IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 28, 2020, accepted August 5, 2020, date of publication August 12, 2020, date of current version August 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3016267

ADMorph: A 3D Digital Microfossil Morphology Dataset for Deep Learning

YEMAO HOU[1,2,3], XINDONG CUI[2,3,4], MARIO CANUL-KU[5], SHICHAO JIN[4,6,7], ROGELIO HASIMOTO-BELTRAN[5], (Member, IEEE), QINGHUA GUO[4,7], AND MIN ZHU[2,3,4]

[1]School of Life Science and Technology, Xidian University, Xi'an 710071, China
[2]Key Laboratory of Vertebrate Evolution and Human Origins of Chinese Academy of Sciences, Institute of Vertebrate Paleontology and Paleoanthropology, Chinese Academy of Sciences, Beijing 100044, China
[3]CAS Center for Excellence in Life and Paleoenvironment, Beijing 100044, China
[4]University of Chinese Academy of Sciences, Beijing 100049, China
[5]Centro de Investigación en Matemáticas (CIMAT), Guanajuato 36023, Mexico
[6]Plant Phenomics Research Center, Nanjing Agricultural University, Nanjing 210095, China
[7]State Key Laboratory of Vegetation and Environmental Change, Institute of Botany, Chinese Academy of Sciences, Beijing 100093, China

Corresponding author: Min Zhu (zhumin@ivpp.ac.cn)

This work was supported in part by the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant XDA19050102 and Grant XDB26000000, in part by the National Natural Science Foundation of China under Grant 41530102, and in part by the Key Research Program of Frontier Sciences, CAS, under Grant QYZDJ-SSW-DQC002.

ABSTRACT Microfossils, tiny fossils whose study requires the use of a microscope, have been widely applied in many fields of earth, life, and environmental sciences. The abundance and high diversity of microfossils, as well as the need for rapid identification, call for automated methods to classify microfossils. In this study, we constructed an open dataset of three-dimensional (3D) microfossils and proposed a deep learning-based approach for microfossil classification. The dataset, named 'Archives of Digital Morphology' (ADMorph), currently contains more than ten thousand 3D models from five classes of 410 million-year-old fishes. The deep learning-based method includes data preprocessing, feature extraction, and 3D microfossil model classification. To assess the method performance and dataset representability, we performed extensive experiments. Compared with multiview convolutional neural networks (MVCNN) (91.54%), PointNet (64.13%), and VoxNet (78.15%), the method proposed herein had higher accuracy (97.60%) on the experimental dataset. We also verified data preprocessing (92.36%) and feature extraction (97.10%). We combined them to obtain the macroaveraging accuracy of 97.60%, the highest accuracy of 100%, and the lowest accuracy of 88.78%. We suggest that the proposed method can be applied to other 3D fossils and biomorphological research fields. The fast-accumulating 3D fossil models might become a source of information-rich datasets for deep learning.

INDEX TERMS Archives of digital morphology, data preprocessing, feature extraction, 3D microfossil model classification, deep learning.

I. INTRODUCTION

Microfossils, which are generally less than 5 mm in size, contain a wealth of information from small rock samples. They are useful to the study of earth, life, and environmental sciences, such as biostratigraphy, paleoclimatology, paleoceanography, hydrocarbon exploration, and evolutionary biology [1]–[3]. The identification and classification of microfossils are fundamental tasks in micropaleontology.

Conventionally, paleontologists qualitatively identify and classify microfossils based on morphological and histological features, using microscopes or electron microscopes and making thin sections. However, these traditional methods are time consuming and require considerable expertise due to the vast quantity and high diversity of microfossils. In some cases, paleontologists apply geometric morphometrics to identify and classify fossils [4]–[6]. These methods are not fully automated, and paleontologists need to define many landmarks and indices for the measurement. Over the past few decades, paleontologists have made efforts to identify and classify microfossils in automated manners. One effort is based on expert inference for identifying microfossils such as the Visual Identification Expert System (VIDES) [7]. The other is applying multivariate

The associate editor coordinating the review of this manuscript and approving it for publication was Ge Wang.

statistical analysis for the recognition of fossil freshwater ostracodes [8]. In recent years, machine learning, especially deep learning, has shown excellent feature extraction achievements in classifying microfossils. A deep learning method can implement an end-to-end classification process, which is more objective and not limited to a specific species [9].

Some researchers have deployed deep learning-based methods to 2D images of microfossils. Pedraza *et al*. [10] applied AlexNet [11] architecture pretrained on ImageNet [12] for diatom identification. In their experiments, the overall accuracy reached 99%. Johansen and Sørensen [13] used a deeper Visual Geometry Group (VGG) 16 network [14] with transfer learning for classification. They achieved 98.5% accuracy on a foraminifera dataset. Rehn *et al*. [15] implemented a U-Net [16] variant to remove image artifacts for charcoal particle segmentation and a VGG16 network for individual particle classification. They attained 96% accuracy for segmentation and 75% accuracy for classification. All these researchers explored convolutional neural network (CNN) technology to classify 2D microfossil images. However, 2D images can only provide limited information on microfossils, possibly leading to incorrect identification of microstructures [17]. Compared to 2D images, 3D models provide more complete fossil data, which are essential for visual understanding [18]. With the widespread use of industrial computed tomography (CT) in paleontology, we can obtain comprehensive microfossil information from the surface to the interior without destroying the microfossil. Carvalho *et al*. [19] used the U-NET [16] method to segment the 2D CT slices of planktonic microorganisms. Their method performs the segmentation and recognition of slice data about the background, porous space, fossil, and rock, not the identification of the species. However, the identification of 3D fossil models at the species level is more useful for paleontological research [4], [20]. The increased interest in the collection has raised motivations for developing an intelligent application of 3D microfossil models such as visualization, classification, and retrieval.

In summary, this study aims to explore an automated system for 3D microfossil model classification, which is the level of the species. There are two main contributions of this article. 1) We built the 'Archives of Digital Morphology (ADMorph)' dataset, an open dataset for 3D microfossil models. It is a 3D model dataset based on paleontological fossils, which is potentially useful for biomorphology research and even for computer vision research. 2) We proposed an automated method for 3D microfossil model classification based on deep neural networks (DNNs) and support vector machine (SVM). It is an attempt to apply the deep learning method to classify 3D microfossils.

II. RELATED WORK

In this section, we review the available fossil datasets, nonfossil 3D model datasets, and deep learning methods for 3D object classification.

A. FOSSIL DATASETS AND NONFOSSIL 3D MODEL DATASETS

Fossil dataset. With deep learning techniques applied to the automated classification of fossils, some fossil datasets have been created, such as the foraminifera dataset [13], charcoal particle dataset [15], and diatom dataset [10], which are composed of 2D images. The 2D images are collected using a digital camera mounted on a microscope or scanning electron microscope. Few other datasets, such as echinoderm datasets [21] and planktonic datasets [19], are used in 3D imaging methods such as CT scanning. However, in these datasets, 3D fossil models except 2D slices are not provided. To date, there is no public 3D model dataset for developing a fossil classification system based on deep learning.

Nonfossil 3D model dataset. In recent years, deep learning algorithms have transformed 2D images into 3D models. The 3D model datasets have been widely used in the fields encompassing computer vision, graphics, and image processing. Object semantic segmentation and classification algorithms have been applied to public datasets such as ModelNet [22] and ShapeNet [23] to evaluate their effectiveness. Although the above datasets have a large number of 3D computer-aided design (CAD) models, they mainly include large objects such as bathtubs, beds, and chairs. Unlike CAD models, microfossil models have complex biomorphic structures. To develop an automated classification system of the 3D microfossil models, we probably need to create a dataset that includes 3D digital morphological data such as fish bones, teeth, and scales.

B. DEEP LEARNING METHODS FOR 3D OBJECT CLASSIFICATION

Different from 2D images, which are regular grids of pixels, 3D shapes are irregular and unstructured. It is not feasible to use traditional convolutional neural networks in 3D shapes. To solve this problem, we reviewed available deep learning methods for 3D object retrieval and classification. The methods comprise two main categories: one is model-based, and the other is view-based.

Model-based methods utilize raw representations to describe 3D shapes such as voxels [24], [25], point clouds [26], and polygon meshes [27]. VoxNet [24] employs a 3D convolutional neural network and deals with 3D volumetric representations directly. The main challenges focus on the remarkable growth of memory overhead and computational complexity with increasing voxel resolution, which is limited to the resolution of $32 \times 32 \times 32$ voxels. Charles *et al*. [26] proposed a novel deep network that operates a raw point cloud without voxelized 3D shapes. PointNet [26] aggregates features using max-pooling, which leads to a great loss of local structure information. The main disadvantages of these model-based methods are the enhanced computational complexity and additional noise in shape representation [28].

