IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 22, 2018, accepted December 6, 2018, date of publication December 14, 2018, date of current version January 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886791

Classification of 3D Archaeological Objects Using Multi-View Curvature Structure Signatures

MARIO CANUL-KU[1], ROGELIO HASIMOTO-BELTRAN[1], (Member, IEEE), DIEGO JIMÉNEZ-BADILLO[2], SALVADOR RUIZ-CORREA[3], (Member, IEEE), AND EDGAR ROMÁN-RANGEL[4]

[1]Centro de Investigación en Matemáticas, Guanajuato 36023, Mexico
[2]Instituto Nacional de Antropología e Historia, Mexico City 06700, Mexico
[3]Instituto Potosino de Investigación Científica y Tecnológica (CNS-IPICYT), San Luis Potosi 78216, Mexico
[4]Digital Systems Department, Instituto Tecnológico Autónomo de México, Mexico City 01080, Mexico

Corresponding author: Rogelio Hasimoto-Beltran (hasimoto@cimat.mx)

This work was supported in part by the Mexican Council of Science and Technology (CONACyT)-FORDECYT-Consorcio de Inteligencia Artificial under Project #296737 and in part by the RedTDPC-CONACyT through the Program of Thematic Research Networks.

ABSTRACT We propose a generalized 3D shape descriptor for the efficient classification of 3D archaeological artifacts. Our descriptor is based on a multi-view approach of curvature features, consisting of the following steps: pose normalization of 3D models, local curvature descriptor calculation, construction of 3D shape descriptor using the multi-view approach and curvature maps, and dimensionality reduction by random projections. We generate two descriptors from two different paradigms: 1) handcrafted, wherein the descriptor is manually designed for object feature extraction, and directly passed on to the classifier and 2) machine learnt, in which the descriptor automatically learns the object features through a pretrained deep neural network model (VGG-16) for transfer learning and passed on to the classifier. These descriptors are applied to two different archaeological datasets: 1) non-public Mexican dataset, represented by a collection of 963 3D archaeological objects from the Templo Mayor Museum in México City that includes anthropomorphic sculptures, figurines, masks, ceramic vessels, and musical instruments; and 2) 3D pottery content-based retrieval benchmark dataset, consisting of 411 objects. Once the multi-view descriptors are obtained, we evaluate their effectiveness by using the following object classification schemes: K-nearest neighbor, support vector machine, and structured support vector machine. Our object descriptors classification results are compared against five popular 3D descriptors in the literature, namely, rotation invariant spherical harmonic, histogram of spherical orientations, signature of histograms of orientations, symmetry descriptor, and reflective symmetry descriptor. Experimentally, we were able to verify that our machine learnt and handcrafted descriptors offer the best classification accuracy (20% better on average than comparative descriptors), independently of the classification methods. Our proposed descriptors are able to capture sufficient information to discern among different classes, concluding that it adequately characterizes the datasets.

INDEX TERMS 3D shape descriptor, multi-class classification, multi-view approach, curvature, transfer learning.

I. INTRODUCTION

Thanks to continuous advances in laser scanning and photogrammetric techniques [67], there has been a significant increase in the production of 3D models of Cultural Heritage (CH) objects, especially in fields like archaeology, museology, conservation and art history. More recently, CH researchers have joined efforts with computer vision and machine learning specialists to identify and classify in an automated manner several ancient artifacts based on their shape. This has led to an explosion of exciting applications to determine, for example, the provenance of Gallo-Roman figurines [7], as well as to characterize the style of artworks, such as Emperor Augustus portraits [57], Chinese terracotta warriors [4] and Classical Roman sculptures [89], among many others.

The growing interest in the production and use of 3D digital models has motivated the need for developing intelligent computer systems for the management, visualization, classification, retrieval, etc; of this kind of data. Globally available tools for the recognition and classification of 3D models from repositories -such as EUROPEANA- would have an enormous impact in the field of CH. It would easily allow to download models to expand their dissemination among scientists, students and the general public [46].

One of the main challenges in building such a system is finding the best combination of algorithms to recognize, retrieve and classify 3D models. So far, most papers have been only focused on improving the discriminatory power of shape descriptors, both at global and local levels to facilitate the classification and retrieval. Bibliographic surveys on the subject have been published in the field of computer vision since the 1980's (see [3], [8], [54], [55], [80], [81]).

3D search engines on the other hand, provide only retrieval functionality. For example, the Princeton 3D Search Engine allows multi-modal searches from sketches and from 3D query models introduced by the user [25] (see also http://shape.cs.princeton.edu/search.html). Another example is the use of FlightField shape descriptors to allow content-based searches [12]. More recently, Filali Ansary *et al.* [1] proposed a solution that allows the user to input photographic images, as well as sketches and 3D query-models to retrieve similar objects. By applying different combinations of shape descriptors and indexing algorithms, these search engines have solved the problem of retrieval with different degrees of success. All of them are able to discriminate between 3D models that have clearly different shapes, a car from a person or a vase from a chair for example. Unfortunately, this level of performance is not enough in the field of CH. As Scopigno *et al.* [72] rightly point out: "shape-based search methodologies do not fulfill the expectations of the CH community. They are able to distinguish among very different objects, but fail on identifying similar objects for which, more advanced methods are required."

In other words, we need to develop methods that are able to distinguish subtle differences in objects of the same category, for example, variations of anthropomorphic figurines, vessels or archaeological masks. It is common that archaeologists rely heavily on subjective perceptions about the artifacts shape, which is reflected in several discrepancies among classifications given in the archaeological community. In this context, it is necessary to go beyond the implementation of description and retrieval algorithms to incorporate classification procedures in a more objective manner based on 3D shape geometry of archaeological artifacts. Gorisse *et al.* [34] tackled this problem producing an interactive classification and retrieval system called "3D-RETIN".

The goal of this research is to develop a generalized Hand-Crafted (HC) and Machine Learnt (ML) 3D shape descriptors (described in section II) based on a Multi-View approach of curvature values, aimed at classifying objectively archaeological artifacts through machine learning techniques (Support Vector Machine-SVM [38], Structured Support Vector Machine-SSVM [83] and K-Nearest Neighbor-KNN [62]). Our results show that our proposed descriptors effectively extracts the most important geometric information of 3D models. They get the highest classification scores among 5 different state of the art descriptors in the literature, such as Rotation Invariant Spherical Harmonic (SPH) [43], Histogram of Spherical Orientations (HoSO) [68], Signature of Histograms of OrienTations (SHOT) [82], Symmetry Descriptor (SYD) [42], and Reflective Symmetry descriptor (RSD) [44]. A total average of 93.6% of classification accuracy is obtained by HC and ML descriptors, independently of the classification scheme and datasets (Mexican and 3D Pottery).

The remaining of this paper is organized as follows. The section II discusses 3D shape descriptors mentioned in the literature. Section III explains our shape representation methodology based on *Multi-View* and curvature map values. In the section IV, we show the experimental set-up and results of the 3D shape representations and their classification accuracy. Finally conclusions are present in the section V.

II. RELATED WORK

Recently, 3D shape representation methods have been based on the development of shape descriptors, which are numerical representations of 3D models. This representation combines mathematical, statistical and efficient computational methods for their construction [80]. The results are 3D shape descriptors, that encode the shape geometry in a numerical array or feature vector $\boldsymbol{x} = [x_1, x_2, \ldots, x_d]$. The resulting features are dependent on the particularities of each 3D object model information, and can be characterized by local or global information.

A global descriptor describes the model as a whole, and local descriptors describe patches in the model. For instance, Figure 1a represents a side-view of a 3D model projected onto a 2D plane (single image), from which local and global descriptors are to be computed. A local descriptor describes a property or metric around a key point (pixel in 2D or

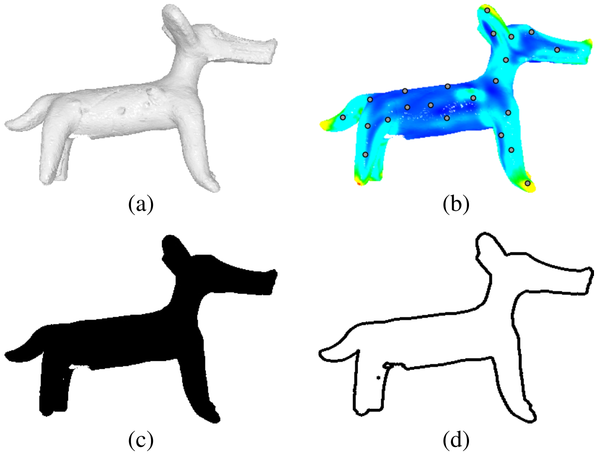


FIGURE 1. Approach examples used to construction of 2D shape descriptors. (a) 2D projection. (b) Local curvature around vertices. (c) Global shape area. (d) Global shape contour.

vertex in 3D) that represents a common behavior of a particular ("small") neighborhood or patch. Depending on the application area, this property can be represented by curvature, gradient orientation, average distance with respect to closes neighbors, among the most common in [18] and [70]. Figure 1a depicts the projected local curvature values of the 3D model that particularly characterizes it from any other similar model. Global descriptors on the other hand, are presented by the shape area and object contour in Figures 1c and 1d respectively, metrics that characterizes the entirely object.

There exist different methods for the construction of 3D shape descriptors, the most popular approaches are based on Vertices [9], [35], [41], [53], [80], Multi-View Projections (MVP), and recently Transfer Learning [13], [20], [29], [59], [88]. The first three approaches can be considered as HC descriptors [40] (manually designed features), whereas Transfer Learning is considered a ML scheme, in which a particular neural network model learns and extracts the database input features or descriptors [36]. Among all descriptors, the vertex approach is the most commonly used in the literature, it encodes geometry properties of points such as areas and angles, and characterizes the shape by means of global or local properties. The following methods are based on the vertex approach:

- **Histogram of Spherical Orientations (HoSO)** [68]. It is a *local* 3D shape descriptor of dimension 1024, that uses concentric spheres around a vertex of interest. The local descriptor is built by a Gaussian kernel in order to obtain a histogram of orientation distances between the interest vertex and its neighborhood. A Bag of Words (BoW) model is then used to created a global descriptor (HoSO-BoW) of the 3D model. The purpose of BoW is to transform local descriptors into global descriptors for object classification and matching [16]. It consists of two main processes, codebook construction and global histogram generation:
    1) *Codebook construction:* Let $\mathcal{D} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N_L}\}$ be local descriptors values of all object in the dataset drawn on the same coordinate system. The local descriptors in $\mathcal{D}$ are clusterized by applying any clustering algorithm, for example K-means. Their corresponding cluster centroids are then used to initialize a new set $\mathcal{M} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K\}$ named "Codebook". This method has a computational complexity of $\mathcal{O}(N_L^{(d\times K)})$ [39], where N_L is the number of *local* descriptors and K the clusters number. So, if N_L is very large the codebook construction is very hard.
    2) *Global descriptor construction:* The number of elements on each cluster centroid or Codebook is used to build a frequency histogram, where each bin summarizes a group of O_i objects with similar values, for $\mathcal{O}(i) = N$ and N being the total number of objects in the dataset.
- **Signature of Histograms of OrienTations (SHOT)** [82]. It is a *local* 3D descriptor close related to HoSO, that makes use of concentric spheres around a vertex of interest. For each vertex, SHOT defines a Normal Vector Reference-NVR (computed from its closest neighbors) and a discrete 3D sphere centered at vertex of interest. Next, the angle differences between NVR and the normal vector from a discrete point over the sphere is computed in order to generate a histogram of angles. This histogram represents the *local* descriptor. The final global descriptor (SHOT-BoW) is created by the BoW model (see HoSO-BoW description).
- **Spherical Harmonic descriptor (SPH)** [43]. In this work, authors make use of 3D surfaces as an approximation $\boldsymbol{f}$ of the real model, which is obtained by a linear combination of 3D harmonic functions $\boldsymbol{f}_i$. The linear coefficients α_i are used as invariant global shape descriptor, as show in Figure 2. The coefficients α_i represent the proportion that each function contributes to the surface. The result is an efficient and invariant descriptor with respect to rotation transforms.

