

Classification of Firn Data Via Topological Features^{*}

Jesse Dimino¹[0009-0008-1776-3780], Matt Jester², Kaitlin Keegan³[0000-0002-1241-2681], Thomas Weighill⁴[0000-0003-2614-0979], and Sarah Day⁵[0009-0009-8047-0187]

¹ Department of Mathematics, CUNY College of Staten Island, Staten Island, NY 10314, United States jesse.dimino@csi.cuny.edu

² Division of Student Success, University of North Carolina at Greensboro, Greensboro, NC 27412, United States mwjester@uncg.edu

³ Nevada Geosciences, Department of Geological Sciences and Engineering, University of Nevada, Reno, Reno, NV 89557, United States kkeegan@unr.edu

⁴ Department of Mathematics and Statistics, University of North Carolina at Greensboro, Greensboro, NC 27412, United States t_weighill@uncg.edu

⁵ Department of Mathematics, William & Mary, Williamsburg, VA 23185, United States sldayx@wm.edu

Abstract. In this paper we evaluate the performance of topological features for generalizable and robust classification of firn image data, with the broader goal of understanding the advantages, pitfalls, and trade-offs in topological featurization. Firn refers to layers of granular snow within glaciers that haven't been compressed into ice. This compactification process imposes distinct topological and geometric structure on firn that varies with depth within the firn column, making topological data analysis (TDA) a natural choice for understanding the connection between depth and structure. We use two classes of topological features, sublevel set features and distance transform features, together with persistence curves, to predict sample depth from microCT images. A range of challenging training-test scenarios reveals that no one choice of method dominates in all categories, and uncovers a web of trade-offs between accuracy, interpretability, and generalizability.

Keywords: Persistent homology · Firn · Computer vision

^{*} Day, Keegan, and Jester were partially supported by the Army Research Office under Grant Number W911NF-20-1-0131. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes not-withstanding any copyright notation herein. Dimino and Weighill were partially supported by NSF REU Site: Complex Data Analysis using Statistical and Machine Learning Tools (DMS-1950549).

1 Introduction

Image classification is a well-studied area of machine learning with important real-world applications. The classification of textures presents a specific challenge, namely that a well-tuned classifier should be roughly invariant under various rigid transformations of the image. For example, a patch of sand should not change its classification label if rotated 180° , while an image of a 6 can become an image of a 9 under such a transformation. This invariance can be incorporated into the machine learning pipeline in various ways. One could add transformed versions of each image to the training set (data augmentation), or one could force the required invariance in the architecture of the classifier itself (invariant machine learning [17,4]). We focus on a third option, namely *invariant featurization* – transforming the data into vectors in an invariant way before feeding them into a standard machine learning pipeline.

For this purpose we draw on the well-suited family of methods called topological data analysis (TDA). TDA is a cluster of data analysis methods based on theory from algebraic topology. One key tool in TDA is *persistent homology* which tracks the appearance and disappearance of topological features (such as connected components and holes) over a range of scales or thresholds. Since its introduction by Edelsbrunner, Letscher and Zomorodian in 2000, persistent homology has had wide-ranging applications in areas such as geospatial data, time series data, image classification, medical diagnostics, dimension reduction, deep learning, and more. Most relevant to our work are applications to texture classification in [7,9].

In this paper, we undertake a particular image classification task on geological data to evaluate and compare the performance of various persistent homology featurizations. Specifically, our dataset consists of firn image data obtained from X-ray micro-computed tomography (micro-CT) scans of shallow ice cores. This dataset has a number of unique features, which are visible in the examples in Figure 1.

- Each image is grayscale and while each is roughly divided into a foreground (pore space) and background (ice space), there is enough variation and noise in each to make extracting a binary image non-trivial.
- The pore space and ice space complex (as opposed to the granular noise) can be viewed as a medium-scale texture in that the pattern does not have any well-defined orientation or boundary.
- Even within the sample images, a clear trend can be noticed as depth increases: the pore space goes from a connected complex taking up roughly half of the image to a set of small bubbles.

