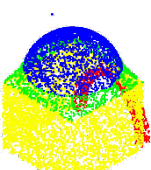
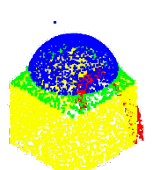
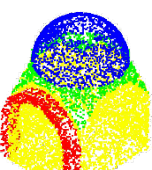
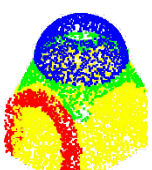
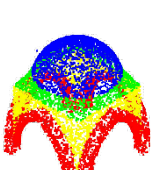
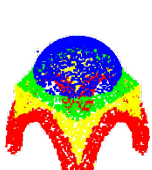
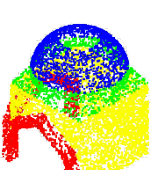
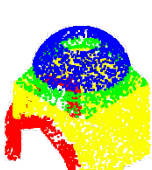
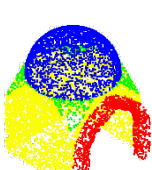
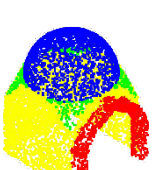
Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

Table 3.27 : Results of P2S on HDD, 10k points.

Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

The Point2Sequence network identifies the boundaries of the dome geometries with a small margin of error in both populations of the real-world data set. Confusion matrices in Figure 3.11 reflect that in 2k real-world test sets the dome points were identified with minimal error. Also, P2S identified other classes in full models with approximately a 25% error rate while the partial model has a more confusion ratio. Also, confusion matrices in Figure 3.12 show that P2S performed better in the 10k partial segmentation task, particularly in the arch class. However, the 2k full model set fails to identify the limits of the arch geometry (Table 3.28). The trained model is

sensitive to the changes in the surface deviations and lacks some parts of the model. The approximate location and borders of the transition elements are recognized but with a low level of local qualities. Additionally in the KM_M model with 10k points, an extraordinary arrangement of the points is labeled as the arch (Table 3.29). Since data is interpreted consecutively it is possible that misinterpretations also impact the following predictions. In the partial models, P2S fails to capture worthwhile outcomes, particularly in the cases where the models do not have wall and arch points (Tables 3.30 and 3.31). Even though it internalizes the sequential relations where all the elements are presents, the model loses all the contextual sensitivity and identifies elements with un-realistic part-whole relations and positions. Also, this drawback shows that the trained sequential model is highly dependent on the training data that consist of all the architectural element labels in the scene for all the models. Additionally, the distribution of the labels in the HDD and real-world full models are similar while partial models have an unbalanced distribution of the classes in the scenes.

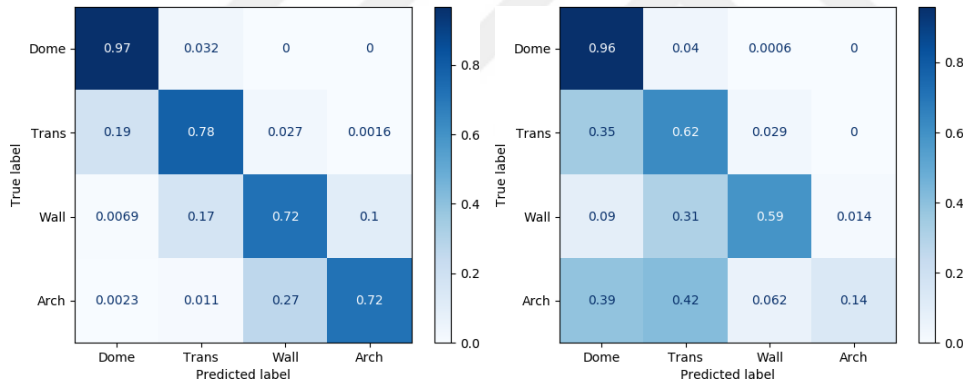


Figure 3.11 : Confusion matrices of P2S (left) full and (right) partial real-world models in 2k.

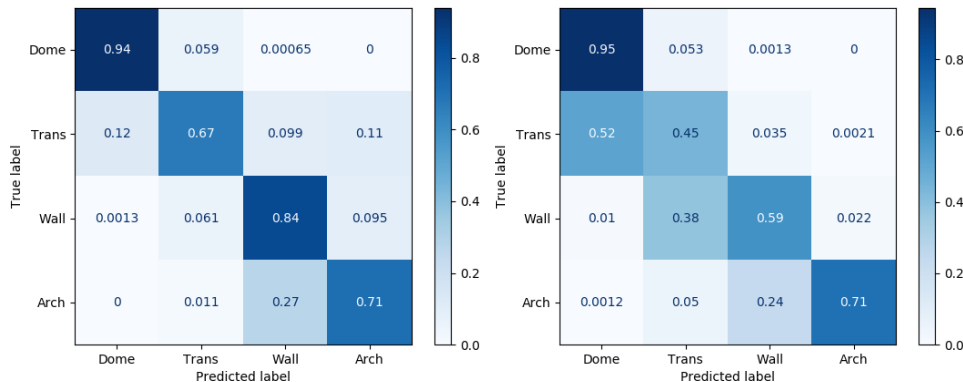


Figure 3.12 : Confusion matrices of P2S (left) full and (right) partial real-world models in 10k.

Table 3.28 : Results of P2S on full models, 2k points.

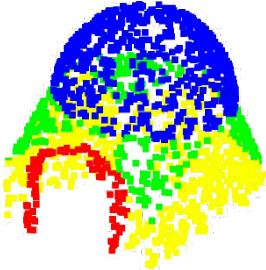
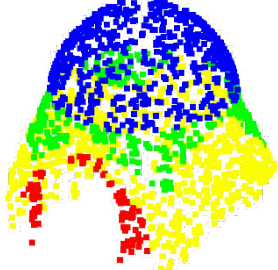
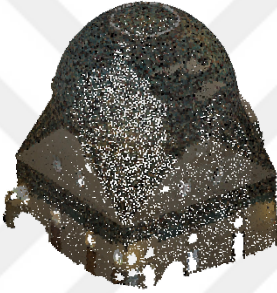
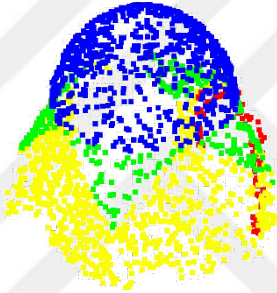
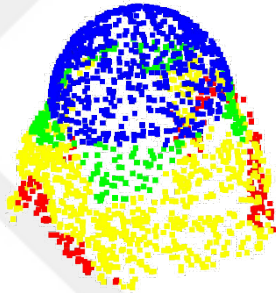
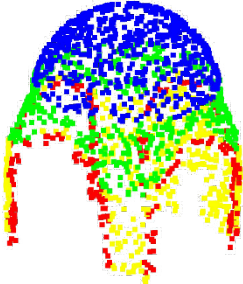
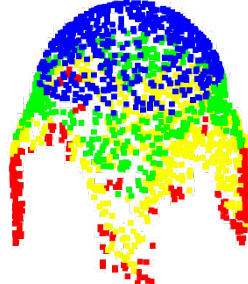
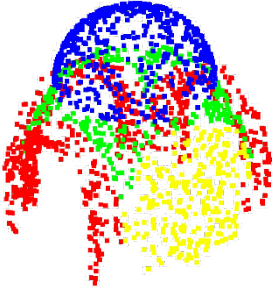
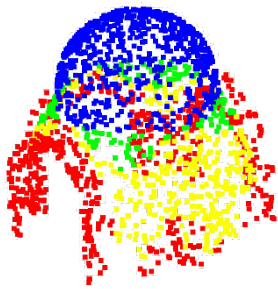
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.29 : Results of P2S on full models, 10k points.

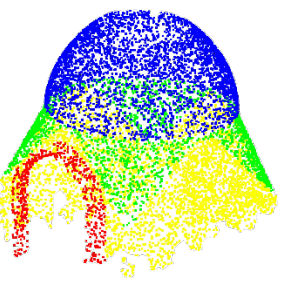
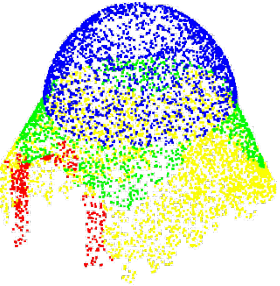
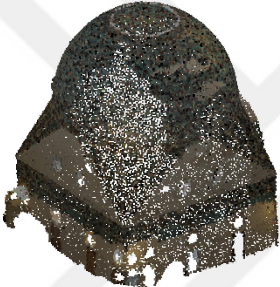
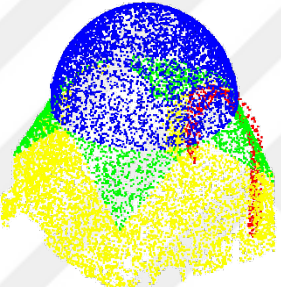
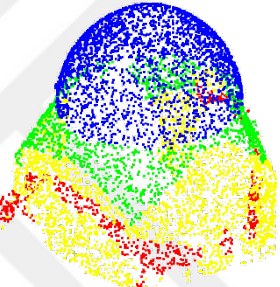
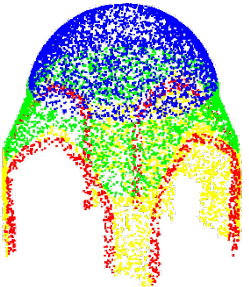
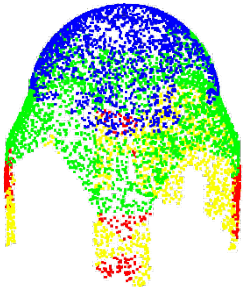
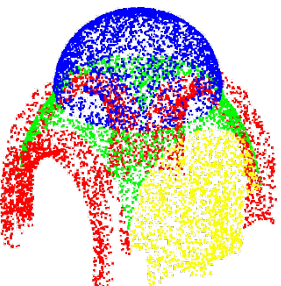
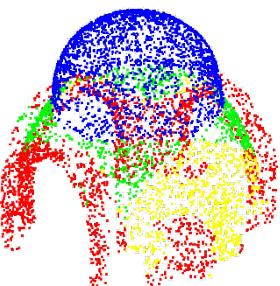
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.30 : Results of P2S on partial models, 2k points.

