

# Topological Descriptors for 3D Surface Analysis

Matthias Zeppelzauer<sup>1</sup>(✉), Bartosz Zieliński<sup>2</sup>(✉), Mateusz Juda<sup>2</sup>,  
and Markus Seidl<sup>1</sup>

<sup>1</sup> Media Computing Group, Institute of Creative Media Technologies,  
St. Poelten University of Applied Sciences, Matthias-Corvinus Strasse 15,  
3100 St. Poelten, Austria

{m.zeppelzauer,markus.seidl}@fhstp.ac.at

<sup>2</sup> The Institute of Computer Science and Computer Mathematics,  
Faculty of Mathematics and Computer Science, Jagiellonian University,  
ul. Łojasiewicza 6, 30-348 Kraków, Poland

{bartosz.zielinski,mateusz.juda}@uj.edu.pl

**Abstract.** We investigate topological descriptors for 3D surface analysis, i.e. the classification of surfaces according to their geometric fine structure. On a dataset of high-resolution 3D surface reconstructions we compute persistence diagrams for a 2D cubical filtration. In the next step we investigate different topological descriptors and measure their ability to discriminate structurally different 3D surface patches. We evaluate their sensitivity to different parameters and compare the performance of the resulting topological descriptors to alternative (non-topological) descriptors. We present a comprehensive evaluation that shows that topological descriptors are (i) robust, (ii) yield state-of-the-art performance for the task of 3D surface analysis and (iii) improve classification performance when combined with non-topological descriptors.

**Keywords:** 3D surface classification · Surface topology analysis · Surface representation · Persistence diagram · Persistence images

## 1 Introduction

With the increasing availability of high-resolution 3D scans, topological surface description is becoming increasingly important. In recent years, methods for sparse and dense 3D scene reconstruction have progressed strongly due to availability of inexpensive, off-the-shelf hardware (e.g. Microsoft Kinect) and the development of robust reconstruction algorithms (e.g. structure from motion techniques, SfM) [5,28]. Since 3D scanning has become an affordable process the amount of available 3D data has increased significantly. At the same time, the reconstruction accuracy has increased strongly, which enables 3D reconstructions with sub-millimeter resolution [27]. The high resolution enables the accurate description of a 3D surface's geometric micro-structure, which opens up new opportunities for search and retrieval in 3D scenes, such as the recognition of objects by their specific surface properties as well as the distinction of different types of materials for improved scene understanding.

In this paper, we investigate the problem of describing and classifying 3D surfaces according to their geometric micro-structure. Two different types of approaches exist for this problem. Firstly, the dense processing of the surface in 3D space and secondly, the processing of the surface geometry in image-space based on depth maps derived from the surface.

For the representation of surface geometry in 3D, descriptors are required that capture the local geometry around a given point or mesh vertex. Different types of local 3D descriptors have been developed recently that are suitable for the description of the local geometry around a 3D point, such as spin images [12], 3D shape context [4], and persistent point feature histograms [24].

The dense extraction of surface geometry by local 3D descriptors, however, becomes a computationally demanding task when several millions of points need to be processed. A computationally more efficient approach is the analysis of 3D surfaces in image space. In such approaches a 3D surface is first mapped to a depth map which represents a height field of the surface. This processing step maps the 3D surface analysis problem to a 2D texture analysis task which can be approached by analyzing the surface by texture descriptors, such as HOG, GLCM, and Wavelet-based features [19, 29, 30].

The presented approach falls into the category of image-space approaches. We first map the surface to image-space by a depth projection. Next, we divide the resulting depth map into patches and describe them with traditional non-topological as well as with topological surface descriptors. For the classification of surface patches we use random undersampling boosting (RUSBoost) [25] due to its high accuracy for imbalanced class distributions [16].

## 2 Topological Approach

By mathematical standards topology, with its 120 years of history, is a relatively young discipline. It grew out of H. Poincares seminal work on the stability of the solar system as a qualitative tool to study the dynamics of differential equations without explicit formulas for solutions [20–22]. Due to the lack of useful analytic methods, topology soon became a purely theoretical discipline. However, in the last few years we observe a rapid development of topological data analysis tools, which open new applications for topology.

