Eurographics Workshop on 3D Object Retrieval (2017)
I. Pratikakis, F. Dupont, and M. Ovsjanikov (Editors)

Exploiting the PANORAMA Representation for Convolutional Neural Network Classification and Retrieval

K. Sfikas[1], T. Theoharis[1] and I. Pratikakis[2]

[1]NTNU - Norwegian University of Science and Technology, Department of Computer Science, Trondheim, Norway
[2]Democritus University of Thrace, Department of Electrical and Computer Engineering, Xanthi, Greece

Abstract

A novel 3D model classification and retrieval method, based on the PANORAMA representation and Convolutional Neural Networks, is presented. Initially, the 3D models are pose normalized using the SYMPAN method and consecutively the PANORAMA representation is extracted and used to train a convolutional neural network. The training is based on an augmented view of the extracted panoramic representation views. The proposed method is tested in terms of classification and retrieval accuracy on standard large scale datasets.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Visible line/surface algorithms I.3.3 [Computer Graphics]: Picture/Image Generation—Viewing Algorithms I.5.1 [Pattern Recognition]: Models—Neural Nets

1. Introduction

This work proposes a novel 3D object classification and retrieval method, based on the PANORAMA [PPTP10] 3D shape descriptor and a convolutional neural network architecture. The proposed method suggests a learning strategy where the panoramic view images of the 3D models are fed to a convolutional neural network (CNN) [LBBH98, HL06].

The 3D models are initially pose normalized, based on their panoramic view images, using the SYMPAN [STP14] pose normalization algorithm which is based on reflective symmetry. Next, the PANORAMA representation is extracted from the pose normalized 3D models on both the spatial and orientation domains. These two distinct representations are concatenated to create an augmented panoramic view, used for training the convolutional neural network. It is further suggested that a reduction in the size of the augmented panoramic view representation actually benefits the training procedure.

The motivation behind the aforementioned method is that the PANORAMA representation is able to bridge the dimensionality gap between the 3D object space and the 2D image input typically required by a convolutional neural network, in a very efficient manner. The PANORAMA representation, previously proposed by our team, has already proven to be a successful hand-crafted 3D model descriptor which has achieved state of the art 3D model retrieval performance in various implementations [PPTP10, STP14, STP13, SPK*16]. One of these implementations is the SYMPAN pose normalization method. SYMPAN has achieved highly accurate results in 3D model pose normalization, as measured via 3D model retrieval performance.

The performance of the proposed method is evaluated in terms of accuracy on both 3D model classification and retrieval tasks. The dataset used for the evaluation is the publicly available Princeton ModelNet 3D CAD model dataset [WSK*15]. This dataset is especially designed for machine learning algorithms, thus containing both training and testing partitions. The ModelNet dataset is organized into two subsets, ModelNet-10, which is categorized into 10 model classes and containing 3D models that are manually cleaned and oriented, and ModelNet-40, categorized into 40 model classes, containing 3D models that are manually cleaned but not oriented.

The remainder of this paper is organized as follows: Section 2 briefly discusses recent works on 3D model classification and retrieval with emphasis on deep neural network methods. Section 3 details the proposed method. Section 4 presents the experimental procedure along with the corresponding results. Finally, in Section 5 conclusions are drawn and discussed.

2. Related Work

One categorization for 3D shape representation methods can be performed based on the dimensionality of the representation data: (a) 2D image-based representations (i.e. planar and panoramic projections), (b) 3D model-based representations (i.e. 3D shapes, point clouds and voxels) and (c) higher levels of data representations (i.e. 3D videos, doxels etc). Recent works of the first two categories will be discussed in the sequel, as these are most relevant to the problem at hand.