View-based methods encode a 3D object as a set of 2D rendered images. A sequence of images is fed into the

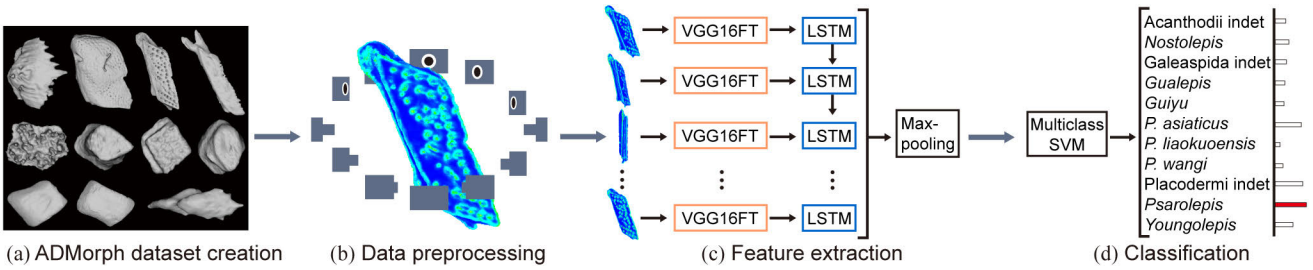

FIGURE 1. The pipeline of 3D microfossil model classification. (a) ADMorph dataset creation including fossil scanning and 3D modeling, (b) Data preprocessing including fossil alignment and fossil coloring, (c) Feature extraction using VGG16FT and LSTM-RNNs, (d) Microfossil classification using SVM.

convolutional operators for feature extraction and classification [29]. View-based methods obtain better performance than many other model-based descriptors [30]. Multiview representation converts irregular 3D shapes to regular 2D images by placing a set of virtual cameras around a 3D object and capturing images from arbitrary viewpoints. The MVCNN [31]. generates the features for different views separately with a weight-shared CNN. The features are merged into a single global shape descriptor by a view-pooling layer. However, the view-pooling layer related to the max-pooling layer and max-out layer retains the maximal activation from one specific view, ignoring the nonmaximal elements. To deal with the limitations of the max-pooling layer, MHBN [32] integrates multiple view features with bilinear pooling to generate a global compact representation. The favorable performance of view-based methods pushes researchers to begin more work on graphics processing unit (GPU)-based methods to learn 3D shapes. GPU acceleration and inverted files (Twice) (GIFT) [33] use GPU-accelerated computing and an inverted file to implement a real-time 3D model search engine. Among the above methods, view-based methods perform better in 3D shape classification and are more suitable for real applications (e.g., microfossils) due to their high flexibility [34].

Although remarkable progress has been achieved in state-of-the-art methods over the past few years, two weaknesses affect view-based methods for 3D microfossil model classification. The first weakness is that the most available view-based approaches [31]–[33] aggregate view features by treating them as an independent set of features, ignoring correlation information among these views. For the 3D microfossil model classification, we moved a virtual camera around the fossil. The microstructures of the fossil may contain temporal sequence information between adjacent views. We needed to efficiently store the sequence information for 3D microfossil model classification. The second weakness is that the MVCNN [31] uses the Phong shading model [35] to render multiview grayscale images depicting each input 3D object model. The microfossils have many local reliefs on their surfaces, which are essential for paleontologists to identify them. We highlighted the local relief features in a corresponding color. To overcome the above two weaknesses, we applied long short-term memory recurrent neural networks (LSTM-RNNs) [36] to fuse the view sequences and generated a curvature map to highlight local reliefs of the 3D microfossil model. We proposed a deep learning method for the task of 3D microfossil model classification.

III. METHODOLOGY

We illustrated the framework of the 3D microfossil model classification in Fig. 1. The pipeline proposed here had four main steps: 1) ADMorph dataset creation, 2) data preprocessing (DP), 3) feature extraction, and 4) 3D microfossil model classification. To accomplish the pipeline, we used microfossils, which were excavated from the muddy limestone of the Kuanti Formation (Ludlow (Silurian)), the muddy limestone of the Xitun Formation (Early Devonian), and the siltstone of the Xishancun Formation (Early Devonian) in Qujing, Yunnan, China.

A. ADMORPH DATASET CREATION

The process of ADMorph dataset creation (Fig. 2) can be summarized as: 1) fossil selection and preparation, 2) fossil scanning, and 3) 3D model calculation.

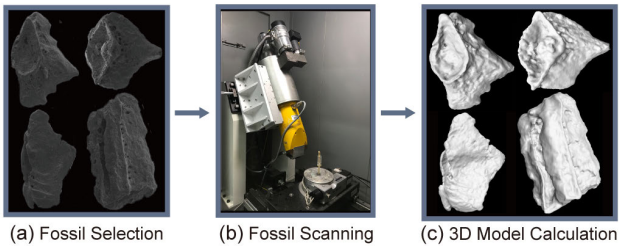

FIGURE 2. The process of ADMorph dataset creation. (a) Fossil selection including field excavation and indoor repair, (b) Fossil scanning using a 225 kV micro-CT system, and (c) 3D model calculation using the threshold segmentation method.

The selected fossils from the three strata had different preservations. The fossils from the Kuanti Formation were articulated and could be directly scanned after simple mechanical preparation using pneumatic air scribes and needles under microscopes. The disarticulated fossils from the Xitun Formation were processed using an acetic acid solution (3%-7%) to dissolve the surrounding matrix and

selected under microscopes. The specimens from the Xishancun Formation were fossil molds that were cavities left in the matrix because of the dissolution of the original organism in taphonomic processes. The rock samples from the Xishancun Formation were cut into cuboids of 1cm length, 1cm width, and approximately 5cm height. All fossils were housed and accessible for examination in the collections of the Institute of Vertebrate Paleontology and Paleoanthropology, Chinese Academy of Sciences (CAS). Then, we scanned the prepared fossils using 225kV micro- CT [37]. Each sample was scanned with a beam energy of 100kV and a flux of 100μA at a resolution of 5.96μm per pixel using a 360° rotation with a step size of 0.25°. A total of 1,440 projections were reconstructed in each 2048 × 2048 image matrix of 1,536 slices using 2D reconstruction software developed by the Institute of High Energy Physics, CAS.

After obtaining the CT slice data, we segmented the fossils out (as masks) by setting proper grayscale thresholds to microfossil CT slices and calculated the 3D models from the mask in the Materialise Interactive Medical Image Control System 18.0 (MIMICS) software. Then, we exported the models in the format of the labeled standard tessellation language (STL) to our dataset.

B. DATA PREPROCESSING

Data preprocessing (DP) can be summarized as: 1) 3D microfossil model pose normalization, 2) curvature map calculation, and 3) 3D microfossil model rendered.

It was impossible to place microfossils scanned by CT in a uniform axis. Pose normalization of the 3D models was required before any other processing. We moved the centroid of the 3D microfossil model to the origin of the canonical or global reference system and then aligned it with one of the axes in the global reference system. We used the principal components analysis (PCA) method [38] to compute a new reference system for the 3D microfossil models (obtaining the directions of the axes without dimensionality reduction). The PCA computed a covariance matrix of all vertex coordinates with three eigenvectors, which were used to rotate the 3D microfossil model to a new coordinate system. The eigenvector with the highest eigenvalue by any projection of the object was the first coordinate (first principal component, Z-axis), the eigenvector with the medial eigenvalue was the second coordinate (second principal component, Y-axis), and the last was the X-axis with the smallest eigenvalue. We translated a set of 3D microfossil models into the centroid and oriented them along with the PCA reference system, as shown in Fig. 3.

Curvatures remain invariant under rigid transform (translation, rotation, and scale) [39]. The curvature values are independent of the 3D object model orientation. Local curvature is a way of measuring or describing the surface concavity−convexity at a point that lies on a curve line. We took advantage of this property as a point of reference for distinguishing different surface types [40]. The concavity-convexity regarded as surface reliefs of the microfossil

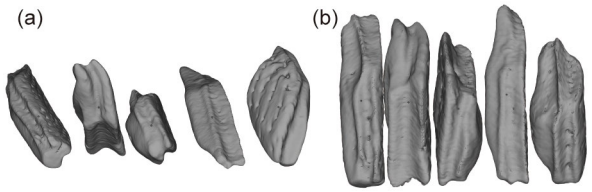

FIGURE 3. 3D microfossil model pose normalization using the PCA method. (a) The pose of the initial models during CT scanning. (b) The pose of aligned models after normalization.

was used as the local feature. The local microstructures (e.g., pores, ridges, and serration) with high curvatures for the paleontologists were generally the diagnostic features for identifying the microfossils. We applied a mean curvature map to highlight the reliefs of the 3D microfossil model. Mean curvature is a single shape indicator that characterizes local variations of a surface. It is widely used in different applications, such as 3D face recognition [41], 3D skeleton shapes [42], and 3D object classification [43]. The mean curvature function is defined as follows:

$$C_{mean} = \frac{C_1 + C_2}{2} \tag{1}$$

where maximum $C_1 = \lambda_1 \|\boldsymbol{k}_1\|$ and minimum $C_2 = \lambda_2 \|\boldsymbol{k}_2\|$ are the magnitudes of the local principal curvature directions at the vertex, and λ_1 and λ_2 are scalars associated with the directions. The principal curvature directions $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ are orthogonal and coplanar defined by the normal vector $\boldsymbol{n}_i$ with respect to the tangent plane at vertex $\boldsymbol{u}_i$, as depicted in Fig. 4.