Topological spaces appearing in data analysis are typically constructed from small pieces or cells. A natural tool in the study of multidimensional images with topological methods are hypercubes (points, edges, squares, cubes etc.), e.g. a pixel in a 2 dimensional image is equivalent to a square, a voxel in a 3 dimensional volume is equivalent to a cube. Hypercubes are building blocks for structures called cubical complexes. Such representations give topology a combinatorial flavour and make it a natural tool in the study of multi-dimensional data sets.

Intuitively, the rank of the  $n$ th homology group, the so called  $n$ th Betti number denoted  $\beta_n$ , counts the number of  $n$ -dimensional holes in the topological space. In particular,  $\beta_0$  counts the number of connected components. As an example consider the image of the digit 8. In this image there is one connected

component and two holes, hence  $\beta_0 = 1$  and  $\beta_1 = 2$ . For a hollow sphere we have  $\beta_0 = 1, \beta_1 = 0, \beta_2 = 1$ . For a tube in a tire we have  $\beta_0 = 1, \beta_1 = 2, \beta_2 = 1$ .

Betti numbers do not differentiate between small and large holes. In consequence, the holes resulting from the noise in the data cannot be distinguished from the holes indicative for the nature of the data. For instance, in a noisy image of the digit 8 one can get easily  $\beta_0 > 1$ . A remedy for this drawback is persistent homology, a tool invented at the beginning of the 20th century [7]. Persistent homology studies how the Betti numbers change when the topological space is gradually built by adding cubes in some prescribed order.

If  $X$  is a cubical complex, one can add cubes step by step. Typically, the construction goes through different scales, starting from the smallest pieces. However, in general an arbitrary function  $f : X \rightarrow \mathbb{R}$ , called the Morse function or measurement function, may be used to control the order in which the complex is built, starting from low values of  $f$  and increasing subsequently. This way we obtain a sequence of topological spaces, called a filtration,

$$\emptyset = X_{r_0} \subset X_{r_1} \subset X_{r_2} \subset \dots \subset X_{r_n} = X,$$

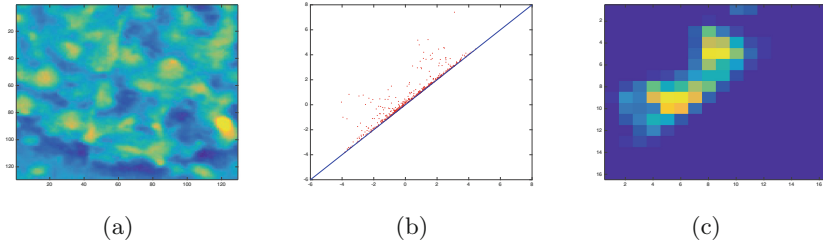
where  $X_r := f^{-1}((-\infty, r])$  and  $r_i$  is a growing sequence of values of  $f$  at which the complex changes. As the space is gradually constructed, holes are born, persist for some time and eventually may die. The length of the associated birth-death intervals (persistence intervals) indicates if the holes are relevant or merely noise. The lifetime of holes is usually visualized by the so called persistence diagram (PD). Persistence diagrams constitute the main tool of topological data analysis. They visualize geometrical properties of a multidimensional object  $X$  in a simple two dimensional diagram.

Figure 1(a) shows a 3D surface as a 2D depth map, where colors corresponds to depth (blue refers to low depth, yellow to high depth). In this case pixels are represented as 2-dimensional cells of a cubical complex. For the complex we can obtain a filtration  $X_r$  using a measuring function which has a value for a 2-dimensional cube equal to height (pixel color). For a lower dimensional cell (a vertex or an edge) we can set the function value as a maximum from the higher-dimensional neighborhoods of the cell. Figure 1(b) shows the persistence diagram for  $X_r$ .

