




## Article

# AMS-Net: An Attention-Based Multi-Scale Network for Classification of 3D Terracotta Warrior Fragments

Jie Liu <sup>1,†</sup>, Xin Cao <sup>1,†</sup>, Pingchuan Zhang <sup>2</sup>, Xueli Xu <sup>1</sup>, Yangyang Liu <sup>1</sup>, Guohua Geng <sup>1,\*</sup>, Fengjun Zhao <sup>1</sup>, Kang Li <sup>1</sup>  and Mingquan Zhou <sup>1</sup>

- <sup>1</sup> School of Information Science and Technology, Northwest University, Xi'an 710068, China; jieliu@stumail.nwu.edu.cn (J.L.); caoxin@nwu.edu.cn (X.C.); xuxueli@stumail.nwu.edu.cn (X.X.); yylliu@nwu.edu.cn (Y.L.); fjzhao@nwu.edu.cn (F.Z.); likang@nwu.edu.cn (K.L.); mqzhou@bnu.edu.cn (M.Z.)  
<sup>2</sup> School of Information Engineering, Henan Institute of Science and Technology, Xinxiang 453003, China; 201910030@stumail.nwu.edu.cn  
\* Correspondence: ghgeng@nwu.edu.cn  
† Those authors contributed equally to this paper.

**Abstract:** As an essential step in the restoration of Terracotta Warriors, the results of fragments classification will directly affect the performance of fragments matching and splicing. However, most of the existing methods are based on traditional technology and have low accuracy in classification. A practical and effective classification method for fragments is an urgent need. In this case, an attention-based multi-scale neural network named AMS-Net is proposed to extract significant geometric and semantic features. AMS-Net is a hierarchical structure consisting of a multi-scale set abstraction block (MS-BLOCK) and a fully connected (FC) layer. MS-BLOCK consists of a local-global layer (LGLayer) and an improved multi-layer perceptron (IMLP). With a multi-scale strategy, LGLayer can parallel extract the local and global features from different scales. IMLP can concatenate the high-level and low-level features for classification tasks. Extensive experiments on the public data set (ModelNet40/10) and the real-world Terracotta Warrior fragments data set are conducted. The accuracy results with normal can achieve 93.52% and 96.22%, respectively. For real-world data sets, the accuracy is best among the existing methods. The robustness and effectiveness of the performance on the task of 3D point cloud classification are also investigated. It proves that the proposed end-to-end learning network is more effective and suitable for the classification of the Terracotta Warrior fragments.

**Keywords:** self-attention; multi-scale; deep neural networks; point cloud classification; Terracotta Warrior fragments



**Citation:** Liu, J.; Cao, X.; Zhang, P.; Xu, X.; Liu, Y.; Geng, G.; Zhao, F.; Li, K.; Zhou, M. AMS-Net: An Attention-Based Multi-Scale Network for Classification of 3D Terracotta Warrior Fragments. *Remote Sens.* **2021**, *13*, 3713. <https://doi.org/10.3390/rs13183713>

Academic Editor:

Joaquín Martínez-Sánchez

Received: 21 July 2021

Accepted: 8 September 2021

Published: 17 September 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

As one of the critical channels for spreading Chinese culture, the Terracotta Warriors have completed the third large-scale excavation study since they were first discovered in 1974. Extensive Terracotta Warriors have been predominantly found in fragments due to the natural environment and human factors (Figure 1). Therefore, it is of great significance to implement the protection and restoration of cultural relics. As the Terracotta Warriors were found in fragments and randomly mixed, the process of traditional manual restoration methods may spend much time and tedious work. Remote sensing can realize rapid multisource data analysis and the dynamic monitoring of cultural relics and their surrounding environments [1]. With the development of remote sensing, remote sensing archaeology has become an increasingly common method for researchers to investigate cultural sites. Light detection and ranging (LiDAR) data are easy to obtain, which includes height and structure information of objects [2]. A point cloud can efficiently reduce secondary damage to cultural relics. The point cloud is of great significance to implement the protection and restoration of cultural relics as it can restore the structural relationship

of the cultural relics well [3]. In recent years, virtual restoration of the Terracotta Warrior fragments can effectively save human resources and time and effectively reduce secondary damage. The research on virtual restoration methods is significant. In general, there are three essential steps in the process. Firstly, digital models of the fragments are obtained through a 3D laser scanning device. Secondly, the Terracotta Warriors fragment should be classified into different categories correctly. Finally, the splicing of the fragments is completed, and the parts with holes are repaired. As the most critical step, 3D shape classification of the Terracotta Warrior fragments plays a vital role in the automatic splicing and restoration of cultural relics.



**Figure 1.** Some unearthened Terracotta Warrior fragments.

Some studies have proposed various traditional approaches to classify fragments of cultural relics. Most of the previous methods are mainly based on color features [4], texture features [5], color and texture features [6–8], texture and shape features [9], multiple-features fusion [10], and other features. The traditional methods usually require experts to design accurate feature description operators and spend much time. Experts manually classify and calibrate fragments of cultural relics with non-salient features or fusion features based on experience. These are the reasons why the traditional methods have relatively low classification accuracy.

With the development of deep learning, convolutional neural networks (CNNs) have shown significant success in image recognition [11,12], video analysis [13,14], speech emotion recognition (SER) [15–18], and other domains. Based on the above work, Wang [19] propose an improved CNN specialized for the classification of Terracotta Warrior fragments for the first time. Compared with traditional methods, the proposed method can reduce the time complexity of the algorithm and improve the efficiency of fragments classification. However, the accuracy of image-based deep learning classification methods of Terracotta Warrior fragments is still relatively low. In recent years, the excellent results of deep neural networks for 2D image processing have motivated a data-driven approach to learning features on 3D models. Unlike the 2D image, several common 3D data representations are volumetric grids, depth images, and point clouds. According to the input data type for networks, 3D shape classification methods can be classified into volumetric-based [20,21], multi-view-based [22,23], and point-based methods. Compared with the former two data types, the point cloud is one of the most straightforward 3D shape representations and has been widely used. However, a key challenge is that the raw point cloud is irregular and unordered. PointNet [24] directly takes point cloud as its input and achieves permutation invariance with a symmetric function as a pioneering work. Inspired by PointNet, Gao et al. [25] present an automatic method combined with template guidance to classify 3D fragments of the Terracotta Warriors. In [26], the proposed method can directly consume the point cloud and texture image of the fragment and outputs its category. Experimental results demonstrate that the two methods perform better than traditional methods. However, the baseline model of the two methods is PointNet, which fails to

capture local features adequately. To capture local structures better, the subsequent works have also been proposed (e.g., PointNet++ [27], PointCNN [28] and DGCNN [29]).

Although the existing deep learning models have shown suitable performances in point cloud classification, there are still some shortcomings. During the classification experiments, we find most existing deep learning models have the following problems:

(1) The receptive fields are fixed-size, which cannot learn complex features by extracting features from different scales in parallel. The characteristics of Terracotta Warrior fragments are different in size and location. Some fragments have a smooth surface and few salient features, while fragments from the body have detailed features of the plackets. For the classification of 3D Terracotta Warrior fragments, the selection and extraction of salient and representative features are still challenging tasks. (2) Capturing long-range dependencies is crucial in deep neural networks. Most of the existing methods use the large receptive fields formed by the deep stacks of convolution to obtain long-range dependencies. However, blindly increasing the depth of the network can reduce the performance of the network. To make matters worse, the network becomes more complex as the depth of the network increases. (3) In most existing deep learning methods for point cloud understanding, the features are abstracted into higher dimensions through the MLP layer and then aggregated by a max/avg-pooling operation. However, the pooling-based feature aggregation methods can hardly encode the correlation between feature vectors in the feature. How to aggregate those learned local region features and their spatial relationships is still a challenging task.

In order to solve the mentioned problems, an end-to-end attention-based multi-scale neural network, named AMS-Net, is introduced to specialize in the classification of the 3D Terracotta Warrior fragments. A multi-scale set abstraction block (MS-BLOCK) is designed to extract local and global features from different scales and capture the long-range dependencies from the input data. In addition, high-level features contain more semantic information but less spatial information. Low-level features have more space coordinates information, but the semantic information is insufficient. The improved multi-layer perceptron (IMLP) can retain both the high-level and low-level features well. Then, aggregated features with abundant information, which, using a skip connection strategy, are fed to a fully connected (FC) layer for further processing. Finally, a softmax classifier is used for the classification. Extensive experiments demonstrate that the proposed network achieves improved performance on the classification of the 3D Terracotta Warrior fragments. The main contributions of this work are summarized as follows:

- A novel hierarchical network called AMS-Net is proposed to enhance the capability of extracting the features of the 3D Terracotta Warrior fragments. In order to decrease the computational cost, our AMS-Net is proposed to extract contextual features in a multi-scale way instead of stacking many layers to increase the receptive field directly. The self-attention model is adopted to integrate the semantic and spatial relationships between features. To the best of our knowledge, this is the first work to apply the multi-scale structure and self-attention strategy to classify 3D cultural relic fragments;
- A local-global module is proposed, which can effectively achieve local region feature aggregation and capture long-range dependencies well. The two main components are local features aggregated cell (LFA-Cell), and global features aggregated cell (GFA-Cell). However, LFA-Cell is proposed to preserve complex local structures, which are explicitly encoded with the spatial locations from the original 3D space. The global geometric features are obtained by GFA-Cell based on self-attention. As one of the important components in LFA-Cell, a self-attention feature aggregation method named attentive aggregation sub-unit (AAS) is proposed. Compared with the traditional max-pooling-based feature aggregation networks, AAS can explicitly learn not only geometric features of local regions but also the spatial relationships among them;
- As the performance of the feature extractor is strongly affected by the dimension of the max-pooling layer, a feature fusion named IMLP is proposed in a targeted manner for

our multi-scale structure, which can aggregate both low-level and high-level features with rich local information;

- Our AMS-Net can explicitly learn not only geometric features of local regions but also the spatial relationships among them. The proposed method is more suitable for the characteristic of the Terracotta Warrior fragments and can achieve a suitable classification result.