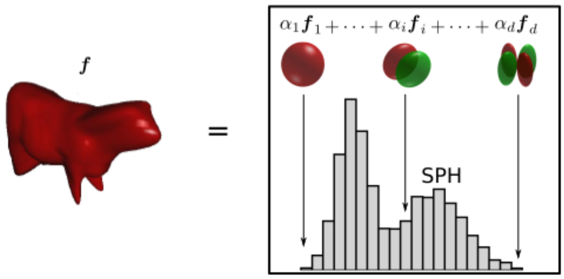

f
=
$\alpha_1 \boldsymbol{f}_1 + \cdots + \alpha_i \boldsymbol{f}_i + \cdots + \alpha_d \boldsymbol{f}_d$
SPH

FIGURE 2. Surface approximation f generated by linear combination of harmonic functions $\boldsymbol{f}_i$.

- **Reflective and Symmetry Descriptors (RSD)**. These methods are based on symmetry distances between a vertex $\boldsymbol{v}_i$ and its reflection $\gamma(\boldsymbol{v}_i)$ with respect to symmetry plane l_d that passes through the origin as depicted in Figure 3. The blue dotted lines represent the symmetric distances between a vertex a its corresponding reflection, which are perpendicular to symmetry plane. Each component of the shape descriptor vector is computed by the sum of all symmetric distances corresponding to particular symmetry plane l_d. After all symmetry planes and symmetric distances have been computed, the resulting descriptor is a global rotational invariant vector with respect all symmetry planes of the 3D model (additional details can be found in [42] and [44]).

Different than the vertices approaches, the view methods make use of *Multi-View* Projections (MVP) in order to abstract the shape of the 3D model [15], [32], [51], [63], [79]. Since the pioneer work of Cyr and Kimia [17] on MVP, several new methods have been proposed in the literature, some of the most recent are:

- **MVP based on Local Visual Features**. One of the most representative methods in this group is the

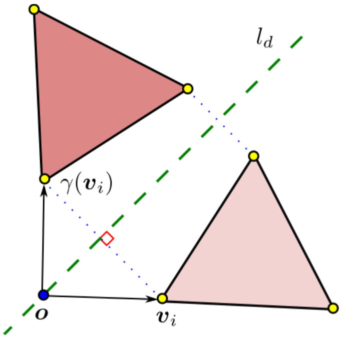


FIGURE 3. Illustration of symmetry between vertices. The blue dotted lines represent the symmetric distances between the vertices and their reflections, with respect to symmetry plane l_d.

"Scale-Invariant Feature Transform (SIFT)", which encodes local features of partial views that are invariant to rotation, scale, zooming, etc [56]. SIFT local features are then combined with the BoW model to create a global descriptor of 3D model. The main drawback of this approach is related to the generation of the codebook (see appendix), which needs to handle large training set of local features. Additional details of this methods can be found in [60] and [61].

- **MVP methods based on probabilistic models**. They model the local features (or descriptors) of the object partial views through a Gaussian Mixture Model (GMM) [86], and calculate how similar two objects are based on a similarity function. For example, in [86], GMM is used to model the Zernike moment descriptors (proposed in [45] and [52]), using as a similarity function the upper bound of the Kullback-Leibler divergence of the corresponding probabilistic models (see also [27]). These previous works the method cannot be applied to complex objects as in our case. We use archaeological artifacts with very complex and distinct geometry.

In this work, we propose a combined method based on curvature values [9], [26], [30], [50], [74], [87], [90] and Multi-View approach, that produces descriptors with reduced distortions and effective classification independently of the classification scheme, as shown in the section IV. As far as we know, there are no methods that consider both curvature and MVP. As part of this study, we are also interested on applying the state-of-the-art Transfer Learning-TL techniques to our proposed descriptor, and analyze its effect on our small-scale and complex datasets: Mexican and 3D Pottery (described in subsection IV-A). TL is novel paradigm that takes in a Deep Neural Network (DNN) pretrained model from a large-scale dataset [20], [64], to transfer this previously learnt knowledge onto a new problem or dataset.

One of the assumptions behind TL is that, the small-scale dataset content be similar to the original large-scale training set for the transfer knowledge to be meaningful or successful (this is a relative term and dependent on the application area). Some of the most relevant applications of TL have been developed for machine-learnt feature extraction in perceptual problems [75], such as emotion recognition [59], 3D matching and registration [29], [88], pavement distress detection [33], and classification of 3D Computer Aided Design (CAD) models [79]. Su *et al.* [79] demonstrated the effectiveness of transfer learning in the classification of 3D CAD models datasets. In particular, they consider a simple 2D color appearance (Phong reflexion model [73]) as input to a pretrained DNN model (VGG-M model) for feature extraction [11]. As far as we know, TL has not been applied to complex 3D shapes repositories of archaeological artifacts.

III. METHODOLOGY

We describe the methodology to obtain 3D shape descriptors based on local curvature values and the MVP model. Our proposed descriptor, can be summarized by the following steps:

1) Pose normalization of the 3D models.
2) Curvature descriptors calculation.
3) 3D shape descriptor construction based on MVP and curvature maps.

The output of the previous steps represents what we call an HC descriptor, a manually designed object feature extractor. To produce the ML descriptor based on TL, we add an additional step as follows:

4) Transformation of the HC descriptor into an ML descriptor by applying a pretrained DNN model obtained from a different large-scale dataset.

Finally, we apply the following step scheme to both HC and ML descriptors:

5) Dimensionality reduction by Random Projections (RP).
6) Classification process.

A. POSE NORMALIZATION

3D models are given in arbitrary scale, position and orientation as a consequence of the degrees of freedom in the digitization process. Therefore, applications in classification, retrieval, and recognition require a pose normalization step before any processing. Pose normalization fix the objects in a common reference frame defined by canonical or global vectors $\boldsymbol{e}_i$, as shown in Figure 4. It consists of three basic transformations, translation, scaling, and rotation, which altogether must not modify the intrinsic geometry of the objects.

Translation transformation moves the centroid of the object to the origin of the canonical or global reference system, as depicted in Figure 4. The centroid position is denoted by $\boldsymbol{o}$, is computed by:

$$\boldsymbol{o} = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{p}_i \tag{1}$$

where $\boldsymbol{p}_i \in \mathcal{P}$ represented by (x_i, y_i, z_i), and $\mathcal{P} = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_n\}$ is the set of vertices. The actual normalization consists in the subtraction of the centroid from the

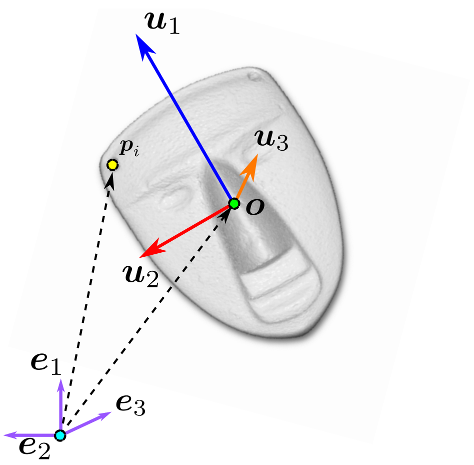


FIGURE 4. **3D model given in arbitrary scale, position and orientation with respect to a global or canonical coordinate system $\{e_1, e_2, e_3\}$.**

position of each vertex in the 3D model, so the new translation invariant vertex is defined as:

$$\boldsymbol{p}'_i = \boldsymbol{p}_i - \boldsymbol{o} \tag{2}$$

The second transformation is scaling, which is performed by first computing the largest distance between the centroid and a vertex as follows:

$$t = \max_{\boldsymbol{p}_i \in \mathcal{P}} \|\boldsymbol{p}_i - \boldsymbol{o}\|_2 \tag{3}$$

we use the inverse of t as a scaling factor, to obtain a new vertex inside the unitary sphere defined as:

$$\boldsymbol{p}''_i = \frac{1}{t} \times \boldsymbol{p}'_i \tag{4}$$

By applying the translation and scaling transformations to the object, we obtain a normalized 3D model as shown in Figure 5.

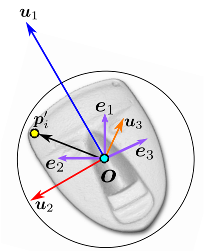


FIGURE 5. **Normalized 3D model in terms of scale (inside the unitary sphere) and translation.**

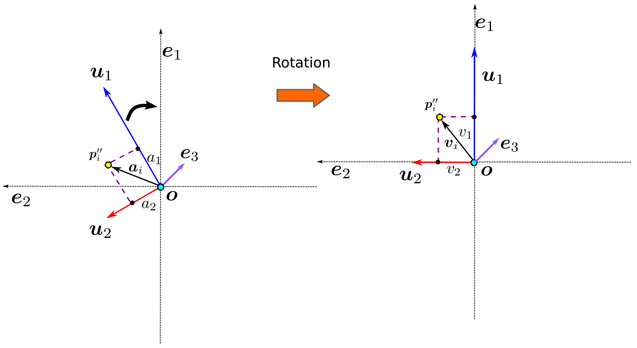


FIGURE 6. **Free body diagrams for the unaligned and canonical coordinate system, before and after rotation around axis e_3.**

Finally, we need to rotate the 3D model with respect to one of the axes in the canonical frame. For this, an additional reference frame is required, which serves to define the orientation of the object, called unaligned frame and represent by $\boldsymbol{u}_i$ vectors. To visualize the relationship between the aligned and unaligned frames, we make use of free body diagrams. These diagrams are graphical illustration to connect bodies with angular momentums (rotations) and forces [37]. Suppose that our 3D model in Figure 6 only requires a rotation with respect to the axis $\boldsymbol{e}_3$, we can see that the vertex $\boldsymbol{p}''_i = (x''_i, y''_i, z''_i)$ is represented in the unaligned frame by $\boldsymbol{a}_i$ (before rotation) and $\boldsymbol{v}_i$ in the new rotated frame, aligned with respect to $\boldsymbol{e}_1$.