There is still no specific answer on how and when the tools of computational topology and machine learning should be used together. A first attempt is to provide a descriptor of a topological space filtration based on elementary statistics of persistence intervals (or equivalently on persistence diagrams). Let

$$I := \{[b_1, e_1], [b_2, e_2], \dots, [b_n, e_n]\}$$

be a set of persistence intervals. Let  $D := \{d_i := (e_i - b_i)\}_{i=1}^n$  be a set of the interval lengths. We build an aggregated descriptor of  $D$ , denoted by PD\_AGG, using following measures: number of elements, minimum, maximum, mean, standard deviation, variance, 1-quartile, median, 3-quartile, and norms  $\sum \sqrt{d_i}, \sum d_i,$  and  $\sum (d_i)^2$ .



**Fig. 1.** Example patch: (a) the original 3D surface as a 2D depth map; (b) the corresponding persistent diagram; (c) and the persistent image with  $\sigma = 0.001$  and resolution  $16 \times 16$ . (Color figure online)

Except the PD-AGG descriptor described above, which can be used with standard classification methods, there are also attempts to use PD directly with appropriately modified classifiers. Reininghaus et al. [23] proposed a multiscale kernel for PDs, which can be used with a support vector machine (SVM). While this kernel is well-defined in theory, in practice it becomes highly inefficient for a large number of training vectors (as the entire kernel matrix must be computed explicitly). As an alternative, Chepushtanova et al. [1] introduced a novel representation of a PD, called a persistence image (PI), which is faster and can be used with a broader range of machine learning (ML) techniques.

A PI is derived from mapping a PD to an integrable function  $G_p : \mathbb{R}^2 \rightarrow \mathbb{R}$ , which is a sum of Gaussian functions centered at each point of the PD. Taking a discretization of a subdomain of  $G_p$  defines a grid. An image can be created by computing the integral of  $G_p$  on each grid box, thus defining a matrix of pixel values. Formally, the value of each pixel  $p$  within a PI is defined by the following equation:

$$PI(p) = \iint_p \sum_{[b_i, e_i] \in I} g(b_i, e_i) \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2} \left( \frac{(x-b_i)^2}{\sigma_x^2} + \frac{(y-e_i)^2}{\sigma_y^2} \right)} dy dx,$$

where  $g(b_i, e_i)$  is a weighting function, which depends on the distance from the diagonal (points close to the diagonal are usually considered as noise, therefore they should have low weights),  $\sigma_x$  and  $\sigma_y$  are the standard deviations of the Gaussians in  $x$  and  $y$  direction. The resulting image (see Fig. 1c) is vectorized to achieve a standardized vectorial representation which is compatible to a broad range of ML techniques.

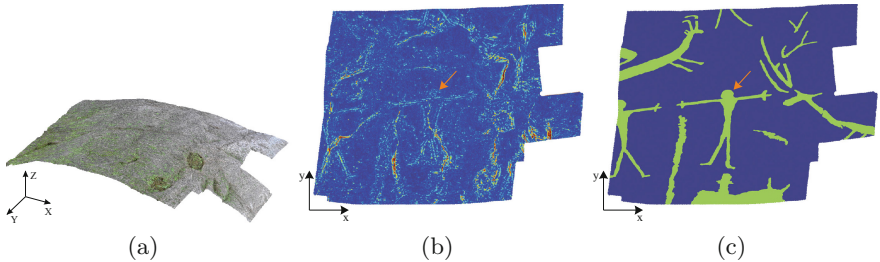
The advantage of PIs compared to PDs descriptor is a high classification accuracy and stability [1]. However, they require numerous parameters like the PI resolution, the weighting function  $g$ , as well as  $\sigma_x$  and  $\sigma_y$ .