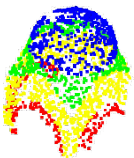
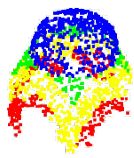
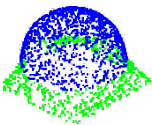
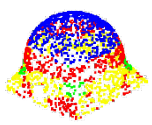
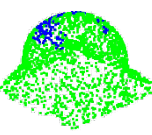
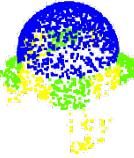
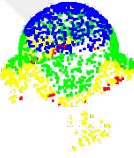
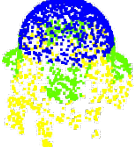
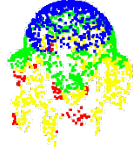
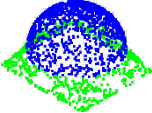
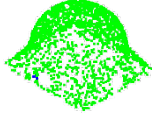
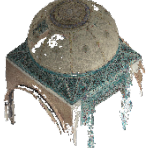
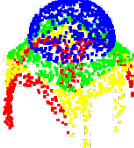
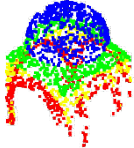
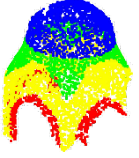
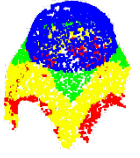
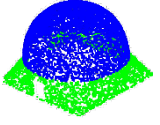
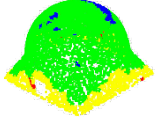
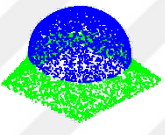
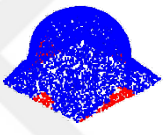
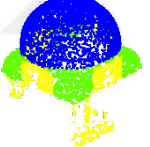
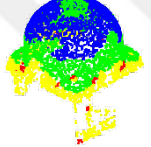
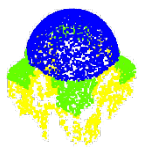
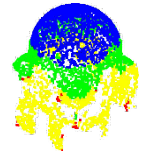
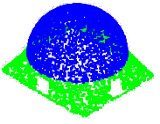
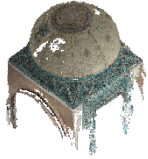
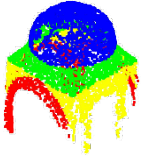
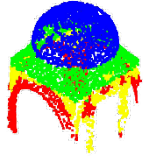
Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

Table 3.31 : Results of P2S on partial models, 10k points.

Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

3.4.5 PointTransformer

The transformer-based architecture labels the point set in the HDD test set with high accuracy. While global relations are examined well, in some examples boundaries of the parts are identified with a small margin of error. Confusion matrices in Figure 3.13 show that PTr tested on 2k HDD examples tend to mislabel the small number of transition elements while 10k HDD examination labels point in all the classes with high accuracy. Specifically, the point at the limit of the transition element and the wall geometries are mixed in the prediction step (Tables 3.32 and 3.33).

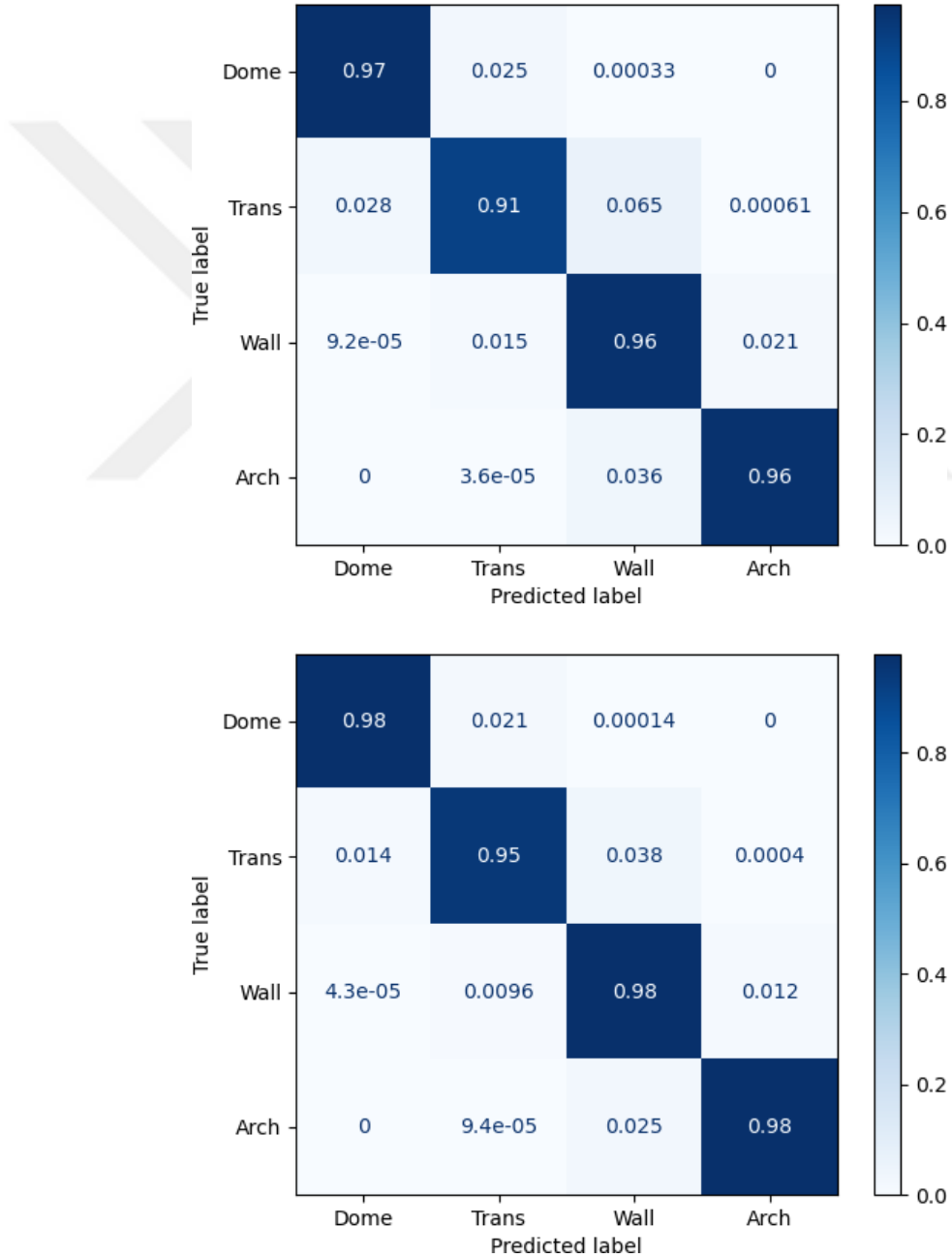


Figure 3.13 : Confusion matrices of PTr on HDD, 2k (top), and 10k (bottom).

Table 3.32 : Results of PTr on HDD, 2k points.

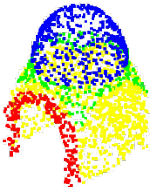
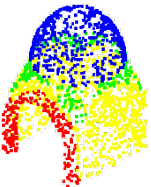
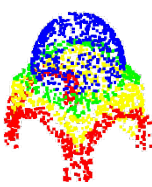
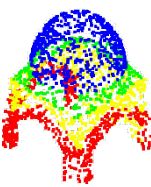
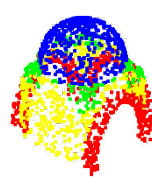
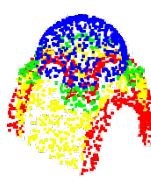
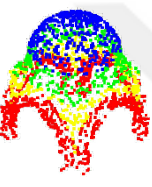
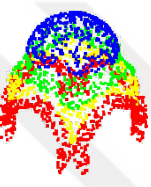
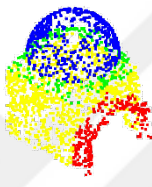
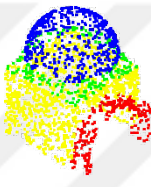
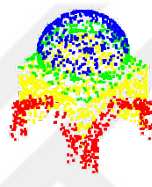
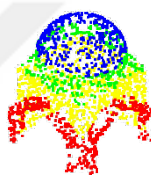
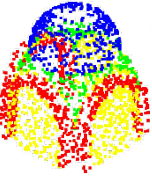
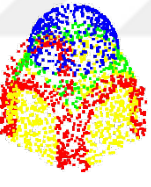
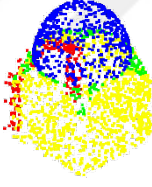
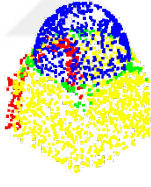
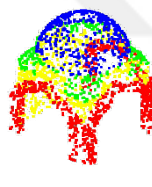
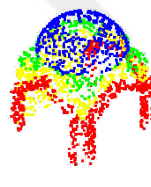
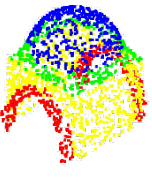
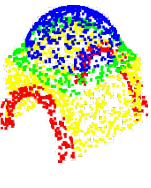
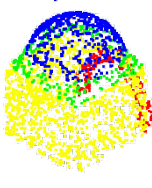
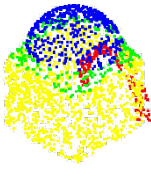
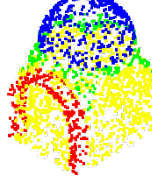
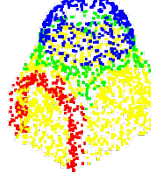
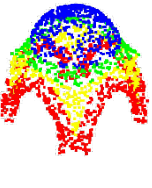
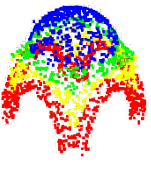
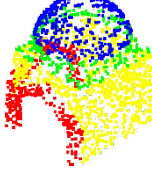
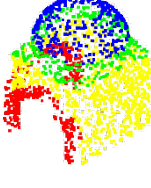
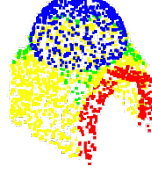
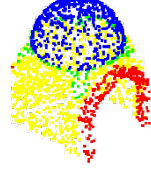
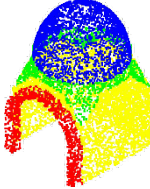
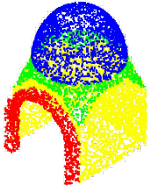
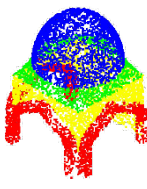
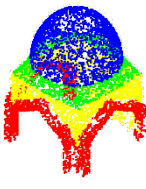
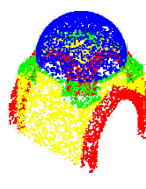
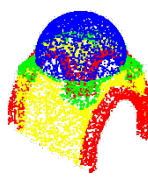
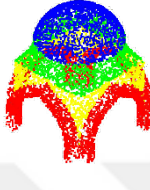
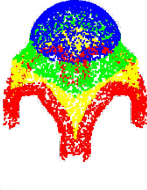
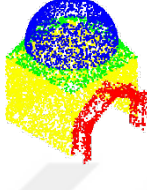
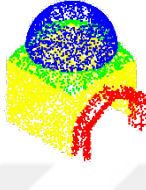
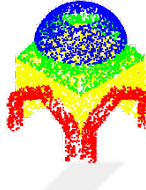
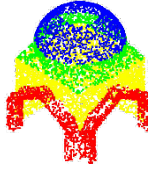
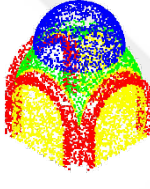
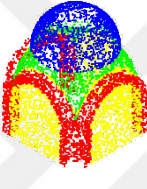
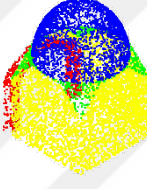
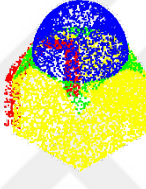
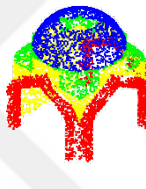
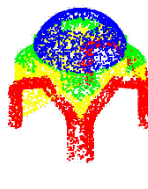
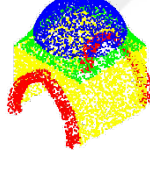
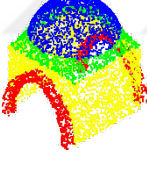
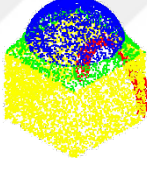
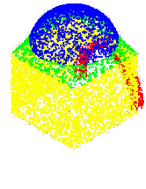
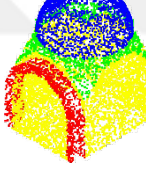
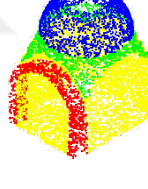
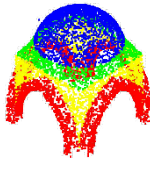
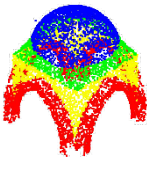
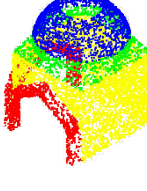
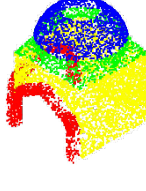
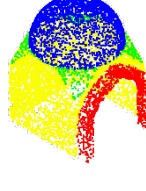
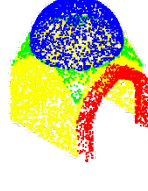
Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