The remainder of this paper is organized as follows: the related work is introduced in Section 2. Then, the detailed overview of the proposed system and its sub-modules are described in detail in Section 3. In Section 4, the data preprocessing method of the Terracotta Warrior fragments and the experimental results are provided. Finally, the conclusions and the limitations of this study and future works are illustrated in Section 5.

## 2. Related Work

### 2.1. Traditional Classification Methods of Terracotta Warrior Fragments

As the most critical step, 3D shape classification of the Terracotta Warrior fragments plays a vital role in the protection and restoration of cultural relics. Many studies have focused on the issue of archaeology to find solutions based on images or 3D models, and some researchers are interested in the proposed methods of classifying fragments. Kampel et al. [4] focus on the classification of two-dimensional fragments based on the properties of color. In contrast, Qi et al. [5] deal with the problem on the basis of surface texture properties. Some researchers have proposed to classify cultural relic fragments with two or more feature description operators. Nada A. Rasheed et al. [6–8] present algorithms that rely on the intersection of the RGB color between the archaeological fragments and extraction of texture features from fragments based on gray-level co-occurrence matrix (GLCM). Wei et al. [9] extract the texture features and shape features by the scale-invariant feature transform (SIFT) algorithm and Hu invariant moments. Combined with the above features, a new method based on a support vector machine (SVM) for the classification of the Terracotta Warrior fragments has been proposed. Zhao et al. [10] extract the fragments' significant region features based on the region and shape features. The earth mover's distance (EMD) method is used to match the region features and the fragments to achieve coarse classification. The shape features are extracted by the Hu invariant moment. The salient features on the surface of Terracotta Warrior fragments are obtained by clustering local surface descriptions introduced by Kang et al. [30]. Lu et al. [31] present a local descriptor used to extract the fragments' rotational projection features and salient local features. The corresponding similarity measure matching method is proposed. The weight of characteristics is adaptively calculated according to the measurement results of each type of feature. Du et al. [32] propose a modified point feature histogram (PFH) descriptor to match fragments with templates. Karasik and Smilansky [33] propose a method that relies on the computerized morphological classification of ceramics.

### 2.2. Deep Learning on Point Clouds

According to the network architecture used for the feature learning of each point, methods can be divided into point-wise MLP [24,27,34], convolution-based [28,35,36], graph-based [29,37], and other methods. PointNet++ [27] improves performance by introducing a hierarchical approach to complete the feature extraction, which can capture local structures better. Due to the irregular format of the point cloud, convolutional kernels for the 3D point cloud are challenging to design. PointCNN [28] is a generalization of CNN into leveraging spatially local correlation from data represented in the point cloud. Relation-shape convolution [35], as a learn-from-relation convolution operator, can explicitly encode the geometric topology constraint among points. Based on the proposed convolution, a hierarchical architecture RS-CNN (relation-shape convolutional neural network) is presented. An SOM (self-organizing map) [36] is built to model the spatial distribution of the input point cloud, which enables hierarchical feature extraction on both individual points and SOM nodes. It can extend regular grid CNN to irregular configuration for achieving

contextual shape-aware learning of point cloud. Wang et al. [37] presented a spectral graph convolution on a local graph and combined it with recursive cluster pooling to make full use of the neighboring points' relative structure and features. The method requires no pre-computation of the graph Laplacian matrix and graph coarsening hierarchy. As there is a lack of large-scale data sets of partial views of real objects, Par3DNet [38] is proposed to fill the gap between synthetic and real data, which can take a partial 3D view of the object as an input and is able to accurately classify it. Hou et al. [39] propose a novel method for detecting gold foil damage on stone carving relics by making use of multi-temporal 3D LiDAR point cloud.

### 2.3. Multi-Scale Structure

Feature extraction is a crucial part, and its performance plays an important role in the quality of the classification results. As a useful technology, research on the multi-scale structure is increasing gradually. Zhao et al. [40] present a novel transfer learning framework based on a deep multi-scale convolutional neural network (MSCNN). MSCNN is applied to the intelligent fault diagnosis of rolling bearings and has excellent performance. Another elegant mechanism to significantly increase the receptive field size is dilated convolution network (DCN). Huang et al. [41] present a workflow for LiDAR point cloud classification, which is combined multi-scale feature extraction with manifold learning-based dimensionality reduction. Mustaqeem et al. [42] propose a one-dimensional dilated convolutional neural network (DCNN) architecture for the SER system. The proposed framework uses the dilated convolution layer (DCL) in order to easily enhance the usage of the features and to improve the current baseline methods.

### 2.4. Attention Mechanism

In recent years, the self-attention mechanism has made remarkable achievements in the field of computer vision. It has become an essential part that can capture long-term dependencies. The self-attention mechanism ignores irrelevant features through the score function and focuses on crucial features. Mustaqeem et al. [43] propose a self-attention module (SAM) for the SER system, which is the first time use attention mechanism in the SER domain. The experiments on speech emotional databases prove the effectiveness of the SER system. Vaswani et al. [44] propose a model architecture, which entirely relies on an attention mechanism to draw global dependencies between input and output. Wang et al. [45] present nonlocal operations to capture long-range dependencies in video sequences and explain that self-attention can be viewed as a form of the nonlocal mean. Inspired by the self-attention, two critical components of PointASNL [46], which are the adaptive sampling (AS) module and local-nonlocal (L-NL) module, are proposed. PointASNL can deal with point cloud with noise effectively and achieve suitable performance by combining local neighbors and global context interaction.

## 3. Methods

Firstly, the multi-scale framework for 3D point cloud classification with hierarchical architecture is presented (in Section 3.1). Secondly, the local-global module, which can effectively extract local and global geometric information (in Section 3.2). The model can be plugged into existing deep neural networks. Thirdly, the local-global layer (LGLayer), which is composed of  $M$  ( $M = 3$ )-independent local-global module, can generate multi-scale features (in Section 3.3). Finally, the improved method called IMLP is explained to obtain more about both low-level and high-level features (in Section 3.4). In the following subsections, each cell in the pipeline is introduced in detail. There are many symbols and notations in each cell. In order to understand, we have added a dedicated table to define all these symbols and notations in supplementary. The notations and definitions are shown in Table A1.

### 3.1. Our Proposed AMS-Net

Motivated by multi-scale structure, a novel attention-based multi-scale neural network named AMS-Net is proposed, which is illustrated in Figure 2. The input of our network is a raw point set  $\chi = \{x_i \in \mathbb{R}^{3+c_{in}}, i = 1, 2, \dots, N\}$ , where  $N$  is the size of point cloud  $\chi$ . Each point is composed of a 3D coordinate  $(x, y, z)$  and other features (e.g., RGB, normal, etc.). The main components of our hierarchical structure AMS-Net are MS-BLOCK and FC layer. On each level, the module MS-BLOCK has two components: LGLayer and IMLP. Firstly,  $N_{FPS}$  points  $\chi_{FPS} = \{x_1, \dots, x_i, \dots, x_{N_{FPS}}\}$  are selected to define the centroids of local regions by the farthest point sampling (FPS). After that, LGLayer is used to capture abundant local geometric information and share geometric features with distant points in each scale, respectively. As shown in Figure 2b, we know that LGLayer is consisted of  $m$ -independent local-global module to generate a multi-scale feature with  $M \times c_{out}$  channels. The output point cloud  $mlg$  is concatenated by the extracted point cloud  $slg$  from each local-global module. The structure of the local-global module is shown in Figure 2c. Then, point cloud  $g1$  and  $g2$  are obtained by the proposed model IMLP, which can contain both low-level and high-level features. Finally, the learned global feature  $G$  is obtained by the connection of the former two levels, which can be applied to shape classification. In summary, the proposed framework can exhibit impressive performance in the point cloud classification by hierarchical multi-layer learning.

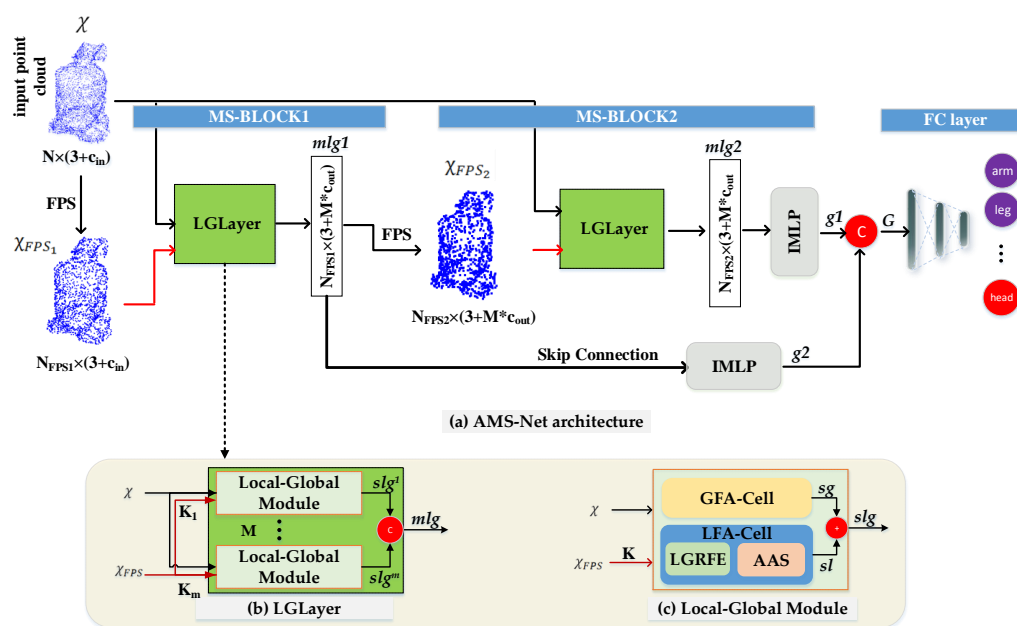
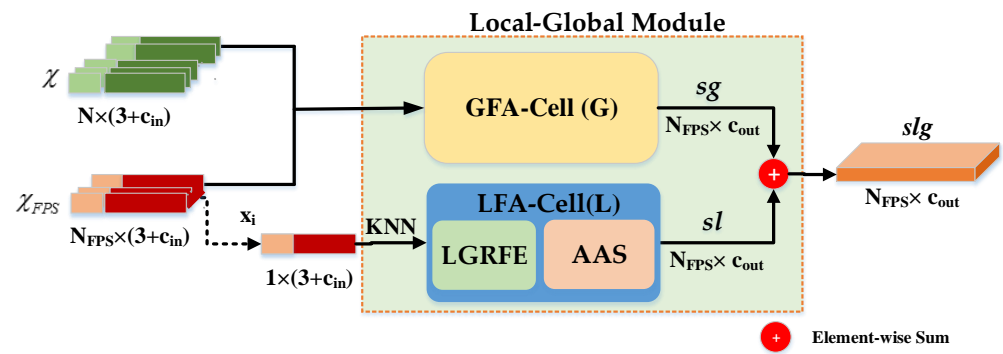


Figure 2. The framework of our AMS-Net to classify the Terracotta Warrior fragments.