Each sample in our dataset comes with a label indicating the depth (on the ice core) at which the scan was taken, with ten possible values corresponding to ten sampling locations. Our classification task is to predict depth from an image. Since the effect of depth on the firn structure is the result of well-studied geological processes, a long term goal is to apply a depth predictor trained on this ice core to other ice core images. By comparing the predictions with ground

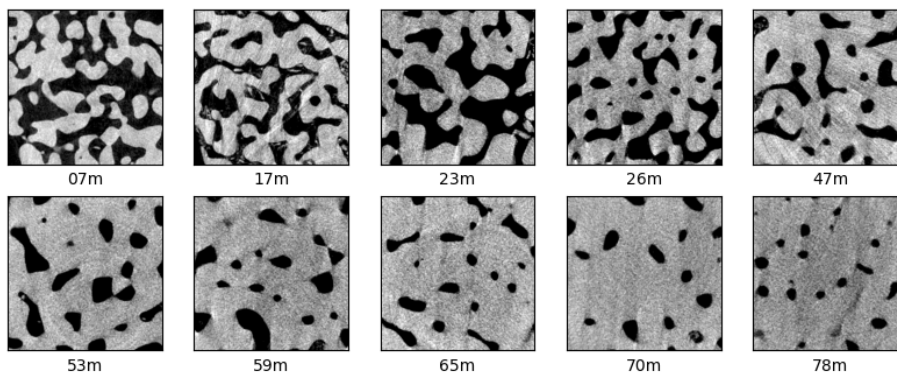


Fig. 1: One sample image per depth for the firn dataset. Gray areas are ice, while the black areas show the pore (air) space.

truth for the alternate core, we can gauge the geological differences between the two sites where the cores were taken. It may be, for example, that the geological forces that produced the 26m sample in our core correspond to much deeper or shallower samples in another core.

Persistent homology features, which are invariant under many different transformations, are naturally suited to this task since they can help mitigate overfitting (say to specific locations within each image) and promote generalizability and robustness. Indeed, TDA methods have already been used to study this dataset [6, 11, 2, 12]. Our study involves two methods of persistent homology featurization: *sublevel set filtration*, which computes topological features at various pixel intensity thresholds, and *distance transforms*, which computes topological features by slowly thickening up the background region of the image.

We set up a range of different prediction and classification tasks, aiming to find a featurization which is accurate, robust to various manipulations of the test set, and which can extrapolate to out-of-sample depth values. Ultimately, we find that none of the featurizations we consider satisfy all these requirements perfectly. Instead, our experiments reveal an interesting web of trade-offs between our desiderata, and insights into the use of topological features on firn images and similar datasets.

We now briefly outline the rest of the paper. We begin in the next section with some necessary background including a more detailed description of the dataset and its geological properties. We then outline our classification methods and experiments in Section 3, and give some results and discussion in Section 4. We conclude the paper with a summary of the lessons learned during the process, and some directions for future work on TDA and firn.

2 Background

2.1 Firn image data

Firn is a porous medium that forms in polar regions where snow does not completely melt during the summer season; instead, it accumulates, densifies, and ultimately becomes glacial ice. Firn comprises a matrix of dense snow with an interconnected pore space, which undergoes densification due to the increasing pressure from the accumulation of snowfall on the surface. The zone at the top of glaciers and ice sheets where the firn is densifying is referred to as the firn column. As depth increases, the ice matrix grows and the pore space constricts until the pores close off into individual bubbles within the ice at the bottom of the firn column (Figure 1).

The process of densification and its impact on the underlying microstructure of firn, has important implications on many glaciological applications. Recent studies suggest that microstructural information is needed to improve empirical models of firn densification [13], but the technique generates large amounts of data with unique challenges. Additionally, there are no universal image processing and analysis methods for extracting useful microstructural information from the micro-CT data. TDA is well-suited for improving firn microstructure classification, since topological features are invariant under rotation, translation and other rigid transformations which should be viewed as a result of changes in the sampling technique and not the underlying geological data.