Table 3.33 : Results of PTr on HDD, 10k points.

Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

The PointTransformer network successfully identifies the positions and order of the elements in three-dimensional space. Nevertheless, it is unsuccessful at capturing the precise limits of the parts. Confusion matrices in Figure 3.14 show that PTr confuses transition elements with dome and wall classes. Also, it mixes arch points with wall and transition element points in a real-world 2k full test set. In partial 2k real-world examinations, PTr confused the transition element and wall points while mixing wall points with transition elements and arch. On the other hand, full 10k real-world models failed to identify elements of the domes systems except for the dome points while

partial 10k real-world cases resulted better in identifying the arch geometries. In full model examinations, the scenes with 2k points (Table 3.34) resulted better than the scenes with 10k points (Table 3.35) due to the simplicity of the model. Particularly, in the SA_H model in 2k transition area is identified very well despite the low accuracy in detecting the narrow arch geometries. Apart from the other networks, the transformer-based model resulted in inconsistent outcomes in detecting the arch geometries in full models. On the other hand, the results of the partial models show that the consecutive relations of the parts are decoded (Tables 3.36 and 3.37). Especially in most cases, the boundaries of the dome geometry are recognized approximately. The gaps in the point clouds cause wrong assessments in scene understanding because the geometric features are affected by these kinds of properties. For example, the cavity of the mihrab in the HK case and the openings that stem from the low quality of the collected data confuse the prediction task.

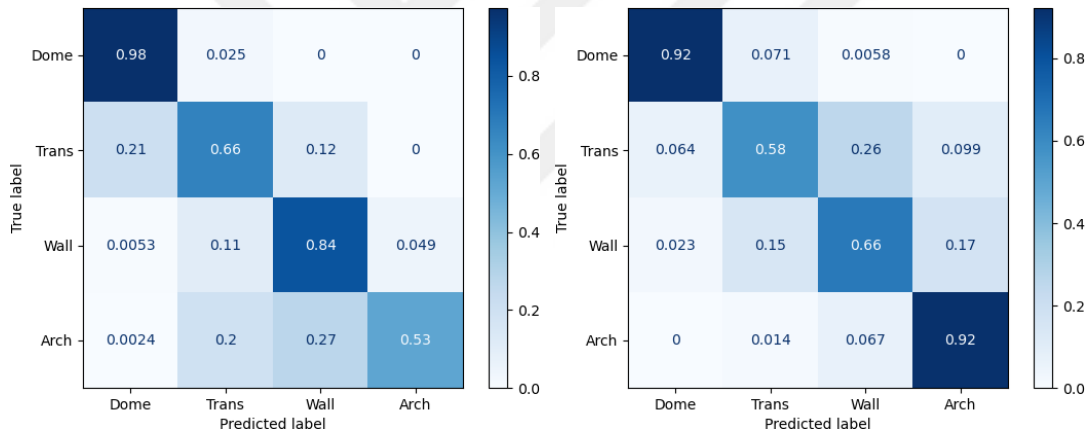


Figure 3.14 : Confusion matrices of PTr (left) full and (right) partial real-world models in 2k.

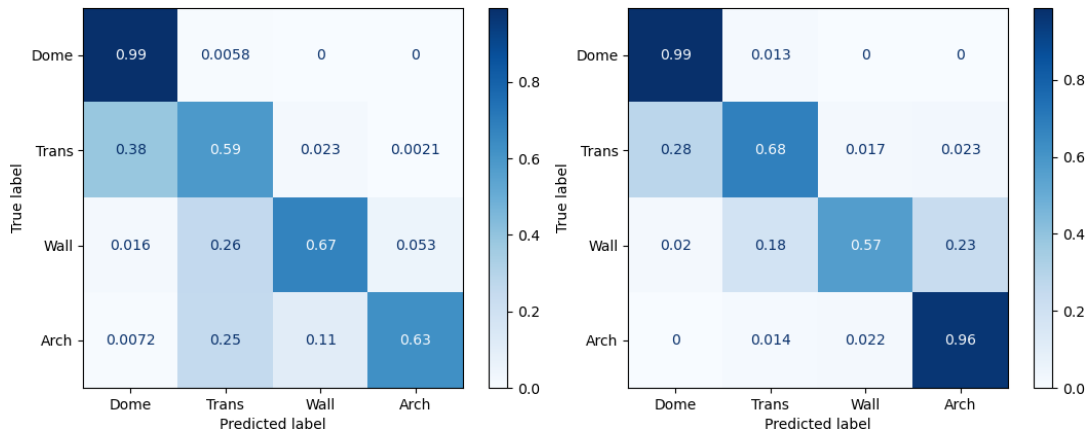


Figure 3.15 : Confusion matrices of PTr (left) full and (right) partial real-world models in 10k.

Table 3.34 : Results of PTR on full models, 2k points.

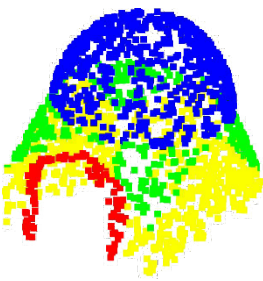
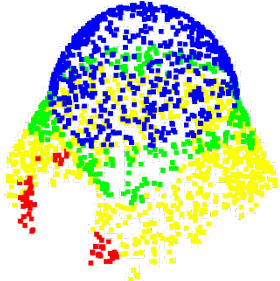
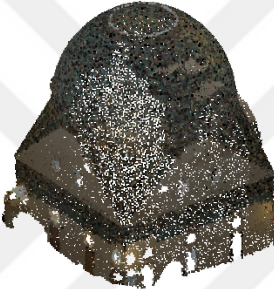
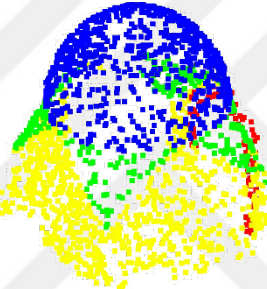
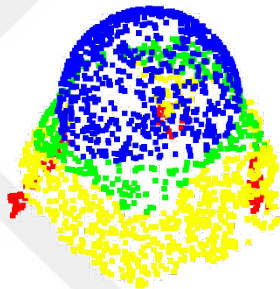
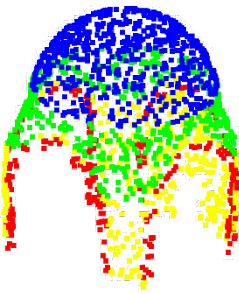
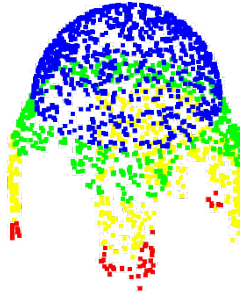
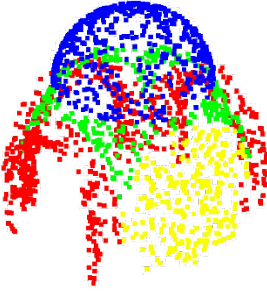
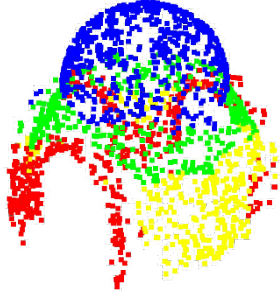
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.35 : Results of PTr on full models, 10k points.

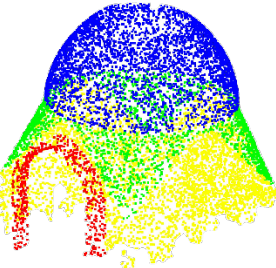
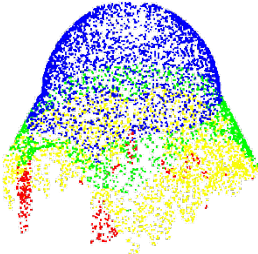
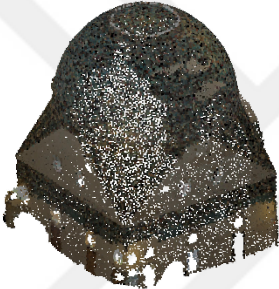
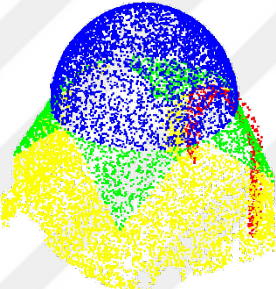
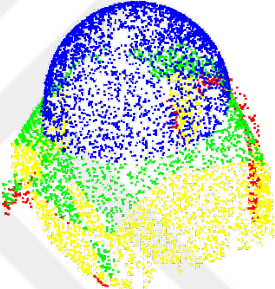
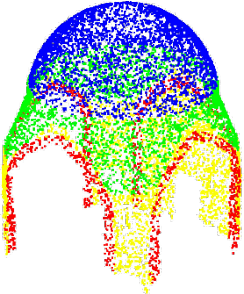
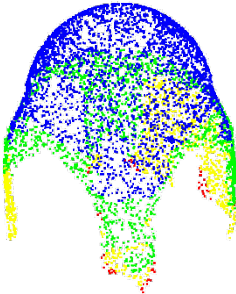
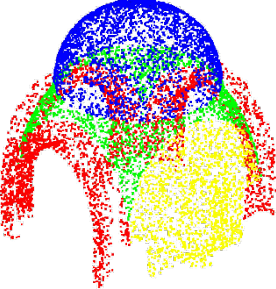
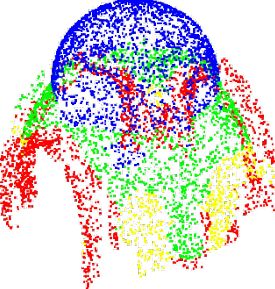
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.36 : Results of PTr on partial models, 2k points.