### 3.2. Local-Global Module

Figure 3 illustrates our proposed local-global module, which has two key components: local features aggregated cell (LFA-Cell)  $\mathcal{L}$  and global features aggregated cell (GFA-Cell)  $\mathcal{G}$  based on self-attention. In LFA-Cell, for each sampled point, the  $K$  nearest neighbor (KNN) searching is employed to find extract the features of all neighbor points. LFA-Cell is composed of two parts: local geometric relation and features encode (LGRFE) and attentive aggregation sub-unit (AAS) and can effectively learn complex local structures. GFA-Cell can capture long-range dependencies. The local-global module can well extract the structural and semantic features in local and global sections. More details are introduced in Sections 3.2.1 and 3.2.2.

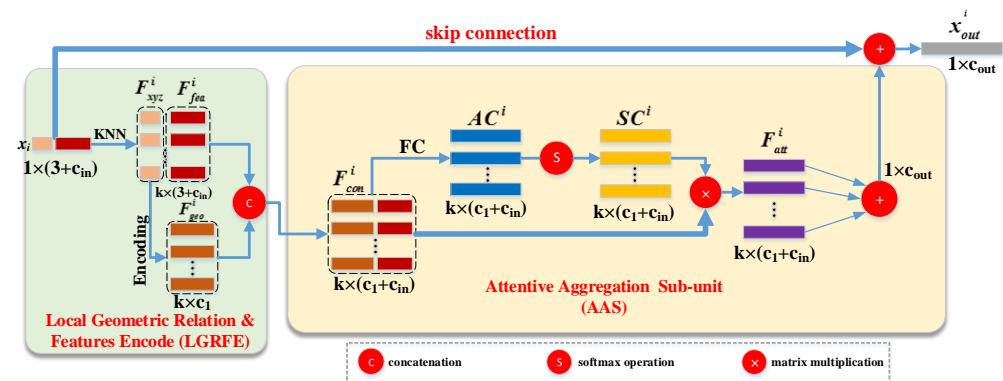


**Figure 3.** Local-global module.  $N$  denotes the size of the point cloud  $\chi$ , and  $N_{FPS}$  denotes the size of sampled point cloud  $\chi_{FPS}$ .  $c_{in}$ ,  $c_{out}$ , and  $c_{final}$  stand for numbers of dimensions except for  $xyz$ -coordinates.

### 3.2.1. Local Features Aggregated Cell (LFA-Cell)

Given the entire point cloud  $\chi = \{x_i \in \mathbb{R}^{3+c_{in}}, i = 1, 2, \dots, N\}$ , where the number 3 denotes the dimensions of 3D coordinates, and  $c_{in}$  is the number of dimensions in per-point features except for coordinates (e.g., RGB, normal, etc.). Here,  $c_{in} = 3$  is used to denote the normal vector as per-point additional features. For each point  $x_i$ , its total feature vector  $F_{\chi}^i = \{f_1^i, \dots, f_j^i, \dots, f_N^i\}$ ,  $f_i \in \mathbb{R}^{3+c_{in}}$  consists of two parts: coordinate feature vector  $F_{xyz}^i = \{f_{xyz_1}^i, \dots, f_{xyz_j}^i, \dots, f_{xyz_N}^i\}$ ,  $f_{xyz_j}^i \in \mathbb{R}^3$  and additional feature vector  $F_{fea}^i = \{f_{fea_1}^i, \dots, f_{fea_j}^i, \dots, f_{fea_N}^i\}$ ,  $f_{fea_j}^i \in \mathbb{R}^{c_{in}}$ . The sampled point cloud can be denoted as  $\chi_{FPS} \in \mathbb{R}^{N_{FPS} \times (3+c_{in})}$ , where  $N_{FPS}$  is the size of the current sampled point cloud.

As illustrated in Figure 4, the LFA-Cell has two key components: local geometric relation and features encode (LGRFE) and attentive aggregation sub-unit (AAS). The overall procedure of LFA-Cell is described as follows:



**Figure 4.** Local features aggregated cell.

Firstly, the  $k$ -neighboring points of the point  $x_i$  are obtained by the KNN method, which is denoted as  $Neig(x_i) = \{x_j^i \mid j = 1, 2, \dots, k\}$ , where  $x_j^i$  is the  $j$ th point of the  $k$ -neighboring points of the point  $x_i$  (namely,  $x_i \in \mathbb{R}^{3+c_{in}} \rightarrow Neig(x_i) \in \mathbb{R}^{k \times (3+c_{in})}$ ). Secondly, the local geometric features are re-encoded from the 3D coordinates of the  $k$ -neighboring points, and the dimensions of geometric features are changed into  $c_1$  (namely,  $Neig(x_i) \in \mathbb{R}^{k \times (3+c_{in})} \rightarrow Neig(x_i) \in \mathbb{R}^{k \times (c_1+c_{in})}$ ). A mid-dimensional feature vector can be denoted as  $f_{con}^i \in \mathbb{R}^{c_1+c_{in}}$ , which is obtained by features of the local geometric positions and its  $k$ -nearest neighbors. Thirdly, a high-dimensional feature is aggregated from the obtained  $k$  mid-dimensional features by using the AAS (namely,  $Neig(x_i) \in \mathbb{R}^{k \times (c_1+c_{in})} \rightarrow x_{out}^i \in \mathbb{R}^{c_{out}}$ ). Finally, the output point  $x_{out}^i$  is obtained by skip connections with the final feature of  $1 \times c_{out}$ . The two sub-units are described in the following:

### 1.→ Local Geometric Relation and Features Encode (LGRFE)

To effectively learn complex local structures, how to represent point cloud should be the primary consideration. For a point  $x_i$ , its absolute and relative positions are incomplete as local neighborhood information. It should also be represented by all the points within its  $k$ -nearest neighbors of point  $x_i$ , and the Euclidean distances between point  $x_i$  and its neighbors. Combining the former four components, more comprehensive local geometric features can be obtained. The  $k$ -nearest neighbors of point  $x_i$  can be denoted as  $\{x_1^i, \dots, x_j^i, \dots, x_k^i\}$ , and the corresponding coordinate feature vector can be denoted as  $F_{xyz}^i = \{f_{xyz_1^i}, \dots, f_{xyz_j^i}, \dots, f_{xyz_k^i}\}$ . The encoded local geometric feature is defined as Equation (1):

$$f_{geo_j^i} = \mathcal{M}\left(C\left(f_{xyz^i}, (f_{xyz_j^i} - f_{xyz^i}), f_{xyz_j^i}, \sqrt{(f_{xyz_j^i} - f_{xyz^i})^2}\right)\right), j = 1, 2, \dots, k \quad (1)$$

where  $C$  denotes concatenation operation, and  $\mathcal{M}$  indicates the function conducted by the MLP. This process contributes to learn more comprehensive local features and obtain suitable performance. Therefore, the encoded local geometric feature vector of point  $x_i$  can be denoted as  $F_{geo}^i$ .

For each neighboring point  $x_j^i$ , a synthesized feature vector  $f_{con_j^i}$  is obtained by connecting the encoded local geometric feature  $f_{geo_j^i}$  with its corresponding additional feature  $f_{fea_j^i}$ . Finally, a new encoded neighboring feature vector  $F_{con}^i = \{f_{con_1^i}, \dots, f_{con_j^i}, \dots, f_{con_k^i}\}$ ,  $f_{con_j^i} \in \mathbb{R}^{1 \times (c_1 + c_m)}$  is formed.

### 2.→ Attentive Aggregation Sub-unit (AAS)

The key idea of AAS is to aggregate the features of  $k$ -neighboring points. Given a set of the feature vector  $F_{con}^i = \{f_{con_1^i}, \dots, f_{con_j^i}, \dots, f_{con_k^i}\}$ , which is extracted from LGRFE. A single fixed output  $x_{out}^i \in \mathbb{R}^{1 \times c_{out}}$  is formed by AAS. The main steps of AAS are as follows:

First, the set of the feature vector  $F_{con}^i$  is fed into a shared function  $\mathcal{T}$ . For less computation,  $\mathcal{T}$  is the form of a linear transformation of point features. A set of the new feature vector  $AC^i = \{ac_1^i, \dots, ac_j^i, \dots, ac_k^i\}$  is obtained by an FC layer without bias. That is,  $ac_j^i = \mathcal{T}(f_{con_j^i}, W) + b$ , where  $W$  is learnable weight and  $b = 0$ . In the above formulation,  $f_{con_j^i} \in \mathbb{R}^{1 \times (c_1 + c_m)}$ ,  $W \in \mathbb{R}^{(c_1 + c_m) \times (c_1 + c_m)}$ , and  $ac_j^i \in \mathbb{R}^{1 \times (c_1 + c_m)}$ .