The firn micro-CT data used in this study were collected by Kaitlin Keegan from a core collected at NEEM, Greenland, and they were processed with a Skyscan 1172 micro-CT scanner housed within a cold laboratory. Each scan produced a stack of approximately 900 2-D cross-sectional images that represent the original 3-D sample. The cross-sectional images, or slices, are grayscale where gray pixels represent the ice-space and black pixels represent the pore-space of the firn. Pixel values range from 0 (black) to 255 (white). For each core, every 5th slice was subsampled. We investigate the microstructure of samples from ten sample depths ranging from 7m to 78 m, roughly spanning the entire firn column at that site. Figure 1 shows one cross-sectional image from each sample depth. Note how the amount of pore space decreases as the depth increases. The result is a transition from a connected, curved pore space to a collection of bubbles as depth increases.

2.2 Persistent homology

For an introduction to the full theory of persistent homology we invite the reader to the paper [15] or to one of the recent book-length treatments such as [5]. In this section we give a brief overview. Persistent homology begins with a nested sequence of spaces

$$X_0 \subseteq X_1 \subseteq \cdots$$

In our case, each X_i will represent a set of pixels in a fixed image, produced in one of two ways (see Section 3), which grows as i increases. As the set of

pixels grows, topological features such as holes (0-th dimensional features) and connected components (1-dimensional features) appear and disappear, and they can be tracked in a well-defined way using the cubical complex implementation of persistent homology in [16]. Each feature has a *birth* and *death* time. For each dimension, every birth-death pair is represented as a point (b, d) in the plane to create a *persistence diagram*. Note that we always have $b < d$, and it is customary to include the $y = x$ line in persistence diagrams.

2.3 Persistence curves

In an ideal world, raw persistence diagrams could serve as the input to a classifier. However, most well-established and efficient supervised machine learning algorithms require vectors as input. Converting persistence diagrams into vectors is unsurprisingly an area of active research. Some standard choices are persistence landscapes [3] and persistence images [1]. For our study, we utilize persistence curves, based on their usefulness for texture classification in [9]. Persistence curves are a general family of methods for producing vectors from persistence diagrams. We select two for this paper, namely the Betti curve and the Gaussian persistence curve. Other choices are possible (for example, lifespan curves), but we choose two curves which represent an interesting trade-off between interpretability and stability. Indeed, the Betti curve has a simple interpretation in terms of counting topological features (as we will see below), but is not as smooth the Gaussian persistence curve [8].

Definition 1 (Betti curve). *Consider a k^{th} -dimensional persistence diagram $P = \{(b_0, d_0), (b_1, d_1), \dots, (b_n, d_n)\}$. For $t \in \mathbb{R}$, consider the subset of the plane given by*

$$F_t = \{(x, y) \in \mathbb{R}^2 \mid x \leq t < y\}$$

which we call a fundamental box at threshold t . We define the k^{th} Betti curve to be the function

$$\beta_k(t) = \#(F_t \cap P)$$

where $\#$ counts the number of elements.

See Figure 2 for a visual exposition of how the Betti curve is created from a persistence diagram. If t ranges over a finite set $\{0, 1, \dots, N\}$, then we can view β_k as a vector in $\mathbb{R}^N$. In this paper, we always concatenate the β_0 and β_1 curves to obtain a vector $\mathbf{v}_{\text{Betti}} \in \mathbb{R}^{2N}$. By the Fundamental Lemma of Persistent Homology [10, p. 118], $\beta_k(t)$ is the k^{th} Betti number of the sublevel set given by the threshold t , hence the name. Recall that the 0^{th} Betti number counts connected components while the 1^{st} Betti number counts holes. In summary: the Betti curves β_0 and β_1 encode the changing topology of a sequence in a straightforward way: $\beta_0(t)$ and $\beta_1(t)$ are the number of connected components and holes respectively in X_t .