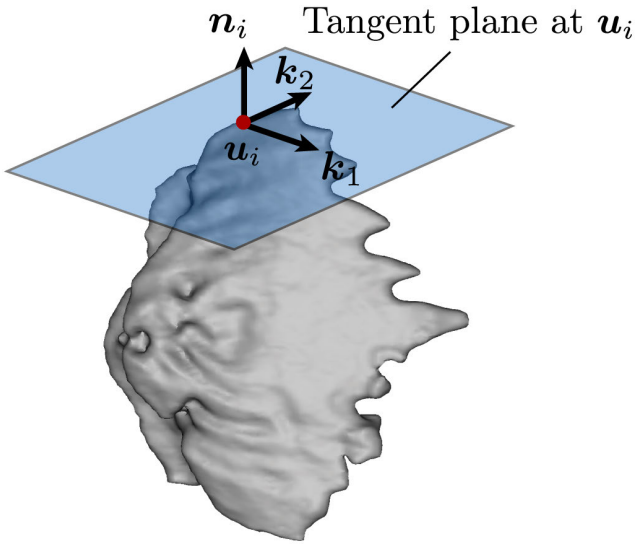

FIGURE 4. Principal curvature directions at vertex $\boldsymbol{u}_i$.

Principal curvature directions are computed by the normal vectors around a local neighborhood of the vertex [44], which are projected to the tangent plane. A normal projection $\hat{\boldsymbol{n}}_i = B\boldsymbol{n}_i$ is obtained by means of a projection matrix B defined as follows:

$$B = I - \boldsymbol{n}_i \boldsymbol{n}_i^T \tag{2}$$

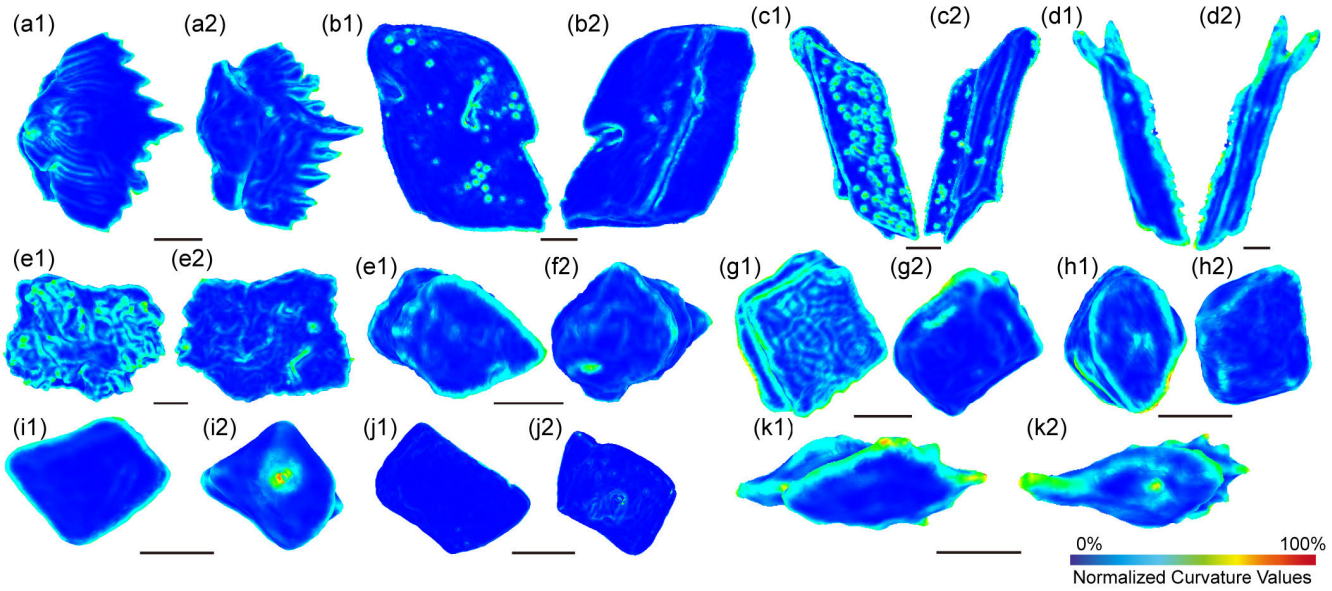

FIGURE 5. Curvature map of 3D microfossil models: (a) *Gualepis*. (b) *Youngolepis*. (c) *Psarolepis*. (d) *Guiyu*. (e) Galeaspida indet. (f) Acanthodii indet. (g) Placodermi indet. (h) *Nostolepis*. (i) *Parathelodus wangi*. (j) *P. liaokuoensis*. (k) *P. asiaticus*. Scale bar, a-c, e-k = 0.2 mm, d = 1 mm.

where I is an identity matrix. Given a spherical neighborhood with N vertices around the vertex $\boldsymbol{u}_i$, we can obtain the normal projections to compute a covariance matrix A defined as follows:

$$A = \frac{1}{N-1}\sum_{k=1}^{N}\left(\hat{\boldsymbol{n}}_k - \bar{\boldsymbol{n}}\right)\left(\hat{\boldsymbol{n}}_k - \bar{\boldsymbol{n}}\right)^T \tag{3}$$

where $\hat{\boldsymbol{n}}_k$ is a projection onto the tangent plane around the neighbor of $\boldsymbol{n}_i$. The vector $\bar{\boldsymbol{n}}$ is the mean of all nearest normal neighbors of vertex $\boldsymbol{u}_i$. According to [45], the two greatest eigenvectors of covariance matrix A are the principal curvature directions $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$. Using the singular value decomposition (SVD) eigendecomposition of matrix A, we obtained the eigenvalues (λ_1, λ_2) and eigenvectors $(\boldsymbol{k}_1, \boldsymbol{k}_2)$.

The eigenvalues and principal curvature directions obtained from Eq. 3 were used to compute the mean curvature values of all vertices in a 3D model, which were normalized to the range (0, 1). Normalized curvature values were used to create curvature maps of the 3D models. The curvature map is a representation that resembles a heat map, in which curvature values are represented by a jet color palette, as shown in Fig. 5. The minimum and maximum values are represented by blue and red, respectively. All intermediate values are linearly interpolated to match a particular color in the palette. Therefore, different curvature magnitudes are represented by different colors.

Finally, we converted the unstructured 3D shapes to regular 2D images. The colored 3D microfossil model was represented by a set of rendered images. In our experiment, the 3D microfossil model was oriented upright along a consistent axis (e.g., y-axis). The virtual cameras were placed every 30 degrees counterclockwise around the y-axis from the plane parallel to the horizontal plane, pointing toward the centroid of the model. A visualization tool created using OpenGL captured 12 views $V = \{v_1, v_2, \cdots, v_{12}\}$ to represent a 3D microfossil model, and each v_i was a $224 \times 224 \times 3$ image matrix, as shown in Fig. 1.

C. FEATURE EXTRACTION

The feature extraction process for rendered images can be summarized by the following steps: 1) VGG16 with fine-tuning (VGG16FT), which adjusted the original output feature vector of VGG16 into a more meaningful vector in accordance with our microfossil dataset, and 2) LSTM-RNNs, which fused the output feature vectors from step 1 into a single and dimensionally reduced feature vector.

In the VGG16FT step, the top three layers of VGG16 were redefined by 2 fully connected (FC) layers and a softmax layer. The FC layers (FC1 and FC2) focused on reducing the dimensions of the feature vectors, going from $1 \times 1 \times 4096$ to $1 \times 1 \times 1048$ with a tanh activation function. The softmax layer learned how to classify the feature vectors extracted from the microfossil dataset. These three layers needed to be trained (using image views V on the training set) to obtain the VGG16FT network.

ADMorph is a new and small dataset for which no pre-trained CNN can be found in the literature. Creating a CNN with such a small dataset may incur overfitting. CNN would only classify the objects on the training set correctly. To work around this problem, we used the transfer learning method, in which the VGG16 (a well-established network trained with thousands of objects) was fine-tuned to modify and adjust their parameters in accordance with our dataset. The fine-tuned VGG16 is called VGG16FT. The last fully connected layer (FC2) in VGG16FT produced the feature

vectors $\boldsymbol{F} = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots \boldsymbol{f}_{12}\}$ from the 12 views V, where $\boldsymbol{f}_i \in \mathbb{R}^{1048}$.