The vectors $\boldsymbol{a}_i = [a_1, a_2, a_3]$ and $\boldsymbol{v}_i = [v_{i,1}, v_{i,2}, v_{i,3}]$ are related by:

$$v_{i,1} = |a_1| \tag{5a}$$
$$v_{i,2} = |a_2| \tag{5b}$$
$$v_{i,3} = |a_3| \tag{5c}$$

The components of $\boldsymbol{a}_i$ are the projections of $\boldsymbol{p}''_i$ over the vectors $\boldsymbol{u}_i$, and corresponds to the component of the rotated vertex $\boldsymbol{p}''_i$ in the canonical reference frame represented by $\boldsymbol{v}_i$. This relationship between components is formulated as:

$$v_{i,1} = |a_1| = \boldsymbol{u}_1^T \boldsymbol{p}''_i \tag{6a}$$
$$v_{i,2} = |a_2| = \boldsymbol{u}_2^T \boldsymbol{p}''_i \tag{6b}$$
$$v_{i,3} = |a_3| = \boldsymbol{u}_3^T \boldsymbol{p}''_i \tag{6c}$$

Using matrix notation, we obtain the transformation between both coordinate systems, represented by:

$$\begin{bmatrix} v_{i,1} \\ v_{i,2} \\ v_{i,3} \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} \begin{bmatrix} x''_i \\ y''_i \\ z''_i \end{bmatrix} \tag{7a}$$
$$\boldsymbol{v}_i = U^T \boldsymbol{p}''_i \tag{7b}$$

where U is the rotation matrix, which in our case is calculated by the method of Principal Components Analysis [2] (PCA). PCA is an orthonormal linear transformation defined by eigenvectors $\boldsymbol{w}_i$, that transfers the data to a new coordinate system such that, the greatest variance by any projection of the data lies on the first coordinate (first principal component), the second greatest variance lies on the second coordinate (second principal component), and so on. For example, in Figure 5 the largest eigenvector $\boldsymbol{w}_1$ is a unit vector in the direction of axis $\boldsymbol{u}_1$ of the mask. Since these vectors are collinear, there exists a relationship of the form $\boldsymbol{u}_1 = \lambda_1 \boldsymbol{w}_1$. The eigenvectors can be interpreted by a geometrical rotation of the canonical axes, and the reference frame will represent the directions of the maximum variability of the vertices in terms of their covariance matrix distances, defined by:

$$\Sigma = \frac{1}{t \times (n-1)} \sum_{i=1}^{n} (\boldsymbol{p}_i - \boldsymbol{o})(\boldsymbol{p}_i - \boldsymbol{o})^T \tag{8}$$

The eigenvectors are obtained by the Singular-Value Decomposition (SVD) of the vertex covariance matrix, which is defined as follows:

$$\Sigma = W\Lambda^2 W^T \tag{9}$$

where the matrix W contains the eigenvectors used as coordinate system, and the diagonal matrix Λ stores the eigenvalues of the covariance matrix. Geometrically speaking, the relation $U = W\Lambda$ corresponds to a matrix factorization of a rotation matrix W and scale matrix Λ, that is, allow us to align 3D models with arbitrary orientation, that is, we applied to each vertex the transformation defined as:

$$\boldsymbol{v}_i = W^T \boldsymbol{p}_i'' \tag{10}$$

Finally, we obtain a normalized set of vertices $\mathcal{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}$, which are used for curvature calculation.

B. CURVATURE DESCRIPTORS

Local curvature, is a way of measuring or describing the concavity of a surface at a point that lies on a curve line. An important property we take advantage of in this work, is that curvature is invariant against rigid transformations (translation, rotation and scale) [49], implying that the same curvature values are obtained independently of the orientation of the 3D model. This property represents a point of reference to distinguish different surface categories [21].

Local curvature descriptors of a vertex $\boldsymbol{v}_i$, can be characterized by the maximum κ_1 and minimum κ_2 curvature values. The directions of κ_1 and κ_2 denoted by $\boldsymbol{u}_1$, $\boldsymbol{u}_2$ respectively, happens to be orthogonal and coplanar with respect to the tangent plane defined by the normal vector $\boldsymbol{n}_i$ at the vertex $\boldsymbol{v}_i$. In Figure 7, we can see that the normal $\boldsymbol{n}_i$ and the orthogonal axes $\boldsymbol{u}_1$, $\boldsymbol{u}_2$ define a reference frame oriented along the principal curvatures κ_1 and κ_2.

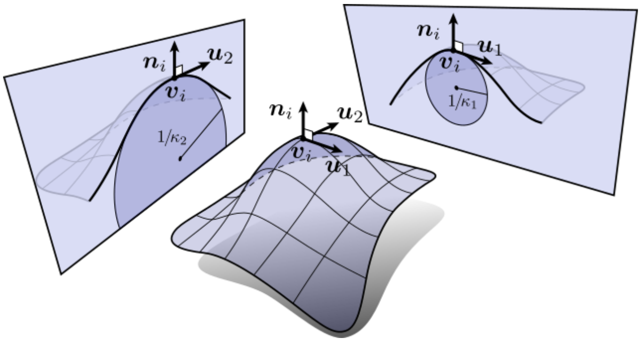


FIGURE 7. **Principal curvatures directions u_1 and u_2 for a vertex v_i (Original image taken from: http://brickisland.net/cs177fa12/?p=214).**

As shown in [65], an eigen-analysis of the covariance matrix of the normal vector projections around a local neighborhood, can be used to compute the local curvature principal components $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ as follows:

$$A = \frac{1}{N_K - 1} \sum_{k=1}^{N_K} (\hat{\boldsymbol{n}}_k - \bar{\boldsymbol{n}})(\hat{\boldsymbol{n}}_k - \bar{\boldsymbol{n}})^T \tag{11}$$

where the point $\bar{\boldsymbol{n}}$ is the centroid of all N_K nearest normal neighbors of $\boldsymbol{v}_i$, and $\hat{\boldsymbol{n}}_k = B\boldsymbol{n}_k$ is the normal projection obtained by means a projection matrix defined as follow:

$$B = I - \boldsymbol{n}_i \boldsymbol{n}_i^T \tag{12}$$

Therefore, given a spherical neighborhood around each vertex $\boldsymbol{v}_i$ of the 3D model, we can obtain the two greatest eigenvalues of covariance matrix A by means the SVD eigendecomposition, just as follow:

$$A = U\Lambda^2 U^T \tag{13}$$

where Λ contains the two greatest eigenvalues corresponding to the principal curvatures [71], which are obtained from equations 11 and 13. By notation, we will refer to local principal curvatures descriptors associated with a vertex $\boldsymbol{v}_i$ as follows:

$$\text{Maximal curvature vertex} \quad \kappa_{i,1} \tag{14a}$$
$$\text{Minimal curvature vertex} \quad \kappa_{i,2} \tag{14b}$$

Local curvature descriptors have been used to create curvature maps [28]. The minimum and maximum local curvatures descriptors are then used for creating curvature maps of the 3D models. Curvature map is a representation that resembles a heat map, in which curvature descriptors are represented with a color palette. For example, Figure 8 shows some samples of the most popular curvature maps: Maximum, Minimum, Gaussian, Mean and Shape Index, where the highest curvature values are represented in red color.

We can see that Mean and Shape Index descriptors, highlight the most prominent geometric features of the 3D model. For this reason, they will be used (along with MVP) in our final shape descriptor representation. The Mean curvature and Shape Index descriptor are defined as follows:

$$\text{Mean curvature} \quad \frac{\kappa_{i,1} + \kappa_{i,2}}{2} \tag{15a}$$
$$\text{Shape index} \quad \frac{2}{\pi} \tan^{-1} \frac{\kappa_{i,1} + \kappa_{i,2}}{\kappa_{i,1} - \kappa_{i,2}} \tag{15b}$$

It is important to point out that the size of the neighborhood (N_k) determines the spatial representation or locality of curvature (shown in [85]), the smaller the size, the higher the locality of the curvature.

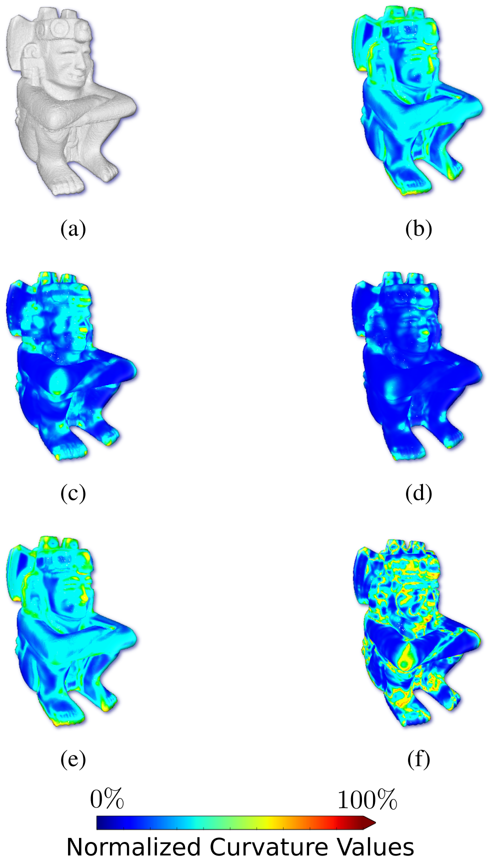


FIGURE 8. Curvature maps associated to different curvature descriptors. (a) Default color. (b) Maximal curvature κ_1. (c) Minimal curvature κ_2. (d) Gaussian. (e) Mean. (f) Shape index.

C. SHAPE DESCRIPTOR CONSTRUCTION BASED ON MVP

Our proposed descriptor, is the result of combining the curvature maps (subsection III-B) with multiple views of the 3D object. The geometric intuition behind our approach, is that curvature descriptors of similar objects are also similar, even in 2D views [10]. Each view is a render image, created by projecting the colored curvature maps of the 3D model over a 2D plane (Figure 9). In computer graphics, the process of automatically creating a render image, is known as "rendering process" [73].

In our methodology the rendering process can be described as follows:

1) For each 3D object, use its center of mass to generate a sphere that fully surrounds the model. Then, subdivide the sphere into 36 regions as depicted in Figure 9. The intersection points (black dots on the sphere) define the uniformly sampled locations of the camera.
2) These 36 camera locations around the sphere correspond to 6 slices defined along the horizontal plane (azimuth axis), and 6 stacks in the vertical direction (zenith axis).
3) Each camera location captures a partial view of the colored 3D model, which is rendered into an RGB image, and converted to gray-scale with dimension $d = W \times L$. This image is scanned in a top-down and left-to-right fashion to obtain a vector $\boldsymbol{I} \in \mathcal{R}^d$ or descriptor for this partial view.
4) The final descriptor is constructed by concatenating all $\boldsymbol{I}_i$ vectors of the 3D model taken from different perspectives (i.e. Multi-View approach):

$$\boldsymbol{x}_{HC} = \{\boldsymbol{I}_1, \boldsymbol{I}_2, \ldots, \boldsymbol{I}_{N_V}\} \tag{16}$$

with dimension $W \times L \times N_v$. For $W = 32$, $L = 32$, and $N_V = 36$ (the number of 2D projections or views), the resulting shape descriptor size is 36864.