Then, the learned attention score vector  $SC^i = \{sc_1^i, \dots, sc_j^i, \dots, sc_k^i\}$  is normalized by softmax operation. The  $j$ th element of  $SC^i$  is defined as:

$$sc_j^i = \frac{\exp(ac_j^i)}{\sum_k \exp(ac_k^i)} \quad (2)$$

Moreover, the feature vector  $F_{att}^i = \{f_{att_1^i}, \dots, f_{att_j^i}, \dots, f_{att_k^i}\}$  is weighted summed as follows:

$$f_{att_j^i} = f_{con_j^i} \times sc_j^i \quad (3)$$

Finally, to avoid losing the low-level features, a skip connection is used to combine the newly aggregated features with the raw features. The final output point  $x_{out}^i$  is obtained with the size of  $1 \times c_{out}$ .

### 3.2.2. Global Features Aggregated Cell (GFA-Cell) Based on Self-Attention

As mentioned in LFA-Cell, the  $\chi_{FPS}$  denotes a sampled point cloud, and the corresponding feature vector is  $F_{\chi_{FPS}} = \{f_1, \dots, f_i, \dots, f_{N_{FPS}}\}$ ,  $f_i \in \mathbb{R}^{3 + c_m}$ . The overall process



of global features aggregated based on general self-attention is shown in Figure 5. The sampled set  $\chi_{FPS}$  with the size of  $N_{FPS}$  and the entire point cloud  $\chi$  can be regarded as query points and key points individually. To reduce the computation of the cell, bottleneck layers are added. In this work, each bottleneck layer's size is set to be half of the output channels ( $c_{mid} = 1/2 c_{out}$ ). We compute the dot products of the query points with key points, with scaling by  $c_{mid}$ , and apply a softmax function to obtain the weights on the values, then aggregate them with function  $\mathcal{A}(\cdot)$ . Therefore, for a sampled point  $x_i$ , the convolutional operation of global features aggregated can be denoted as:

$$Atten(x_i, F_\chi) = \mathcal{A}\left(\text{softmax}\left(\frac{g(f_i)^T h(f_j)}{\sqrt{c_{mid}}}\right)\right) r(f_j), \forall f_j \in F_\chi \quad (4)$$

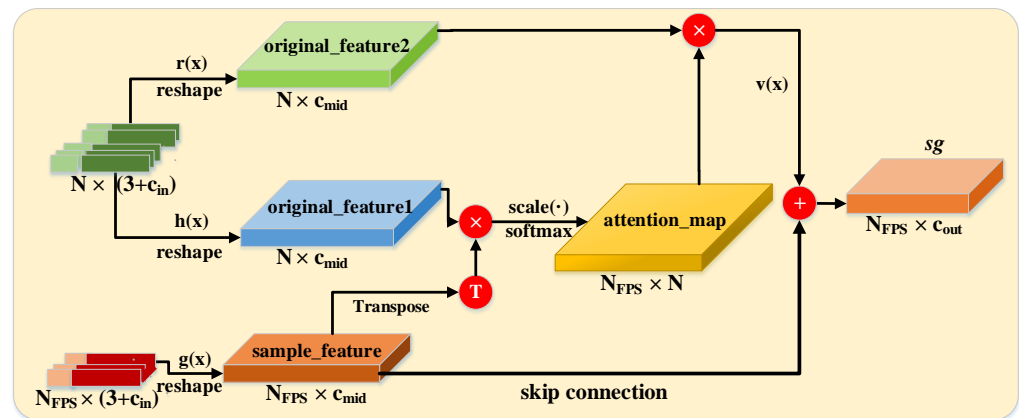


Figure 5. Global features are aggregated based on self-attention.

For simplicity,  $g$  is considered in the form of a linear, that is  $g(f_i) = W_g \cdot f_i$ , where  $W_g$  is learnable weight, “ $\cdot$ ” denotes element-wise multiplication.  $h(\cdot)$  and  $r(\cdot)$  are also linear functions. The updated global feature of the sampled point  $x_i$  can be written as:

$$f_{glo_j}^i = v(Atten(x_i, F_\chi)) \quad (5)$$

where  $v(\cdot)$  is a nonlinear function. In the last step, to ensure the same dimension as the output of LFA-Cell, the global features are fused by  $v$ . The skip connection is also used to combine the generated global features with the raw features. The output vector with  $N_{FPS} \times c_{out}$  is obtained. Therefore, GFA-Cell can break the limitations of local regions and capture more long-range dependencies.

### 3.3. LGLayer

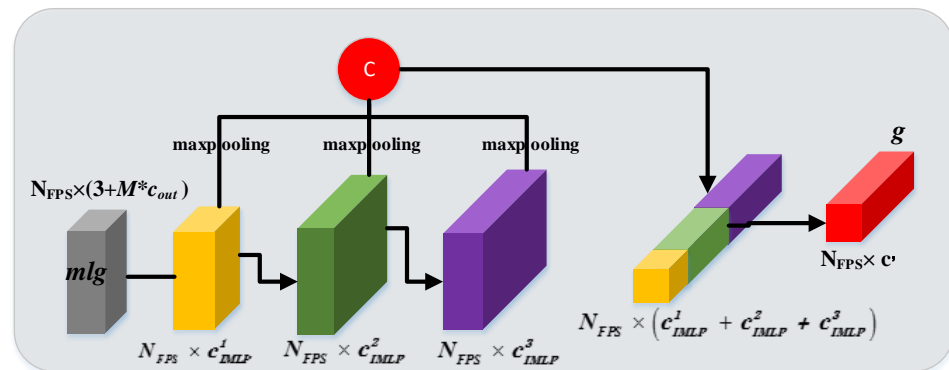
According to the explanation in Section 3.2, the output feature of the local-global module can be written as:

$$f_{slg} = \delta\left(\sum_{i \in N_{FPS}} \sum_{j \in Neig(x_i)} (f_{att_j}^i + f_{glo_j}^i)\right) \quad (6)$$

where  $\delta(\cdot)$  is a nonlinear activation function,  $f_{slg}$  with size of  $N_{FPS} \times c_{out}$ . In order to obtain sufficient structural information and stabilize the network, the  $M$ -independent local-global modules are concatenated to generate a multi-scale feature with  $M \times c_{out}$  channels.  $M$  is the total number of scales, and we set  $M = 3$  in this study. As shown in Figure 6, the output of LGLayer is a multi-scale feature that concatenates the structural and semantic features, both local and global. Finally, the multi-scale feature is defined as:

$$f_{mlg} = C(f_{slg1}, \dots, f_{slgm}), m = 1, \dots, M \quad (7)$$

where  $C$  denotes concatenation operation.  $f_{slgm}$  is a concatenated feature of the  $m$ -th local-global module.



**Figure 6.** The architecture of IMLP.

### 3.4. IMLP

The input point cloud is with a size of  $N_{FPS} \times (3 + M \cdot c_{out})$ , which is obtained by multi-scale local-global modules and  $xyz$ -coordinates. The output point cloud is with a size of  $N_{FPS} \times c'$ . As the MLP can only extract the maximum value from the last layer. The maximum value is regarded as the global feature of the point cloud. However, it does not make full use of the information contained in the low-level and mid-level. The low-level features include the rich geometric structure of the original point cloud. IMLP is proposed to solve the above problems, which can aggregate features effectively. The details of IMLP are shown in Figure 6.

Different from MLP, we perform three different scales of convolutions on the features obtained from the previous layer and maximize the three-scale feature vectors' outputs. As shown in Figure 6, each point can be encoded into the dimensions of  $(c_{IMLP}^1, c_{IMLP}^2, c_{IMLP}^3)$ . The feature vector of different layers can be denoted as  $c_{IMLP}^i$ , and the combined vector  $D$  is obtained by concatenating all the  $c_{IMLP}^i$ ,  $i = 1, 2, 3$ . The feature vector  $D$  with a size of  $(c_{IMLP}^1 + c_{IMLP}^2 + c_{IMLP}^3)$ , which includes low-level, mid-level, and high-level features. Finally, the dimension of the feature is changed to  $c'$  through a convolution operation.

## 4. Experiments and Results

### 4.1. Data Set and Implementation Detail

**Data set** In this section, to demonstrate the proposed framework's effectiveness and efficiency, experiments are conducted on a benchmark of point cloud classification, which is the ModelNet40/10 data set of CAD models. ModelNet40 and ModelNet10 comprise 9843/3991 training objects and 2468/908 test objects in 40 and 10 classes, respectively.

For the real-world data set, the Terracotta Warrior fragments' quantity is large, and the structure is complex. Meanwhile, the fragments always vary in shape. To prevent secondary damage to the Terracotta Warrior fragments during the restoration process, the point cloud models of the Terracotta Warrior fragments are obtained by using Creaform VIU 718 handheld 3D scanners from Canada. Figure 7 shows some point cloud models we used in the experiments. However, the scanning data generally have holes and noise, and data preprocessing must be performed to ensure the accuracy of fragments classification. Data preprocessing usually includes three steps: noise removal, hole filling, and simplification. We first use Geomagic software (Geomagic Wrap, Shenzhen, China) to remove noise points and repair holes manually. For example, 006192-Arm-38 is chosen from the models of Terracotta Warrior fragments. The preprocessed model of 006192-Arm-38 is shown in Figure 8a, and then the mesh model is converted to a point cloud model (see Figure 8b). A total of 26,923 points are redundant for processing, and the raw point model should be reduced by random sampling. Figure 8c shows the sampled point cloud, which has been down to 40%. In this experiment, 11,996 point cloud patches train the network

extracted from 40 whole Terracotta Warriors. The proposed network has a limited size of the point cloud (e.g., 2048 points or 1024 points), and ten thousands of points in model 006192-Arm-38 cannot be directly used to be input point cloud. The sampled model (e.g., model in Figure 8c) should be divided into patches; each patch contains fixed numbers of 2048 by uniform sampling. Figure 9 presents several patches of different parts in model 006192-Arm-38. For the Terracotta Warrior fragments data set, there are four categories: Arm, Body, Head, and Leg. Among them, 10,144 patches for training (Arm: 2656, Body: 2720, Head: 2272, Leg: 2496) and remained 1852 for testing (testArm: 476, testBody: 504, testHead: 428, testLeg: 444). All the training and testing data are the same preprocessed data set as [24]. For training, we sample 1024 points and normalize them into a unit ball as input. The point cloud is augmented by randomly rotating, jittering each point's position by Gaussian noise with zero mean and random dropout 20% points.