Definition 2 (Gaussian persistence curve). Consider a k^{th} -dimensional persistence diagram $P = \{(b_0, d_0), (b_1, d_1), \dots, (b_n, d_n)\}$. Given a weighting function $\kappa : \mathbb{R}^2 \rightarrow [0, \infty)$ satisfying $\kappa(x, x) = 0$ for all $x \in \mathbb{R}$ and a 2×2 covariance matrix Σ , we define the k^{th} -dimensional persistence surface to be the function

$$\rho(P) = \sum_{(b,d) \in P} \kappa(b, d) g_{(b,d), \Sigma}$$

where $g_{(b,d), \Sigma}$ is a bivariate normal distribution with mean (b, d) and covariance matrix Σ . We define the k^{th} Gaussian persistence curve to be the function

$$\gamma_k(t) = \int_{F_t} \rho(P) dx dy$$

where F_t is the fundamental box at threshold t .

The Gaussian persistence curve can be viewed as a smoothed and re-weighted version of the Betti curve: points are replaced by Gaussians, an idea originating with persistence images [1], and points far from the diagonal are up-weighted using κ . In our experiments, we use $\kappa(b, d) = \frac{d-b}{\sum_{(b', d') \in P} (d' - b')}$ throughout. Note that this weighting function depends on the diagram, so that it is normalized in a weak sense. We use $\Sigma = 100I$ for the sublevel set featurization and $\Sigma = 25I$ for the distance transform featurization (described in the next section), since the latter has a smaller maximum value. As with the Betti curves, we concatenate 0 and 1-dimensional curves to obtain a vector $\mathbf{v}_{\text{Gaussian}} \in \mathbb{R}^{2N}$.

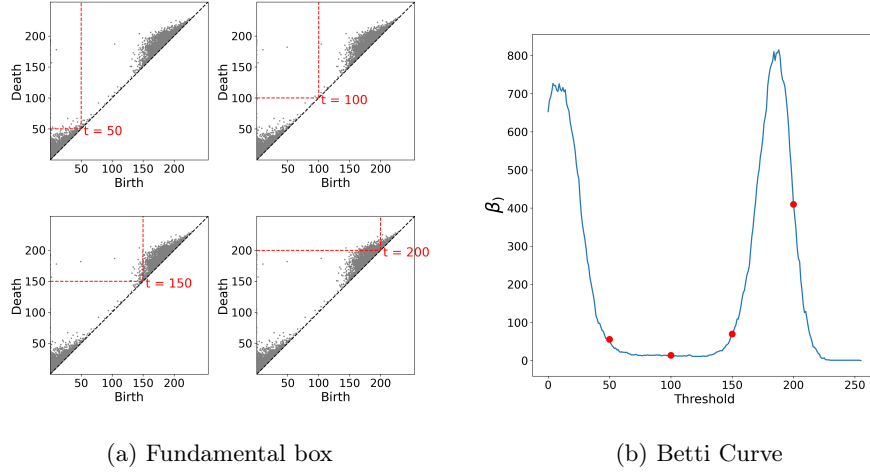


Fig. 2: Computing the 0^{th} Betti curve (right) from the 0^{th} -dimensional persistence diagram (left). The red dashed lines indicate the extent of the fundamental box at each threshold t .

2.4 Random forests

Random forests are built using decision trees. Decision trees assign outcomes (in our case, depth values) to inputs using axis-aligned decision boundaries, where the quality of a particular set of decision boundaries is evaluated through a loss function. Decision trees can be used to predict a scalar outcome using linear regression as the final stage, or they can be used to classify into predefined categories. We will use both versions in our experiments. Decision trees are non-parametric so they have low bias but high variability. A standard solution to try to minimize the total prediction error is to aggregate the results using a set of distinct decision trees, which together are called a *random forest*. To create the random forest, new decision trees are created using bootstrapped samples of the data and a random subsample of features at each node, rather than using the entire dataset at every possible stage. We use random forests as our final stage in our classification pipeline since they require very little parameter tuning and also are capable of producing explainable results.

3 Methods

In this section, we describe the two topological featurization methods used on the firn images as preprocessing steps, which we evaluate and compare in the next section. In each case, persistent homology was compute using the `gudhi` library's implementation of the cubical complex method in [16]. A visual overview of each method is given in Figure 3.