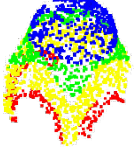
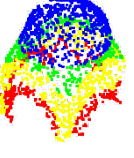
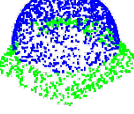
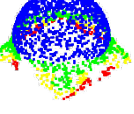
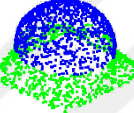
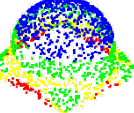
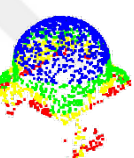
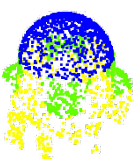
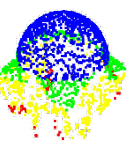
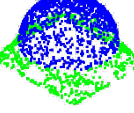
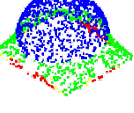
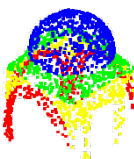
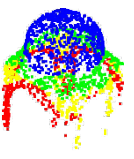
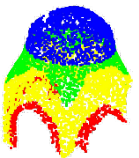
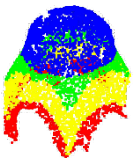
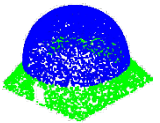
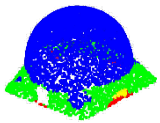
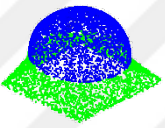
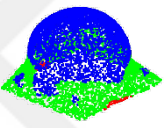
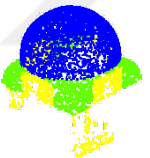
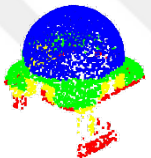
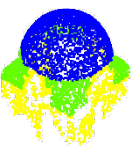
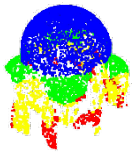
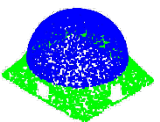
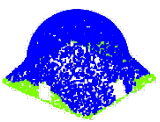
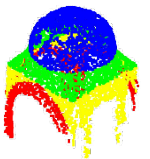
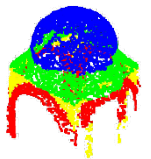
Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

Table 3.37 : Results of PTr on partial models, 10k points.

Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

3.4.6 Point-BERT

The state-of-the-art method, Point-BERT(PB) labels the point in the synthetic data set with the best accuracy in the exemplified networks. All the parts are identified almost with minimal errors in the boundaries of the elements. The confusion matrices in Figure 3.16 represents that all of the classes in both the 2k and 10k HDD test resulted in minimal confusion. In the segmentation process, both the 2k and 10k samples resulted in similar qualities visually and numerically (Tables 3.38 and 3.39). Unlike the former networks, the network has an initial training step that focuses on guessing the masked partitions of the provided data set. This additional feature targets a better understanding of unseen parts. Also, the network aims to decode the scenes by exploiting both sequential and global relations using the bidirectional architecture of the deep learning network.

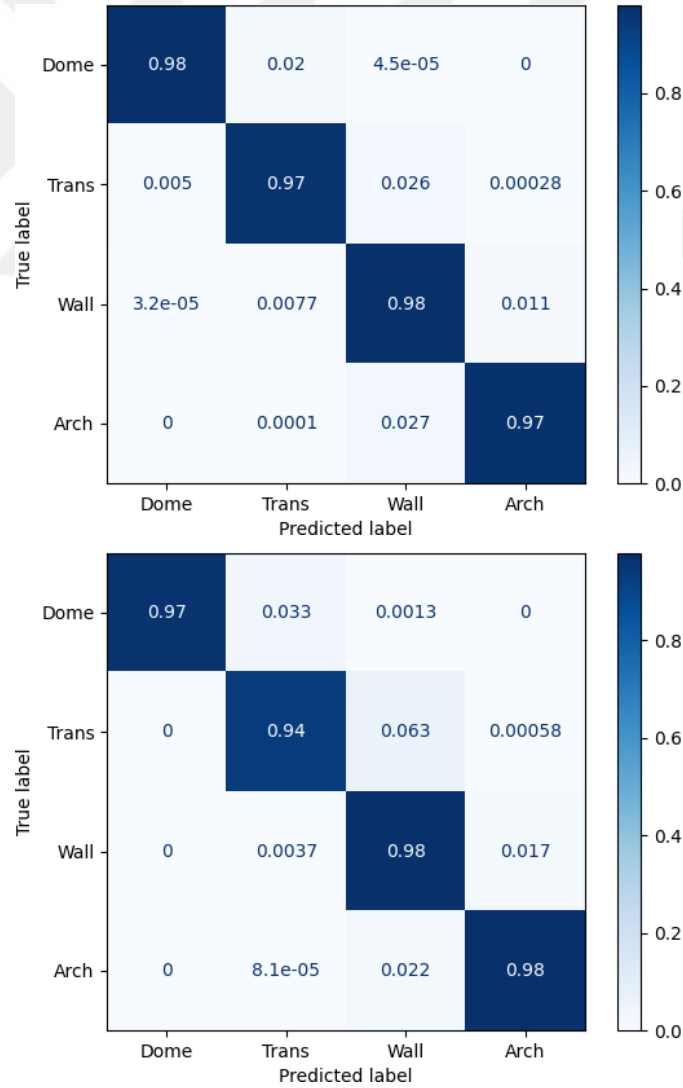


Figure 3.16 : Confusion matrices of PB on HDD, 2k (top) and 10k (bottom).

Table 3.38 : Results of PB on HDD, 2k points.

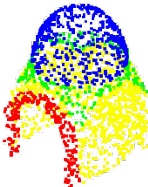
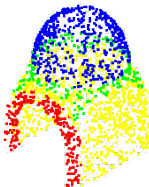
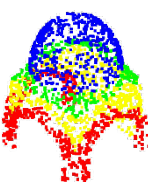
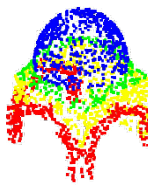
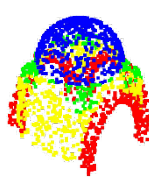
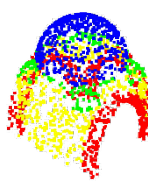
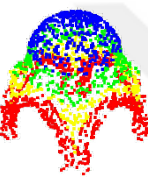
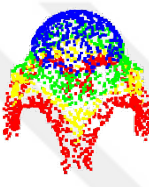
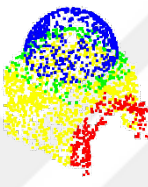
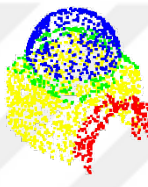
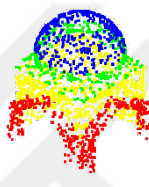
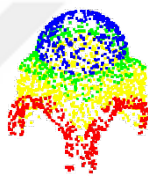
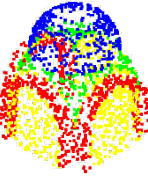
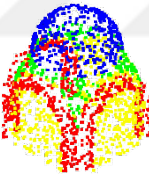
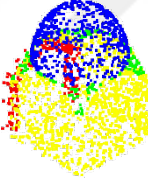
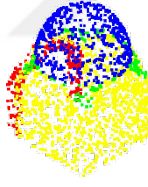
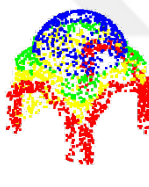
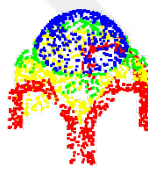
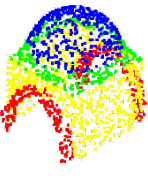
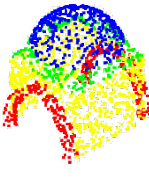
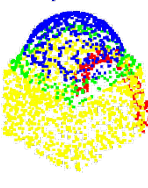
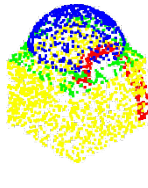
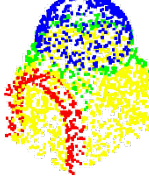
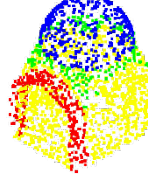
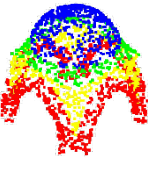
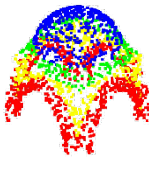
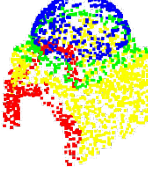
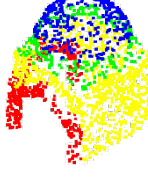
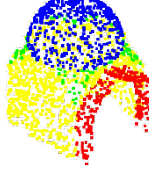
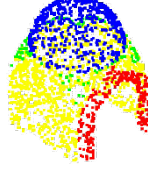
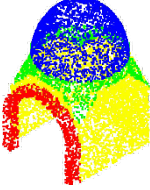
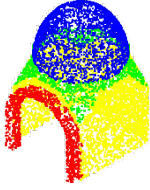
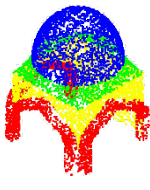
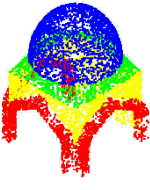
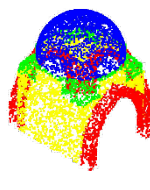
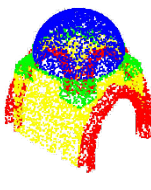
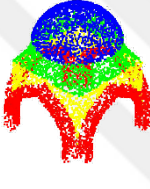
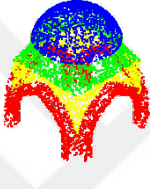
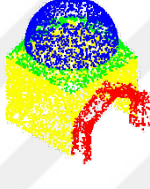
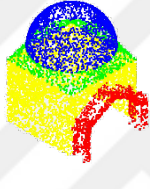
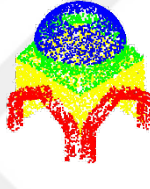
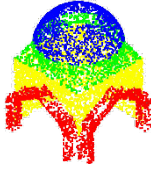
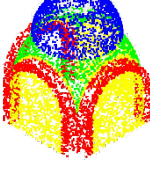
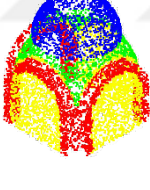
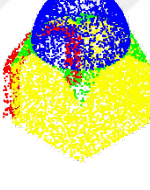
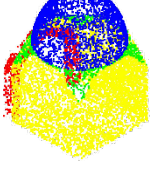
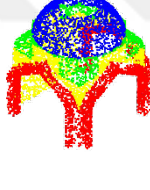
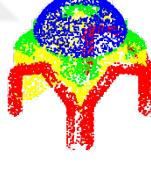
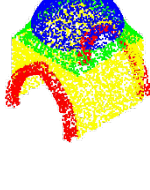
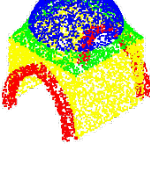
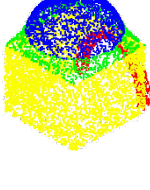
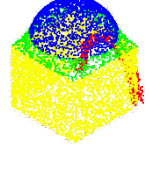
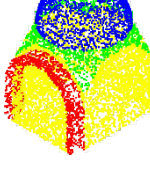
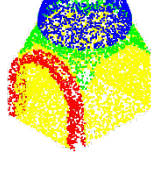
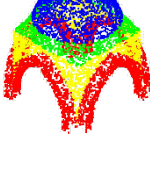
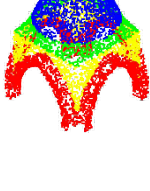
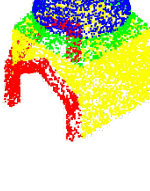
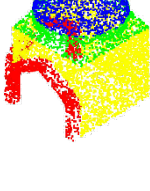
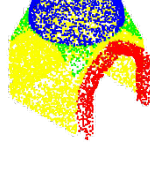
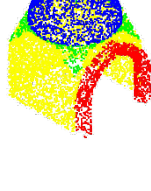
Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