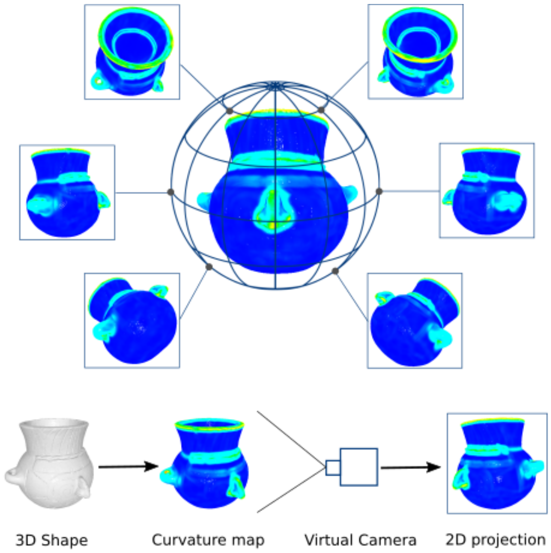


FIGURE 9. 3D *Multi-View* object descriptor construction, using the Mean curvature map.

D. TRANSFER LEARNING FOR DESCRIPTOR CONSTRUCTION BASED ON MVP

Deep Learning (DL) methods have been very popular in the field of machine learning. This is related to several factors, among them are the increase on computing performance (clusters, cpus, gpus), larger datasets that allow more robust training of such complex models, which in turn lead to high accuracy classification rates [20]. However (as mentioned before), on small-scale datasets (as the ones we are dealing with) DL methods cannot guarantee correct classification of new input data (data not used during the training process) due to the "overfitting problem" [31], [77]. A good alternative for applying DNN on small datasets, is to reuse a pretrained model with a large-scale dataset and use the learnt network parameters for the generation of new descriptors (this is called Transfer Learning-TL). The descriptors' outputs preserve important features learnt during the training process with the large-scale dataset. For the pretrained network we make use of the VGG-16 DNN (see Figure 10), developed by the Oxford University-Visual Geometry Group (VGG) [76] as a test bed for image classification systems for the ImageNet Large-scale Visual Recognition Challenge (ILSVRC) [69]. ImageNet consists of 1.4 million of RGB images with dimension of 224 × 224 pixels, partitioned into 1000 classes.

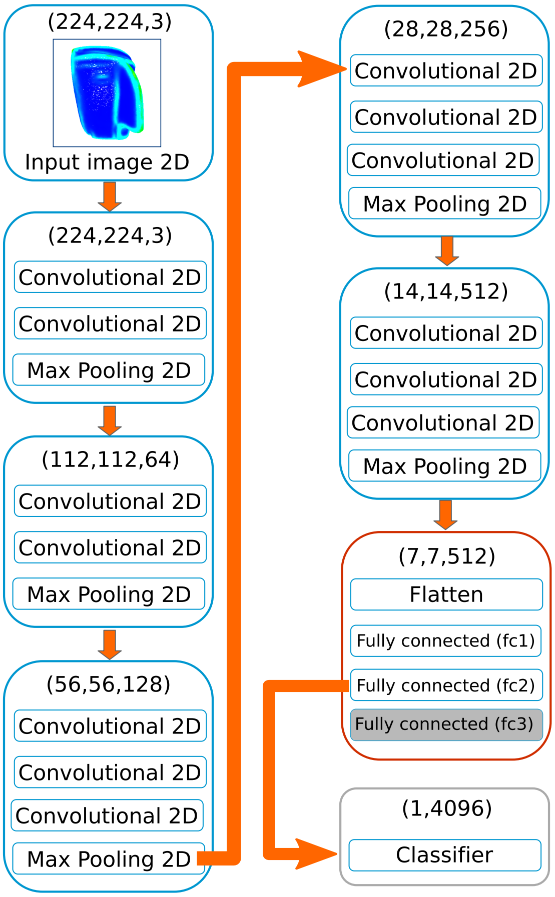


FIGURE 10. VGG-16 neural network architecture employed for the 3D shape Multi-View feature extraction, fc3 layer is replaced by the SSVM, SVM and KNN classifiers in our experiments.

An expanded version of our RGB HC-descriptor (from 32 × 32 to 224 × 224) is taken as input to the VGG-16 model, which is processed by thirteen convolutional 2D layers split into 5 blocks and two fully connected layers fc1 and fc2 that performs dimensionality reduction. The output $\boldsymbol{t}_i$ of the fully connected layer fc2, represents our ML vector descriptor with length $d = 4096$. Instead of going into fc3 classification layer in the VGG-16 (it performs the object classification based on the pretrained large-scale dataset), it is passed on to the SVM, SSVM, and KNN for the classification process based on our experimental datasets (Mexican and 3D Pottery).

The final descriptor is constructed by concatenating all $\boldsymbol{t}_i$ vectors (i.e. 3D shape *Multi-View* feature extraction approach) as follows:

$$\boldsymbol{x}_{ML} = \{\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_{N_V}\} \tag{17}$$

where $N_V = 36$, and the resulting shape descriptor size is 4096 × 36.

E. DIMENSIONALITY REDUCTION BY RANDOM PROJECTIONS (RP)

Our method creates high-dimensional descriptors (HC = [32 × 32 × 36] and ML = [4096 × 36]) with redundant information, that comes from the overlapping regions during the Multi-View scanning process, as can be appreciated in Figure 9. It is possible however, to efficiently reduce descriptors' length without (significantly) affecting important information of the 3D models, that is curvature values in our case.

This problem has been widely studied in [78], commonly known as dimensional reduction technique. In particular, Random Projection (RP) [19] is an efficient and simple technique that transforms a dataset through a random projection matrix R. This transformation has the property to preserve pairwise distances and similarities of the input vectors or descriptors, and more importantly, it is independently of the dataset. The matrix entries $R_{i,j}$, are random variables from a Gaussian distribution with mean $\mu = 0$ and variance $\sigma^2 = \frac{1}{d}$ (where d is the descriptors' length), which produces an orthogonal matrix in a highly dimensional space.

The matrix $R \in \mathcal{R}^{m \times d}$ maps each vector $z_i \in \{\boldsymbol{I}_i, \boldsymbol{t}_i\}$ to a lower dimensional vector $z_i^{RP} \in \mathcal{R}^m$ $(m << d)$, as follows:

$$z_i^{RP} = Rz_i \tag{18}$$

and the new shape descriptor is defined as:

$$\boldsymbol{x}^{RP} = \{z_1^{RP}, z_2^{RP}, \ldots, z_{N_V}^{RP}\} \tag{19}$$

We apply the above dimensionality reduction process to our proposed HC and ML descriptors based on both shape index and mean curvature values. For comparative purposes, we make a distinction between these two descriptors for HC and ML before and after applying the random projection, i.e., the HC descriptors before RP are named Hand-Crafted Multi-View Shape Index Descriptor (HC-MVSID) and Hand-Crafted Multi-View Curvature Descriptor (HC-MVMCD), whereas HC-MVSID-RP and HC-MVMCD- RP are the names after RP. The same nomenclature is applied for he ML descriptors, that is ML-MVMCD, ML-MVSID, ML-MVMCD-RP, and ML-MVSID-RP. The final step in our methodology is the classification process.

F. CLASSIFICATION PROCESS

Supervised learning is a machine learning approach that considers prior information for categorazing an object (words, voice, images, 3D models, etc) within a certain class (that is, objects with similar features). When the problem is the classification of multiple objects over multiple classes, the problem is known as a Multi-Class Classification (MCC) problem [5]. The output of MCC can be in the form of a label or number associated to the class membership of the input object, or can be a more complex structure (structured prediction) [6] of objects themselves, relationship among objects, meanings of a sentence, a graph with words associated to a certain topic etc. The MCC input information is given in the form of descriptors denoted by the set X, with corresponding labelset (or classes) output Y. The correspondences between X and the right output Y is defined by $\mathcal{S} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_N, y_N)\}$, which is a subset of the cartesian product $X \times Y$.

Classification methods assign correspondences between X and Y by a prediction function $F : X \rightarrow Y$, which is determined during the training process. Next, we describe the three classifiers used on our experimental section.

1) K-NEAREST NEIGHBORS (KNN)

KNN [62] is a simple geometric method that makes use of distances between the input descriptor and the ones in the dataset for object classification. For example, Figure 11 depicts the closest neighbors of an archaeological mask given as query. The training process consist on the construction of a search-tree data structure from the dataset. This classifier looks for the nearest neighbors to the input descriptor, before assigning the object class. That is, given a query $\boldsymbol{x}_q$, the search-tree returns the subset $\mathcal{S}_q \subseteq \mathcal{S}$ defined by:

$$\mathcal{S}_q = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{S} : \ \|\boldsymbol{x}_i - \boldsymbol{x}_q\|_2 \leq \|\boldsymbol{x}_{i+1} - \boldsymbol{x}_q\|_2, \\ \forall i, \ 1 \leq i \leq k\} \quad (20)$$

where k is the number of nearest neighbors, and $\mathcal{S}_q$ is the set of closest descriptors to $\boldsymbol{x}_q$, along their respective labels or class information. $\mathcal{S}_q$ is used in a prediction function that finds the class $\boldsymbol{x}_q$ belongs to. The prediction function is defined as follows:

$$F(\boldsymbol{x}_q) = \underset{\boldsymbol{y}_c \in Y}{\operatorname{argmax}} \sum_{\boldsymbol{y}_i \in \mathcal{S}_q} \mathbb{1}\{\boldsymbol{y}_c = \boldsymbol{y}_i\} \quad (21)$$

where $\boldsymbol{y}_c$ is an element of labelset (or class), and the prediction label $\hat{\boldsymbol{y}} = F(\boldsymbol{x}_q)$ corresponds to the class with the highest frequency label. Figure 11 shows $k = 5$ closest neighbors from 2 different classes, blue and red with two and three members respectively. In this case, the prediction function assigns $\boldsymbol{x}_q$ to the red class. The disadvantage of this method is the search time, which is in the order of $\mathcal{O}(N \log N)$ (where N is the size of the dataset). Therefore KNN is no recommended for large-scale datasets.

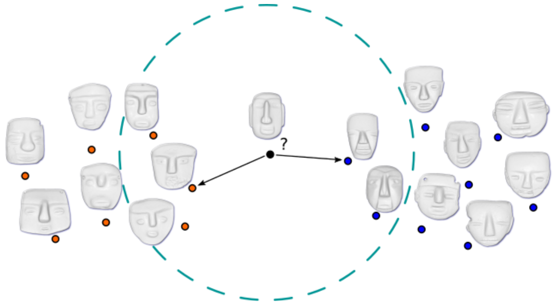


FIGURE 11. Illustration of K-Nearest Neighbors (KNN) method for shape classification.