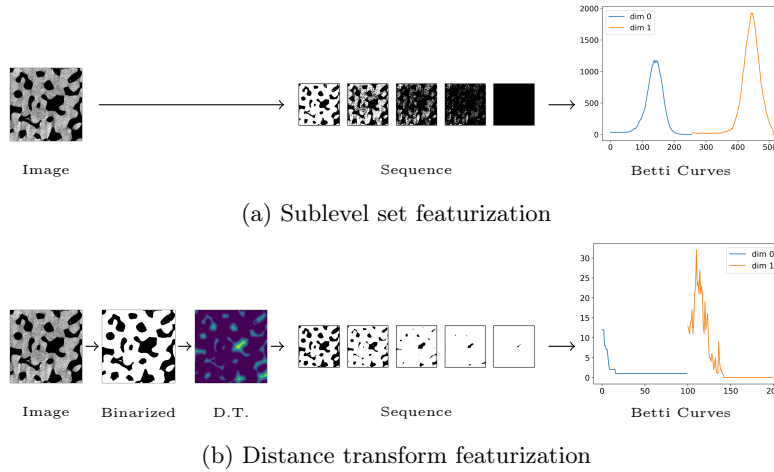


Fig. 3: Visual representations of the two topological featurizations for an example image, using Betti curves as an example.

3.1 Sublevel set featurization

Our first featurization method is inspired by the texture classification method in [7]. Each pixel in a grayscale image carries an intensity value from 0 to 255. Let X_t denote the set of pixels with intensity at most t , so that X_{255} is the entire image. Each image gives rise to a sequence of expanding pixel regions

$$X_0 \subseteq X_1 \subseteq \cdots \subseteq X_{255}$$

This data gives rise to two Betti curves (for 0^{th} and 1^{st} dimension) as described in Sections 2.2 and Sections 2.3 above, concatenated to give a Betti curve vector $\mathbf{v}_{\text{Betti}}^{SS} \in \mathbb{R}^{512}$ and a Gaussian persistence curve $\mathbf{v}_{\text{Gaussian}}^{SS} \in \mathbb{R}^{512}$.

3.2 Distance transform featurization

Our second method is an implementation of the idea in [11] to use a distance transform (DT) instead of raw pixel intensities. Each image is binarized using Gaussian smoothing and Otsu’s binarization method [14] so that pixels are assigned to either ice space or pore space. We define a function f on the image as follows. On an ice space pixel, f is defined to be zero. On a pore space pixel, f is defined to be the Euclidean distance (in pixel space) to the nearest ice space pixel. Define X_{DT}^t to be the set of pixels with f value less than t :

$$X_{DT}^t = \{(i, j) \mid f((i, j)) \leq t\}$$

This definition gives rise to a sequence of spaces

$$X_{t_1} \subseteq X_{t_2} \subseteq \cdots \subseteq X_{t_N}$$

where $t_1 < t_2 < \cdots < t_N$ are the values in the range of f , rounded to integers. Note that this sequence is different to the sequence in the previous section. This sequence gives rise to a persistence curves, which we denote by $\mathbf{v}_{\text{Betti}}^{DT}$ and $\mathbf{v}_{\text{Gaussian}}^{DT}$ for Betti and Gaussian persistence curves respectively. In practice, we note that f has a upper bound of 100, so our resulting vectors lie in $\mathbb{R}^{200}$.

3.3 Experiments

Our input data consists of the images of firn at various depths, and the outcome to be predicted is the depth. As input features, we consider four options: the sublevel set featurization $\mathbf{v}^{SS}$ or the distance transform featurization $\mathbf{v}^{DT}$, with either Betti or Gaussian curves for each choice. On all featurizations, we use one of two types of predictor: a random forest regression method to predict depth as a scalar variable, or random forest classifier method to predict depth as a categorical variable with ten choices, using default `sklearn` parameters in both cases.