Image views V of a 3D object had redundancies that may result from overlapped views and similar features in different views. We modeled these redundancies as a directed acyclic graph and made inferences from it. The LSTM-RNNs were added into our methodology to model graph dependencies of sequential data efficiently [36]. The LSTM-RNNs can learn features from sequential data [46]–[48]. Similar to a Markov chain, a transition state of LSTM-RNNs was used to process a current feature vector with respect to the previous feature vector. A transition state, denoted as C_t, which contains inner variables is defined as follows:

$$\begin{aligned}
\boldsymbol{h}_t &= \sigma\left(W_{h,t}\boldsymbol{f}_t + \Theta_{f,t}z_{t-1} + b_n\right)\\
\boldsymbol{i}_t &= \sigma\left(W_{i,t}\boldsymbol{f}_t + \Theta_{i,t}z_{t-1} + b_i\right)\\
\boldsymbol{o}_t &= \sigma\left(W_{o,t}\boldsymbol{f}_t + \Theta_{o,t}z_{t-1} + b_o\right)\\
\boldsymbol{r}_t &= \boldsymbol{h}_t \odot \boldsymbol{r}_{t-1} + \boldsymbol{i}_t \odot \sigma\left(W_{r,t}\boldsymbol{f}_t + \Theta_{r,t}z_{t-1} + b_r\right)\\
z_t &= \boldsymbol{o}_t \odot \sigma\left(\boldsymbol{r}_t\right)
\end{aligned} \tag{4}$$

where $T_t = [\boldsymbol{r}_t, z_t]$ are transition variables between states, $\sigma(\cdot)$ is a sigmoid function, $\odot$ denotes elementwise multiplication between vectors (Hadamard product), and $z_t \in \mathbb{R}^{1048}$ is the output of each state or cell C_t. The variables W_*, Θ_*, and b_* are the weights of each cell based on recurrent connections.

The outputs of the LSTM-RNNs were $Z = \{z_1, z_2, \cdots, z_{12}\}$ and $z_i \in \mathbb{R}^{1048}$. The max-pooling layer was used to convert Z into a single vector $\boldsymbol{x}_i \in \mathbb{R}^{1048}$. We obtained the components x_j of vector $\boldsymbol{x_i}$ as follows:

$$x_j = \max\left(z_{0,j}, z_{1,j}, \ldots, z_{11,j}\right) \tag{5}$$

The processes (VGG16FT and LSTM-RNNs) built a global feature vector representing the 3D model as multiview images, as shown in Fig. 6. The 3D model dataset was converted into a new dataset $X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\}$, where $\boldsymbol{x}_i$ is a global vector of the ith model, $y_i \in \{1, \cdots, K\}$ is its class label assigned to a 3D microfossil model, and K is the number of categories. Finally, the X dataset was fed into a classifier during the training and testing phases.

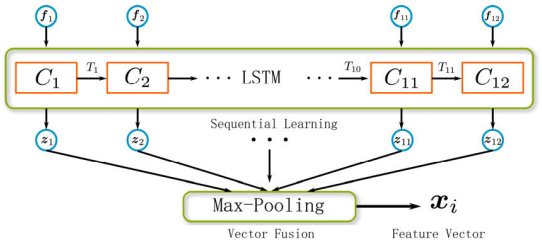

FIGURE 6. Exploiting LSTM-RNNs for aggregating sequential views.

D. CLASSIFICATION PROCESS

The SVM is more effective for image classification than other classifiers [49], and it is well suited for the classification of complex but small or medium-sized datasets [50]. We chose an SVM as the classifier in our new dataset X. The SVM computed hyperplanes to separate one class from other classes. Each hyperplane parametrization was denoted by $\boldsymbol{w}_j$. In total, we needed $\boldsymbol{W} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k\}$ parameters (which were unknown) to classify dataset X. The unknown parameters $\mathbf{W}$ for SVM multiclass [51] were computed by a quadratic optimization problem defined by:

$$\begin{aligned}
\underset{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k}{argmin} \quad & \frac{1}{2}\sum_{j=1}^{k}\boldsymbol{w}_j^T\boldsymbol{w}_j + C\sum_{i=1}^{n}\sum_{j\neq y_i}^{k}\xi_i^j\boldsymbol{w}_{y_i}^T\Phi\left(\boldsymbol{x}_i\right)\\
s.t \quad & + b_{y_i} \geq \boldsymbol{w}_j^T\Phi\left(\boldsymbol{x}_i\right) + b_j + 2 - \xi_i^j\\
& \xi_i^j \geq 0, j \in \{1, \ldots, K\} \setminus y_i\\
& \forall i = 1, \ldots, n
\end{aligned} \tag{6}$$

where ξ_i^j is a slack variable that controls the prediction error margins, $\boldsymbol{w}_{y_i}$ is a parametrization of the hyperplane class y_i, b_{y_i} represents an offset with respect to the data space, $\Phi\left(\boldsymbol{x}_i\right)$ is a feature transformation function, and C is a regularization parameter associated with the margins.

The one-vs-rest multiclass approach computed the distances from the global vector $\boldsymbol{x}_i$ to the hyperplanes, which separated one class from other classes. The prediction function was defined as follows:

$$\begin{aligned}
\hat{y}_i &= argmax\ \boldsymbol{w}_j^T\boldsymbol{x}_i\\
\boldsymbol{w}_j &\in \{\boldsymbol{w}_1, \boldsymbol{w}_{2,\cdots,}\boldsymbol{w}_k\}
\end{aligned} \tag{7}$$

The class corresponded to the maximum margin hyperplane, which was used to assign the label $\hat{y}_i$ to the vector $\boldsymbol{x}_i$. The operation $\boldsymbol{w}_j^T\boldsymbol{x}_i$ represented the distance from a global vector $\boldsymbol{x}_i$ to each hyperplane as parameter $\boldsymbol{w}_j$, and K was the number of classes.

E. CLASSIFICATION ALGORITHM AND COMPLEXITY CALCULATION

We calculated the parameter complexity of each step in our method. It was split as follows:

The complexity of the PCA and curvature map depended on the total number of vertices and the total number of 3D models. PCA computed a covariance matrix $R^{3\times3}$ to rotate each model. In total, we obtained n covariance matrices, where n was the total number of 3D models. The mean curvature value for a vertex $\boldsymbol{u}_i$ was computed by means of the maximum and minimum curvature directions of the covariance matrix $A \in R^{3\times3}$ (Eq. 3). To create a curvature map for a 3D model, we needed a total number of $2\times m$ curvature values for each model, where m was the total number of vertices in a 3D model.

The feature extraction based on the VGG16FT architecture used a total of 27,394,416 parameters. The LSTM-RNNs were employed to fuse the feature vectors using 8,786,432 parameters for the outputs of VGG16FT.

The classification process was the final step in our method, which depended on the dimension d of the global feature

vector and the classes K of all the 3D models. The total number of 11,528 parameters for the SVM classifier was obtained by the operation $d \times K$, which depended on the dimension of the global feature vector $d = 1048$ and the number of classes $K = 11$ to classify all the 3D models.

We added the classification algorithm description to summarize our method as shown in Alg. 1

Algorithm 1 Feature Extraction and Classification Algorithm From Views

Input: Dataset of views $V = \left\{\left\{v_{i,j}\right\}_{j=1}^{12}\right\}_{i=1}^{n}$.
repeat
  $\boldsymbol{F} \leftarrow \emptyset$
    repeat
    $\boldsymbol{f}_i = VGG16\left(v_{i,j}\right)$ // feature extraction
    $\boldsymbol{F} \leftarrow \boldsymbol{F} \cup \left\{\boldsymbol{f}_i\right\}$
    until $v_{i,j} \in V_i$;
  $\boldsymbol{x}_i = LSTM\ (\boldsymbol{F})$ // vector fusion
  $\hat{y}_i = SVM\ (\boldsymbol{x}_i)$ // classification
until $V_i = \left\{v_{i,j}\right\}_{j=1}^{12} \in V$;
return $\hat{y}_i$

IV. EXPERIMENT SETTINGS

A. EXPERIMENTAL DATASET

We selected 2,010 scales of 11 groups from the ADMorph dataset as the experimental dataset and labeled each fish scale according to morphology and histology. The dataset and its details used in this study has be published online in a publicly accessible repository (ADMorph) at http://www.admorph.org/. We randomly split our annotated experimental dataset into two subsets: a training set (70%) and a test set (30%). We randomly extracted 20% (per class) of instances from the training set to form a cross-validation set. According to the abovementioned ratio, we randomly split the experimental dataset a total of 30 times to obtain global statistics such as the mean, variance and confidence intervals, which proved the effectiveness of our proposed method.