2) SUPPORT VECTOR MACHINES

The SVM and SSVM compute hyperplanes that split the dataset into C regions or classes. Figure 12 shows an example of a three-region hyperplane defined by perpendicular parameters $\boldsymbol{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3]$ (which are unknown). The parallel bars (dashed lines) represent the prediction error margins for each hyperplane. $\boldsymbol{W}$ are used to formulate the prediction function $F : (X, \boldsymbol{W}) \rightarrow Y$, defined as follows:

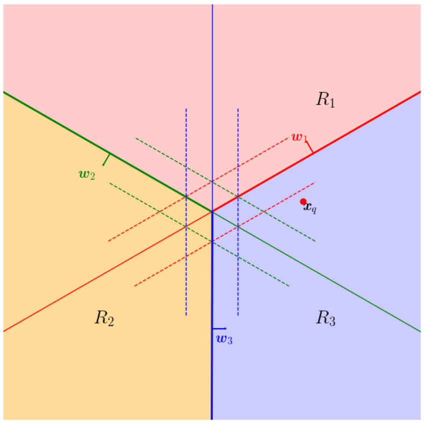


FIGURE 12. Regions splited by hyperplanes and associated to three classes.

$$\begin{aligned} F(\boldsymbol{x}_q, \boldsymbol{W}) &= \underset{c \in \{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_3\}}{\operatorname{argmax}} \ \boldsymbol{w}_c^T \boldsymbol{x}_q \\ &= \underset{\boldsymbol{w}_c \in \boldsymbol{W}}{\operatorname{argmax}} \ \boldsymbol{W}^T \boldsymbol{x}_q \end{aligned} \quad (22)$$

the operation $\boldsymbol{w}_c^T \boldsymbol{x}_q$ is the distance of $\boldsymbol{x}_q$ to each hyperplane $\boldsymbol{w}_c$, where the maximum distance defines the class of $\boldsymbol{x}_q$. For the SVM classifier [38], the prediction function in equation 22 is named ‘‘one-vs-rest approach’’ [23].

The unknown parameter $\boldsymbol{W}$, is computed by a quadratic optimization problem defined by:

$$\begin{aligned} & \underset{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_C}{arg\,min} \ \frac{1}{2} \sum_{c=1}^{C} \boldsymbol{w_c}^T \boldsymbol{w_c} + \gamma \sum_{i=1}^{N} \sum_{c \neq y_i}^{C} \xi_i^c \\ & \text{s.t } \boldsymbol{w_{y_i}}^T \Phi(\boldsymbol{x}_i) + b_{y_i} \geq \boldsymbol{w_c}^T \Phi(\boldsymbol{x}_i) + b_c + 2 - \xi_i^c \\ & \quad \xi_i^c \geq 0, \quad c \in \{1, \ldots, C\} \setminus y_i \\ & \quad \forall i = 1, \ldots, N \end{aligned} \quad (23)$$

where ξ_i^c is a slack variable that controls the prediction error margins, $\boldsymbol{w_{y_i}}$ is the normal vector to the hyperplane, b_{y_i} represents an offset with respect to the data space, $\Phi(\boldsymbol{x}_i)$ is a feature transformation function, and γ is regularization parameter associated to the margins.

SVM classifier output is represented by a label or a class number (simple prediction). In addition to simple predictions, SSVM (a generalization of the SVM) can be used for more complex structured predictions rather than a single class, that is it can be related to an image, concept, semantic meaning, etc. [6], [83]. The optimization problem for SSVM is defined as:

$$\begin{aligned} & \underset{\boldsymbol{W}, \boldsymbol{\xi}}{arg\,min} \ \frac{1}{2} \|\boldsymbol{W}\|_F^2 + \gamma \sum_{i=1}^{N} \xi_i \\ & \text{s.t } \Delta(\boldsymbol{y}, \boldsymbol{y}_i) + \langle \boldsymbol{W}, \Psi(\boldsymbol{x}, \boldsymbol{y}) - \Psi(\boldsymbol{x}, \boldsymbol{y}_i) \rangle_F \leq \xi_i \\ & \quad \xi_i \geq 0 \\ & \quad \forall i = 1, \ldots, N \\ & \quad \forall \boldsymbol{y} \in Y \setminus \boldsymbol{y}_i \end{aligned} \quad (24)$$

where $\Delta(\boldsymbol{y}, \boldsymbol{y}_i)$ is the loss function (Ex. Hamming loss function [84]) and $\Psi(\boldsymbol{x}, \boldsymbol{y})$ is a junction feature function that weights the feature vector with respect to the labelset Y.

IV. EXPERIMENTAL RESULTS

Our proposed Multi-View descriptors are aimed at extracting the most relevant features that characterize 3D archaeological objects. These features should contribute to a good classification independently of the classification algorithm being applied. To test their performance for the object classification problem, we have selected three multi-class classification methods for the experiments: SVM, SSVM and KNN. The first two classifiers were chosen because of their robustness to overfitting [5], while KNN classifier is the most intuitive to code, as well as the most used learning algorithms [62]. More complex algorithms such as DNN [20], [77], require large databases (in the order of thousands of objects) for the training phase. Our non-public archaeological database contains less than 1000 objects, with unbalance number of classes in terms of number of instances, therefore training with DNN will incure in overfitting [31], [48], [59].

Next, we present the experimental protocol used to validate our methodology, and discuss our results.

A. DATASETS

We evaluated our propose descriptors on two different 3D models of archaeological datasets:

- **Mexican Dataset**. We use a sample of 963 digitizations of archaeological artifacts from the collection of the Templo Mayor museum. The sample includes 3D model of different objects such as anthropomorphic sculptures, figurines, masks, ceramic vessels, musical instruments, etc. The original artifacts were elements of ritual offerings deposited for religious purposes in the main ceremonial complex of Tenochtitlan, the capital of the Mexica Empire. Due to its variety and symbolic significance, the collection is one of the most important research resources for specialists of the Mexica (i.e. Mexican) civilization. This dataset is partitioned into 10 categories (in compliance with Mexican Archaeologists classification), whose distribution is showed in Figure 13, and a sample of 3D models (one per class) are shown in Figure 14.

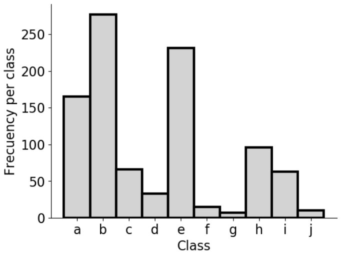


FIGURE 13. Class distribution of the Mexican dataset.

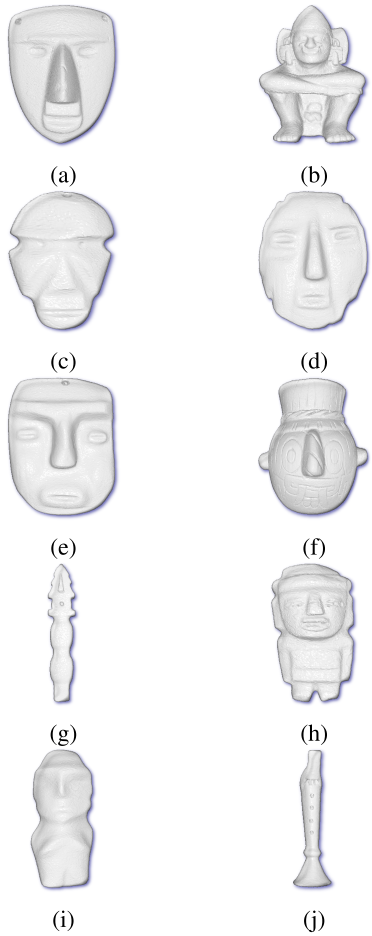


FIGURE 14. Random sample of shapes (one per class) chosen from the Mexican dataset. (a) Sultepec. (b) Xiuhtecuhtli. (c) Mezcala. (d) Chontal Typical. (e) Chontal nose ball. (f) Tláloc vessel. (g) Canes. (h) Flat statuettes. (i) Cylindrical statuettes. (j) Flutes.

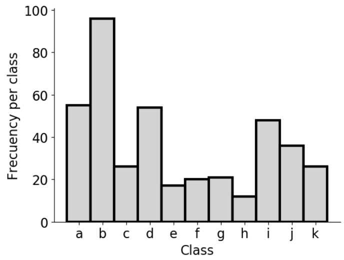


FIGURE 15. Class distribution of the 3D Pottery dataset.

- **3D Pottery Dataset**. This dataset, own by the Hampson Archaeological Museum in Arkansas [47] contains 993 models, partitioned into 23 categories. Since the number of artifacts per class is very sparse, we selected 411 models partitioned in 11 classes, which have an homogeneous geometry, whose distribution and sample of 3D models (one per class) are visualized on the Figures 15 and 16 respectively.

B. EVALUATION

The effectiveness of our proposed descriptors (HC-MVMCD, HC-MVSID, HC-MVMCD-RP, HC-MVSID-RP,

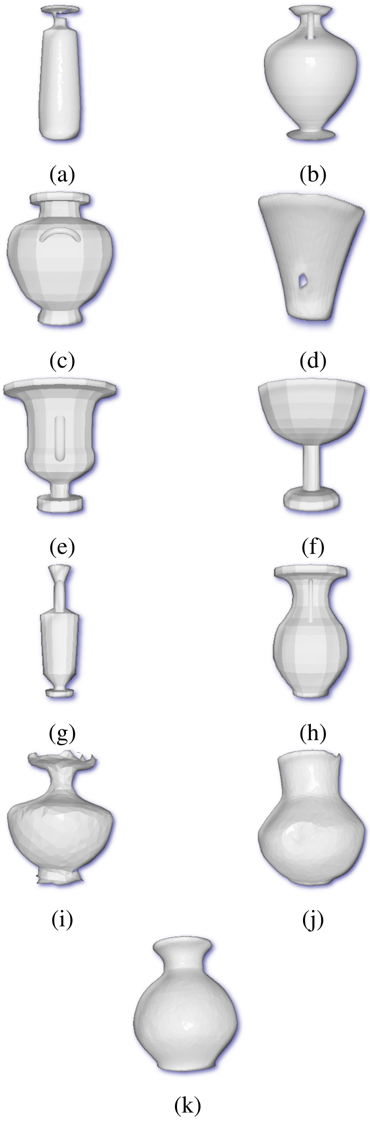


FIGURE 16. Random sample of shapes chosen (one per class) from the 3D Pottery dataset. (a) Alabastron. (b) Amphora. (c) Hydria. (d) Kalathos. (e) Krater. (f) Kylix. (g) Lekythos. (h) Pelike. (i) Psykter. (j) Native American. (k) Picher Shaped.

TABLE 1. Specification of 3D shape descriptors.