In order to test the generalizability of each setup, we create a range of challenging test-set scenarios:

- **Whole:** We make a random 75%/25% train-test split of all images.
- **Split:** Each image is split into four quadrants. We then make a random 75%/25% test-train split of the set of all quadrant images. See Figure 4.
- **Split BR:** Each image is split into four quadrants. Instead of a random train-test split, we assign all bottom right quadrants to the test set, and all others to the training set.
- **Blurred:** We make a random 75%/25% train-test split of all images. We then blur *only* the test set with a 3×3 Gaussian blur.
- **Missing depths:** The test set consists of full images from depths 23m, 53m and 70m, while the training set consists of all other images.

The **Split BR** and **Blurred** scenarios are designed to gauge the likelihood that the method would give reliable results on a different firn core in the future. The **Missing depths** scenario allows us to test the ability of the regression-based methods to predict out-of-sample depths.

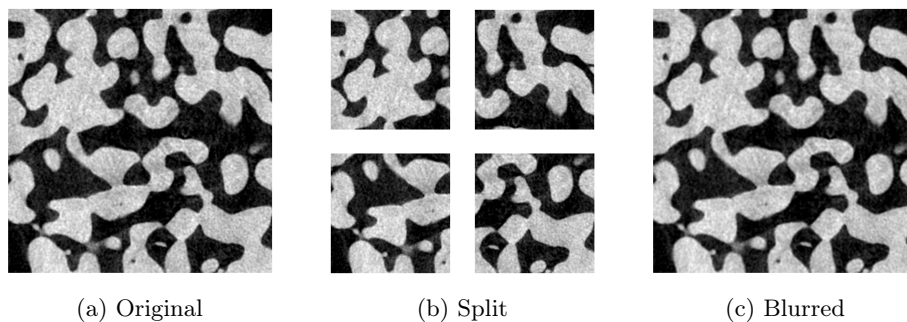


Fig. 4: The original image and two manipulations used to evaluate predictor generalizability. The original image is shown in (a). In (b), we show the quadrants used for the **Split** and **Split BR** scenarios. The blurred version of the image (applied to the test set in **Blurred**) is shown in (c).

4 Results and Discussion

Table 1 shows the error (for scalar prediction) and accuracy (for category prediction) for the featurizations and train-test scenarios considered in this paper. For each setup, we run ten trials and report the mean and standard deviation. Overall, we see among the four featurizations, $\mathbf{v}_{\text{Betti}}^{SS}$, $\mathbf{v}_{\text{Gaussian}}^{SS}$, $\mathbf{v}_{\text{Betti}}^{DT}$, and $\mathbf{v}_{\text{Gaussian}}^{DT}$, none dominate in every category. Instead, the experiments reveal a number of interesting trade-offs. We list some of these here with brief explanations.

Sublevel set features give very high accuracy on unblurred images.

This is likely due to the fact that the images in each depth category are highly correlated due to being slices taken close together. The sublevel set filtration

Scalar prediction (mean absolute error in metres)				
	$\mathbf{v}_{\text{Betti}}^{SS}$	$\mathbf{v}_{\text{Gaussian}}^{SS}$	$\mathbf{v}_{\text{Betti}}^{DT}$	$\mathbf{v}_{\text{Gaussian}}^{DT}$
Whole	0.74 ± 0.12	0.90 ± 0.14	4.01 ± 0.29	3.86 ± 0.25
Split	2.10 ± 0.10	1.92 ± 0.10	7.74 ± 0.15	8.36 ± 0.18
Split BR	2.40 ± 0.04	2.21 ± 0.05	7.35 ± 0.02	7.86 ± 0.03
Blurred	14.61 ± 0.78	11.26 ± 0.93	3.97 ± 0.27	3.97 ± 0.25
Missing depths	9.93 ± 0.22	10.15 ± 0.26	6.20 ± 0.04	7.69 ± 0.10
Category prediction (% accuracy)				
	$\mathbf{v}_{\text{Betti}}^{SS}$	$\mathbf{v}_{\text{Gaussian}}^{SS}$	$\mathbf{v}_{\text{Betti}}^{DT}$	$\mathbf{v}_{\text{Gaussian}}^{DT}$
Whole	99.03 ± 0.49	98.25 ± 0.63	71.72 ± 1.55	69.72 ± 2.24
Split	95.95 ± 0.72	96.72 ± 0.27	45.98 ± 1.18	42.82 ± 0.93
Split BR	92.02 ± 0.31	93.59 ± 0.21	46.22 ± 0.56	39.93 ± 0.43
Blurred	50.59 ± 4.51	81.75 ± 1.61	71.34 ± 1.77	68.91 ± 2.04