One of the most acknowledged methods for 3D object retrieval, based on the extraction of features from 2D representations of the 3D objects, was the Light Field descriptor, proposed by Chen et al. [CTSO03]. This descriptor comprises Zernike moments and Fourier coefficients computed on a set of projections taken at the vertices of a dodecahedron. Su et al. [SMKLM15] present a CNN architecture that combines information from multiple views of a 3D shape into a single and compact shape descriptor. They show that this descriptor is able to achieve higher recognition performance than single image recognition architectures. Papadakis et al. in [PPTP10] proposed PANORAMA, a 3D shape descriptor that uses a set of panoramic views of a 3D object which describe the position and orientation of the object's surface in 3D space. For each view the corresponding 2D Discrete Fourier Transform and the 2D Discrete Wavelet Transform are computed. Shi et al. in [SBZB15], convert each 3D shape into a panoramic view, namely a cylinder projection around its principle axis. Then, a variant of CNN is used for learning the representations directly from these views. A row-wise max-pooling layer is inserted between the convolution and fully-connected layers, making the learned representations invariant to the rotation around a principal axis. In [SBZB15], the authors use panoramic views that feed a CNN for 3D model categorization and retrieval. Although similar to PANORAMA, the authors do not use the two different representations of PANORAMA (one distance based and one angle-based), nor the three standard projection axes. Furthermore, although rotation invariance on one axis is achieved through a specially designed layer of the proposed CNN architecture, it is not described how the key problem of pose normalization is solved. (Panoramic views change drastically as the orientation of a 3D model varies).

In [KFR03] Kazhdan et. al. proposed the Spherical Harmonic Representation, a rotation invariant representation of spherical functions in terms of the energies at different frequencies. This descriptor is a volumetric representation of the Gaussian Euclidean Distance Transform of a 3D object, expressed by norms of spherical harmonic frequencies. Wu et al. [WSK*15] propose to represent a geometric 3D shape as a probability distribution of binary variables on a 3D voxel grid, using a Convolutional Deep Belief Network. Sinha et al. [SBR16] propose an approach of converting the 3D shape into a 'geometry image' so that standard CNNs can directly be used to learn 3D shapes, thus bridging the associated representation gap. They create geometry images using an authalic parametrization on a spherical domain. This spherically parameterized shape is then projected and cut to convert the original 3D shape into a flat and regular geometry image.

A summarized categorization of the aforementioned methods based on their usage of Machine Learning (ML) and their representation dimensionality is presented in Table 1.

Throughout the current literature, view-based representation methods have proven to be more accurate in both 3D model classification and retrieval (see also Section 4). The (view-based) method that was employed in this work is based on the successful hand-crafted PANORAMA descriptor representation, extending its usage based on CNNs. The proposed method, in a manner similar to, but in many ways extending [SBZB15], feeds a CNN with the PANORAMA representation (both distance and angular) for the three principal projection axes. In addition, the proposed method uses a PANORAMA-based pose normalization method ([STP14]), in order to normalize the panoramic views of the 3D models, a necessary step that ensures uniformity among the descriptors. In [SBZB15] the authors only use a single panoramic view (distance) for one projection axis. Furthermore, the authors do not clarify how the problem of pose normalization is handled (except for the rotation invariance of the 2D panoramic image projection axis). Table 2 summarizes the differences between the proposed method and [SBZB15].

	Methods using ML	**Methods not using ML**
2D	MVCNN [SMKLM15] DeepPano [SBZB15]	LFD [CTSO03] PANORAMA [PPTP10]
3D	3D ShapeNets [WSK*15] Geometry Image [SBR16]	SPH [KFR03]

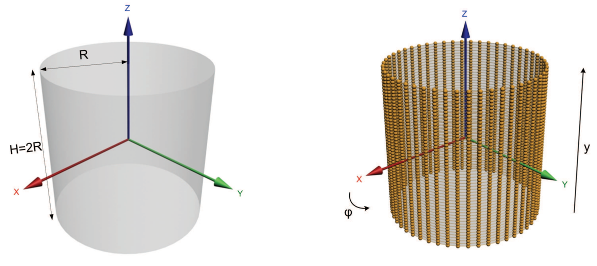
Table 1: *Method categorization based on the usage of Machine Learning (ML) and dimensionality of the descriptor (2D or 3D).*

	PANORAMA-NN	DeepPano
2D Image Representation	**angle, distance**	distance
Projection Axes	$\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$	one axis
Pose Normalization	$\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$	one axis

Table 2: *Differences between the proposed method (PANORAMA-NN) and the method in [SBZB15] (DeepPano).*

3. Methodology

In this section, the proposed 3D model classification and retrieval method, based on the PANORAMA representation views and convolutional neural networks is presented. For completion, in the background subsection, the key methodologies used and previously developed by our team (i.e. the PANORAMA representation and the SYMPAN pose normalization method) are briefly discussed.