### 3 Experimental Setup

In our experiments we investigate the robustness and expressiveness of the topological descriptors presented in Sect. 2 for 3D surface analysis and compare and

combine them with traditional non-topological descriptors. For our experiments, we employ a dataset of high-resolution 3D reconstructions from the archaeological domain with a resolution below 0.1 mm [30]. The dimension of the scanned surfaces ranges from approx.  $20 \times 30$  cm to  $30 \times 50$  cm. The reconstructions represent natural rock surfaces that exhibit human-made engravings (so-called rock-art). The engravings represent symbols and figures (e.g. animals and humans) engraved by humans in ancient times. See Fig. 2 for an example surface. The engraved regions in the surface exhibit a different surface geometry than the surrounding natural rock surface. In our experiments we aim at automatically separating the engraved areas from the natural rock surface. The corresponding ground truth is depicted in Fig. 2c.

The employed dataset contains 4 surface reconstructions with a total number of 12.3 millions of points. For each surface a precise ground truth has been generated by domain experts that labels all engravings on the surface. The dataset contains two classes of surface topographies: class 1 represents engraved areas and class 2 represents the natural rock surface. Class priors are imbalanced. Class 1 represents 16.6% of the data and is thus underrepresented.



**Fig. 2.** Example data: (a) the 3D point cloud of the surface; (b) the depth projection of the surface with compensated global curvature; (c) ground truth labeling that specifies areas with different topography, such as the human-shaped figure in the center whose head is marked with an arrow.

For each scan we perform depth projection and preprocessing as described in [30]. The result is a depth map that reflects the geometric micro-structure of the surface, see Fig. 2b. This representation is the input to feature extraction.

From the depth map we extract a number of non-topological image descriptors in a block-based manner that serve as a baseline in our experiments. The block size is  $128 \times 128$  pixels (i.e.  $10.8 \times 10.8$  mm) and the step size between two blocks is 16 pixels (1.35 mm). The baseline features include: MPEG-7 Edge Histogram (EH) [11], Dense SIFT (DSIFT) [17], Local Binary Patterns (LBP) [18], Histogram of Oriented Gradients (HOG) [6], Gray-Level Co-occurrence Matrix (GLCM) [10], Global Histogram Shape (GHS), Spatial Depth Distribution (SDD), as well as manually modified enhanced versions of GHS and SDD (short EGHS and ESDD) that apply additional enhancements to the depth map described in [30].

Additionally to the baseline descriptors, we extract persistent homology descriptors in the same block-wise manner. For each patch, we compute a persistence diagram and derive the 12-dimensional aggregated descriptor (PD\_AGG) as described in Sect. 2. Additionally, we extract persistence images (PIs) for different resolutions (8, 16, 32, 64) and standard deviations (0.00025, 0.0005, 0.001, 0.002) with and without weighting (see Sect. 2).

Alternatively, we first extract Completed LBP (CLBP) features [9] from the depth map as proposed in [15, 23] and then extract PD\_AGG and PIs from the CLBP\_S and CLBP\_M maps.

For the discrimination of different surface topographies we employ supervised machine learning. All employed descriptors are represented by numerical vectors of fixed dimension and are thus suitable for statistical classification. As mentioned above, the class priors in our dataset are imbalanced. Skewed datasets pose problems to most classification techniques and often yield suboptimal models as one class dominates the other classes. A classifier especially designed for imbalanced datasets is Random Undersampling Boosting (RUSBoost) [25]. RusBoost builds upon AdaBoost [8] which is an ensemble method that combines the weighted decisions of weak classifiers to obtain a final decision for a given input sample. RUSBoost extends this concept by a data sampling strategy that enforces similar class priors. During each training iteration the majority class in the training set is undersampled in a random fashion to balance the resulting class priors. In this manner, the weak classifiers can be learned from balanced datasets without being biased from the skewed class distribution.

For training the RUSBoost classifier we split the entire dataset into independent training and evaluation sets. The training set contains image patches from scans 1 and 2 from the dataset. Scans 3 and 4 make up the evaluation set. From the training set we randomly select 50% of the blocks from class 1 (2962 blocks) and 30% from class 2 (7592 blocks). From this subset of 9654 samples we train the RUSBoost classifier. For training we apply 5-fold cross-validation to estimate suitable classifier parameters (primarily the number of weak classifiers of the ensemble). The best parameters are used to train the classifier on the entire training set. The trained classifier is finally applied to the independent evaluation set of 27192 patches.