B. NETWORK TRAINING

We performed experiments on a machine with an NVIDIA GeForce RTX 2080Ti GPU, Intel Xeon Silver 4114 CPU, and 64 GB RAM. Our proposed framework was coded in the Keras [52] platform, a popular deep learning library. Our training network was divided into three phases: 1) the VGG16FT extracted the feature vector, 2) the LSTM-RNNs fused the feature vectors, and 3) the SVM classified 3D microfossil models.

1) VGG16FT FEATURE VECTOR EXTRACTION

The first step was the training and optimization of the VGG16FT network (last three layers). The rendered images on the training set and cross-validation set were reshaped to a $224 \times 224 \times 3$ image matrix to train the VGG16FT network. We loaded a VGG16 model pretrained on ImageNet without the top layer and froze the weights of all layers. As mentioned in the methodology, the FC layers focused on reducing the dimension of the feature vectors with a tanh activation function from $1 \times 1 \times 4096$ to $1 \times 1 \times 1048$. The output of the softmax layer was fixed $K = 11$ according to the number of classes. On the training set, each view v_i was classified as one of the classes through the softmax layer [53]. The VGG16FT model was trained using the Adam optimization method [54]. The initial learning rate was set to 0.0001, and the number of epochs was set to 100 with a batch size of 4. The loss function L was cross-entropy with the L_2 regularization parameter. The training and cross-validation processes of the VGG16FT adjusted the parameters in the FC1 and FC2 layers during the fine-tuning phase.

2) FUSION OF THE FEATURE VECTORS

We trained LSTM-RNNs to optimize the network, which fused the feature vectors. We added the max-pooling and FC layer after the LSTM-RNNs to set the trainable parameters. For the LSTM-RNN cell configuration, we set the hidden state size $H = 1048$ with the forward units. The initial learning rate was set to $\gamma = 0.001$, and momentum was set to 0.95. The number of epochs was set to 100 with a batch size of 36 in the training process. We aimed to find a learning model that contained the optimized parameters of the LSTM-RNNs. The LSTM-RNN parameters W_*, Θ_* and b_* (Eq. 4) were estimated by means of stochastic gradient descent (SGD) [55]. We used the training set (feature vectors) and cross-validation set to adjust the shared weights in the LSTM-RNNs. We used cross-entropy as the loss function L, which is defined as:

$$L = -\frac{1}{n}\sum_{i=1}^{n}\left\{y_i \log\left(\hat{y}_i\right) + (1 - y_i)\log\left(1 - \hat{y}_i\right)\right\}$$
$$\hat{y}_i = \arg\max_{i \in \{1,\ldots,k\}} \frac{\exp\{\boldsymbol{s}_i\}}{\sum_{j=1}^{k}\exp\{\boldsymbol{s}_i\}} \tag{8}$$

where n is the total number of 3D microfossil models on the training set, $\boldsymbol{s}_i$ is obtained with a sigmoid function from the feature vector output $\boldsymbol{x}_i$ of the LSTM-RNNs, and $\hat{y}_i$ is the predicted category of the LSTM-RNNs.

3) SVM CLASSIFIED 3D MICROFOSSIL MODEL

To obtain better classification results, we trained the parameters of the one-vs-rest multiclass linear SVM classifier. The maximum number of iterations was 3,500, and the regularization parameter C was 0.1, which was included in Eq. 6. The SVM was trained to find the optimized normal vectors of hyperplanes that separated one class from the other classes, resulting in a predicted label. We aimed to find an SVM learning model to evaluate the classification accuracy on the test set.

C. NETWORK TESTING

The whole pipeline was evaluated on the test set. We extracted the 1048-dimensional feature vector by VGG16FT from rendered images on the test set. The view sequences extracted in the previous step were fused using the LSTM-RNNs, and a one-vs-rest multiclass linear SVM classifier was used to obtain the predicted label. Then, we evaluated the proposed method by comparing the predicted label with the actual label on the test set.

D. EVALUATION CRITERIA

To assess the effectiveness of the proposed method, we used the confusion matrix and macroaveraging accuracy as the evaluation criteria.

The confusion matrix (also called the contingency table) is an important method for evaluating the performance of a classifier. It is computed as a $K \times K$ matrix that displays the numbers of correct and incorrect predictions made by the classifier compared with the actual labels on the test set. K represents the number of classes.

Alternatively, the macroaveraging accuracy P_{macro} of a classification model on a test set may be defined as follows:

$$P_{macro} = \frac{1}{K} \sum_{i=1}^{K} P_i \tag{9}$$

where P_i is the accuracy of each class and P_{macro} is the average of the total category accuracy. A correct classification means that the learned model predicts the same class as the original class of the test case.

E. COMPARISON WITH OTHER METHODS

To validate the effectiveness and robustness of our DNNs and SVM method, we compared it with three classification methods, VoxNet [24], PointNet [26], and MVCNN [31], on the experimental dataset.

We used the same data structure mentioned in the literature on the experimental dataset. First, VoxNet [24] used a shallow volumetric CNN with the probabilistic occupancy grid. The binvox command-line toolset [56] converted each mesh data to binvox data in a volume of $32 \times 32 \times 32$ for CNN computation. The volumetric grid was a 3D data structure. The memory overhead and computational complexity with voxel resolution increased exponentially. Due to the computational and memory limitations, we selected $32 \times 32 \times 32$ voxels, which is the same data structure as the VoxNet [24] method. Second, PointNet [26] processed the raw point clouds directly and extracted point features with multi-layer perception (MLP) networks. We uniformly downsampled each point cloud to 2048 using the Monte Carlo method. Each point cloud was zero mean and normalized into a unit sphere. We augmented the point cloud by randomly rotating the 3D model upward and jittering the position of each point by Gaussian noise with zero mean and 0.01 standard deviation. Finally, the MVCNN [31] exploited the grayscale image $224 \times 224 \times 3$ matrix depicting each input 3D model. The virtual cameras were placed at every 30° around the 3D model. They were placed parallel to the horizontal plane, pointing toward the centroid of the mesh data.

F. CONTRIBUTIONS OF DIFFERENT COMPONENTS

To compare the contributions of different components, we combined various components of our network to perform 3D model classification on the experimental dataset. First, the MVCNN [31] denotes the typical view-based method, which was the baseline model for the 3D microfossil model classification. The framework of the MVCNN method is illustrated in Fig. 10 of the appendix section. Second, we preprocessed 3D microfossil models before feature extraction. The data preprocessing (DP) included the calculation of PCA and curvature map (CM). We fed the preprocessed 3D models into the MVCNN method. Then, DNNs including VGG16FT and LSTM-RNNs were a modified version of the MVCNN. The VGG16FT retained the initial layers of VGG16 intact and retrained the later layers to fit our dataset. We used LSTM-RNNs to replace the view-pooling of the typical MVCNN for the sequence-based view aggregation. We fed the 3D source models into DNN models. Finally, we proposed our method, which combined data preprocessing with DNN models. We compared the performance of the SVM classifier and softmax classifier on the experimental dataset.

V. RESULTS

A. DATASET CREATION

We constructed an ADMorph dataset, a large-scale 3D microfossil model dataset. It contains over ten thousand 3D virtual models of fish bones, teeth, and scales from 5 main classes of early fishes (agnathans, placoderms, osteichthyans, chondrichthyans, and acanthodians). In this study, we selected 2,010 labeled scale virtual models (Fig. 7) among the datasets to verify the effectiveness of the proposed method. We labeled the early fish scales into different categories according to their morphology and histology, some of which are shown in Fig. 8. The virtual models contained agnathans (Galeaspida indet, 105; *Parathelodus asiaticus, 57; P. liaokuoensis, 100; P. wangi, 100*), placoderms (Placodermi indet, 106), osteichthyans (*Guiyu*, 145; *Youngolepis*, 163; *Psarolepis*, 170), chondrichthyans (*Gualepis*, 888), and acanthodians (Acanthodii indet, 34; *Nostolepis*, 142).

B. MICROFOSSIL CLASSIFICATION

The classification results for the six hundred three 3D models (30% on the experimental dataset for testing) of early fish scales from 11 groups were analyzed both quantitatively and visually. We achieved a macroaveraging accuracy of 97.60% on the test set. The confusion matrix of the classifier evaluated on the test set is shown in Fig. 9. The diagonal elements represent the classification accuracy for each category, while

TABLE 1. Classification accuracy (%) for our method and other methods on the test set.