(a)

(b)

Figure 1: *(a) A projection cylinder for the acquisition of a 3D model's panoramic view and (b) the corresponding discretization of its lateral surface to the set of points* $s(\phi_u, y_v)$

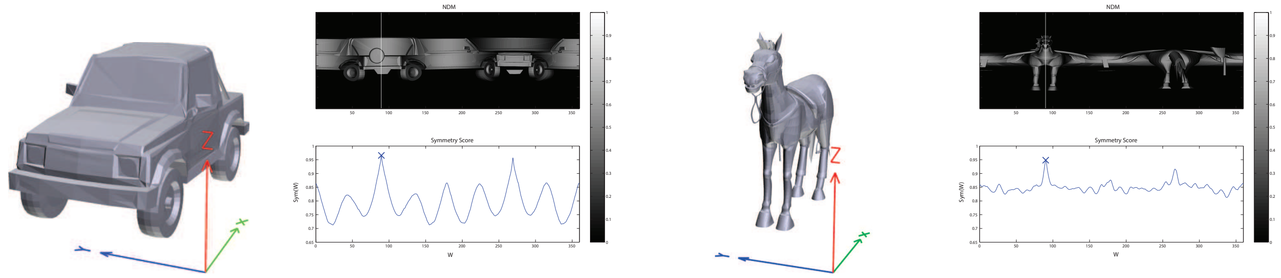


Figure 2: *Two sample 3D models along with their panoramic views and symmetry plane estimations as these are employed in the SYMPAN pose normalization method.*

3.1. Background

3.1.1. PANORAMA Representation Extraction

Let a projection cylinder be defined as a cylinder whose axis is aligned with one of the principal axes of space (e.g. the z axis), as described by Papadakis et al. [PPTP10]. To obtain a panoramic view for a 3D model, it is projected onto the lateral surface of a cylinder of radius R and height $H = 2R$, centered at the origin, with its axis parallel to one of the principal axes of space, see Fig. 1a. The value of R is set to $2 * d_{max}$ where d_{max} is the maximum distance of the model's surface from its centroid.

In the following, the lateral surface of the cylinder is parameterized using a set of points $s(\phi, y)$ where $\phi \in [0, 2\pi]$ is the angle in the XY plane, $y \in [0, H]$ and the ϕ and y coordinates are sampled at rates $2B$ and B, respectively (B is set to be equal to 180). The ϕ dimension is sampled at twice the rate of the y dimension to account for the difference in length between the perimeter of the cylinder's lateral surface and its height. Although the perimeter of the cylinder's lateral surface is $2\pi \simeq 3$ times its height, the sampling rates are set at $2B$ and B, respectively, as these values were experimentally found to give good results. Thus, the set of points $s(\phi_u, y_v)$ are obtained, where $\phi_u = u * 2\pi/(2B)$, $y_v = v * H/B$, $u \in [0, 2B-1]$ and $v \in [0, B-1]$. These points are shown in Fig. 1b.

Next, the value at each point $s(\phi_u, y_v)$ of the panoramic view shall be determined. The computation is carried out iteratively for $v = 0, 1, ..., B-1$, each time considering the set of coplanar $s(\phi_u, y_v)$ points, i.e. a cross section v of the cylinder at height y_v and for each cross section rays are cast from its center c_v in the ϕ_u directions.

The cylindrical projections are used to capture two different characteristics of a 3D model's surface; (i) the position of the model's surface in 3D space, (referred to as **Spatial Distribution Map** or **SDM**), and (ii) the orientation of the model's surface, (referred to as **Normals' Deviation Map** or **NDM**). To capture these characteristics two kinds of cylindrical projections $s_1(\phi_u, y_v)$ and $s_2(\phi_u, y_v)$ are used.

To capture the position of the model's surface, for each cross section at height y_v, the distances from c_v of the intersections of the model's surface are computed with the rays at each direction ϕ_u. Let $pos(\phi_u, y_v)$ denote the distance of the furthest from c_v point of intersection between the ray emanating from c_v in the ϕ_u direction and the model's surface; then $s_1(\phi_u, y_v) = pos(\phi_u, y_v)$. The value of a point $s(\phi_u, y_v)$ lies in the interval $[0, R]$, where R denotes the radius of the cylinder.