As a performance measure we employ the Dice Similarity Coefficient (DSC). DSC measures the mutual overlap between an automatic labeling  $X$  of an image and a manual (ground truth) labeling  $Y$ :

$$\text{DSC}(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|}.$$

DSC is between 0 and 1 where 1 means a perfect segmentation.

Each classification experiment is repeated 10 times with 10 different randomly selected subsets from the training set to reduce the dependency from the training data. From the 10 resulting DSC values we provide median and standard deviation as the final performance measures.

Aside from quantitative evaluations we investigate the following questions:

- Can persistent homology descriptors outperform descriptors like HOG, SIFT, and GLCM for surface classification?
- How does aggregation of the PD (PD\_AGG) influence performance compared to non-aggregated representations like PI?
- Is CLBP a suitable input representation for persistent homology descriptors?
- How sensitive is PI to its parameters (resolution, sigma, weighting)?
- Do persistent homology descriptors add beneficial or even necessary information to the baseline descriptors in our classification task?

The experiment was implemented in Matlab. Most of the descriptors were extracted with VLFeat library [26], except PD\_AGG and PI. We compute persistence intervals of the images using CAPD::RedHom library [13,14] with the PHAT [2,3] algorithm for persistence homology.

## 4 Results

We start our evaluation with the aggregated descriptor PD\_AGG. The descriptor applied to our surfaces yields a DSC of  $0.6528 \pm 0.0118$  and represents a first baseline for further comparisons. Next, we apply PI with different resolutions, sigmas with and without weighting. Results are summarized in Table 1. All results for PI outperform that of PD\_AGG. We assume the reason is that PD\_AGG neglects the information about the points’ localization, which is preserved in PI. The best result for PI is a DSC of  $0.7335 \pm 0.0024$  without weighting. The difference between the best weighting and no weighting result is statistically significant<sup>1</sup> with  $p - value = 0.006$ . This result is surprising as it is contrary to the results of [1] where artificial datasets were used for evaluation. Results in Table 1 further show that PI has low sensitivity to different resolutions and sigmas.

Next, we evaluate the performance of PD\_AGG and PI with CLBP as input representation, see Table 2. The best result for PD\_AGG ( $0.6874 \pm 0.0030$ ) is

**Table 1.** DSC for PI descriptors depending on the sigma of the Gaussian function ( $\sigma$ ) and resolution ( $res$ ). Bold represents the best results for PI with and without weighting.

		$res = 8 \times 8$	$res = 16 \times 16$	$res = 32 \times 32$	$res = 64 \times 64$
weighting	$\sigma = 0.00025$	$0.714 \pm 0.005$	$0.718 \pm 0.007$	$0.715 \pm 0.007$	$0.709 \pm 0.008$
	$\sigma = 0.0005$	$0.718 \pm 0.005$	$0.715 \pm 0.005$	$0.715 \pm 0.006$	$0.714 \pm 0.004$
	$\sigma = 0.001$	$0.715 \pm 0.006$	$0.716 \pm 0.005$	$0.718 \pm 0.004$	<b><math>0.718 \pm 0.005</math></b>
	$\sigma = 0.002$	$0.706 \pm 0.003$	$0.719 \pm 0.005$	$0.715 \pm 0.004$	$0.710 \pm 0.005$
no wghting.	$\sigma = 0.001$	$0.724 \pm 0.004$	<b><math>0.734 \pm 0.002</math></b>	$0.732 \pm 0.004$	$0.733 \pm 0.004$

<sup>1</sup> Statistical significance is computed with the Wilcoxon signed rank test, as most of the samples do not pass the Shapiro-Wilk normality test.