Descriptors	Abbrev.	Dimensionality
ML Multi-View Shape Index Descriptor	ML-MVSID	4096×36
ML Multi-View Mean Curvature Descriptor	ML-MVMCD	4096×36
ML Multi-View Shape Index Descriptor Random Projection	ML-MVSID-RP	64×36
ML Multi-View Mean Curvature Descriptor Random Projection	ML-MVMCD-RP	64×36
HC Multi-View Shape Index Descriptor	HC-MVSID	$32 \times 32 \times 36$
HC Multi-View Mean Curvature Descriptor	HC-MVMCD	$32 \times 32 \times 36$
HC Multi-View Shape Index Descriptor Random Projection	HC-MVSID-RP	$8 \times 8 \times 36$
HC Multi-View Mean Curvature Descriptor Random Projection	HC-MVMCD-RP	$8 \times 8 \times 36$
Spherical Harmonic [43]	SPH	256
Symmetry Descriptor [42]	SYD	1922
Reflective Symmetry Descriptor [44]	RSD	3844
Histogram of Spherical Orientations [68]	HoSO-BoW	100
Signature of Histograms of OrienTations (SHOT) [82]	SHOT-BoW	100

ML-MVSID, ML-MVMCD, ML-MVSID-RP, ML-MVMCD-RP), is compared against the following descriptors in the literature (Table 1) : HoSO-BoW [68], SHOT-BoW [82], SPH [43], SYD [42] and RSD [44], described in section II. HoSO-BoW is of recent creation, with proven efficiency with respect to SHOT-BoW, Fast Point Feature Histograms (FPFH) [70] and Scale-invariant features (SISI) [18], among others. The other descriptors we compare with (SPH, SYD and RSD), have been referenced in numerous works on matching and ranking tasks [80].

The evaluation process consists in splitting each dataset into two random sets. The first set, groups 70% of the 3D objects for knowledge acquisition or training, while the second set (remaining 30%) is used in the test phase or classification process. The computation of the mean curvature and shape index in our descriptors is slightly different depending on the dataset involved. Since Mexican dataset comes with higher resolution models than 3D Pottery (lower resolution models), we found that a spherical neighborhood with a normalized radius = 0.06 is sufficient to compute the mean curvature and shape index around vertex $\boldsymbol{v}_i$. For the 3D Pottery dataset on the other hand, a neighborhood of $N_k = 30$ vertices around $\boldsymbol{v}_i$ provides enough resolution to analyze the archaeological artifacts in our experiments.

For the construction of ML descriptors, we use the pre-trained VGG-16 model from Keras [14]. Regarding object classification task, we use scikit-learn [66] implementation of the KNN and SVM (one-vs-rest approach) methods, and pystruct [58] for the SSVM implementation. SVM and SSVM where implemented with linear kernel and regularization parameter $\gamma = 0.1$ in all our experiments. Classification results involve 30 randomly generated training and testing sets, for which the *classification accuracy* (ACC) [22], [84] along with its *confidence intervals* is reported as statistical proof.

Finally, in order to visualize our descriptors dispersion (or similarity) among intra and inter classes, we compute a similarity matrix for each dataset (Mexican and 3D Pottery), as discuss in subsection IV-D.

C. CLASSIFICATION RESULTS

A comparison of the classification accuracy and confidence intervals for the proposed and discussed descriptors are summarized in Tables 2-4. Overall, our descriptors ML and HC get the highest ACC scores in all classification tests and datasets, which are on average 20.2% better than the rest of compared descriptors SPH, SYD, RSD, HoSO-BoW, and SHOT-BoW. Among our own descriptors, ML descriptors are the clear winners with approximately 3.1% better performance than HC descriptor ones. Individually, there is not a clear winner among the ML descriptors regarding classification scheme and dataset, that is the first and forth places only differ by 0.57%. What is important to point out is that, dimensionality reduction by Random Projections (from 4096 × 36 to 64 × 36) does not affect the performance accuracy of the classifiers in both ML and HC descriptors. The average ACC difference between original length (ML-MVMCD and ML-MVSID) and dimensionally reduced

TABLE 2. SSVM descriptors' classification accuracy and confidence interval for both Mexican and 3D Pottery datasets.

	SSVM	
Descriptor	Mexican	3D Pottery
ML-MVSID	$\mathbf{0.978 \pm 0.013}$	0.933 ± 0.038
ML-MVMCD	0.977 ± 0.014	$\mathbf{0.937 \pm 0.040}$
ML-MVSID-RP	$\mathbf{0.978 \pm 0.012}$	0.922 ± 0.045
ML-MVMCD-RP	0.975 ± 0.013	0.927 ± 0.047
HC-MVSID	0.956 ± 0.015	0.886 ± 0.038
HC-MVMCD	0.947 ± 0.021	0.901 ± 0.042
HC-MVSID-RP	0.962 ± 0.014	0.879 ± 0.043
HC-MVMCD-RP	0.964 ± 0.017	0.885 ± 0.038
SPH	0.915 ± 0.023	0.839 ± 0.046
SYD	0.929 ± 0.027	0.789 ± 0.048
RSD	0.854 ± 0.018	0.706 ± 0.065
HoSO-BoW	0.608 ± 0.030	0.499 ± 0.003
SHOT-BoW	0.495 ± 0.080	0.334 ± 0.033

TABLE 3. Same as Table 2, for the SVM classifier.

	SVM	
Descriptor	Mexican	3D Pottery
ML-MVSID	$\mathbf{0.978 \pm 0.014}$	0.934 ± 0.041
ML-MVMCD	0.977 ± 0.014	$\mathbf{0.938 \pm 0.042}$
ML-MVSID-RP	$\mathbf{0.978 \pm 0.013}$	0.921 ± 0.047
ML-MVMCD-RP	0.975 ± 0.012	0.925 ± 0.047
HC-MVSID	0.964 ± 0.016	0.904 ± 0.040
HC-MVMCD	0.967 ± 0.012	0.883 ± 0.046
HC-MVSID-RP	0.962 ± 0.017	0.890 ± 0.051
HC-MVMCD-RP	0.965 ± 0.011	0.875 ± 0.045
SPH	0.940 ± 0.016	0.840 ± 0.050
SYD	0.944 ± 0.027	0.679 ± 0.129
RSD	0.904 ± 0.021	0.725 ± 0.061
HoSO-BoW	0.543 ± 0.012	0.542 ± 0.053
SHOT-BoW	0.314 ± 0.001	0.248 ± 0.014

TABLE 4. Same as Table 2, for the KNN classifier.

	KNN	
Descriptor	Mexican	3D Pottery
ML-MVSID	$\mathbf{0.974 \pm 0.015}$	0.922 ± 0.037
ML-MVMCD	0.970 ± 0.013	$\mathbf{0.932 \pm 0.004}$
ML-MVSID-RP	0.970 ± 0.015	0.918 ± 0.043
ML-MVMCD-RP	0.970 ± 0.013	0.926 ± 0.044
HC-MVSID	0.955 ± 0.015	0.876 ± 0.058
HC-MVMCD	0.945 ± 0.015	0.880 ± 0.036
HC-MVSID-RP	0.950 ± 0.016	0.861 ± 0.055
HC-MVMCD-RP	0.941 ± 0.016	0.875 ± 0.037
SPH	0.945 ± 0.021	0.853 ± 0.052
SYD	0.945 ± 0.019	0.752 ± 0.063
RSD	0.942 ± 0.016	0.764 ± 0.056
HoSO-BoW	0.955 ± 0.016	0.838 ± 0.057
SHOT-BoW	0.922 ± 0.028	0.557 ± 0.068

TABLE 5. Total average classification performance below the best ranked descriptor ML-MVMCD.

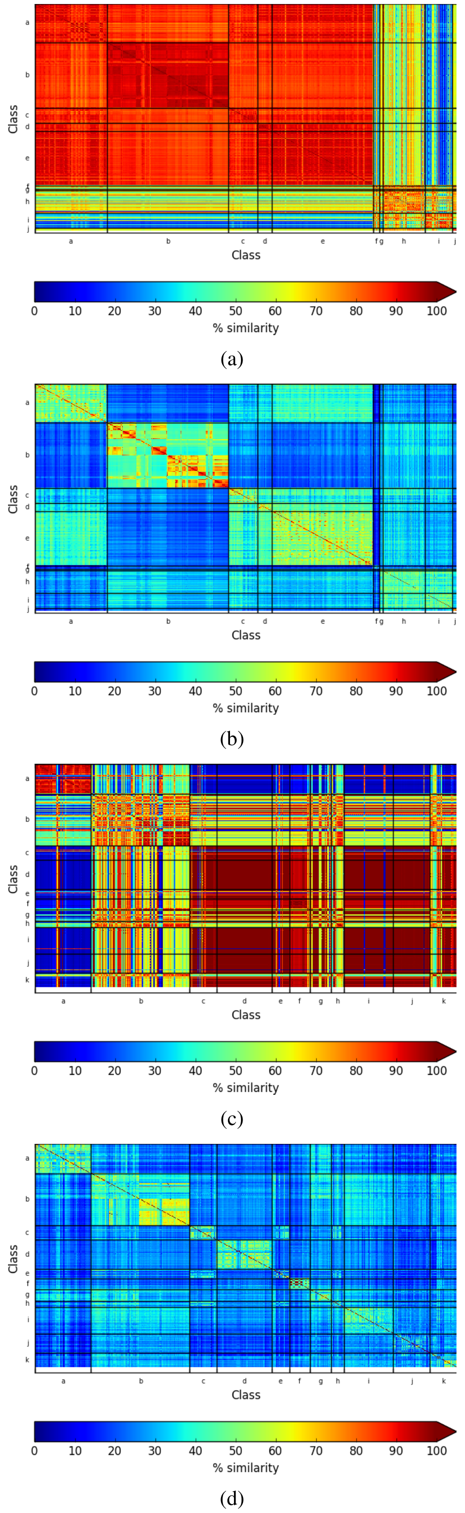
Descriptor	%
ML-MVSID	0.20
ML-MVMCD-RP	0.57
ML-MVSID-RP	0.57
HC-MVSID	3.01
HC-MVMCD	3.62
HC-MVMCD-RP	3.94
HC-MVSID-RP	3.96
SPH	6.87
SYD	12.09
RSD	14.58
HoSO-BoW	30.46
SHOT-BoW	45.54

(ML-MVMCD-RP and ML-MVSID-RP) descriptors considering all classifiers and datasets is around 0.47% (not a significant difference). Similar behavior is found on HC descriptors, with average ACC difference of 0.64%. Considering now all descriptors in Table 1, ML-MVMCD and ML-MVSID get the highest ACC total average scores (across all classifiers and datasets) of 0.955 and 0.953 respectively. The third-best descriptor is ML-MVMCD-RP, with a total average accuracy of 0.95, representing only 0.57% below the first place descriptor ML-MVMCD (as shown in Table 5). SPH descriptor came out to be in ninth place (fifth place if RP descriptors are not considered), with a total average of 0.89, corresponding to 6.9% below ML-MVMCD. SPH and SYD obtained adequate classification accuracy, showing a more stable behavior than RSD, HoSO-BoW, and SHOT-BoW. It is surprising that HoSO-BoW and SHOT-BoW got high score in the KNN classifier, but its performance on SSVM and SVM was low and unstable, ending up with the worst classification scores. An additional problem not related to HoSO-BoW and SHOT-BoW but to KNN classifier, is the time required to search for the nearest neighbors in reference to the object descriptor in question [62]. This search is performed on a tree data structure in the order of $\mathcal{O}(N \log N)$ (where N is the size of the dataset). Therefore, KNN is not recommended for high-dimensional datasets. In contrast, SSVM and SVM methods (in which our proposed schemes were the best) only need a Matrix-vector product in the order of $\mathcal{O}(d \times C)$ (where $C << N$ is the number of classes), representing a much lower classification times. Furthermore, SVM and SSVM classifier offer better data generalization, which do not occur with KNN method (it is more sensitive to overfitting).