Table 3.39 : Results of PB on HDD, 10k points.

Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth	Prediction
					
					
					
					
					

Even though the synthetic test results are highly promising, Point-BERT cannot generalize the learned information on the real-world models. Particularly, it failed the semantic segmentation of the partial data due to the tendency of finding all types of elements even if one or two of them does not exist in the scene (Tables 3.40, 3.41, 3.42, and 3.43). The results of the examinations on full models show that the deep learning model recognizes slanted and circular drum areas in the IM_M, KM_M, and SA_H cases even if the training models do not have a definition for circular and slanted drums. Although the aforementioned networks interpret the sloped drums as a part of the dome geometry the Point-BERT network assesses the drum zones that are not vertical and not placed on the polygon base as a part of transition elements. Even though, the acquired result is not ideal according to the definition of HDD it is significant that Point-BERT identified a different and unseen type of drum geometries. Additionally, results of the partial data cases show that the network internalized the relations of the items regarding the context better in 2k cases rather than 10k models. The confusion matrices in Figure 3.17 demonstrates that the wall and arch points are mixed while some transition elements are labeled as wall class in 2k full model examinations.

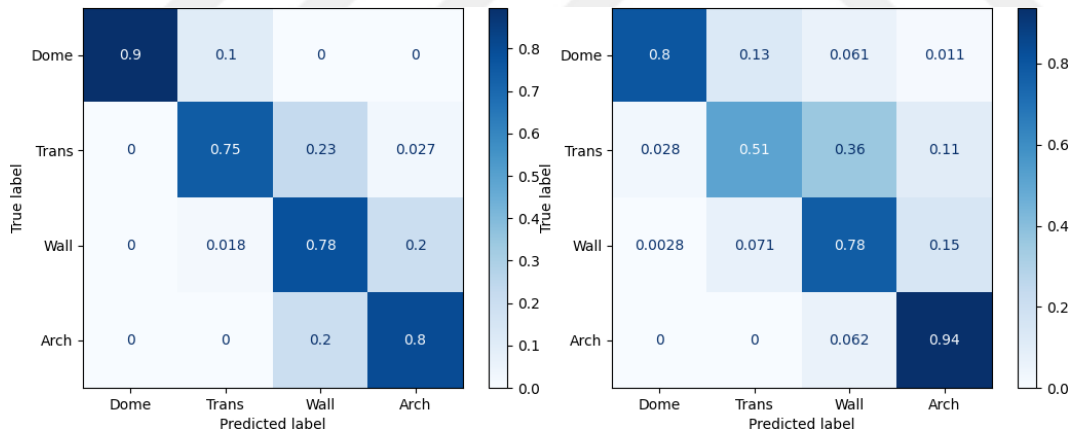


Figure 3.17 : Confusion matrices of PB (left) full and (right) partial real-world models in 2k.

Except for the arch points, other labels display similar error pattern in 10k full real-world tests. The matrices in Figure 3.18 shows both full and partial real-world tests in 10k are consistent with the 2k tests even though the error rates are bigger. The 10k models find nonexistent elements in the model. Instead of confusing only the borders of the elements some cases show that all the labels are fitted in a scene even if the geometric properties do not match. On the other hand, the partial models with 2k points

segmented the dome system meaningfully in most of the cases. Thus, even though the BERT-based algorithm outperforms the other networks in both 2k and 10k synthetic test accuracies, it could not generalize the learned information into real-world scenes. As a specific type of transformer network, Point-BERT might perform better with an augmented version of HDD that includes partial samples and more challenging cases. The overall accuracies in synthetic tests show that the network has potential for successful segmentation of parts of a dome system, yet the solution should be optimized for real-world problems in future studies.

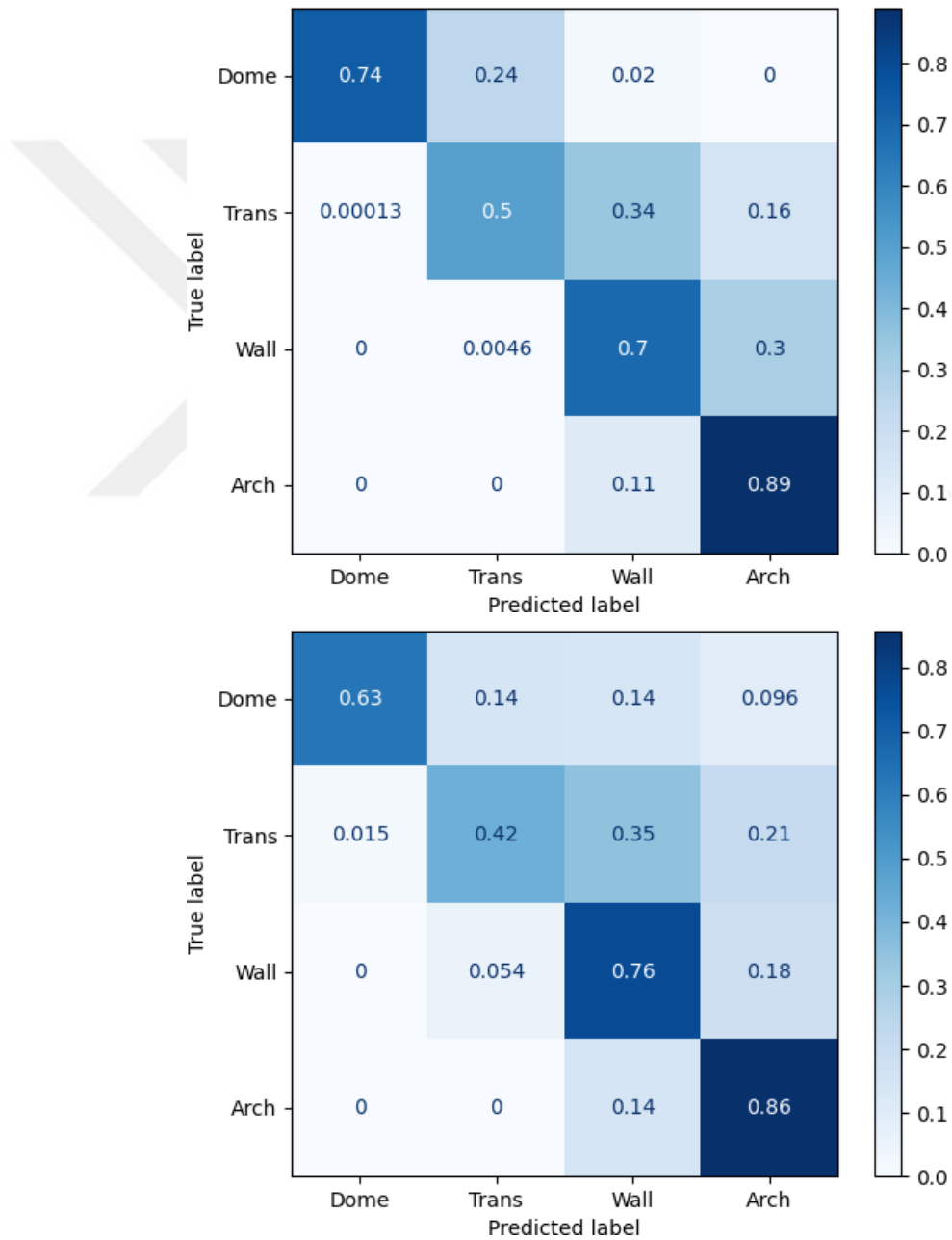


Figure 3.18 : Confusion matrices of PB (top) full and (bottom) partial real-world models in 10k.

Table 3.40 : Results of PB on full models, 2k points.

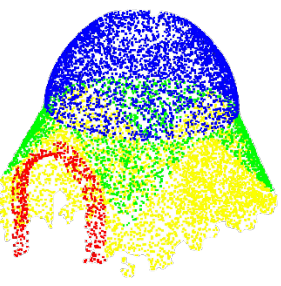
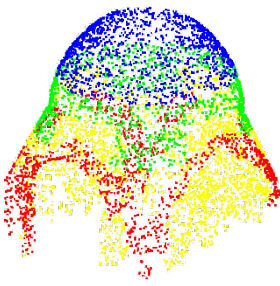
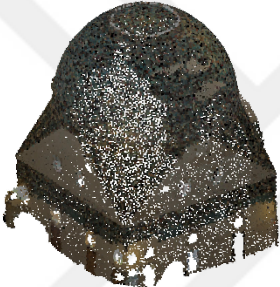
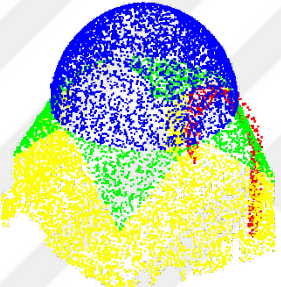
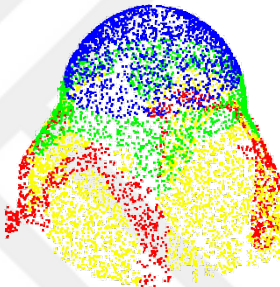
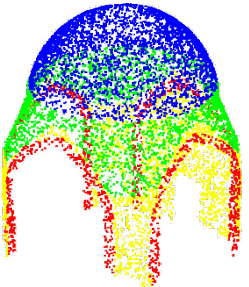
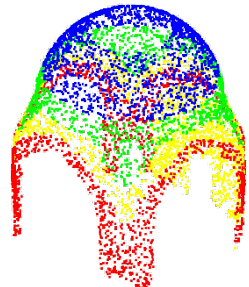
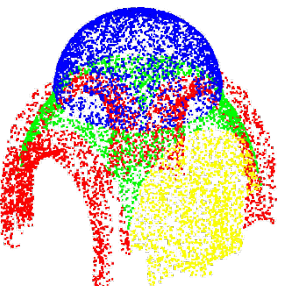
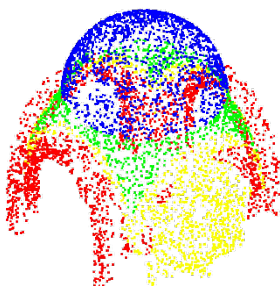
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.41 : Results of PB on full models, 10k points.