**Table 2.** DSC for PD\_AGG and PI descriptors extracted from the CLBP\_S and CLBP\_M maps. We consider two encodings for CLBP: rotation invariant uniform (riu2) and rotation invariant (ri) and vary radius  $r$  and the number of samples  $n$ . Bold numbers represent the best results for PD\_AGG and PI.

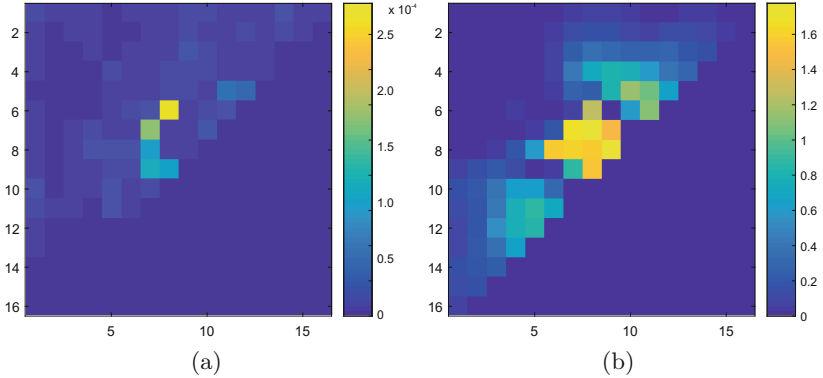
Descriptor	CLBP type		$n = 8$	$n = 16$
PD_AGG	riu2	$r = 3$	$0.613 \pm 0.009$	$0.625 \pm 0.005$
		$r = 5$	$0.654 \pm 0.003$	$0.636 \pm 0.010$
	ri	$r = 3$	$0.632 \pm 0.009$	$0.666 \pm 0.007$
		$r = 5$	$0.681 \pm 0.004$	<b><math>0.687 \pm 0.003</math></b>
PI	riu2	$r = 3$	$0.688 \pm 0.005$	$0.702 \pm 0.004$
		$r = 5$	$0.704 \pm 0.002$	$0.717 \pm 0.003$
	ri	$r = 3$	$0.699 \pm 0.002$	$0.699 \pm 0.002$
		$r = 5$	$0.703 \pm 0.003$	<b><math>0.703 \pm 0.003</math></b>

obtained for the rotation invariant CLBP maps with radius 5 and number of samples 16. This improvement is statistically significant, with  $p$ -value = 0.002 (compared to the PD\_AGG without CLBP). For PI we do not observe an improvement. This was confirmed by further experiments, where we combined PI obtained for the original depth map with PI on CLBP maps. The resulting DSC equals  $0.7178 \pm 0.0034$ . This shows not only that CLBP brings no additional information for PI, but further indicates that it can even be confusing for the classifier. The expressiveness of PI seems to be at a level where CLBP is not able to add additional information. Whereas PD\_AGG is less expressive and thus benefits from the additional processing.

As a next step we investigate which locations of PI are the most important ones for classification. For this purpose we computed Gini importance measure for each location of the PI, see Fig. 3a. The most important pixels are located in the middle of the PI. It is worth noting that there are only few very important pixels, while the others are more than 10 time less important. Moreover, there are few important pixels near to the center of the diagonal. To get a more complete picture, we compute the Fisher discriminant for each location of the PI, see Fig. 3b. The result is to a large degree consistent with the Gini measure and confirms our observation.

Finally, we investigate the performance of topological vs. non-topological descriptors and their combinations. The DSC for baseline descriptors and for their combination with PD\_AGG and PI are presented in Table 3. Our experiments show that both topological descriptors contribute additional valuable information to the baseline descriptors and improve the classification accuracy. All combinations with PD\_AGG are significantly better than the baseline itself. Moreover, PI works significantly better than PD\_AGG with all of the baseline descriptors (except for GHS, GHS+SDD, EGHS+ESDD where the improvement is not significant).