Even though our work is focused on the comparison of our proposed descriptors against state-of-the-art descriptors in the literature, we were curious about the performance of SVM and SSVM classifier involved in the experiment. The ML descriptors get exactly the same ACC values for the SVM and SSVM, while for the HC descriptors SVM is better than SSVM by 0.7%. Considering the rest of descriptors analyzed (SPH, SYD, RSD, HoSO-BoW, and SHOT-BoW), the order of performance changes, yielding SSVM (0.760), and SVM (0.734). It is clear that SSVM can be applied to wider variety of problems than the SVM, but regarding to classification problems, they are (theoretically speaking) very similar (the only change is in the cost function, particularly in the constraints' formulation), so we do not find a clear winner between these two classifiers, in this work.

D. SIMILARITY RESULTS

A similarity matrix (SM), is a pairwise graphical representation of data (or descriptors in our case) [24], that measures their similarity. The similarity value is based on the Euclidean distance between descriptors, which is normalized between



(a)

(b)

(c)

(d)

FIGURE 17. **Similarity matrix for best (ML-MVMCD) and worst (SHOT-BoW) descriptors used in classification for Mexican and 3D Pottery datasets. (a) SHOT-BoW for Mexican dataset. (b) ML-MVMCD for Mexican dataset. (c) SHOT-BoW for 3D Pottery dataset. (d) ML-MVMCD for 3D Pottery dataset.**

[0 – 100], where a similarity close to 100 means that the descriptors are very similar. Figure 17, shows the similarity matrices of ML-MVMCD and SHOT-BoW, the best and worst ranked descriptors respectively for Mexican and 3D Pottery datasets. Areas on red reflect descriptors with high similarity, while deep-blue color represents descriptors with low similarity. The diagonal red line represents a perfect match, since descriptors are being compared against themselves. Within the same class (or intra-class), high similarity values are expected (yellow-red color), otherwise low similarity (closer to blue) must be present. Looking at the similarity matrices for the Mexican and 3D Pottery datasets for ML-MVMCD and SHOT-BoW, our descriptor discerns much better the inter-class descriptors than SHOT-BoW. On the contrary, SHOT descriptors on both datasets show high similarity values among different classes; for this reason, it may get confused during the classification process, yielding poor results.

V. CONCLUSIONS

We have proposed two Multi-View descriptors based on mean local curvature (HC-MVMCD) and shape index (HC-MVSID) for 3D archaeological objects classification. Based on these descriptors, we created two different sets of descriptors, the first set was built using a VGG-16 pretrained model under the transfer learning paradigm, while the second set was hand-crafted, that is without automatic training for feature extraction. Our experiments are based on two archaeological datasets, Mexican (non-public) and 3D Pottery, which consist of 963 and 411 3D objects. Our proposed descriptors are compared against state-of-the-art descriptors in the literature, such as HoSO-BoW [68], SHOT-BoW [82], SPH [43], SYD [42], and RSD [44]. Our proposed 3D Multi-View descriptors (trained and untrained) obtained the highest classification scores, showing that efficiently capture the main structure of the models in both the Mexican and 3D Pottery datasets. They provide excellent classification accuracy independently of the classification method being used.

REFERENCES

[1] T. F. Ansary, M. Daoudi, and J.-P. Vandeborre, "A Bayesian 3-D search engine using adaptive views clustering," *IEEE Trans. Multimedia*, vol. 9, no. 1, pp. 78–88, Jan. 2007.

[2] B. Bellekens, V. Spruyt, R. Berkvens, and M. Weyn, "A survey of rigid 3D pointcloud registration algorithms," in *Proc. 4th Int. Conf. Ambient Comput., Appl., Services Technol. (AMBIENT)*, Rome, Italy, Aug. 2014, pp. 8–13.

[3] P. J. Besl and R. C. Jain, "Three-dimensional object recognition," *ACM Comput. Surv.*, vol. 17, no. 1, pp. 75–145, 1985.

[4] A. Bevan *et al.*, "Computer vision, archaeological classification and China's terracotta warriors," *J. Archaeological Sci.*, vol. 49, pp. 249–254, Sep. 2014.

[5] C. Bishop, *Pattern Recognition and Machine Learning* (Information Science and Statistics). New York, NY, USA: Springer, 2006.

[6] M. B. Blaschko and C. H. Lampert "Learning to localize objects with structured output regression," in *Computer Vision—ECCV*, D. Forsyth, P. Torr, and A. Zisserman, Eds. Berlin, Germany: Springer, 2008, pp. 2–15.

[7] A. Bourdeu and D. Pitzalis, "Geometric morphometrics for provenance determination of Gallo–Roman white clay figurines," in *Proc. 11th Int. Conf. Virtual Reality, Archaeology Cultural Heritage*, 2010, pp. 25–31.

[8] B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranic, "Automatic selection and combination of descriptors for effective 3D similarity search," in *Proc. IEEE 6th Int. Symp. Multimedia Softw. Eng.*, Dec. 2004, pp. 514–521.

[9] 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.

[10] M. Chaouch and A. Verroust-Blondet, "A new descriptor for 2D depth image indexing and 3D model retrieval," in *Proc. IEEE Int. Conf. Image Process.*, vol. 6, Sep./Oct. 2007, pp. VI-373–VI-376.

[11] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. (2014). "Return of the devil in the details: Delving deep into convolutional nets." [Online]. Available: https://arxiv.org/abs/1405.3531

[12] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On visual similarity based 3D model retrieval," *Comput. Graph. Forum*, vol. 22, no. 3, pp. 223–232, 2003.

[13] T. Chen, I. Goodfellow, and J. Shlens. (2015). "Net2net: Accelerating learning via knowledge transfer." [Online]. Available: https://arxiv.org/abs/1511.05641

[14] F. Chollet *et al.* (2015). *Keras*. [Online]. Available: https://keras.io

[15] D. Chrysostomou, A. Gasteratos, L. Nalpantidis, and G. C. Sirakoulis, "Multi-view 3D scene reconstruction using ant colony optimization techniques," *Meas. Sci. Technol.*, vol. 23, no. 11, pp. 114002, 2012.

[16] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. 8th Eur. Conf. Comput. Vis. (ECCV)*, Prague, Czech Republic, May 2004, pp. 1–2.

[17] C. M. Cyr and B. B. Kimia "3D object recognition using shape similiarity-based aspect graph," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1, Jul. 2001, pp. 254–261.

[18] T. Darom and Y. Keller, "Scale-invariant features for 3-D mesh models," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2758–2769, May 2012.

[19] S. Dasgupta, "Experiments with random projection," in *Proc. 16th Conf. Uncertainty Artif. Intell.* Burlington, MA, USA: Morgan Kaufmann, 2000, pp. 143–151.

[20] L. Deng and D. Yu, "Deep learning: Methods and applications," *Found. Trends Signal Process.*, vol. 7, nos. 3–4, pp. 197–387, Jun. 2014.

[21] 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.

[22] C. Elkan, "Evaluating classifiers," Univ. California, San Diego, CA, USA, Tech. Rep., 2012.

[23] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Aug. 2008.

[24] J. Foote, "Visualizing music and audio using self-similarity," in *Proc. 7th ACM Int. Conf. Multimedia*, 1999, pp. 77–80.

[25] T. Funkhouser *et al.*, "A search engine for 3D models," *ACM Trans. Graph.*, vol. 22, no. 1, pp. 83–105, 2003.

[26] R. Gal and D. Cohen-Or, "Salient geometric features for partial shape matching and similarity," *ACM Trans. Graph.*, vol. 25, no. 1, pp. 130–150, Jan. 2006.

[27] Y. Gao *et al.*, "Camera constraint-free view-based 3-D object retrieval," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2269–2281, Apr. 2012.

[28] T. Gatzke, C. Grimm, M. Garland, and S. Zelinka, "Curvature maps for local shape comparison," in *Proc. Int. Conf. Shape Modeling Appl.*, Jun. 2005, pp. 244–253.

[29] G. Georgakis, S. Karanam, Z. Wu, J. Ernst, and J. Kosecka, "End-to-end learning of keypoint detector and descriptor for pose invariant 3D matching," in *Proc. CVPR*, 2018, pp. 1965–1973.

[30] A. Gilboa, A. Tal, I. Shimshoni, and M. Kolomenkin, "Computer-based, automatic recording and illustration of complex archaeological artifacts," *J. Archaeological Sci.*, vol. 40, no. 2, p. 1329– 1339, 2013.

[31] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.

[32] S. Goel, A. Jain, P. Singh, S. Bagga, S. Batra, and U. Gaur, "Computer vision aided pottery classification and reconstruction," in *Proc. INDO US Sci. Technol. Forum Digit. Archaeol.*, 2005, pp. 1–14.

[33] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction Building Mater.*, vol. 157, pp. 322–330, Dec. 2017.

[34] D. Gorisse, M. Cord, M. Jordan, S. Philipp-Foliguet, and F. Precioso, "3D content-based retrieval in artwork databases," in *Proc. 3DTV Conf.*, May 2007, pp. 1–4.

[35] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and N. M. Kwok, "A comprehensive performance evaluation of 3D local feature descriptors," *Int. J. Comput. Vis.*, vol. 116, no. 1, pp. 66–89, 2016.

[36] J. Heaton, "An empirical analysis of feature engineering for predictive modeling," in *Proc. SoutheastCon*, Mar./Apr. 2016, pp. 1–6.

[37] J. R. Holton and G. J. Hakim, *An Introduction to Dynamic Meteorology*. Amsterdam, The Netherlands: Elsevier, 2004.

[38] 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.

[39] M. Inaba, N. Katoh, and H. Imai, "Applications of weighted Voronoi diagrams and randomization to variance-based k-clustering," in *Proc. 10th Annu. Symp. Comput. Geometry*, 1994, pp. 332–339.

[40] L. Jin, S. Gao, Z. Li, and J. Tang, "Hand-crafted features or machine learnt features? Together they improve RGB-D object recognition," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2014, pp. 311–319.

[41] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.