Figure 7. Several fragments displayed by category.

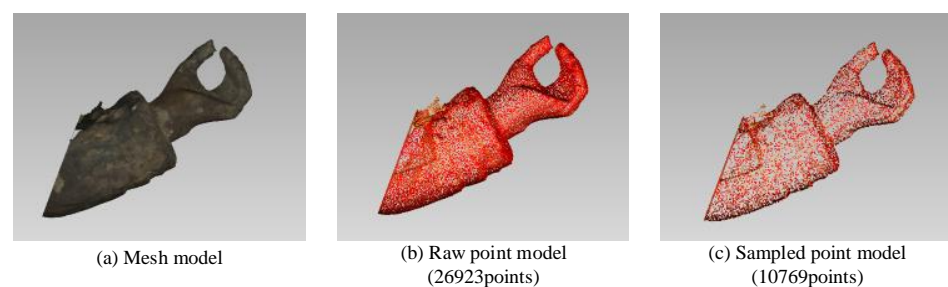
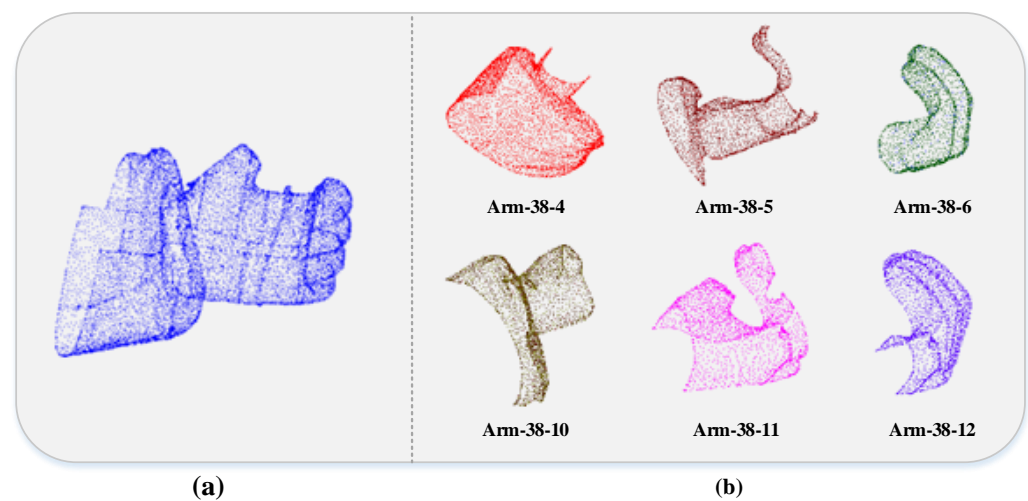


Figure 8. Sampling results of model 006192-Arm-38.



**Figure 9.** Point cloud patches of model 006192-Arm-38. (a) Raw point cloud model of 006192-Arm-38; (b) patch models of 2048 points.

**Architecture** The 3-layer network architecture for classification is shown in Figure 2. In the first two layers, the input point cloud needs to be down-sampled by the FPS, and MS-BLOCK extracts features from different scales. In IMLP, a global feature vector is obtained by concatenating features from IMLP. Finally, the final features are encoded by 3-layer MLPs of size (512, 256, 4) to perform point cloud classification. Random dropout is applied to the last two layers with a keep-ratio of 0.4. The configuration of our AMS-Net in point cloud classification is shown in Table 1.

**Table 1.** Architecture configurations.  $N_{down}$  denotes the number of sampled points by FPS;  $K$  is the number of group neighbors in LFA-Cell;  $mlp$  indicates a list for MLP construction in layers;  $c_{out}$  is the dimension of output (in Figures 4 and 5), which determines the size of bottleneck and scale;  $mlp_{adv}$  denotes a list for IMLP.

Layer	$N_{down}$	$K$	$mlp$	$c_{out}$	$mlp_{adv}$
1st. MS-BLOCK1	512	16	(32, 32, 6)	64	-
		32	(64, 64, 128)	128	-
		64	(64, 96, 128)	128	-
2nd. MS-BLOCK2	256	32	(64, 64, 128]	128	-
		64	(128, 128, 256)	256	-
		128	(128, 128, 256)	256	-
3rd.	1	-	-	-	(320, 384, 512)
					(640, 768, 1024)

**Training** All experiments are implemented in the following hardware: a 3.2 GHz AMD Ryzen 7 2700 Eight-Core Processor with 16 GB of Kingston Impact 2666 MHz and CL10 DDR4 RAM on an Asus TUF GAMING B550M-PLUS motherboard. We trained our AMS-Net for 251 epochs on an NVIDIA GTX 1080Ti GPU and TensorFlow v1.13 using Adam optimizer with an initial learning rate of 0.001, the decay rate of 0.1 every 500 K steps, the momentum of 0.9, and a batch size of 8. The decay rate for batch normalization starts at 0.5 and is gradually increased to 0.99. Batch-normalization and ReLU activation are applied after each layer except the last fully connected layer.

#### 4.2. ModelNet40/10 Classification

##### 4.2.1. Comparing with Other Methods

We compare our AMS-Net with several methods in 3D shape classification on ModelNet 40/10, respectively. The representations of input data are voxel or point cloud. As

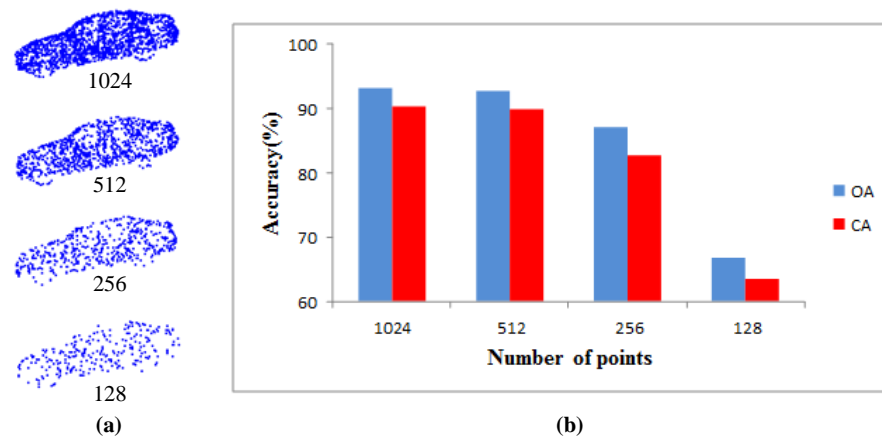
known to all, the computational costs increase exponentially when 3D data are rasterized into voxel representations, and most computations are redundant as the sparsity of 3D data. As a form of scalability, regular structures of octree are suitable for deep learning techniques. However, as illustrated in Table 2, our method outperforms the networks using octree or voxel grids as input by 2.34% and 1.61%, respectively, in terms of instance accuracy for ModelNet40 (e.g., O-CNN (90.6%) and VRN (91.33%)). Interestingly, our AMS-Net (92.94%) using 1024 points still outperforms the previous models such as Kd-Net (32k points, 91.8%), DeepSets (5k points, 90.0%), SO-Net (2k points, 90.9%). Compared with the *xyz*-input networks with 1024 point cloud, our AMS-Net also shows suitable performance. Our AMS-Net outperforms PointCNN, PointASNL by 0.74% and 0.04%, respectively. For using a normal vector, our method outperforms the methods shown in Table 2 except for RS-CNN and point transformer. Our AMS-Net achieves competitive performance with 0.32% higher accuracy than PointASNL and surpasses the methods that include more points (5k), such as PointNet++ and SO-Net.

**Table 2.** Classification results on ModelNet40/10. (“CA” stands for per-class accuracy; “OA” stands for overall accuracy; “pnt” stands for *xyz*-coordinates of point and “nor” stands for surface normal vector; “-” stands for unknown.).

Method	Representation	Input	ModelNet10 (%)		ModelNet40 (%)	
			CA	OA	CA	OA
O-CNN [47]	Octree	$64^3$	-	-	-	90.60
VRN [21]	Voxel	-	-	93.61	-	91.33
Kd-Net [48] depth = 15	pnt.	$2^{15} \times 3$ (32k)	93.50	94.0	88.50	91.80
DeepSets [49]	pnt.	$5000 \times 3$	-	-	-	90.00
SO-Net [36]	pnt.	$2048 \times 3$	-	-	-	90.90
PointNet [24]	pnt.	$1024 \times 3$	-	-	86.20	89.20
PointNet++ [27]	pnt.	$1024 \times 3$	-	-	-	90.70
PointCNN [28]	pnt.	$1024 \times 3$	-	-	-	92.20
RS-CNN [35]	pnt.	$1024 \times 3$	-	-	-	93.60
PointASNL [46]	pnt.	$1024 \times 3$	-	95.70	-	92.90
AMS-Net (Ours)	pnt.	$1024 \times 3$	-	95.83	-	92.94
PointNet++ [27]	pnt., nor.	$5000 \times 6$	-	-	-	91.90
SO-Net [36]	pnt., nor.	$5000 \times 6$	95.50	95.70	90.80	93.40
Point transformer [50]	pnt., nor.	$1024 \times 6$	-	-	90.6	93.70
PointASNL [46]	pnt., nor.	$1024 \times 6$	-	95.90	-	93.20
AMS-Net (Ours)	pnt., nor.	$1024 \times 6$	-	95.91	-	93.52

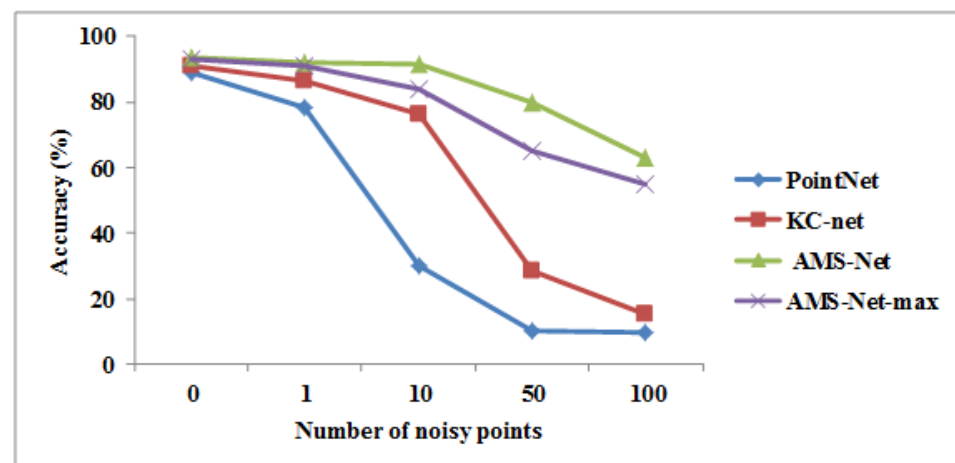
#### 4.2.2. Robustness Test

To evaluate the robustness of our AMS-Net on point cloud density, we train our network with 1024 points and test it with different sizes of sparser points. Random input points obtain the numbers 512, 256, and 128 of test data drop out. As shown in Figure 10a, it is hard to identify the overall shape and obtain geometrical and locational relations of the point cloud when the points become sparse. Figure 10b indicates that the number of points is reduced by half; the model can still obtain suitable results. If there are too few points (e.g., less than 256), the accuracy drops sharply.