**Fig. 3.** Importance of the PI's pixels obtained with Gini importance measure and Fisher discriminant.

**Table 3.** DSC for baseline descriptors (B) and their combination with PD\_AGG and PI descriptors (B + PD\_AGG and B + PI, respectively). Asterisks (\*) correspond to  $p$ -values  $< 0.01$  when comparing B to B + PD\_AGG and B + PD\_AGG to B + PI.

Descriptor	Baseline (B)	B + PD_AGG	B + PI
EH	$0.641 \pm 0.007$	$0.669 \pm 0.015^*$	$0.696 \pm 0.015^*$
LBP	$0.452 \pm 0.020$	$0.531 \pm 0.023^*$	$0.587 \pm 0.027^*$
DSIFT	$0.486 \pm 0.003$	$0.739 \pm 0.004^*$	$0.764 \pm 0.004^*$
HOG	$0.503 \pm 0.008$	$0.712 \pm 0.007^*$	$0.732 \pm 0.003^*$
GLCM	$0.645 \pm 0.003$	$0.706 \pm 0.002^*$	$0.732 \pm 0.002^*$
GHS	$0.301 \pm 0.048$	$0.470 \pm 0.038^*$	$0.476 \pm 0.066$
SDD	$0.692 \pm 0.003$	$0.735 \pm 0.003^*$	$0.767 \pm 0.004^*$
GHS+SDD	$0.399 \pm 0.028$	$0.426 \pm 0.027^*$	$0.454 \pm 0.029$
EGHS	$0.650 \pm 0.008$	$0.683 \pm 0.003^*$	$0.690 \pm 0.004^*$
ESDD	<b><math>0.743 \pm 0.002</math></b>	<b><math>0.763 \pm 0.002^*</math></b>	<b><math>0.790 \pm 0.002^*</math></b>
EGHS+ESDD	$0.728 \pm 0.005$	$0.740 \pm 0.003^*$	$0.743 \pm 0.005$

## 5 Conclusion

We have presented an investigation of topological descriptors for 3D surface analysis. Our major conclusions are: (i) the aggregation of persistence diagrams removes important information which can be retained by using PI descriptors, (ii) PIs are expressive and robust descriptors that are well-suited to include topological information into ML pipelines, and (iii) topological descriptors are complementary to traditional image descriptors and represent necessary information to obtain peak performance in 3D surface classification. Furthermore, we observed that short intervals in the PD contribute more to classification accuracy than expected. This will be subject to future research.

**Acknowledgements.** Parts of the work for this paper has been carried out in the project 3D-Pitoti which is funded from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 600545; 2013-2016.

## References

1. Adams, H., Chepushtanova, S., Emerson, T., Hanson, E., Kirby, M., Motta, F., Neville, R., Peterson, C., Shipman, P., Ziegelmeier, L.: Persistent images: A stable vector representation of persistent homology (2015). arXiv preprint [arXiv:1507.06217](https://arxiv.org/abs/1507.06217)
2. Bauer, U., Kerber, M., Reininghaus, J.: Phat - persistent homology algorithms toolbox (2013). <https://code.google.com/p/phat/>
3. Bauer, U., Kerber, M., Reininghaus, J., Wagner, H.: PHAT - Persistent homology algorithms toolbox. In: Hong, H., Yap, C. (eds.) ICMS 2014. LNCS, vol. 8592, pp. 137–143. Springer, Heidelberg (2014). [http://dx.doi.org/10.1007/978-3-662-44199-2\\_24](http://dx.doi.org/10.1007/978-3-662-44199-2_24)
4. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(4), 509–522 (2002)
5. Crandall, D., Owens, A., Snavely, N., Huttenlocher, D.: Discrete-continuous optimization for large-scale structure from motion. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3001–3008. IEEE (2011)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893. IEEE (2005)
7. Edelsbrunner, H., Letscher, D., Zomorodian, A.: Topological persistence and simplification. *Discrete Comput. Geom.* **28**, 511–533 (2002)
8. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **55**(1), 119–139 (1997)
9. Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process.* **19**(6), 1657–1663 (2010)
10. Haralick, R.M., Shanmugam, K., Dinstein, I.H.: Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* **6**, 610–621 (1973)
11. ISO-IEC: Information Technology - Multimedia Content Description Interface.15938, ISO/IEC, Moving Pictures Expert Group, 1st edn. (2002)
12. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(5), 433–449 (1999)
13. Juda, M., Mrozek, M., Brendel, P., Wagner, H., et al.: CAPD::RedHom (2010–2015). <http://redhom.ii.uj.edu.pl>
14. Juda, M., Mrozek, M.: CAPD:RedHom v2 - homology software based on reduction algorithms. In: Hong, H., Yap, C. (eds.) ICMS 2014. LNCS, vol. 8592, pp. 160–166. Springer, Heidelberg (2014)
15. Li, C., Ovsjanikov, M., Chazal, F.: Persistence-based structural recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2003–2010. IEEE (2014)
16. López, V., Fernández, A., García, S., Palade, V., Herrera, F.: An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Inf. Sci.* **250**, 113–141 (2013)
17. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)

18. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. *Pattern Recogn.* **29**(1), 51–59 (1996)
19. Othmani, A., Lew Yan Voon, L., Stolz, C., Piboule, A.: Single tree species classification from terrestrial laser scanning data for forest inventory. *Pattern Recogn. Lett.* **34**(16), 2144–2150 (2013)
20. Poincaré, H.J.: Sur le probleme des trois corps et les équations de la dynamique. *Acta Math.* **13**, 1–270 (1890)
21. Poincaré, H.J.: Les méthodes nouvelles de la mécanique céleste. Gauthiers-Villars, Paris (1892, 1893, 1899)
22. Poincaré, H.J.: Analysis situs. *J. Éc. Polytech.*, ser. 2 **1**, 1–123 (1895)
23. Reininghaus, J., Huber, S., Bauer, U., Kwitt, R.: A stable multi-scale kernel for topological machine learning (2014). arXiv preprint [arXiv:1412.6821](https://arxiv.org/abs/1412.6821)
24. Rusu, R.B., Marton, Z.C., Blodow, N., Beetz, M.: Persistent point feature histograms for 3D point clouds. In: Proceedings of the 10th International Conference on Intel Autonomous System (IAS-10), Baden-Baden, Germany, pp. 119–128 (2008)
25. Seiffert, C., Khoshgoftaar, T.M., Van Hulse, J., Napolitano, A.: Rusboost: A hybrid approach to alleviating class imbalance. *IEEE Trans. Syst. Man Cybern. Part A: Syst. Hum.* **40**(1), 185–197 (2010)
26. Vedaldi, A., Fulkerson, B.: Vlfeat: An open and portable library of computer vision algorithms. In: Proceedings of the International Conference on Multimedia, pp. 1469–1472. ACM (2010)
27. Wohlfeil, J., Strackenbrock, B., Kossyk, I.: Automated high resolution 3D reconstruction of cultural heritage using multi-scale sensor systems and semi-global matching. *Int. Arch. Photogrammetry Remote Sens. Spat. Inf. Sci. XL-4 W 4*, 37–43 (2013)
28. Wu, C.: Towards linear-time incremental structure from motion. In: 2013 International Conference on 3DTV, pp. 127–134. IEEE (2013)
29. Zeppelzauer, M., Poier, G., Seidl, M., Reinbacher, C., Breiteneder, C., Bischof, H., Schuster, S.: Interactive segmentation of rock-art in high-resolution 3D reconstructions. In: 2015 Digital Heritage, vol. 2, pp. 37–44, September 2015. doi:[10.1109/DigitalHeritage.2015.7419450](https://doi.org/10.1109/DigitalHeritage.2015.7419450)
30. Zeppelzauer, M., Seidl, M.: Efficient image-space extraction and representation of 3D surface topography. In: Proceedings of the IEEE International Conference on Image Processing (ICIP). IEEE, Quebec, Canada (2015). <http://arXiv.org/pdf/1504.08308v3.pdf>