[42] M. Kazhdan, B. Chazelle, D. Dobkin, A. Finkelstein, and T. Funkhouser, "A reflective symmetry descriptor," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2002, pp. 642–656.

[43] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," in *Proc. Eurograph./ACM SIGGRAPH Symp. Geometry Process. (SGP)*. Aire-la-Ville, Switzerland: Eurographics, 2003, pp. 156–164.

[44] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Symmetry descriptors and 3D shape matching," in *Proc. Eurograph./ACM SIGGRAPH Symp. Geometry Process. (SGP)*, New York, NY, USA, 2004, pp. 115–123.

[45] A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 5, pp. 489–497, May 1990.

[46] D. Koller, B. Frischer, and G. Humphreys, "Research challenges for digital archives of 3D cultural heritage models," *J. Comput. Cultural Heritage*, vol. 2, no. 3, p. 7, 2009.

[47] A. Koutsoudis, G. Pavlidis, V. Liami, D. Tsiafakis, and C. Chamzas, "3D pottery content-based retrieval based on pose normalisation and segmentation," *J. Cultural Heritage*, vol. 11, no. 3, pp. 329–338, 2010.

[48] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[49] J. M. Lee, *Riemannian Manifolds: An Introduction to Curvature*, vol. 176. New York, NY, USA: Springer, 2006.

[50] H. Li, J.-M. Morvan, and L. Chen, "3D facial expression recognition based on histograms of surface differential quantities," in *Advanced Concepts for Intelligent Vision Systems*, J. Blanc-Talon, R. Kleihorst, W. Philips, D. Popescu, and P. Scheunders, Eds. Berlin, Germany: Springer, 2011, pp. 483–494.

[51] X.-X. Li, Q. Cao, and S. Wei, "3D object retrieval based on multi-view convolutional neural networks," *Multimedia Tools Appl.*, vol. 76, no. 19, pp. 20111–20124, Oct. 2017.

[52] M. Liu, Y. He, and B. Ye, "Image zernike moments shape feature evaluation based on image reconstruction," *Geo-Spatial Inf. Sci.*, vol. 10, no. 3, pp. 191–195, 2007.

[53] Z. Liu, C. Zhao, X. Wu, and W. Chen, "An effective 3D shape descriptor for object recognition with RGB-D sensors," *Sensors*, vol. 17, no. 3, p. 451, 2017.

[54] S. Loncaric, "A survey of shape analysis techniques," *Pattern Recognit.*, vol. 31, no. 8, pp. 983–1001, 1998.

[55] G. L. López, A. P. N. P. Negrón, A. D. A. Jiménez, J. R. Rodríguez, and R. I. Paredes, "Comparative analysis of shape descriptors for 3D objects," *Multimedia Tools Appl.*, vol. 76, no. 5, pp. 6993–7040, 2017.

[56] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[57] M. Lu *et al.*, "Portrait sculptures of augustus: Categorization via local shape comparison," in *Proc. Digit. Heritage Int. Congr. (DigitalHeritage)*, vol. 1, Oct./Nov. 2013, pp. 661–664.

[58] A. C. Müller and S. Behnke, "Pystruct–learning structured prediction in python," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 2055–2060, 2014.

[59] H.-W. Ng, V. D. Nguyen, V. Vonikakis, and S. Winkler, "Deep learning for emotion recognition on small datasets using transfer learning," in *Proc. ACM Int. Conf. Multimodal Interact.*, 2015, pp. 443–449.

[60] R. Ohbuchi and T. Furuya, "Scale-weighted dense bag of visual features for 3D model retrieval from a partial view 3D model," in *Proc. IEEE 12th Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Sep./Oct. 2009, pp. 63–70.

[61] R. Ohbuchi, K. Osada, T. Furuya, and T. Banno, "Salient local visual features for shape-based 3D model retrieval," in *Proc. IEEE Int. Conf. Shape Modeling Appl.*, Jun. 2008, pp. 93–102.

[62] S. M. Omohundro, "Five balltree construction algorithms," Int. Comput. Sci. Inst., Berkeley, CA, USA, Tech. Rep. TR-89-063, 1989.

[63] R. Pan and G. Taubin, "Automatic segmentation of point clouds from multi-view reconstruction using graph-cut," *Vis. Comput.*, vol. 32, no. 5, pp. 601–609, 2016.

[64] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[65] M. Pauly, M. Gross, and L. P. Kobbelt, "Efficient simplification of point-sampled surfaces," in *Proc. Conf. Vis.*, 2002, pp. 163–170.

[66] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[67] F. Remondino and S. El-Hakim, "Image-based 3D Modelling: A review," *Photogramm. Rec.*, vol. 21, no. 115, pp. 269–291, 2006.

[68] E. Roman-Rangel, D. Jimenez-Badillo, and S. Marchand-Maillet, "Classification and retrieval of archaeological potsherds using histograms of spherical orientations," *J. Comput. Cultural Heritage*, vol. 9, no. 3, pp. 17:1–17:23, Sep. 2016.

[69] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[70] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2009, pp. 3212–3217.

[71] 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, 2008.

[72] R. Scopigno *et al.*, "3D models for cultural heritage: Beyond plain visualization," *Computer*, vol. 44, no. 7, pp. 48–55, Jul. 2011.

[73] G. Sellers, R. S. Wright, Jr., and N. Haemel, *OpenGL Superbible: Comprehensive Tutorial and Reference*. Reading, MA, USA: Addison-Wesley, 2013.

[74] A. Shamir, "A survey on mesh segmentation techniques," *Comput. Graph. forum*, vol. 27, no. 6, pp. 1539–1556, 2008.

[75] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 512–519.

[76] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[77] D. Soekhoe, P. van der Putten, and A. Plaat, "On the impact of data set size in transfer learning using deep neural networks," in *Advances in Intelligent Data Analysis XV*, H. Boström, A. Knobbe, C. Soares, and P. Papapetrou, Eds. Cham, Switzerland: Springer, 2016, pp. 50–60.

[78] C. O. S. Sorzano, J. Vargas, and A. P. Montano. (2014). "A survey of dimensionality reduction techniques." [Online]. Available: https://arxiv.org/abs/1403.2877

[79] 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.*, Dec. 2015, pp. 945–953.

[80] J. W. H. Tangelder and R. C. Veltkamp, "A survey of content based 3D shape retrieval methods," in *Proc. Shape Modeling Appl.*, Jun. 2004, pp. 145–156.

[81] P. Theologou, I. Pratikakis, and T. Theoharis, "A review on 3D object retrieval methodologies using a part-based representation," *Comput.-Aided Des. Appl.*, vol. 11, no. 6, pp. 670–684, 2014.

[82] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2010, pp. 356–369.

[83] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *J. Mach. Learn. Res.*, vol. 6, pp. 1453–1484, Sep. 2005.

[84] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int. J. Data Warehousing Mining*, vol. 3, no. 3, pp. 1–13, 2007.

[85] R. Unnikrishnan, "Statistical approaches to multi-scale point cloud processing," M.S. thesis, Brown Univ., Providence, RI, USA, 2008.

[86] M. Wang, Y. Gao, K. Lu, and Y. Rui, "View-based discriminative probabilistic modeling for 3D object retrieval and recognition," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1395–1407, Apr. 2013.

[87] R. Zatzarinni, A. Tal, and A. Shamir, "Relief analysis and extraction," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 136:1–136:9, Dec. 2009.

[88] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3DMatch: Learning local geometric descriptors from RGB-D reconstructions," in *Proc. CVPR*, Jul. 2017, pp. 199–208.

[89] Y. Zhang *et al.*, "Classical sculpture analysis via shape comparison," in *Proc. Int. Conf. Culture Comput.*, Sep. 2013, pp. 57–61.

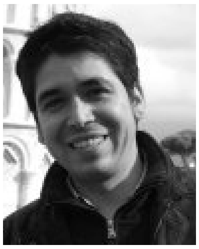
[90] K. Zou, Z. Zhang, J. Zhang, and Q. Zhang "A 3D model feature extraction method using curvature-based shape distribution," in *Proc. 12th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Aug. 2015, pp. 1809–1813.

MARIO CANUL-KU received the B.S. degree in computer science from the University of Yucatan, in 2011, and the M.Sc. degree in computer science from the Centro de Investigación en Matemáticas, Mexico, in 2013, where he is currently pursuing the Ph.D. degree in computer science. His research interests include machine learning, deep Learning, computer vision, augmented reality, 3D shape analysis, and image processing.

ROGELIO HASIMOTO-BELTRAN received the Ph.D. degree in computer and electrical engineering from the University of Delaware, 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, during 2009–2010. 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 over 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.

DIEGO JIMÉNEZ-BADILLO is currently a Senior Researcher with the Instituto Nacional de Antropología e Historia (INAH), Mexico. He is the Director of the project "Developing Computer Applications in Archaeology." He is the Chairman of Research Network for the Application of Digital Technologies to Cultural Heritage (RedTDPC). He implemented a 3D shape recognition system for analyzing archaeological artifacts, created the relative neighborhood method of spatial analysis (i.e., RN-method), applied a new methodology for the analysis of archaeological artifacts based on spectral clustering, and developed a database information system for managing records of archaeological objects within a museum environment. His first book is *Ofrendata: Un sistema de base de datos para controlar una colección arqueológica* (1998, INAH, Mexico). More recently, he has edited two books: *El Patrimonio Cultural y las tecnologías digitales: Experiencias recientes desde México* (2016, INAH, Mexico) and *Arqueología computacional: Nuevos enfoques para la documentación, análisis y difusión del patrimonio cultural* (2017, INAH, Mexico). His research areas include computer vision and machine learning applied to cultural heritage, spatial analysis using GIS, implementation of archaeological databases and data mining, development of new quantitative methods to analyze cultural heritage data, 3D modeling of archaeological sites, and artifacts.

SALVADOR RUIZ-CORREA received the Ph.D. degree in electrical engineering from the University of Washington. He is currently an Adjunct Professor and a Researcher with the Instituto Potosino de Investigación Científica y Tecnológica, where he leads the Youth Innovation Laboratory (You-i Lab). He co-directs the Center for Mobile Life (Ce Mobili), a research initiative in México. His research interests range from machine learning and computer vision applications to social computing in development contexts, data for social good, and citizen innovation. He is a member of the Sistema Nacional de Investigadores of Consejo Nacional de Ciencia y Tecnología.



EDGAR ROMÁN-RANGEL received the Ph.D. degree from the École Polytechnique Fédérale de Lausanne for his work in computer vision and pattern recognition applied to imagery of Mayan hieroglyphs, in 2013. From 2013 to 2017, he was a Postdoctoral Researcher with the CVM Lab, University of Geneva, with a break in 2014 and 2015 during which he was a Visiting Researcher at the National Institute of Anthropology and History, Mexico. He is a Visiting Researcher at the Instituto Tecnológico Autónomo de México. He has experience also in the banking sector, where he has provided consulting services in statistics and machine learning. His research interests include intersection between computer vision, deep learning, and representation learning, applied to cultural and societal challenges.