Table 1: Performance measures for topological feature-based depth prediction. When predicting depth as a scalar value we use mean absolute error (MAE). Note that true values range from 7m to 78m. For predicting depth as one of ten predefined levels (see Figure 1), we use accuracy. Ten trials are performed to obtain each entry, with the mean and standard deviation shown.

method is able to leverage the small-scale information in each image to accurately predict the depth category.

The performance of sublevel set features degrades on blurred test sets (with one exception).

To understand why, see Figure 5 which shows the mean $\mathbf{v}_{\text{Betti}}^{SS}$ vector by depth. We see separation between depths, but observe that curves peak at values above 1,000. Recall that the Betti curve counts the number of holes or connected components at particular thresholds. Visually, we see that the pore space never has this many connected components in any sample, so the Betti curves are being dominated by smaller-scale structures. Through Figure 5(b), we see that the Betti curve is detecting the granular noise in the ice space of the image. This noise is altered by even small amounts of blurring, which explains the high error on blurred test images. The exception to this is $\mathbf{v}_{\text{Gaussian}}^{SS}$ for categorical prediction. This is likely because the weighting function for the Gaussian persistence curve introduces a normalization factor which can account for the blurring.

Distance transform errors for scalar prediction are stable across tasks, and classification accuracy is robust to blurring.

For the distance transform features, we see that the mean absolute error is around 7 for most tasks, an error which corresponds roughly to being misassigned to the nearest depth category for most images. Blurring does not substantially alter the accuracy values for distance transforms, likely because the binarization is minimally affected.

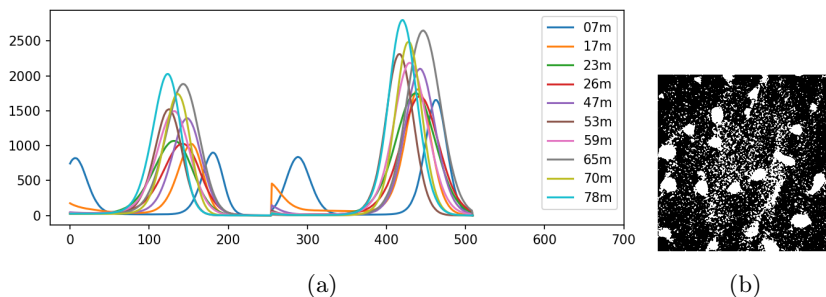


Fig. 5: On the left, the average Betti curve vector $\mathbf{v}_{\text{Betti}}^{SS}$ by depth for the sublevel set featurization, showing good separation but conspicuously high peak values. On the right, we show a sample image from the 78m depth thresholded at $t = 120$, near the peak of the average curve for that depth. We see that the topology of the image at this threshold is dominated by granular noise.

Distance transform performance is worse on quadrant images.

This is likely because slicing up the image reduces the number of air pockets and increases the border effects, both of which can affect the distance transform and give less information to learn from.

Missing depths are better modeled by distance transform features than by sublevel set features.

Note that this is despite the sublevel set features being far superior on the other unblurred test scenarios. This shows that the features detected by the distance transform features are easier to linearly extrapolate than the (likely smaller-scale) features detected by sublevel sets.

5 Conclusion

Topological features have many advantages when attempting to analyze the kind of images in this paper, namely that they are by design invariant to translation, rotation, and other transformations. In this paper, we sought to understand better what kinds of featurizations are better than others at various tasks. We find that in a straightforward classification task, sublevel set filtrations achieve almost perfect accuracy, but that their reliance on small scale features hinders their robustness to noise and their generalizability to out-of-sample data (i.e. missing depths). This demonstrates that while the “noise” near the diagonal of persistence diagrams are often viewed as unimportant, a classifier based on topological features can be susceptible to overfitting to it. The distance transform, on the other hand, captures much coarser information than the sublevel set filtration, and also depends on a separate binarization phase during preprocessing. This

results in a controlled but significant error. On the other hand, this error is more stable to blurring effects and more readily predicts out-of-sample depths.