**Figure 10.** Test results of robustness. (a) Point cloud with random point dropout; (b) Our AMS-Net results with sparser points of (a) as the input, and our model is trained with 1024 points.

To further verify the ability to deal with noise, we do the experiment like PointNet [24] and KC-Net [28] on random noise in the input point cloud. To achieve fair comparisons, the same training set and test set are the same with KC-Net, which replaces a certain number of randomly selected points with random noise ranging  $(-1.0, 1.0)$  during testing. The comparisons with PointNet and KC-Net are shown in Figure 11. The accuracy of PointNet drops 58.6% when 10 points are replaced with random noise, and KC-Net drops 23.8%, while our AMS-Net only drops 4.08% (from 93.52% to 91.6%). As shown in Figure 11, our AMS-Net is relatively robust to noise. The decrease in accuracy becomes larger when AAS is replaced with max-pooling in the LFA-Cell.



**Figure 11.** Classification results of different models with noisy points. AMS-Net-max model is replaced the AAS with max-pooling in the LFA-Cell.

#### 4.2.3. Complexity Analysis

To demonstrate the effectiveness of our model, we compare complexity to other methods in terms of model size and forward time. The forward time is recorded under the same conditions with batch size 8, a single GTX 1080 Ti, and 1024 points as the input. Table 3 shows AMS-Net can achieve a tradeoff between the model complexity and computational complexity. However, due to the multi-scale strategy, the forward time of our AMS-Net and PointNet++ (MSG) is longer than PointNet and PointNet++ (SSG), which are based on single-scale grouping. However, our AMS-Net achieves the pleased classification accuracy among the models listed in Table 3.

**Table 3.** Complexity and forward time of different models on ModelNet40.

Method	Model Size (MB)	Time (MS)
PointNet	40.1	17.6
PointNet++ (SSG)	8.3	82.4
PointNet++ (MSG)	12.0	165.0
AMS-Net (Ours)	17.2	112.8

### 4.3. Results of Real-World Data

#### 4.3.1. Shape Classification

To further verify that our AMS-Net can obtain a state-of-the-art classification result on 3D Terracotta Warrior fragments, some methods are employed to be the baseline. As shown in Table 4, the highest mean accuracy of the existing traditional methods is 87.64%. Our AMS-Net without normal (95.68%) can achieve competitive performance with 8.04% higher accuracy than the best traditional classification method, which shows its great potential for real applications. Compared with PointNet, our AMS-Net improves accuracy by 6.75%. With the normal vector, the mean accuracy is up to 96.22%. In [26], a dual-modal that incorporated geospatial and texture information of the fragments is proposed. However, the accuracy only reaches 91.41% with the complex algorithm. Our AMS-Net is not only simple and effective but also improves the accuracy by 4.27%, which is attributed to the two strategies of multi-scale and self-attention mechanism. Results can prove that our proposed method is more suitable for the characteristic of the Terracotta Warrior fragments and can achieve a suitable classification result.

**Table 4.** Comparison with the methods proposed in the references.

Method	Input Data Type	Deep Model	OA (%)
Method in [9]	image	F	74.66
Method in [31]	image	F	84.34
Method in [10]	image	F	86.86
Method in [19]	image (cnn-based)	T	89.54
Method in [32]	pnt.	F	87.64
PointNet [24]	pnt.,	T	88.93
Method in [25]	pnt.,	T	90.94
Method in [26]	pnt., image	T	91.41
Ours	pnt.	T	95.68
Ours	pnt., nor.	T	96.22

Without normal, the classification accuracies of the four classes are 98.1% (Body), 98.0% (Head), 94.2% (Leg), and 92.4% (Arm). Figure 12 show some representative fragments of the four classes. From the results, we know that the accuracy of class Body is the highest, while the accuracy of class Arm is the lowest. The main reason is that most of the body parts are wearing armor, or the clothes have more folds. The characteristics of class Body are more obvious in general (see in Figure 12). As shown in Figure 13, there are many distinctive characteristics in eyes, nose and headwear. The result is slightly lower than class Body. The characteristics of class Arm are similar to class Leg. The two fragments in the upper row of Figure 14 are from class Arm. The two fragments in the bottom row are from class Leg. The features of class Leg are relatively smooth. If the fragment from class Arm with smooth, it would be misclassified as Leg. The result is shown in Figure 14e.

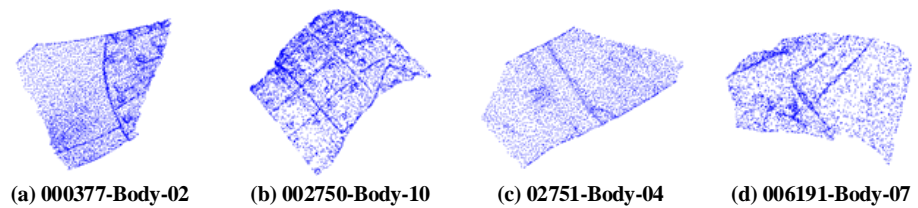


Figure 12. Some fragments of class Body.

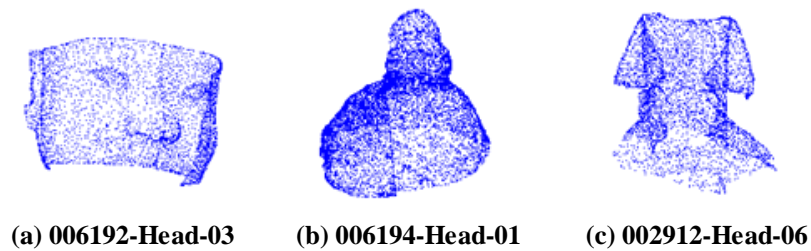


Figure 13. Some fragments of class Head.

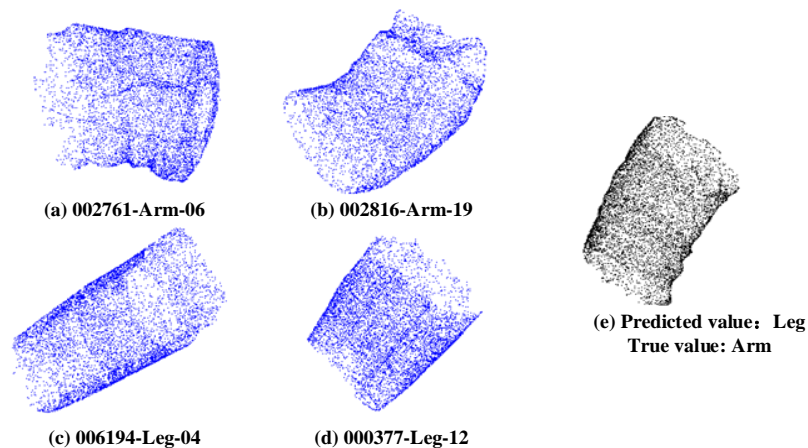


Figure 14. Some fragments of class Arm (a,b) and Leg (c,d). (e) Misclassification result.

#### 4.3.2. Shape Classification with Noise

Even though the coordinate value of Terracotta Warrior fragments is not normalized, the accuracy of the classification is considerable. The above method of adding random noise of  $(-1.0, 1.0)$  to Modelnet40 does not apply to the 3D Terracotta Warrior fragments. So the Gaussian noise with a mean of 0 and a variance of 1 is added to the input point cloud. Figure 15 shows the arm with 10 points being replaced with random noise. The accuracy of our AMS-Net is 91.6% when 1 point is replaced with noise point ( $N_{noise} = 1$ ), where  $N_{noise}$  is the number of noise points. The remaining results are 90.82% ( $N_{noise} = 10$ ), 79.3% ( $N_{noise} = 50$ ) and 67.27% ( $N_{noise} = 100$ ). The accuracy drops only 4.86% when 10 points are replaced with noise points (from 95.68% to 90.82%).

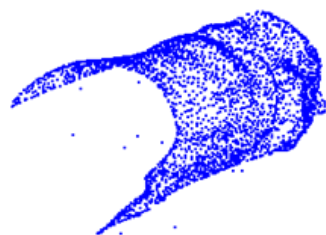


Figure 15. Point cloud with some points being replaced with Gaussian noise.



#### 4.4. Ablation Study

The subsection conducts ablation experiments on 3D Terracotta Warrior fragments to further evaluate each cell's effectiveness in our framework. The input data is a point cloud with a normal vector.

##### 4.4.1. Experiments of Partial Detail Setting in LFA-Cell

###### 1. Ablation Studies on LGRFE

As a critical cell for extracting local information, LGRFE concatenates much spatial information to obtain local relationships. Five different forms of encoding local geometric features are tested, and the representative symbols are the same as those defined  $f_{geo}^i$  in Equation (1). As summarized in Table 5, we can see that model E has excellent classification performance, which contains full spatial information. The relative distance significantly influences obtaining the local information, so model D is the suboptimal model. On the contrary, the coordinates of a single point cannot show the local spatial relationship well; hence model A has the lowest accuracy.