To capture the orientation of the model's surface, for each cross section at height y_v, the intersections of the model's surface with the rays at each direction ϕ_u are computed and the angle between a ray and the normal vector of the triangle that is intersected is measured. To determine the value of a point $s_2(\phi_u, y_v)$ the cosine of the angle between the ray and the normal vector of the furthest from c_v intersected triangle of the model's surface is used. If $ang(\phi_u, y_v)$ denotes the aforementioned angle, then the values of the $s(\phi_u, y_v)$ points are given by $s_2(\phi_u, y_v) = |\cos(ang(\phi_u, y_v))|^n$.

The nth power of $|\cos(ang(\phi_u, y_v))|$ is taken, where $n \geq 2$, since this setting enhances the contrast of the produced cylindrical projection. It has been experimentally found that setting n to a value in the range $[4, 6]$ gives the best results. Also, taking the absolute value of the cosine is necessary to deal with inconsistently oriented triangles along the model's surface.

A cylindrical projection can be viewed as a 2D gray-scale image where pixels correspond to the $s_k(\phi_u, y_v)$ intersection points in a manner reminiscent of cylindrical texture mapping and their values are mapped to the $[0, 1]$ space.

3.1.2. SYMPAN: PANORAMA-based Pose Normalization

Pose normalization is performed using the SYMPAN method [STP14] which uses the **SDM** and the **NDM** extracted in PANORAMA. Pose normalization is significant in order to maintain integrity between the corresponding panoramic view representations of the 3D models. The choice of SYMPAN as the pose normalization method is due to its high integration with the PANORAMA representation and the fact that the majority of real-life 3D models (e.g. CAD objects, human and animal objects, etc) actually exhibit reflective symmetry, to a certain a degree. Methods that exploit symmetries have exhibited high performance, both in terms of pose normalization and retrieval accuracy (see [STP11, KCD*02, CVB09]).

Initially, a 3D model, having arbitrary pose, is normalized in terms of translation and scaling using standard techniques. More specifically, translation normalization is achieved though the definition of the 3D model's centroid and the displacement of this centroid to the coordinate system origin. Consecutively, the 3D model is scaled so that it becomes exactly inscribed inside the unit sphere.

The estimation of a plane of symmetry of a 3D model corresponds to the detection of a line of reflective symmetry in its panoramic view. Since translation normalization has been performed, the plane of symmetry of the 3D object will pass through the origin of the coordinate system. The aim is to rotate the symmetry plane so that it includes the z axis; then the plane of symmetry will be detectable in the panoramic image.

Once a plane of symmetry is defined, the first principal axis of the model is set to be the normal to that plane of symmetry (see Fig. 2). The remaining two principal axes have yet to be estimated. The 3D model can thus be rotated so that its symmetry plane coincides with one of the principal planes of space (e.g. the XY plane).

To complete the rotation normalization task, the 3D model is projected onto the surface of a projection cylinder whose axis is one of the principal axes of space, perpendicular to the symmetry plane's normal. The 3D model is iteratively rotated around the normal axis to the symmetry plane and the corresponding **SDM** images are calculated. For each **SDM** image, the variance of its pixel values is computed and the rotation that minimizes this variance, is defined as the rotation which aligns the principal axis of the 3D model to the axis of the projection cylinder.

3.2. Minimized Augmented Panoramic Views Construction

In order to efficiently train an artificial neural network using the PANORAMA representation, an augmented schema is employed based on the panoramic views produced with respect to the three principal axes.

More specifically, for each principal axis, both SDM and NDM cylindrical view representations are computed, resulting in a total of six cylindrical view representations for each 3D model. Half of each view is appended at the end in order to have no 'wrap-around' gaps in the representation.

The six representations are then stacked together in the following order: NDM(X), NDM(Y), NDM(Z), SDM(X), SDM(Y), SDM(Z) (see Fig. 3). The augmented image representations define the input of the convolutional neural network. The total size of each 3D model's augmented view is 1,5 * 360 = 540 pixels width by 180 * 6 = 1080 pixels height.

Once the augmented view has been montaged, its size is reduced to 10% of its original size, namely 54 × 108 pixels. This is performed using bicubic interpolation. Although a significant amount of detail of the original representation is lost due to the resolution reduction, it has been experimentally found that the minimized representation is sufficient to achieve high performance on the classification task while maintaining feasible neural network training times (see Section 4).

3.3. Convolutional Neural Network Architecture

The convolutional neural network architecture selected in the proposed implementation is based on a standard scheme, namely an input layer followed by a set of convolutional layers and finally by the fully connected layer(s) of the output. This architecture was originally proposed by Krizhevsky et al. [KSH12] and has demonstrated state-of-the-art performance in image classification.