Method	Representation	Core operator	Accuracy (%)
PointNet	point	Pointwise MLP	64.13±2.04%
VoxNet	volume	3D convolution	78.15±4.31%
MVCNN	image	2D convolution	91.54±1.58%
Our method	**image**	**DNNs+SVM**	**97.60±1.92%**

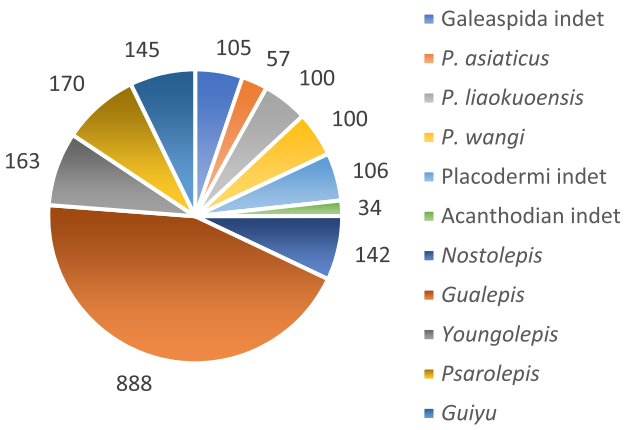

FIGURE 7. Class distribution of the experimental dataset.

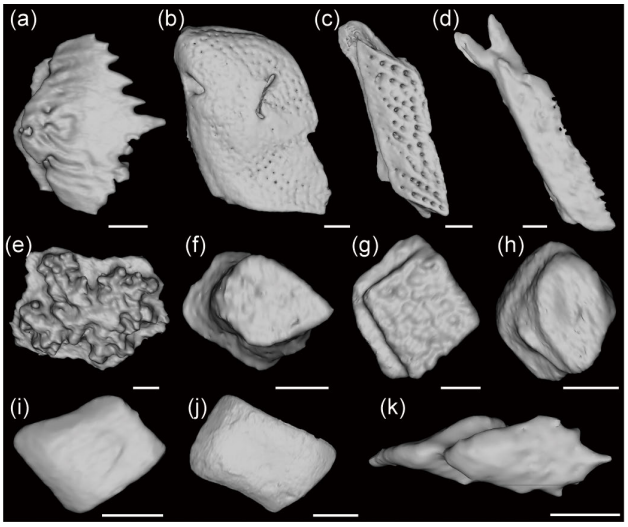

FIGURE 8. A random sample of shapes (one per class) was chosen from the experimental dataset. (a) *Gualepis*. (b) *Youngolepis*. (c) *Psarolepis*. (d) *Guiyu*. (e) Galeaspida indet. (f) Acanthodii indet. (g) Placodermi indet. (h) *Nostolepis*. (i) *Parathelodus wangi*. (j) *P. liaokuoensis*. (k) *P. asiaticus*. Scale bar, a-c, e-k = 0.2 mm, d = 1 mm.

off-diagonal elements are the percentage of those mislabeled by the classifier.

C. COMPARISONS WITH OTHER METHODS

The classification accuracies were compared between our method and other prevalent methods using Student's t-test based on confidence intervals with a significance level of $\alpha = 0.5\%$. It included the baseline view-based method (MVCNN) and the model-based method (PointNet and VoxNet), as illustrated in Table 1. The view-based method performed better than the model-based method on the experimental dataset. Compared with VoxNet, PointNet, and MVCNN, the accuracy of our method was better by 19.45%, 33.47%, and 6.06%, respectively. This accuracy reflects that the proposed method played a crucial role in improving classification accuracy. The proposed method effectively utilized DNNs and SVM to recognize 3D microfossil models imitating a paleontologist classification process. The ability of the proposed method was related to local and global shape representation and fused intrinsic correlation with each viewpoint of the 3D microfossil model.

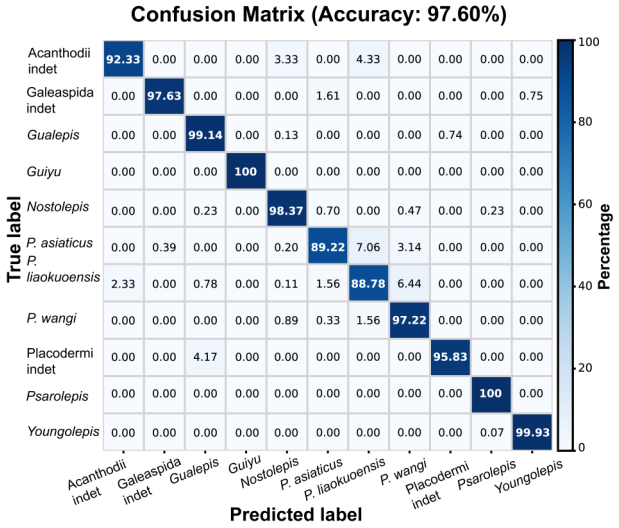

FIGURE 9. Confusion matrix of the classification results on the test set.

D. CLASSIFICATION ACCURACY OF DIFFERENT COMPONENTS

In this work, we compared the classification accuracy for different components on the experimental dataset. Our approach significantly outperformed the baseline view-based method MVCNN. The results demonstrated two advantages of our method. First, we preprocessed each 3D model on the experimental dataset, including the PCA and CM calculations. The macroaveraging accuracy was improved by 0.82% over that of MVCNN. Second, we applied the VGG16FT and LSTM-RNN models. The macroaveraging accuracy increased by 5.56%. Finally, we proposed our method based on DNNs and SVM. We combined data preprocessing with the improved feature extraction method. Compared with

TABLE 2. Classification results for the different components on the test set.

Network	Core operator	Accuracy (%)
MVCNN (SVM)	CNN+SVM	91.54±1.58%
MVCNN (softmax)	CNN+softmax	90.87±2.16%
MVCNN+ DP	PCA+ CM+SVM	92.36±1.83%
DNNs	FT+ LSTM-RNNs+SVM	97.10±2.55%
DNNs+DP (softmax)	PCA+CM+FT+LSTM-RNNs+softmax	97.16±1.16%
Our method	**PCA+ CM+ FT+ LSTM-RNNs+SVM**	**97.60±1.92%**

MVCNN, our approach increased the macroaveraging accuracy by 6.06%. We also compared the performance of the SVM and softmax classifier on the experimental dataset. SVM obtained a slightly higher accuracy than softmax in the MVCNN method (0.67%). Similarly, SVM achieved slightly better results in our method (0.44%). The results based on different components are shown in Table 2.

VI. DISCUSSION

Our dataset is currently focused on 3D models of early fishes, including agnathans, placoderms, osteichthyans, chondrichthyans, and acanthodians. It was constructed based on high-resolution biomorphology. The 3D fossil model classification is potentially useful for biomorphology research and even for computer vision research. Categories and 3D model data are accumulating rapidly. So far, this is a preliminary dataset, which is not well balanced as *Gualepis* occupies a large proportion (888) of our dataset, and Acanthodii indet is least in number (34). We proceeded with an FT strategy to adjust the VGG16 model for our dataset [57]. The classification results showed that the accuracy rate satisfied the preliminary requirements of paleontologists.

We proposed the DNNs and SVM framework, which obtained a macroaveraging accuracy of 97.60%. Our method effectively recognized some categories such as *Youngolepis* and *Gualepis*, even though intraclass data varied in morphology. However, the accuracy rates of different species of the same genus were relatively low (Fig. 9), which might be caused by the morphological similarities and the lack of histological information that is key for the intraspecific classification of fishes [17], [20], [58].

We also compared our method with other view-based and model-based methods. For the model-based method, we chose classic PointNet and VoxNet. PointNet used 2,048 points and could not fully represent the general and local information of the microfossil. Similarly, VoxNet, using $32 \times 32 \times 32$ voxels, could not fully represent the 3D structure of microfossils [30]. For the view-based method, we chose MVCNN as the comparison method. The view-based method achieved better classification accuracy on the experimental dataset than the model-based methods. The mature 2D CNN technology was extended to view-based 3D microfossil model classification [59]. For the microfossils, their biomorphic structures are naturally formed and much more complicated than the CAD models of ModelNet. We proposed a method of microfossil classification and chose the multiview method as our basic framework. We assessed the improvement effectiveness of the different components. By data preprocessing, including the PCA and CM calculations, the experimental result increased 0.82%. Evidence has shown that accuracy improves using a CM when classifying archeological data [43]. Toward the improvement of feature extraction, including the FT and LSTM-RNN models [60], the accuracy of 3D microfossil model classification increased by 5.56%. Both of the above components increased the accuracy. We combined data preprocessing with the improved feature extraction model, making our proposed method suitable for 3D microfossil model classification. We compared the impact of the SVM (97.60%) classifier and softmax (97.16%) classifier on the experimental dataset. We chose deep neural networks (DNNs) and a support vector machine (SVM) framework for 3D microfossil model classification.