**Table 5.** The results (%) of five forms of encoding local geometric features. model A applies only the coordinates of the point  $x_i$  as geo; model B uses the coordinates of the point  $x_i$  and its neighboring points  $x_j^i$ ; model C adds the Euclidean distance to model B; model D adds the relative distance to model B; model E contains all spatial information mentioned above.

Model	Local Geometric Feature Definition	Channels	Acc.
A	$(f_{xyz}^i)$	3	93.82
B	$(f_{xyz}^i, f_{xyz_j}^i)$	6	93.95
C	$(f_{xyz}^i, f_{xyz_j}^i, \sqrt{(f_{xyz_j}^i - f_{xyz}^i)^2})$	7	94.87
D	$(f_{xyz}^i, f_{xyz_j}^i, (f_{xyz_j}^i - f_{xyz}^i))$	9	95.05
E	$(f_{xyz}^i, f_{xyz_j}^i, (f_{xyz_j}^i - f_{xyz}^i), \sqrt{(f_{xyz_j}^i - f_{xyz}^i)^2})$	10	95.68

###### 2. Ablation Studies on AAS

To verify the proposed AAS unit's effect on aggregation features, there are two symmetric functions: max-pooling (Max) and average-pooling (Avg). As shown in Table 6, we can see that our AAS achieves the best performance. The reason may be that our AAS uses the attention mechanism to combine all local point features. By comparison, other methods often lead to most information lost, and they are more difficult to aggregate neighborhood features. The experimental results also show the effectiveness of the self-attention mechanism.

**Table 6.** Accuracy results (%) of different aggregation feature methods.

Aggregation	Max	Avg	AAS
Acc (%)	94.27	94.06	95.68

##### 4.4.2. Experiment of IMLP

IMLP can make full use of the low-level and high-level information of the original point cloud. To demonstrate the effectiveness of IMLP, two extractors are compared with IMLP. The results are shown in Table 7. MLP denotes the widely used way in PointNet; CMLP is proposed in [51]. Compared with the CMLP, our IMLP adds a feature fusion step to better use high-level and low-level information. As shown in Table 7, we can see that our IMLP obtains the best accuracy of 95.68%.

**Table 7.** Accuracy results (%) of IMLP VS. MLP/CMLP.

	MLP	CMLP	IMLP
Acc (%)	93.64	95.15	95.68

#### 4.4.3. Single-Scale vs. Multi-Scale

To verify the effectiveness of multi-scale, Table 8 shows the single-scale and multi-scale model's accuracy results. ASS-Net denotes the single-scale model, which has a similar structure to our MS-Net. The specific parameters of ASS-Net are defined as follows: the number of sampled points is 512 in the first level, the neighbor point  $k_1$  is 32, and a set of MLPs is (64, 64, 128) to abstract features into higher dimensions; in the second level, the sampled point is 128,  $k_2 = 64$ , and MLPs is (128, 128, 256). It can be seen from Table 8 that our approach outperforms ASS-Net by 1.69%, which is owing to our network can extract multi-scale detail features effectively.

**Table 8.** Accuracy results (%) of ASS-Net vs. AMS-Net.

	ASS-Net	AMS-Net
Acc (%)	93.99	95.68

## 5. Conclusions and Future Direction

As one of the great discoveries in the history of archaeology in the 20th century, Terracotta Warriors have become an important channel for spreading Chinese culture. To avoid secondary damage caused by manual repair, virtual splicing and repair have become a research hotspot. However, most of the current methods are based on traditional methods and have low accuracy. As a critical step in cultural relic restoration, the accuracy of the computer-aided fragments classification can directly affect the matching and splicing efficiency.

In this paper, combined the self-attention mechanism with a multi-scale structure, we proposed a dynamic fusion framework, which mainly focuses on improving the classification accuracy by using the complex local structures and long-range dependencies. Firstly, an effective method of local feature aggregation that can capture the local geometric features is proposed. The proposed local operator combines four types of geometric features, e.g., the coordinates of a point, the relative distance, and the Euclidean distance. Thus, LFA-Cell can contain rich local information. Furthermore, to obtain more about the point cloud model's overall structure, GFA-Cell based on self-attention is presented. Then, the local-global module is integrated by the above two cells, which can be plugged into the existing deep neural networks. LGLayer consists of  $M$ -independent local-global module, which can obtain multi-scale features. We evaluate our AMS-Net (with normal) over ModelNet40, and real-world Terracotta Warrior fragments data set, which achieves 93.52% and 96.22% accuracy, respectively. The experimental results show that the suggested model outperforms many previous methods and can obtain a state-of-the-art classification accuracy for the 3D Terracotta Warrior fragments. In summary, our AMS-Net can achieve improved performance on point cloud classification. Meanwhile, it is the first attempt to apply our AMS-Net to the real-world Terracotta Warrior fragments data set. Experiments have verified the suitability for real-world Terracotta Warrior fragments applications. We also hope this work can provide a new way for the classification of cultural relics.

However, there are still shortcomings in our method. Although the classification accuracy of our AMS-Net has been improved to a certain extent, the approach is only able to be trained and operate over a fixed-size point cloud, which is generally 2048 or 1024. When the point number is larger than the fixed size, it must be sampled to a new sparse point cloud. This will lose important geometric information, which is not conducive to learning local features. In addition, manually labeled data require the high cost of human labor and may limit the generalization ability of the learned models.

In the future, we can further design an efficient and lightweight encoder, which can directly extend to any size of large-scale point clouds without preprocessing steps such as sample to a fixed size and can effectively obtain local information. Unsupervised learning is an attractive direction to obtain generic and robust representations for the 3D Terracotta Warrior fragments. Learning useful features from unlabeled data is a challenging problem for the virtual restoration of cultural relics and is also our next main work.

**Author Contributions:** Formal analysis, X.C., P.Z., X.X., and Y.L.; Funding acquisition, Y.L. and G.G.; Investigation, G.G. and F.Z.; Methodology, F.Z. and K.L.; Software, P.Z. and X.X.; Supervision, X.C. and K.L.; Writing—original draft, J.L.; Writing—review and editing, M.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the National Key Research and Development Program of China (2019YFC1521103, 2019YFC1521102); National Natural Science Foundation of China (61701403, 61731015); Key R&D Projects in Shaanxi Province (2019ZDLSF07-02, 2019ZDLGY10-01); Key R&D Projects in Qinghai Province (2020-SF-140); China Post-doctoral Science Foundation (2018M643719); Young Talent Support Program of the Shaanxi Association for Science and Technology (20190107); Scientific Research Program Funded by Shaanxi Provincial Education Department (18JK0767).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author. The data are not publicly available due to that the Terracotta Warriors involve a policy of secrecy over cultural heritage.

**Acknowledgments:** We thank the Emperor Qinshihuang's Mausoleum Site Museum for providing the Terracotta Warriors data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

AAS	Attentive Aggregation Sub-Unit
AMS-Net	Attention-based Multi-Scale Neural Network
AS	Adaptive Sampling
CNNs	Convolutional Neural Networks
DCL	Dilated Convolution Layer
DCN	Dilated Convolution Network
DCNN	Dilated Convolutional Neural Network
EMD	Earth Mover'S Distance
FC	Fully Connected
FPS	Farthest Point Sampling
GFA-Cell	Global Features Aggregated Cell
IMLP	Improved Multi-Layer Perceptron
KNN	K Nearest Neighbor
LFA-Cell	Local Features Aggregated Cell
LGLayer	Local-Global Layer
LGRFE	Local Geometric Relation and Features Encode
L-NL	Local-Nonlocal
MLP	Multi-Layer Perceptron
MS-BLOCK	Multi-Scale Set Abstraction Block
MSCNN	Multi-Scale Convolutional Neural Network
PFH	Point Feature Histogram
SAM	Self-Attention Module
SER	Speech Emotion Recognition
SIFT	Scale-Invariant Feature Transform
SOM	Self-Organizing Map
SVM	Support Vector Machine

## Appendix A

**Table A1.** Notations and definitions.

Notation	Definitions
$\chi$	Raw point cloud
$\chi_{FPS}$	Sampled point cloud
$N, N_{FPS}$	The number of the point cloud $\chi$ and sampled point cloud $\chi_{FPS}$ , respectively
$c_{in}$	The channel number of $\chi$ except for $xyz$ -coordinate, e.g., color, normal the number of channels
$c_{mid}$	The output channel number of each bottleneck layer's size in Figure 4
$c_{out}$	The output channel number of LFA-Cell and GFA-Cell, respectively
$c_{IMLP}^i$	The channel number of features in different layers
$c_1$	The output channel number of feature encoding
$M$	The number of scales (in Figure 2)
$F_{\chi}^i$	The total local feature vector of point $x_i$
$F_{xyz}^i, F_{fea}^i$	The coordinate feature vector and additional feature vector of point $x_i$ , respectively
$F_{geo}^i$	The encoded local geometric feature of point $x_i$
$F_{con}^i$	The new encoded neighboring feature vector of point $x_i$
$AC^i$	The new feature vector obtained by an FC layer
$SC^i$	Learned attention score vector
$x_{out}^i$	The final output point of LFA-Cell
+	Element-wise sum
c	Concatenation
s	Softmax operation
$\times$	Matrix multiplication
T	Transpose
$\mathcal{M}$	MLP (Equation (1))
$\mathcal{A}$	Aggregate function, e.g., max/avg
$g, r$ and $v$	The activation functions