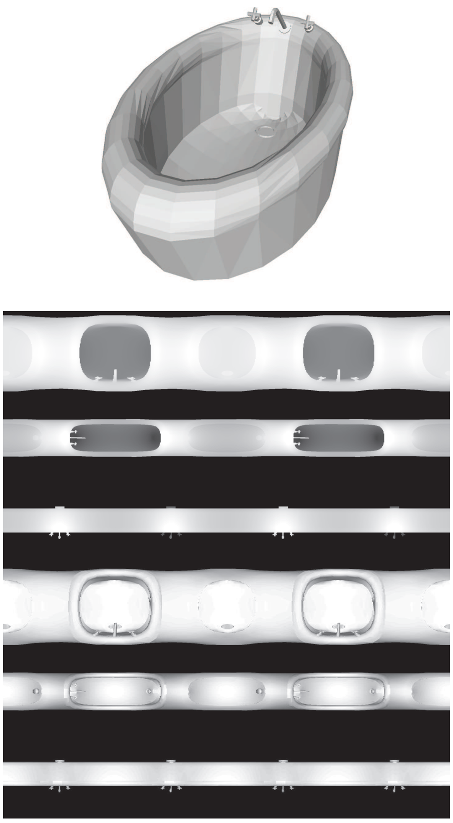


Figure 3: *Sample augmented panoramic view of a 3D model. The order of stacked representations are the following: NDM(X), NDM(Y), NDM(Z), SDM(X), SDM(Y), SDM(Z).*

The above network has been simplified for efficiency reasons. More specifically, three convolutional layers were used and the corresponding feature maps are 64, 256 and 1024 respectively. The kernel size is respectively set to 5, 5, 3 and the padding is set to 2 for all the layers. After each convolutional layer both a ReLU and a 2 × 2 max-pooling layer are inserted.

The output of the architecture consists of one fully connected layer followed by a dropout layer [SHK*14] used to reduce overfitting. Finally, a softmax layer outputs class probabilities for a given input 3D model. The class with the highest probability is considered as the predicted class for the 3D model.

The network is trained using the stochastic gradient descent method (SGDM) with momentum set to 0.9.

The aforementioned convolutional neural network architecture, along with the complete pipeline of the proposed method, is shown in Fig. 4.

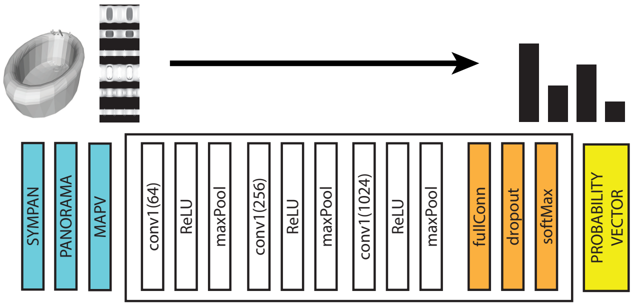


Figure 4: *Illustration of the proposed method pipeline, including the convolutional neural network architecture.*

4. Experiments

4.1. Dataset

The dataset used for evaluating the proposed method is the Princeton ModelNet large scale 3D CAD model dataset [WSK*15]. ModelNet comprises of 127,915 CAD models split into 662 object categories. The ModelNet dataset offers two subsets, ModelNet-10 and ModelNet-40, with both training and testing.

ModelNet-10 comprises 4,899 CAD models split into 10 categories. The models have been manually cleaned and pose normalized in terms of translation and rotation. The train and test subsets of ModelNet-10 consist of 3,991 and 908 models, respectively.

ModelNet-40 comprises 12,331 CAD models split into 40 categories. The models have been manually cleaned but they are not pose normalized. The train and test subsets of ModelNet-40 consist of 9,843 and 2,468 models, respectively.

4.2. 3D Model Classification

The proposed method is evaluated on the task of classification of the test subset 3D models for both ModelNet-10 and ModelNet-40 datasets. The performance is measured via the average binary categorical accuracy (a value of 1 corresponds to the case where the category of the test 3D model is correctly predicted and 0 otherwise).