Currently, our dataset is not comparable to the CAD model datasets in terms of quantity and species. However, we still have a large number of 3D microfossil models yet to be identified by paleontologists, and a large number of microfossils excavated from the field have not been digitized. We will use our system to help experts identify microfossils and augment our dataset. We will also accelerate the digitization of microfossils. To improve the low accuracy rate of the intraspecific classification of fish scales, we will consider the development of the cross-domain classification method. The method will combine the slice information with the fossil surface information. We will identify and employ unique and distinctive information and effectively combine them with deep learning methods to improve their classification accuracy.

VII. CONCLUSION

In this article, we constructed the ADMorph dataset, an open large-scale 3D model dataset of fossil vertebrates, and proposed a classification method based on DNNs and SVM. The pipeline of the automated classification of 3D microfossil models was also released. In this method, a deep learning network automatically learns the shape features of 3D microfossil models and generates a classification. The experimental results demonstrate that the accuracy of the proposed method is higher than that of other comparative classification methods. It is an attempt to apply the deep learning method to classify 3D microfossils. As such, this method will significantly increase the accuracy and efficiency of the classification compared with the traditional approaches in micropaleontology. The automated classification of 3D microfossil models will

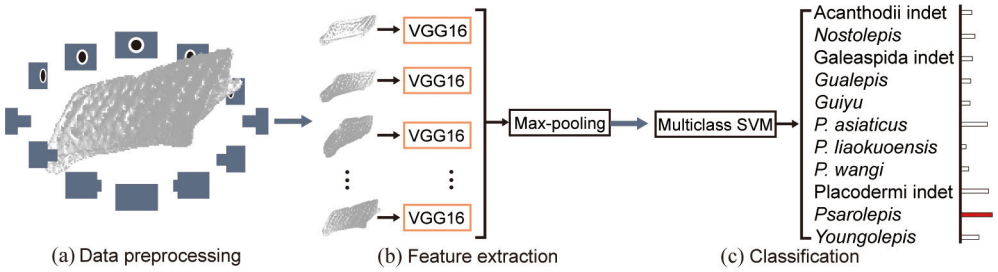

FIGURE 10. The framework of the MVCNN method. (a) 3D source models rendered by 12 virtual cameras, (b) Feature extraction using VGG16, (C) Microfossil classification using SVM.

help paleontologists proceed further into applications such as stratigraphic subdivision and correlation. Along with the accumulation of 3D fossil models in the ADMorph dataset, a large number of fossil specimens excavated from the field can be initially clustered to attempt the unsupervised learning method. Hopefully, more deep learning networks will be applied to paleontology in the future.

ACKNOWLEDGMENT

The authors would like to thank Liping Dong and Zhikun Gai for discussions and Liantao Jia, Pengfei Yin, Qiang Li, and Xianghong Lin for specimen preparation and data capture.

APPENDIX

See Fig. 10.

REFERENCES

[1] B. U. Haq and A. Boersma, *Introduction to Marine Micropaleontology*. Singapore: Elsevier Science, 1998, pp. 1–17.
[2] J. E. Van Hinte, "Geohistory analysis—Application of micropaleontology in exploration geology," *AAPG Bull.*, vol. 62, no. 2, pp. 201–222, 1978.
[3] P. K. Saraswati and M. Srinivasan, *Micropaleontology: Principles and Applications*. Cham, Switzerland: Springer, 2016, pp. 35–77.
[4] X. Cui, T. Qiao, and M. Zhu, "Scale morphology and squamation pattern of Guiyu oneiros provide new insights into early osteichthyan body plan," *Sci. Rep.*, vol. 9, no. 1, pp. 4411–4423, Mar. 2019.
[5] R. J. Brocklehurst, E. R. Schachner, and W. I. Sellers, "Vertebral morphometrics and lung structure in non-avian dinosaurs," *Roy. Soc. Open Sci.*, vol. 5, no. 10, Oct. 2018, Art. no. 180983.
[6] W. C. H. Parr, S. Wroe, U. Chamoli, H. S. Richards, M. R. McCurry, P. D. Clausen, and C. McHenry, "Toward integration of geometric morphometrics and computational biomechanics: New methods for 3D virtual reconstruction and quantitative analysis of finite element models," *J. Theor. Biol.*, vol. 301, pp. 1–14, May 2012.
[7] P. A. Swaby, "VIDES: An expert system for visually identifying microfossils," *IEEE Expert*, vol. 7, no. 2, pp. 36–42, Apr. 1992.
[8] C. D. Burke, W. E. Full, and R. E. Gernant, "Recognition of fossil fresh water ostracodes: Fourier shape analysis," *Lethaia*, vol. 20, no. 4, pp. 307–314, Oct. 2007.
[9] Q. Guo, S. Jin, M. Li, Q. Yang, K. Xu, Y. Ju, J. Zhang, J. Xuan, J. Liu, Y. Su, Q. Xu, and Y. Liu, "Application of deep learning in ecological resource research: Theories, methods, and challenges," *Sci. China Earth Sci.*, vol. 63, pp. 1–18, Mar. 2020.
[10] A. Pedraza, G. Bueno, O. Deniz, G. Cristóbal, S. Blanco, and M. Borrego-Ramos, "Automated diatom classification (Part B): A deep learning approach," *Appl. Sci.*, vol. 7, no. 5, pp. 460–485, May 2017.
[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.
[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[13] T. Haugland Johansen and S. Aagaard Sørensen, "Towards detection and classification of microscopic foraminifera using transfer learning," 2020, *arXiv:2001.04782*. [Online]. Available: http://arxiv.org/abs/2001.04782
[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[15] E. Rehn, A. Rehn, and A. Possemiers, "Fossil charcoal particle identification and classification by two convolutional neural networks," *Quaternary Sci. Rev.*, vol. 226, pp. 106038–106044, Dec. 2019.
[16] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.
[17] Q. Qu, S. Sanchez, M. Zhu, H. Blom, and P. E. Ahlberg, "The origin of novel features by changes in developmental mechanisms: Ontogeny and three-dimensional microanatomy of polyodontode scales of two early osteichthyans," *Biol. Rev.*, vol. 92, no. 2, pp. 1189–1212, May 2017.
[18] J. A. Cunningham, I. A. Rahman, S. Lautenschlager, E. J. Rayfield, and P. C. J. Donoghue, "A virtual world of paleontology," *Trends Ecol. Evol.*, vol. 29, no. 6, pp. 347–357, Jun. 2014.
[19] L. E. Carvalho, G. Fauth, S. B. Fauth, G. Krahl, A. C. Moreira, C. P. Fernandes, and A. Von Wangenheim, "Automated microfossil identification and segmentation using a deep learning approach," 2019, *BioRxiv:661694*, [Online]. Available: https://doi.org/10.1101/661694
[20] X. Cui, Q. Li, T. Qiao, and M. Zhu, "New material of thelodonts from Lochkovian (Lower Devonian) of Qujing, Yunnan, China," *Vertebrata PalAsiatica*, vol. 58, no. 1, pp. 1–15, Jan. 2020.
[21] M. Reid, E. M. Bordy, W. L. Taylor, S. G. L. Roux, and A. D. Plessis, "A micro X-ray computed tomography dataset of fossil echinoderms in an ancient obrution bed: A robust method for taphonomic and palaeoecologic analyses," *GigaScience*, vol. 8, no. 3, pp. 1–8, Dec. 2018.
[22] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.
[23] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*. [Online]. Available: http://arxiv.org/abs/1512.03012
[24] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.
[25] S. Jin, Y. Su, S. Gao, F. Wu, Q. Ma, K. Xu, Q. Ma, T. Hu, J. Liu, S. Pang, H. Guan, J. Zhang, and Q. Guo, "Separating the structural components of maize for field phenotyping using terrestrial LiDAR data and deep convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2644–2658, Apr. 2020.
[26] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.
[27] L. Yi, H. Su, X. Guo, and L. Guibas, "SyncSpecCNN: Synchronized spectral CNN for 3D shape segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6584–6592.