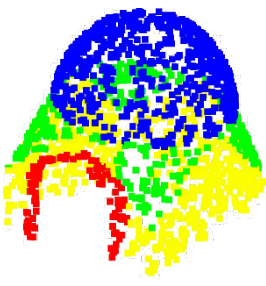
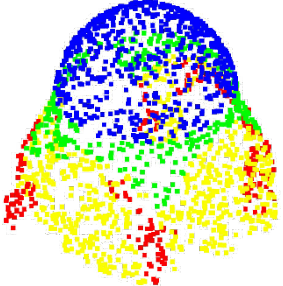
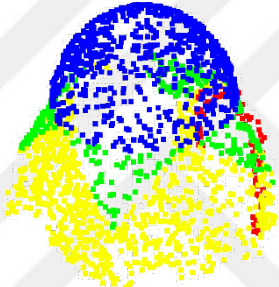
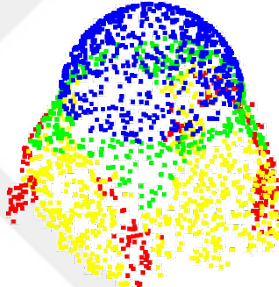
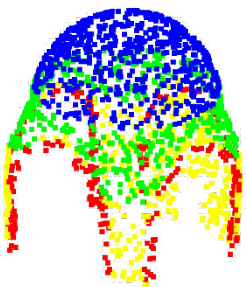
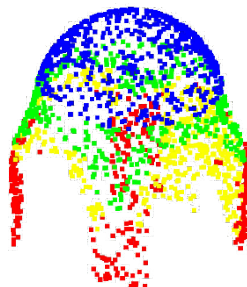
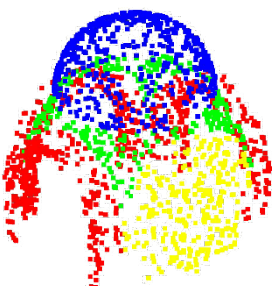
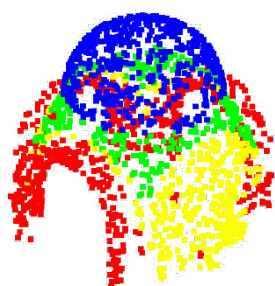
Case	Source Model	Ground Truth	Prediction
IM_M			
KM_M			
SA_H			
HK			

Table 3.42 : Results of PB on partial models, 2k points.

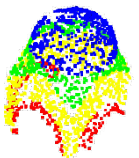
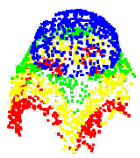
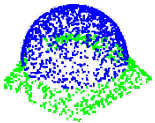
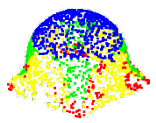
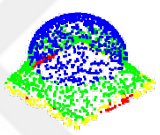
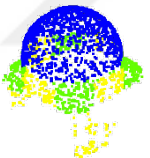
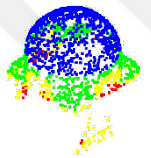
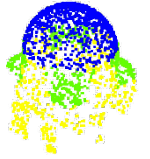
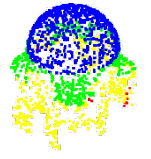
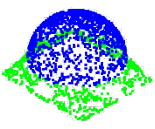
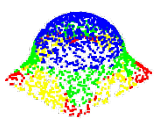
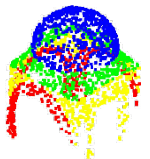
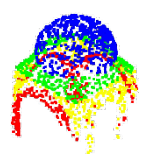
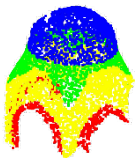
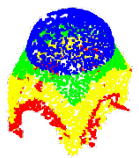
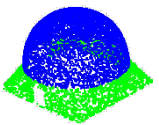
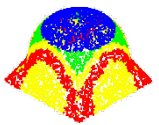
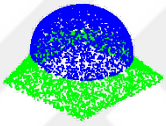
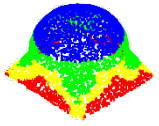
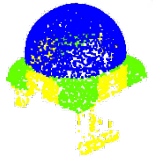
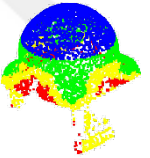
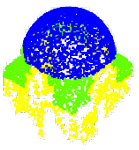
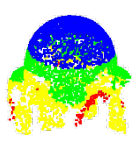
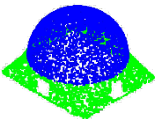
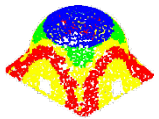
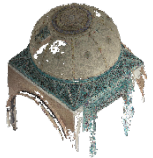
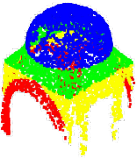
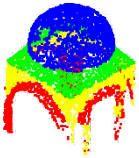
Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			

Table 3.43 : Results of PB on partial models, 10k points.

Case	Source Model	Ground Truth	Prediction
AK_ULU			
KM_S			
SA_T			
AB_M			
AK_GUD			
KM_N			
AL			



4. CONCLUSION

In cultural heritage documentation semi-automating the documentation process, relying on semantic architectural knowledge using deep neural networks has the potential of providing accurate and meaningful results. This study examines the impacts of different semantic segmentation approaches considering the relations of the parts as a section of the documentation problem of the dome systems of the monumental Anatolian architecture. By using diverse methods in a documentation process, this study defines an interdisciplinary workflow that combines architectural and computational knowledge to segment the meaningful parts of the domes that can be used in further documentation steps. The study not only uses deep neural networks designed for three-dimensional semantic segmentation assignments but also inspects the usage of data in the deep learning models to get the best results in terms of identifying the labels of points that represent units of architectural elements.

The presented Historic Dome Data Set (HDD) defines the connections of the investigated architectural parts for the deep learning methods to make them internalize the semantic relations. So the initial definition of the data set and the semantic relations of the architectural parts impact the process. The data set should be well-designed and include various valid combinations of the investigated parts of the whole.

The study shows that the models which learn both local and contextual information perform well in terms of labeling the points with semantic knowledge. Interpreting the bi-directional sequential information improved the results of the Point-BERT, specifically on the synthetic benchmark. Moreover, despite the archaic, simple deep learning architecture, PointNet++ performed decently in general. Because of pointwise neighborhood interpretation, DGCNN identified the parts of the system well. Also, RSNET, the network that uses the volumetric slices as sequential units to be interpreted, performed well despite the drawbacks of the RNN architecture in the synthetic data set. On the other hand, the PointTransformer network showed potential in identifying the part of the dome in challenging partial real-world cases with a ten thousand points examinations. To conclude, even though the deep learning algorithms

successfully label the points of the synthetic data, the models tended to remember some solutions instead of generalizing the learning solutions. Specifically, the networks failed to classify the points where the point clouds have noisy and split sections.

The presented deep neural networks use different types of units to interpret the sequential relations or interrelated connections of the parts of the architectural elements. However, points acquired using sliced volumes, farthest sampled points, or a fixed number of neighbor points would not contain accurate boundaries of the parts that would fit the definition of the architectural elements. In other words, since the input data is unlabeled, the network splits the data into patches regardless of the size or shape of the elements in the scene. The sequence or connections used in the algorithms are based on irregular point patches. Specifically, brick and stone elements are predominant in historical Anatolian architecture; thus, a method that would find similar units in the networks and learn their relations might improve the overall quality of the workflow. So, examinations with a task or content-aware sampling features might provide another quality to the process. Additionally, to semantic segmentation, instance-based part-segmentation also would be beneficial to understanding the existence of the specific elements in the scenes.

The HDD includes models in a low level of detail for generating numerous generic solutions without contradicting inner definitions of the geometries. The Historic Dome Data Set can be easily implemented into the other historic dome detection or generation tasks. Additionally, to the presented historic dome models, the following studies would enrich the data set by defining different specific shapes of various styles and their relations in the three-dimensional space.

The potential performance of deep neural networks is highly related to the provided data set in the models. Even though HDD supplies over 1500 point clouds with generic details of domes of historical Anatolian buildings, the dataset is still limited to user-defined properties. For example, complex geometrical elements such as muqarnas and triangular bands with corner groups are discarded from the data set due to the difficulty of expressing the geometries accurately. Similarly, some real-world examples such as the North Dome of Karatay Madrasa (KM_N), have window openings in the transition area, nonetheless, the synthetic historic dome data set does not include them due to the preferred level of detail. Additionally, to the four main parts of the dome systems, monumental Anatolian buildings connect spaces and enlargen the rectangular floor

plans using the iwans and vaults. The inclusion of these two elements into the HDD would be a welcome addition for a holistic understanding of the scene with minimal simplifications. For specific cases such as Sahip Ata Hanikah (SA_H) that do not fit regular rectangular floor plans with the walls in the corners instead of the edges of the floor plan, alternative definitions could be defined in the HDD and improve the quality of the segmentation process.

The color information is discarded in the procedural modeling process; however, in some real-world examples, a change of color and materials defines the border of the architectural elements. In Karatay Madrasah Main Dome (KM_M), Sahip Ata Tomb Chamber (SA_T), and the Ince Minareli Madrasah (IM_M), the drum is emphasized with a transition of material and color. Since these drums are defined as a part of a dome in these cases, they follow geometrical features of the domes with a circular base instead of placing on a polygonal base and vertically erecting. Turan and Yıldız (2018, p. 176) identify drums as a part of a dome shape. On the other hand, it is known that some dome systems have vertically built polygonal drums that aim to make structures spacious by stepping up a level of a circular dome base (Okçuoğlu, 1995, pp. 17-18). In the context of this thesis, circular and slanted drums are considered as a part of a dome while the polygonal and upright designs are considered a part of transition elements. A detailed investigation of drum geometry might clarify the dualities between the synthetic and real-world definitions. Alternatively, drum geometry can be determined as an independent geometry.

Definition of HDD is based on the results of an investigation of previous manual data collection processes. Manually created drawings parametric rule-based algorithms can be used to collect relevant training information semi-automatically from existing two-dimensional drawings. A projected algorithm can analyze the provided sections and drawings to extract the general dimensions of the domes and inner parameters of the architectural and structural elements. The extracted parameters can be used as inputs for the HDD generator.