The proposed method is compared against the LightField descriptor [CTSO03] (LFD, 4,700 dimensions), the Spherical Harmonics descriptor [KFR03] (SPH, 544 dimensions), the 3D ShapeNets (ML) [WSK*15], the DeepPano descriptor (ML) [SBZB15], the Multi-view Convolutional Neural Networks (ML) [SMKLM15] (MVCNN) and the Geometry Image (ML) descriptor [SBR16]. Descriptors indicated with (ML), they do involve machine learning.

The proposed method outperforms all aforementioned methods, in both ModelNet-10 and ModelNet-40, as shown in Table 3.

4.3. 3D Model Retrieval

Another evaluation of the proposed method was performed on the task of 3D model retrieval. The same datasets were used and the accuracy was measured via the *Mean Average Precision* (mAP) metric.

Method	ModelNet-10	ModelNet-40
PANORAMA-NN (ML)	**0.9112**	**0.9070**
LFD	0.7987	0.7547
SPH	0.7979	0.6823
3D ShapeNets (ML)	0.8354	0.7732
DeepPano (ML)	0.8866	0.8254
MVCNN (ML)	N/A	0.9010
Geometry Image (ML)	0.8840	0.8390

Table 3: *Classification accuracies on the ModelNet-10 and ModelNet-40 datasets.*

To perform the retrieval task, the activations of the fully connected layer of the convolutional neural network for each input 3D model was used as the corresponding descriptor for that object. A 3D model descriptor is compared against the rest of the 3D model descriptors using the L_1 distance metric.

Table 4 shows the results of the retrieval experiment, comparing the methods of the previous paragraph plus the complete PANORAMA descriptor. Fig. 6 illustrates the Precision-Recall plots for the proposed and the compared methods on the two ModelNet subsets. The proposed method outperforms the competition in both datasets.

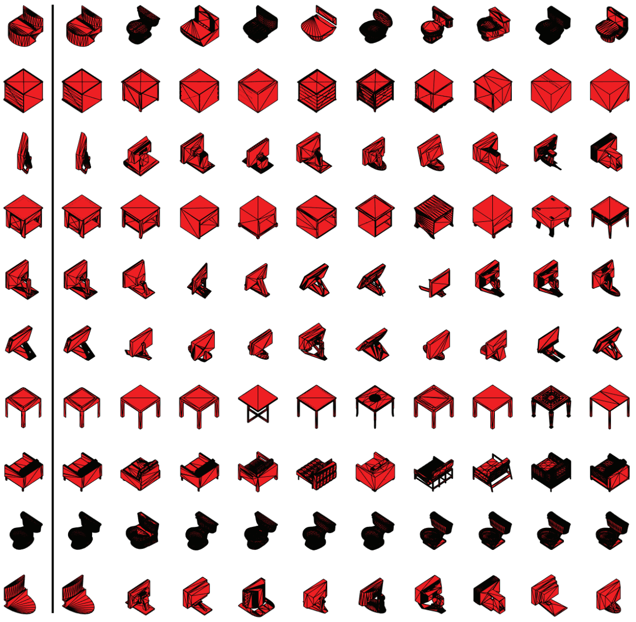


Figure 5: *Retrieval examples for the proposed method on the ModelNet-10 dataset. First column illustrates the queries while the remaining columns illustrate the corresponding retrieved models in rank order. Note that the first retrieved model is the query model in all cases.*

Fig. 5 illustrates qualitative retrieval results for 10 sample query models. The first column indicates the query and the remaining columns (left-to-right in retrieval order) indicate the top 10 retrieved 3D models from the ModelNet-10 dataset. Note that the first