## References

- Liu, Y.Z.; Tang, Y.W.; Jing, L.H.; Chen, F.L.; Wang, P. Remote Sensing-Based Dynamic Monitoring of Immovable Cultural Relics, from Environmental Factors to the Protected Cultural Site: A Case Study of the Shunji Bridge. *Sustainability* **2021**, *13*, 6042. [\[CrossRef\]](#)
- Vinci, G.; Bernardini, F. Reconstructing the protohistoric landscape of Trieste Karst (north-eastern Italy) through airborne LiDAR remote sensing. *J. Archaeol. Sci. Rep.* **2017**, *12*, 591–600. [\[CrossRef\]](#)
- Liu, Y. The Application Research of Laser Scanning System in Cultural Relic Reconstruction and Virtual Repair Technology. Master's Thesis, Chang'an University, Xi'an, China, 2012.
- Kampel, M.; Sablatnig, R. Color classification of archaeological fragments. In Proceedings of the International Conference on Pattern Recognition (ICPR), Barcelona, Spain, 3–7 September 2000; pp. 771–774.
- Qi, L.Y.; Wang, K.G. Kernel fuzzy clustering based classification of Ancient-Ceramic fragments. In Proceedings of the International Conference on Information Management and Engineering, Chengdu, China, 16–18 April 2010; pp. 348–350.
- Rasheed, N.A.; Nordin, M.J. Archaeological Fragments Classification Based on RGB Color and Texture Features. *J. Theor. Appl. Inf. Technol.* **2015**, *76*, 358–365.
- Rasheed, N.A.; Nordin, M.J. Using Both HSV Color and Texture Features to Classify Archaeological Fragments. *Res. J. Appl. Sci. Eng. Technol.* **2015**, *10*, 1396–1403. [\[CrossRef\]](#)
- Rasheed, N.A.; Nordin, M.J. Classification and reconstruction algorithms for the archaeological fragments. *J. King Saud Univ.-Comput. Inf. Sci.* **2020**, *32*, 883–894. [\[CrossRef\]](#)
- Wei, Y.; Zhou, M.Q.; Geng, G.H.; Zou, L.B. Classification of Terra-Cotta Warriors fragments based on multi-feature and SVM. *J. Northwest Univ. (Nat. Sci. Ed.)* **2017**, *47*, 497–504.
- Zhao, F.Q.; Geng, G.H. Fragments Classification Method of Terracotta Warriors Based on Region and Shape Features. *J. Geomat. Sci. Technol.* **2018**, *35*, 584–588.

11. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
12. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 3–8 December 2012; pp. 1097–1105.
13. Christoph, R.; Pinz, F.A. Spatiotemporal residual networks for video action recognition. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 9–10 December 2016; pp. 3468–3476.
14. Bian, Y.L.; Gan, C.; Liu, X.; Li, F.; Long, X.; Li, Y.D. Revisiting the Effectiveness of off-the-shelf Temporal Modeling Approaches for Large-scale Video Classification. *arXiv* **2017**, arXiv:1708.03805.
15. Mustaqeem; Kwon, S. CLSTM: Deep Feature-Based Speech Emotion Recognition Using the Hierarchical ConvLSTM Network. *Mathematics* **2020**, *8*, 2133. [[CrossRef](#)]
16. Mustaqeem; Sajjad, M.; Kwon, S. Clustering-Based Speech Emotion Recognition by Incorporating Learned Features and Deep BiLSTM. *IEEE Access* **2020**, *8*, 79861–79875. [[CrossRef](#)]
17. Mustaqeem; Kwon, S. 1D-CNN: Speech Emotion Recognition System Using a Stacked Network with Dilated CNN Features. *Comput. Mater. Contin.* **2021**, *67*, 4039–4059. [[CrossRef](#)]
18. Mustaqeem; Kwon, S. A CNN-Assisted Enhanced Audio Signal Processing for Speech Emotion Recognition. *Sensors* **2019**, *20*, 183. [[CrossRef](#)] [[PubMed](#)]
19. Wang, Y. Research on the Classification Algorithm of Terracotta Warrior Fragments Based on the Optimization Model of Convolutional Neural Network. Master's Thesis, Northwest University, Kirkland, WA, USA, 2019.
20. Maturana, D.; Scherer, S. Voxnet: A 3D convolutional neural network for real-time object recognition. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 922–928.
21. Brock, A.; Lim, T.; Ritchie, J.M.; Weston, N. Generative and Discriminative Voxel Modeling with Convolutional Neural Networks. *arXiv* **2016**, arXiv:1608.04236v04232.
22. Su, H.; Maji, S.; Kalogerakis, E.; Learned-Miller, E. Multi-view Convolutional Neural Networks for 3D Shape Recognition. In Proceedings of the International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 945–953.
23. Zhang, L.; Sun, J.; Zheng, Q. 3D Point Cloud Recognition Based on a Multi-View Convolutional Neural Network. *Sensors* **2018**, *18*, 3681. [[CrossRef](#)]
24. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
25. Gao, H.J.; Geng, G.H. Classification of 3D Terracotta Warrior Fragments Based on Deep Learning and Template Guidance. *IEEE Access* **2019**, *8*, 4086–4098. [[CrossRef](#)]
26. Yang, K.; Cao, X.; Geng, G.H.; Li, K.; Zhou, M.Q. Classification of 3D terracotta warriors fragments based on geospatial and texture information. *J. Vis.* **2021**, *24*, 251–259. [[CrossRef](#)]
27. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 5099–5108.
28. Li, Y.Y.; Bu, R.; Sun, M.C.; Wu, W.; Di, X.H.; Chen, B.Q. PointCNN: Convolution On X-Transformed Points. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montréal, QC, Canada, 3–8 December 2018; pp. 820–830.
29. Wang, Y.; Sun, Y.B.; Liu, Z.W.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic Graph CNN for Learning on Point Clouds. *ACM Trans. Graph.* **2019**, *38*, 146. [[CrossRef](#)]
30. Kang, X.Y.; Zhou, M.Q.; Geng, G.H. Classification of Cultural Relic Fragments Based on Salient Geometric Features. *J. Graph.* **2015**, *36*, 551–556.
31. Lu, Z.; Li, C.; Geng, G.; Zhou, P.; Li, Y.; Liu, Y. Classification of Cultural Fragments Based on Adaptive Weights of Multi-Feature Descriptions. *Laser Optoelectron. Prog.* **2020**, *57*, 321–329.
32. Du, G.Q.; Zhou, M.Q.; Yin, C.L.; Wu, Z.K.; Shui, W.Y. Classifying fragments of Terracotta Warriors using template-based partial matching. *Multimedia Tools Appl.* **2018**, *77*, 19171–19191. [[CrossRef](#)]
33. Karasik, A.; Smilansky, U. Computerized morphological classification of ceramics. *J. Archaeol. Sci.* **2011**, *38*, 2644–2657. [[CrossRef](#)]
34. Geng, G.H.; Liu, J.; Cao, X.; Liu, Y.Y.; Zhou, M.Q. Simplification Method for 3D Terracotta WarriorFragments Based on Local Structure and Deep Neuralnetworks. *J. Opt. Soc. Am. A* **2020**, *37*, 1711–1720. [[CrossRef](#)]
35. Liu, Y.C.; Fan, B.; Xiang, S.M.; Pan, C.H. Relation-Shape Convolutional Neural Network for Point Cloud Analysis. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 8887–8896.
36. Li, J.; Chen, B.M.; Lee, G.H. SO-Net: Self-Organizing Network for Point Cloud Analysis. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 9397–9940.
37. Wang, C.; Samari, B.; Siddiqi, K. Local spectral graph convolution for point set feature learning. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 52–66.
38. Gomez-Donoso, F.; Escalona, F.; Cazorla, M. Par3DNet: Using 3DCNNs for Object Recognition on Tridimensional Partial Views. *Appl. Sci.* **2020**, *10*, 3409. [[CrossRef](#)]

39. Hou, M.L.; Li, S.K.; Jiang, L.L.; Wu, Y.H.; Hu, Y.G.; Yang, S.; Zhang, X.D. A New Method of Gold Foil Damage Detection in Stone Carving Relics Based on Multi-Temporal 3D LiDAR Point Clouds. *Int. J. Geo-Inf.* **2016**, *5*, 60. [[CrossRef](#)]
40. Zhao, B.; Zhang, X.M.; Zhan, Z.H.; Shuiquan, P. Deep multi-scale convolutional transfer learning network: A novel method for intelligent fault diagnosis of rolling bearings under variable working conditions and domains. *Neurocomputing* **2020**, *407*, 24–38. [[CrossRef](#)]
41. Huang, R.; Hong, D.F.; Xu, Y.S.; Yao, W.; Stilla, U. Multi-Scale Local Context Embedding for LiDAR Point Cloud Classification. *IEEE Geosci. Remote Sens. Lett.* **2019**, *17*, 721–725. [[CrossRef](#)]
42. Mustaqeem; Kwon, S. MLT-DNet: Speech emotion recognition using 1D dilated CNN based on multi-learning trick approach. *Expert Syst. Appl.* **2021**, *167*, 114177. [[CrossRef](#)]
43. Mustaqeem; Kwon, S. Att-Net: Enhanced emotion recognition system using lightweight self-attention module. *Appl. Soft Comput.* **2021**, *102*, 107101. [[CrossRef](#)]
44. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 5998–6008.
45. Wang, X.L.; Girshick, R.; Gupta, A.; He, K.M. Non-Local\_Neural\_Networks. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
46. Yan, X.; Zheng, C.D.; Li, Z.; Wang, S.; Cui, S.Q. PointASNL: Robust Point Clouds Processing using Nonlocal Neural Networks with Adaptive Sampling. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 5588–5597.
47. Wang, P.S.; Liu, Y.; Guo, Y.X.; Sun, C.Y.; Tong, X. O-CNN: Octree-based Convolutional Neural Networks for 3D Shape Analysis. *arXiv* **2017**, arXiv:1712.01537. [[CrossRef](#)]
48. Klovov, R.; Lempitsky, V. Escape from Cells: Deep Kd-Networks for the Recognition of 3D Point Cloud Models. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
49. Ravanbakhsh, S.; Schneider, J.; Póczos, B. Deep Learning with Sets and Point Clouds. *arXiv* **2016**, arXiv:1611.04500v04501.
50. Zhao, H.S.; Jiang, L.; Jia, J.Y.; Torr, P.; Koltun, V. Point Transformer. *arXiv* **2020**, arXiv:2012.09164.
51. Huang, Z.T.; Yu, Y.K.; Xu, J.W.; Ni, F.; Le, X.Y. PF-Net: Point Fractal Network for 3D Point Cloud Completion. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.