Further studies can implement the architectural elements such as openings and floors into the data set to augment the quality of detected elements while acquiring more information from the site. This augmentation in the data would help generalize the learned information because, currently, there is a gap in level-of-detail between the created synthetic historic dome data set and the real-world examples. Also, subsequent

studies would combine synthetic data with real-world Anatolian dome models or the models that correspond to a similar context in the training process. For a detailed evaluation of the examined semantic segmentation algorithms, using the same setup would help assess the training and test durations. In addition to all numerical and visual assessments, a qualitative evaluation by a specialist of architectural documentation or cultural heritage preservation in future studies may be invaluable to reassure the practical contribution of the research.



REFERENCES

- 3DFlow, 3DF Zephyr.** (Version 6.5) [Computer Software] (2022), Retrieved from <https://www.3dflow.net>, date retrieved May 07, 2022
- Agisoft PhotoScan Standard** (Version 1.8.3) [Computer Software]. (2022). Retrieved from <http://www.agisoft.com/downloads/installer>, date retrieved May 07, 2022
- Aliberti, L., & Alonso-Rodríguez, M. Á.** (2017). Geometrical analysis of the coffers of the pantheon's dome in Rome. *Nexus Network Journal*, 19(2), 363-382.
- Andriasyan, M., Moyano, J., Nieto-Julián, J. E., & Antón, D.** (2020). From point cloud data to building information modelling: An automatic parametric workflow for heritage. *Remote Sensing*, 12(7), 1094.
- Anil, E. B., Tang, P., Akinci, B., & Huber, D.** (2011). Assessment of the quality of as-is building information models generated from point clouds using deviation analysis. In *Three-Dimensional Imaging, Interaction, and Measurement*, 7864, 136-148.
- Anil, E. B., Tang, P., Akinci, B., & Huber, D.** (2013). Deviation analysis method for the assessment of the quality of the as-is Building Information Models generated from point cloud data. *Automation in Construction*, 35, 507-516.
- Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., & Savarese, S.** (2016). 3d Semantic Parsing of Large-Scale Indoor Spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1534-1543.
- Asgari Taghanaki, S., Abhishek, K., Cohen, J. P., Cohen-Adad, J., & Hamarneh, G.** (2021). Deep semantic segmentation of natural and medical images: a review. *Artificial Intelligence Review*, 54(1), 137-178.
- Ashkan, M., & Ahmad, Y.** (2009). Persian domes: history, morphology and typologies. *ArchNet-IJAR*, 3(3).
- Atik, M. E., & Duran, Z.** (2021). Classification of aerial photogrammetric point cloud using recurrent neural networks. *Fresenius Environ. Bull*, 30, 4270-4275.
- Autodesk, ReCap Pro** [Computer Software] (2022), Retrieved from <https://www.autodesk.com/products/recap/overview> Accessed on May 07, 2022
- Badrinarayanan, V., Kendall, A., & Cipolla, R.** (2017). Segnet: A Deep Convolutional Encoder-Decoder Architecture For Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12), 2481-2495.

- Bahdanau, D., Cho, K., & Bengio, Y.** (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
- Banfi, F.** (2020). HBIM, 3D drawing and virtual reality for archaeological sites and ancient ruins. *Virtual Archaeology Review*, 11(23), 16-33.
- Barazzetti, L.** (2016). Parametric as-built model generation of complex shapes from point clouds. *Advanced Engineering Informatics*, 30(3), 298-311.
- Barontini, A., Alarcon, C., Sousa, H. S., Oliveira, D. V., Masciotta, M. G., & Azenha, M.** (2021). Development and demonstration of an HBIM framework for the preventive conservation of cultural heritage. *International Journal of Architectural Heritage*, 1-23.
- Baş, T.** (2008). *Anadolu Selçuklu dönemi Konya mahalle mescitlerinin restorasyon sorunları / The restoration problems of Konya district masjids in Seljukian Period of Anatolia*. [Master's thesis, Selçuk University], Konya, Turkey
- Bello, S. A., Yu, S., Wang, C., Adam, J. M., & Li, J.** (2020). Deep learning on 3D point clouds. *Remote Sensing*, 12(11), 1729.
- BIMForum.** (2021). Level of Development Specification. Available online: <http://bimforum.org/lod>, date retrieved 24 April 2022.
- Bianchini, C.** (2020). A methodological approach for the study of domes. *Nexus Network Journal*, 22(4), 983-1013.
- Blomley, R., Weinmann, M., Leitloff, J., & Jutzi, B.** (2014). Shape distribution features for point cloud analysis-a geometric histogram approach on multiple scales. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3), 9.
- Brumana, R., Banfi, F., Cantini, L., Previtali, M., & Della Torre, S.** (2019). Hbim level of detail-geometry-Accuracy and survey analysis for architectural preservation. In *2nd International Conference of Geomatics and Restoration*, GEORES 2019 42(2), 293-299
- Capone, M., & Lanzara, E.** (2019a). Scan-to-BIM vs 3D ideal model HBIM: Parametric tools to study domes geometry. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 219-226.
- Capone, M., & Lanzara, E.** (2019b). 3d data interpretation using treatises geometric rules to built coffered domes. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*.
- Capturing Reality, Reality Capture.** [Computer Software] (2022), Retrieved from <https://www.capturingreality.com/realitycapture>, date retrieved May 07, 2022
- Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., ... & Yu, F.** (2015). Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012.
- Che, E., Jung, J., & Olsen, M. J.** (2019). Object recognition, segmentation, and classification of mobile laser scanning point clouds: A state of the art review. *Sensors*, 19(4), 810.

- Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H.** (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 801-818).
- Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y.** (2014b). On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y.** (2014a). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
- Chomsky, N.** (1956). Three models for the description of language. *IRE Transactions on information theory*, 2(3), 113-124.
- Croce, V., Caroti, G., De Luca, L., Jacquot, K., Piemonte, A., & Véron, P.** (2021). From the semantic point cloud to heritage-building information modeling: A semiautomatic approach exploiting machine learning. *Remote Sensing*, 13(3), 461.
- Croce, V., Caroti, G., Piemonte, A., & Bevilacqua, M. G.** (2019). Geomatics for Cultural Heritage Conservation: Integrated survey and 3D modeling. In *Proceedings of the IMEKO TC4 International Conference on Metrology for Archaeology and Cultural Heritage*, MetroArchaeo, Florence, Italy 271-276.
- Cutler, A.** (1966). Structure and aesthetic at Hagia Sophia in Constantinople. *The Journal of Aesthetics and Art Criticism*, 25(1), 27-35.
- Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L.** (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* 248-255.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K.** (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N.** (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.
- Duan, Y., Zheng, Y., Lu, J., Zhou, J., & Tian, Q.** (2019). Structural relational reasoning of point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 949-958.
- Elberink, S. O., & Vosselman, G.** (2011). Quality analysis on 3D building models reconstructed from airborne laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*. 66(2), 157-165.
- Elman, J. L.** (1990). Finding structure in time. *Cognitive Science*, 14(2), 179-211.
- Girshick, R., Donahue, J., Darrell, T., & Malik, J.** (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 580-587.

- Griffiths, D., & Boehm, J.** (2019). A review on deep learning techniques for 3D sensed data classification. *Remote Sensing*, 11(12), 1499.
- Grilli, E., Özdemir, E., & Remondino, F.** (2019). Application of machine and deep learning strategies for the classification of heritage point clouds. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*.
- Griwodz, C., Gasparini, S., Calvet, L., Gurdjos, P., Castan, F., Maujean, B., Lillo, G. D., & Lanthony, Y.** (2021). AliceVision Meshroom: An open-source 3D reconstruction pipeline. *Proceedings of the 12th ACM Multimedia Systems Conference-MMSys '21*. <https://doi.org/10.1145/3458305.3478443>
- Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., & Bennamoun, M.** (2020). Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12), 4338-4364.
- Güneş, M. C., Mertan, A., Sahin, Y. H., Unal, G., & Özkar, M.** (2021). Synthesizing Point Cloud Data Set for Historical Dome Systems. In *International Conference on Computer-Aided Architectural Design Futures*. 538-554. Springer, Singapore.
- He, K., Gkioxari, G., Dollár, P., & Girshick, R.** (2017). Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, 2961-2969.
- He, K., Zhang, X., Ren, S., & Sun, J.** (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* 770-778.
- Hochreiter, S., & Schmidhuber, J.** (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.
- Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., & Murphy, K.** (2017). Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE Conference On Computer Vision And Pattern Recognition* (pp. 7310–7311).
- Huang, Q., Wang, W., & Neumann, U.** (2018). Recurrent slice networks for 3d segmentation of point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2626-2635).
- Kadi, H., & Anouche, K.** (2020). Knowledge-based parametric modeling for heritage interpretation and 3D reconstruction. *Digital Applications in Archaeology and Cultural Heritage*.
- Kedzierski, M., & Fryskowska, A.** (2014). Terrestrial and aerial laser scanning data integration using wavelet analysis for the purpose of 3D building modeling. *Sensors*, 14(7), 12070-12092.
- Khalil, A., & Stravovavdis, S.** (2019). H-BIM and the domains of data investigations of heritage buildings current state of the art. In *2nd International Conference of Geomatics and Restoration (GEORES 2019)* (Vol. 42, No. W11, pp. 661-667).

- Krizhevsky, A., Sutskever, I., & Hinton, G. E.** (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25.
- Lanzara, E., & Capone, M.** (2020). Domes Decorative Apparatuses: Parametric Coffers and Tiles in Naples. In *Congreso Internacional de Expresión Gráfica Arquitectónica* (pp. 699-712). Springer, Cham.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P.** (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, 86(11), 2278-2324.
- Lee, J. S., Park, J., & Ryu, Y. M.** (2021). Semantic segmentation of bridge components based on hierarchical point cloud model. *Automation in Construction*, 130, 103847.
- Liu, X., Han, Z., Liu, Y. S., & Zwicker, M.** (2019). Point2sequence: Learning the shape representation of 3d point clouds with an attention-based sequence to sequence network. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 33, No. 01, pp. 8778-8785).
- Long, J., Shelhamer, E., & Darrell, T.** (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3431-3440).
- Loverdos, D., Sarhosis, V., Adamopoulos, E., & Drougkas, A.** (2021). An innovative image processing-based framework for the numerical modelling of cracked masonry structures. *Automation in Construction*, 125, 103633.
- Malinverni, E. S., Pierdicca, R., Paolanti, M., Martini, M., Morbidoni, C., Matrone, F., & Lingua, A.** (2019). Deep learning for semantic segmentation of 3D point cloud. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*.
- Masi, F., Stefanou, I., & Vannucci, P.** (2018). On the origin of the cracks in the dome of the Pantheon in Rome. *Engineering Failure Analysis*, 92, 587-596.
- Milani, G., & Cecchi, A.** (2013). Compatible model for herringbone bond masonry: Linear elastic homogenization, failure surfaces and structural implementation. *International Journal of Solids and Structures*, 50(20-21), 3274-3296.
- Mo, K., Zhu, S., Chang, A. X., Yi, L., Tripathi, S., Guibas, L. J., & Su, H.** (2019). Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 909-918).
- Mo, Y., Wu, Y., Yang, X., Liu, F., & Liao, Y.** (2022). Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing*, 493, 626-646.
- Moradi, A.** (2020). Transition zone in Seljuq dome chambers. *Journal of Islamic Architecture*, 6(1), 24-32.
- Morbidoni, C., Pierdicca, R., Paolanti, M., Quattrini, R., & Mammoli, R.** (2020). Learning from synthetic point cloud data for historical buildings

semantic segmentation. *Journal on Computing and Cultural Heritage* (JOCCH), 13(4), 1-16.