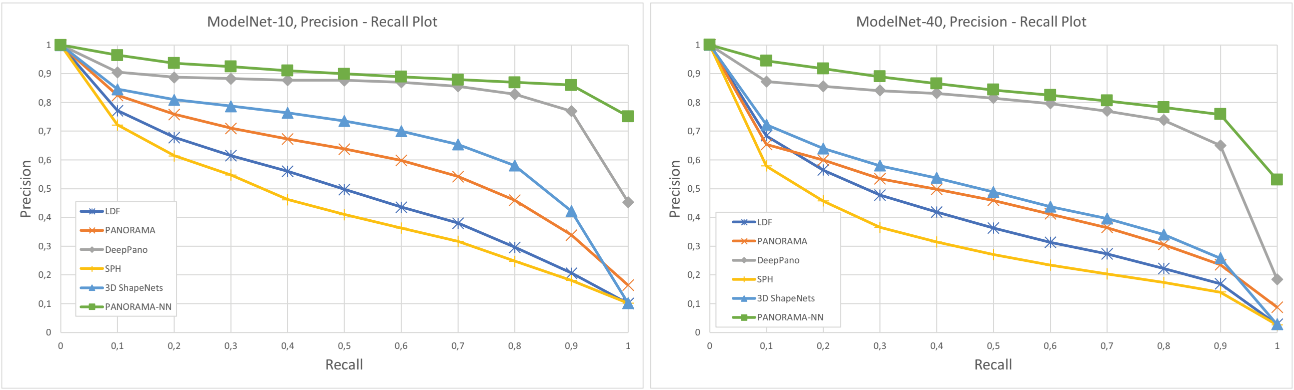


Figure 6: *Precision-Recall plots for ModelNet-10 (left) and ModelNet-40 (right) datasets. Illustrated are the proposed method (PANORAMA-NN) compared to five other retrieval methods.*

retrieved 3D model is the query model itself while all the retrieved 3D models belong to the same class as the query.

Method	ModelNet-10	ModelNet-40
PANORAMA-NN (ML)	**0.8739**	**0.8345**
PANORAMA	0.6032	0.4613
LFD	0.4982	0.4091
SPH	0.4405	0.3326
3D ShapeNets (ML)	0.6826	0.4923
DeepPano (ML)	0.8418	0.7681
MVCNN (ML)	N/A	0.7950
Geometry Image (ML)	0.7490	0.5130

Table 4: *Retrieval accuracies measured in mAP on the ModelNet-10 and ModelNet-40 datasets.*

Fig. 7 illustrates qualitative retrieval failure cases for 4 sample query models. The first column indicates the query and the remaining columns (left-to-right in retrieval order) indicate the top 4 retrieved 3D models from the ModelNet-10 dataset. As illustrated in the figure, although the retrieved models do not belong to the same class as the query model, their structure is similar. For example, in the fourth row the query is from the *desk* class and the results from the *table* class. These two classes contain models whose structure is very similar. It is therefore considered safe to conclude that these failure cases are some of the hardest retrieval examples.

4.4. Implementation

The proposed method was tested on an Intel (R) Core (TM) i7 @ 3.60GHz CPU system, with 32GB of RAM and a discrete NVIDIA (R) TITAN X with 12GB RAM GPU. The system was running Matlab R2016b. The PANORAMA representation extraction method was developed in a hybrid Matlab/C++/OpenGL architecture while the pose normalization procedure was developed in Matlab. The artificial neural network was implemented on the Matlab Deep Neural Network toolbox, accelerated using the CUDA instruction set on the GPU.

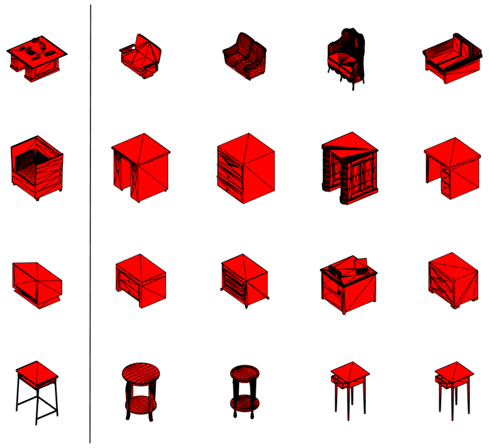


Figure 7: *Sample failure cases, in terms of retrieval accuracy, for the proposed method. First column illustrates the queries while the remaining columns illustrate the corresponding retrieved models in rank order.*

The average PANORAMA representation view extraction for a 10,000 face 3D model is 350 ms. The average pose normalization time for the same typical model is 1,850ms. The artificial neural network training procedure requires approximately 15 minutes to converge. When image representations of higher resolution were used (reduction to 20% of the original size, namely 108 × 216 pixels) the performance gain was considered insignificant (approximately +0.005%) while the training procedure doubled in time (approximately 30 minutes).

5. Conclusion

A novel convolutional neural network based method for the classification and retrieval of 3D models has been presented. The proposed method is presented as a complete pipeline, defining the input, as well as the parameters and structure of the CNN employed.