[28] Z. Li, C. Xu, and B. Leng, "Rethinking loss design for large-scale 3D shape retrieval," 2019, *arXiv:1906.00546*. [Online]. Available: http://arxiv.org/abs/1906.00546
[29] S. Jin, Y. Su, S. Gao, F. Wu, T. Hu, J. Liu, W. Li, D. Wang, S. Chen, Y. Jiang, S. Pang, and Q. Guo, "Deep learning: Individual maize segmentation from terrestrial lidar data using faster R-CNN and regional growth algorithms," *Frontiers Plant Sci.*, vol. 9, pp. 866–875, Jun. 2018.
[30] L. Luo, Z.-X. Yang, L. Tang, and K. Zhang, "An ELM-embedded deep learning based intelligent recognition system for computer numeric control machine tools," *IEEE Access*, vol. 8, pp. 24616–24629, 2020.
[31] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 945–953.
[32] T. Yu, J. Meng, and J. Yuan, "Multi-view harmonized bilinear network for 3D object recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 186–194.
[33] S. Bai, X. Bai, Z. Zhou, Z. Zhang, and L. J. Latecki, "GIFT: A real-time and scalable 3D shape search engine," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5023–5032.
[34] A.-A. Liu, W.-Z. Nie, and Y.-T. Su, "3D object retrieval based on multi-view latent variable model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 868–880, Mar. 2019.
[35] B. T. Phong, "Illumination for computer generated pictures," *Commun. ACM*, vol. 18, no. 6, pp. 311–317, Jun. 1975.
[36] S. Hochreiter and J. Schmidhuber, "LSTM can solve hard long time lag problems," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 1997, pp. 473–479.
[37] Y. F. Wang, C. Wei, J. Que, W. Zhang, C. Sun, Y. Shu, Y. Hou, J. Zhang, R. Shi, and L. Wei, "Development and applications of paleontological computed tomography," *Vertebrata PalAsiatica*, vol. 57, no. 1, pp. 84–92, Jan. 2019.
[38] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer-Verlag, 2006, pp. 561–569.
[39] J. M. Lee, *Riemannian Manifolds: An Introduction to Curvature*. New York, NY, USA: Springer, 2006, pp. 115–128.
[40] C. Dorai and A. K. Jain, "COSMOS—A representation scheme for 3D free-form objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 10, pp. 1115–1130, Oct. 1997.
[41] A. Ceron, A. Salazar, and F. Prieto, "Relevance analysis of 3D curvature-based shape descriptors on interest points of the face," in *Proc. 2nd Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Jul. 2010, pp. 452–457.
[42] A. Tagliasacchi, I. Alhashim, M. Olson, and H. Zhang, "Mean curvature skeletons," *Comput. Graph. Forum*, vol. 31, no. 5, pp. 1735–1744, Aug. 2012.
[43] M. Canul-Ku, R. Hasimoto-Beltran, D. Jimenez-Badillo, S. Ruiz-Correa, and E. Roman-Rangel, "Classification of 3D archaeological objects using multi-view curvature structure signatures," *IEEE Access*, vol. 7, pp. 3298–3313, 2019.
[44] M. Pauly, M. Gross, and L. P. Kobbelt, "Efficient simplification of point-sampled surfaces," in *Proc. IEEE Vis.*, May 2014, pp. 163–170.
[45] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D point cloud based object maps for household environments," *Robot. Auto. Syst.*, vol. 56, no. 11, pp. 927–941, Nov. 2008.
[46] Y. Yang, J. Zhou, J. Ai, Y. Bin, A. Hanjalic, H. T. Shen, and Y. Ji, "Video captioning by adversarial LSTM," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5600–5611, Nov. 2018.
[47] Y. Ma, H. Peng, and E. Cambria, "Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 5876–5883.
[48] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, "Predicting human eye fixations via an LSTM-based saliency attentive model," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5142–5154, Oct. 2018.
[49] Y. Tang, "Deep learning using linear support vector machines," 2013, *arXiv:1306.0239*. [Online]. Available: http://arxiv.org/abs/1306.0239
[50] A. Géron, *Hands-On Machine Learning With Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. Sebastopol, CA, USA: O'Reilly, 2017, pp. 153–174.
[51] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.
[52] F. Chollet. (2015). *Keras*. [Online]. Available: https://keras.io
[53] Z. Han, M. Shang, Z. Liu, C.-M. Vong, Y.-S. Liu, M. Zwicker, J. Han, and C. L. P. Chen, "SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 658–672, Feb. 2019.
[54] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[55] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. AISTATS*, Mar. 2010, pp. 249–256.
[56] F. S. Nooruddin and G. Turk, "Simplification and repair of polygonal models using volumetric techniques," *IEEE Trans. Vis. Comput. Graphics*, vol. 9, no. 2, pp. 191–205, Apr. 2003.
[57] A.-A. Liu, S. Xiang, W.-Z. Nie, and D. Song, "End-to-End visual domain adaptation network for cross-domain 3D CPS data retrieval," *IEEE Access*, vol. 7, pp. 118630–118638, 2019.
[58] P. S. Andreev, W. Zhao, N.-Z. Wang, M. M. Smith, Q. Li, X. Cui, M. Zhu, and I. J. Sansom, "Early Silurian chondrichthyans from the Tarim Basin (Xinjiang, China)," *PLoS ONE*, vol. 15, no. 2, Feb. 2020, Art. no. e0228589.
[59] A.-A. Liu, H.-Y. Zhou, M.-J. Li, and W.-Z. Nie, "3D model retrieval based on multi-view attentional convolutional neural network," *Multimedia Tools Appl.*, vol. 79, nos. 7–8, pp. 4699–4711, Mar. 2019.
[60] W. Nie, K. Wang, H. Wang, and Y. Su, "The assessment of 3D model representation for retrieval with CNN-RNN networks," *Multimedia Tools Appl.*, vol. 78, no. 12, pp. 16979–16994, Jun. 2019.

YEMAO HOU received the B.S. and M.S. degrees in software engineering from Northeastern University, Shenyang, China, in 2007 and 2009, respectively. He is currently pursuing the Ph.D. degree in bioinformatics science and technology with Xidian University, Xi'an, China. His research interests include graph and image processing, machine learning, and computer vision.

XINDONG CUI received the B.S. degree in geology from the China University of Geosciences, Wuhan, China, in 2016. He is currently pursuing the Ph.D. degree in paleontology and stratigraphy with the University of Chinese Academy of Sciences, Beijing, China. His research interests include scale morphology and histology of early vertebrates, and the evolution of early vertebrates.

MARIO CANUL-KU received the B.S. degree in computer science from the University of Yucatan in 2011, and the M.Sc. and Ph.D. degrees in computer science from the Centro de Investigación en Matemáticas, Mexico, in 2013 and 2019, respectively. He is currently a Postdoctoral Researcher with the Department of Computer Science, CIMAT, Mexico. His research interests include oil spill detection, machine learning, deep learning, computer vision, augmented reality, 3D shape analysis, and image processing.

SHICHAO JIN received the B.S. degree in forestry from Huazhong Agricultural University, Wuhan, China, in 2016, and the Ph.D. degree in ecology from the Institute of Botany, Chinese Academy of Sciences, Beijing, in 2020. He is currently an Associate Professor with the Plant Phenomics Research Center, Nanjing Agricultural University, Nanjing. His research focuses on using deep learning and LiDAR technology to solve phenotyping related challenges.

ROGELIO HASIMOTO-BELTRAN (Member, IEEE) received the Ph.D. degree in computer and electrical engineering from the University of Delaware, Newark, DE, USA, in 2001. He spent two years with Akamai Technologies (a leader enterprise in multimedia content delivery) as a Senior Software Engineer. He was a Visiting Associate Professor with the University of Illinois at Chicago, from 2009 to 2010. He has been a Senior Scientific Researcher with the Department of Computer Science, Center for Research in Mathematics (CIMAT), Mexico, since 2003. He has published more than 40 technical papers in refereed conferences and journals in the area of image processing, computer vision, and multimedia networks. His current research interests include robust multimedia communication, error concealment, face detection and recognition, and chaotic encryption.

QINGHUA GUO received the B.S. degree in environmental science and the M.S. degree in remote sensing and geographic information systems (GIS) from Peking University, Beijing, China, in 1996 and 1999, respectively, and the Ph.D. degree in environmental science from the University of California at Berkeley, Berkeley, CA, USA, in 2005. He is currently a Professor with the Institute of Botany, Chinese Academy of Sciences, Beijing. He is also an Adjunct Professor and a member of the Founding Faculty, School of Engineering, University of California at Merced, Merced, CA, USA. His research interests include GIS and remote sensing algorithm development and their environmental applications, such as object-based image analysis, geographic one-class data analysis, and LiDAR data processing.

MIN ZHU received the Ph.D. degree in vertebrate paleontology from the Chinese Academy of Sciences (CAS). He conducted Postdoctoral Research at the National Museum of Natural History in Paris and at the Museum of Natural Science in Berlin. Through more than 100 scientific articles and books, he is recognized for revitalizing the quest for the origin of osteichthyans (bony fishes and tetrapods) from nonosteichthyan gnathostome groups, backed by his field work to unveil many early fossils with exceptional character complement, and for making unique contributions to the research on the fish-tetrapod transition and the jaw origin as well. He was the Director of the Institute of Vertebrate Paleontology and Paleoanthropology, CAS, Beijing, from December 1999 to February 2008. He is currently the Director of the Key CAS Laboratory of Vertebrate Evolution and Human Origins, and a Professor with the University of Chinese Academy of Sciences and Nanjing University.