- Murtiyoso, A., Grussenmeyer, P., Koehl, M., & Freville, T.** (2016). Acquisition and processing experiences of close range UAV images for the 3D modeling of heritage buildings. *In Euro-Mediterranean Conference* (pp. 420-431). Springer, Cham.
- Murtiyoso, A., Koehl, M., Grussenmeyer, P., & Freville, T.** (2017). Acquisition and processing protocols for UAV images: 3D modeling of historical buildings using photogrammetry. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4, 163.
- Nagakura, T., Tsai, D., & Choi, J.** (2015). Capturing History Bit by Bit-Architectural Database of Photogrammetric Model and Panoramic Video. *In Proceedings of the 33rd eCAADe Conference*, Vol. 1, 685–694
- Ochmann, S., Vock, R., & Klein, R.** (2019). Automatic reconstruction of fully volumetric 3D building models from oriented point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 151, 251-262.
- Okçuoğlu, T.** (1995). *Anadolu Selçuklu mescitlerinde kubbeye geçiş alanının değerlendirilmesi*. (Master's thesis, Istanbul University), Istanbul, Turkey
- Oses, N., Dornaika, F., & Moujahid, A.** (2014). Image-based delineation and classification of built heritage masonry. *Remote Sensing*, 6(3), 1863-1889.
- Özcan, A.** (2008). *14.15. yüzyıl Bursa camii ve mescitlerinde kubbeye geçiş elemanları* (Master's thesis, Erciyes University), Kayseri, Turkey
- Özdemir, E., & Remondino, F.** (2019). Classification of aerial point clouds with deep learning. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 103-110.
- Pavoni, G., Giuliani, F., Falco, A., Corsini, M., Ponchio, F., Callieri, M., & Cignoni, P.** (2020). Another Brick in the Wall: Improving the Assisted Semantic Segmentation of Masonry Walls. *In Eurographics Workshop on Graphics and Cultural Heritage. The Eurographics Association*.
- Pellis, E., Murtiyoso, A., Masiero, A., Tucci, G., Betti, M., & Grussenmeyer, P.** (2022). An Image-Based Deep Learning Workflow for 3d Heritage Point Cloud Semantic Segmentation. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 46, 429-434.
- Pierdicca, R., Mameli, M., Malinverni, E. S., Paolanti, M., & Frontoni, E.** (2019, June). Automatic generation of point cloud synthetic dataset for historical building representation. *In International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (pp. 203-219). Springer, Cham.
- Pierdicca, R., Paolanti, M., Matrone, F., Martini, M., Morbidoni, C., Malinverni, E., Frontoni, E., & Lingua, A.** (2020). Point cloud semantic

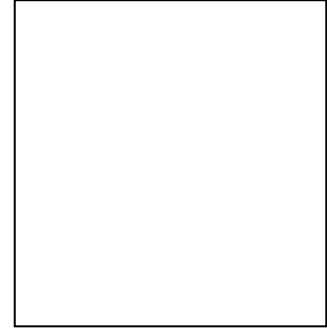
- segmentation using a deep learning framework for cultural heritage. *Remote Sensing*, 12(6), 1005.
- Powers, D. M.** (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. arXiv preprint arXiv:2010.16061.
- Qi, C. R., Su, H., Mo, K., & Guibas, L. J.** (2017a). Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 652-660).
- Qi, C. R., Yi, L., Su, H., & Guibas, L. J.** (2017b). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in Neural Information Processing Systems*, 30.
- Quattrini, R., Malinverni, E. S., Clini, P., Nespeca, R., & Orlietti, E.** (2015). From tils to hbim. High quality semantically-aware 3d modeling of complex architecture. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*.
- Ren, S., He, K., Girshick, R., & Sun, J.** (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28.
- Ronneberger, O., Fischer, P., & Brox, T.** (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 234-241). Springer, Cham.
- Sahin, Y. H., Mertan, A., & Unal, G.** (2022). ODFNet: Using orientation distribution functions to characterize 3D point clouds. *Computers & Graphics*, 102, 610-618.
- Santagati, C., Lo Turco, M., & Garozzo, R.** (2018). Reverse information modeling for historic artefacts: towards the definition of a level of accuracy for ruined heritage. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42(2).
- Schönberger, J. L., & Frahm, J.-M.** (2016). Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*
- Sestras, P., Roşca, S., Bilaşco, Ş., Naş, S., Buru, S., Kovacs, L., Spalevic, V., & Sestras, A.** (2020). Feasibility assessments using unmanned aerial vehicle technology in heritage buildings: Rehabilitation-restoration, spatial analysis and tourism potential analysis. *Sensors*, 20(7), 2054.
- Simonyan, K., & Zisserman, A.** (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
- Simşek, H.** (2010). *Erken osmanlı mimarisinde kubbeye geçiş sistemlerinden üçgenler kuşağı / Of the transition systems to dome, generation of triangles in Early Period Ottoman architecture*. (Master's thesis, Yüzüncü Yıl University), Van, Turkey
- Stiny, G., & Gips, J.** (1971). Shape grammars and the generative specification of painting and sculpture. In *IFIP congress* (2) (2,3, pp. 125-135).

- Teruggi, S., Grilli, E., Russo, M., Fassi, F., & Remondino, F.** (2020). A hierarchical machine learning approach for multi-level and multi-resolution 3D point cloud classification. *Remote Sensing*, 12(16), 2598.
- Tiberti, S., & Milani, G.** (2020). 3D homogenized limit analysis of non-periodic multi-leaf masonry walls. *Computers & Structures*, 234, 106253.
- Turan, Ş. N., & Yıldız, E.** (2018). XIII. Yüzyıl Konya mahalle mescitlerinde kullanılan kubbeye geçiş elemanları. *Türk İslam Medeniyeti Akademik Araştırmalar Dergisi*, 13(26), 163-187.
- Turan, Ş.N.** (2018). *Türk mimarisinde kullanılan kubbeye geçiş elemanları; 13.Yy. Anadolu Selçuklu dönemi Konya mahalle mescitleri örneği / Transition elements into dome in Turkish architecture; 13th century anatolian seljuks period example of Konya neighbourhood mosques*. (Master's thesis, Necmettin Erbakan University), Konya, Turkey
- Valero, E., Bosché, F., & Forster, A.** (2018). Automatic segmentation of 3D point clouds of rubble masonry walls, and its application to building surveying, repair and maintenance. *Automation in Construction*, 96, 29-39.
- Valero, E., Forster, A., Bosché, F., Hyslop, E., Wilson, L., & Turmel, A.** (2019). Automated defect detection and classification in ashlar masonry walls using machine learning. *Automation in Construction*, 106, 102846.
- Valibeig, N., Rahravi Poodeh, S., & Rahimi Ariaei, A.** (2017). Structural and geometric analysis of discontinuous double-shell Persian domes in Isfahan and Nain dome-building schools. *International Journal of Architectural Heritage*, 11(8), 1101-1120.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser Ł., & Polosukhin, I.** (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.
- Wang, X., He, J., & Ma, L.** (2019). Exploiting local and global structure for point cloud semantic segmentation with contextual point representations. *Advances in Neural Information Processing Systems*, 32.
- Wang, Y., Lv, H., Deng, R., & Zhuang, S.** (2020). A Comprehensive Survey of Optical Remote Sensing Image Segmentation Methods. *Canadian Journal of Remote Sensing*, 46(5), 501-531.
- Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., & Solomon, J. M.** (2019). Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5), 1-12.
- Weinmann, M., Jutzi, B., & Mallet, C.** (2013). Feature relevance assessment for the semantic interpretation of 3D point cloud data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 313-318.
- Xiong, X., Adan, A., Akinci, B., & Huber, D.** (2013). Automatic creation of semantically rich 3D building models from laser scanner data. *Automation in Construction*, 31, 325-337.
- Xu, Z., Wu, T. H., Shen, Y., & Wu, L.** (2016). Three dimensional reconstruction of large cultural heritage objects based on uav video and tls data. *The*

- Yang, X., Lu, Y. C., Murtiyoso, A., Koehl, M., & Grussenmeyer, P.** (2019). HBIM modeling from the surface mesh and its extended capability of knowledge representation. *ISPRS International Journal of Geo-Information*, 8(7), 301.
- Yi, L., Kim, V., Ceylan, D., Shen, I.C., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., & Guibas, L.** (2016). A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (ToG)*, 35(6), 1–12.
- Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., & Lu, J.** (2021). Point-BERT: Pre-training 3D Point Cloud Transformers with Masked Point Modeling. arXiv preprint arXiv:2111.14819.
- Zhang, J., Zhao, X., Chen, Z., & Lu, Z.** (2019). A review of deep learning-based semantic segmentation for point cloud. *IEEE Access*, 7, 179118–179133.
- Zhao, H., Jiang, L., Jia, J., Torr, P. H., & Koltun, V.** (2021). Point transformer. *In Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 16259–16268).
- Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J.** (2017). Pyramid scene parsing network. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2881–2890).
- Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X.** (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11), 3212–3223



CURRICULUM VITAE



Name Surname : Mustafa Cem GÜNEŞ

Place and Date of Birth :

E-Mail :

EDUCATION :

- **B.A.** : 2019, Istanbul Technical University, Architecture, Department of Architecture

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2019-2020 Researcher, Analysis of Design Processes of Geometric Pattern Based Muqarnas by Developing Computational Design Tools, KADİR Has University, TÜBİTAK.
- 2020-2022 Researcher, Developing Computational Methods for Modeling Building Components Using Photogrammetry Data in Documenting Historical Buildings, ISTANBUL Technical University, TÜBİTAK.

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- Güneş, M. C., Mertan, A., Sahin, Y. H., Unal, G., & Özkar, M. 2021. Synthesizing Point Cloud Data Set for Historical Dome Systems. *In International Conference on Computer-Aided Architectural Design Futures* (pp. 538-554). Springer, Singapore.

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- Altun, S., & GÜNEŞ, M. C. (2020) Classification of Historic Ornaments with CNN: Issues for Interdisciplinary Studies. *Journal of Computational Design*, 1(3), 115-130.