Initially, the 3D models of the dataset are pose normalized using the SYMPAN algorithm. This is a crucial step since not all dataset 3D models are guaranteed to be pose normalized (e.g. as in the case of ModelNet-40). Next, for each 3D model, the full set of PANORAMA representations is extracted and concatenated into an augmented panoramic view structure; note that PANORAMA assumes that the models are pose normalized. This structure is minimized in terms of dimensionality (i.e. resized to 10% if its original size) and used as input to a convolutional neural network; the latter performs the classification task or produces the shape descriptor for retrieval purposes.

PANORAMA, in addition to being a good shape descriptor, was able to bridge the gap between the initial 3D model representation and the 2D input required by convolutional neural networks. The SYMPAN pose normalization method works with reflective symmetries and this could partially explain the high accuracy achieved on the ModelNet datasets, since the latter consist of CAD models that contain several such symmetries. The ModelNet datasets used for evaluation were specifically designed for deep neural network classification applications.

The proposed method was compared against six published works on the tasks of 3D model classification and retrieval and was able to outperform them by a significant margin.

References

[CTSO03] Chen D.-Y., Tian X.-P., Shen Y.-T., Ouhyoung M.: On visual similarity based 3D model retrieval. In *Computer graphics forum* (2003), vol. 22, Wiley Online Library, pp. 223–232. 2, 5

[CVB09] Chaouch M., Verroust-Blondet A.: Alignment of 3D models. *Graphical Models 71*, 2 (2009), 63–76. 3

[HL06] Huang F. J., LeCun Y.: Large-scale learning with svm and convolutional for generic object categorization. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on* (2006), vol. 1, IEEE, pp. 284–291. 1

[KCD*02] Kazhdan M., Chazelle B., Dobkin D., Finkelstein A., Funkhouser T.: A reflective symmetry descriptor. In *European Conference on Computer Vision* (2002), Springer, pp. 642–656. 3

[KFR03] Kazhdan M., Funkhouser T., Rusinkiewicz S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Symposium on geometry processing* (2003), vol. 6, pp. 156–164. 2, 5

[KSH12] Krizhevsky A., Sutskever I., Hinton G. E.: Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105. 4

[LBBH98] LeCun Y., Bottou L., Bengio Y., Haffner P.: Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*, 11 (1998), 2278–2324. 1

[PPTP10] Papadakis P., Pratikakis I., Theoharis T., Perantonis S.: PANORAMA: a 3D shape descriptor based on panoramic views for unsupervised 3D object retrieval. *International Journal of Computer Vision 89*, 2-3 (2010), 177–192. 1, 2, 3

[SBR16] Sinha A., Bai J., Ramani K.: Deep learning 3D shape surfaces using geometry images. In *European Conference on Computer Vision* (2016), Springer, pp. 223–240. 2, 5

[SBZB15] Shi B., Bai S., Zhou Z., Bai X.: Deeppano: Deep panoramic representation for 3D shape recognition. *IEEE Signal Processing Letters 22*, 12 (2015), 2339–2343. 2, 5

[SHK*14] Srivastava N., Hinton G. E., Krizhevsky A., Sutskever I., Salakhutdinov R.: Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*, 1 (2014), 1929–1958. 4

[SMKLM15] Su H., Maji S., Kalogerakis E., Learned-Miller E.: Multi-view convolutional neural networks for 3D shape recognition. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 945–953. 2, 5

[SPK*16] Sfikas K., Pratikakis I., Koutsoudis A., Savelonas M., Theoharis T.: Partial matching of 3D cultural heritage objects using panoramic views. *Multimedia Tools and Applications 75*, 7 (2016), 3693–3707. 1

[STP11] Sfikas K., Theoharis T., Pratikakis I.: ROSy+: 3D Object Pose Normalization Based on PCA andÂăReflective Object Symmetry with Application in 3D Object Retrieval. *International Journal of Computer Vision 91*, 3 (2011), 262–279. 3

[STP13] Sfikas K., Theoharis T., Pratikakis I.: 3D object retrieval via range image queries in a bag-of-visual-words context. *The Visual Computer 29*, 12 (2013), 1351–1361. 1

[STP14] Sfikas K., Theoharis T., Pratikakis I.: Pose normalization of 3D models via reflective symmetry on panoramic views. *The Visual Computer 30*, 11 (2014), 1261–1274. 1, 2, 3

[WSK*15] Wu Z., Song S., Khosla A., Yu F., Zhang L., Tang X., Xiao J.: 3D ShapeNets: a deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1912–1920. 1, 2, 5