We propose that future work focus on the distance transform featurization, make improvements to create features that depend less on an arbitrary binarization step, and use more complex downstream classifier stages which might improve accuracy. Moreover, an examination of distance transform featurizations could reveal an explainable/measurable feature of firn images which is directly predictive of depth. This would be of interest to geologists in their study of firn formation processes.

6 Acknowledgements

We would like to thank Xiaoli Gao for her crucial input and guidance on an earlier version of this project.

7 Conflicts of Interest

The authors report no known conflicts of interest that could affect the outcomes of this study.

References

1. Adams, H., Emerson, T., Kirby, M., Neville, R., Peterson, C., Shipman, P., Chepushtanova, S., Hanson, E., Motta, F., Ziegelmeier, L.: Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research* **18**(8), 1–35 (2017)
2. Belotti, V., Jester, M., Keegan, K., Chung, Y.M., Day, S.: Topological estimation of 2d image data via subsampling and application to firn. *IEEE Access* **9**, 10348–10356 (2021)
3. Bubenik, P., et al.: Statistical topological data analysis using persistence landscapes. *J. Mach. Learn. Res.* **16**(1), 77–102 (2015)
4. Cahill, J., Iverson, J.W., Mixon, D.G., Packer, D.: Group-invariant max filtering. *Foundations of Computational Mathematics* pp. 1–38 (2024)
5. Carlsson, G., Vejdemo-Johansson, M.: *Topological data analysis with applications*. Cambridge University Press (2021)
6. Chung, Y.M., Day, S.: Topological fidelity and image thresholding: A persistent homology approach. *Journal of Mathematical Imaging and Vision* **60**, 1167–1179 (2018)
7. Chung, Y.M., Hull, M., Lawson, A.: Smooth summaries of persistence diagrams and texture classification. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. pp. 840–841 (2020)
8. Chung, Y.M., Hull, M., Lawson, A., Pritchard, N.: Gaussian persistence curves. *Journal of Statistical Theory and Practice* **20**(1), 18 (2026)
9. Chung, Y.M., Lawson, A.: Persistence curves: A canonical framework for summarizing persistence diagrams. *Advances in Computational Mathematics* **48**(1), 6 (2022)

10. Edelsbrunner, H., Harer, J.L.: Computational topology: an introduction. American Mathematical Society (2022)
11. Hu, C.S., Lawson, A., Chung, Y.M., Keegan, K.: Two-parameter persistence for images via distance transform. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4176–4184 (2021)
12. Lawson, A., Hoffman, T., Chung, Y.M., Keegan, K., Day, S.: A density-based approach to feature detection in persistence diagrams for firm data. *Foundations of Data Science* **4**(4), 623–639 (2022)
13. Lundin, J.M., Stevens, C.M., Arthern, R., Buizert, C., Orsi, A., Ligtenberg, S.R., Simonsen, S.B., Cummings, E., Essery, R., Leahy, W., et al.: Firm model intercomparison experiment (firmice). *Journal of Glaciology* **63**(239), 401–422 (2017)
14. Otsu, N., et al.: A threshold selection method from gray-level histograms. *Automatica* **11**(285-296), 23–27 (1975)
15. Otter, N., Porter, M.A., Tillmann, U., Grindrod, P., Harrington, H.A.: A roadmap for the computation of persistent homology. *EPJ Data Science* **6**, 1–38 (2017)
16. Wagner, H., Chen, C., Vućini, E.: Efficient computation of persistent homology for cubical data. In: *Topological methods in data analysis and visualization II: theory, algorithms, and applications*, pp. 91–106. Springer (2011)
17. Wood, J., Shawe-Taylor, J.: Representation theory and invariant neural networks. *Discrete applied mathematics* **69**(1-2), 33–60 (1996)