Topological and Dynamical Representations for Radio Frequency Signal Classification

Audun Myers [1] **Timothy Doster** [1] **Colin Olson** [2] **Tegan Emerson** [1 3 4]

Editors: *S. Vadgama, E.J. Bekkers, A. Pouplin, S.O. Kaba, H. Lawrence, R. Walters, T. Emerson, H. Kvinge, J.M. Tomczak, S. Jegelka*

Abstract

Radio Frequency (RF) signals are found throughout our world, carrying over-the-air information for both digital and analog uses with applications ranging from WiFi to the radio. One area of focus in RF signal analysis is determining the modulation schemes employed in these signals which is crucial in many RF signal processing domains from secure communication to spectrum monitoring. This work investigates the accuracy and noise robustness of novel Topological Data Analysis (TDA) and dynamic representation based approaches paired with a small convolution neural network for RF signal modulation classification with a comparison to state-of-the-art deep neural network approaches. We show that using TDA tools, like Vietoris-Rips and lower star filtrations, and the Takens' embedding in conjunction with a standard shallow neural network we can capture the intrinsic dynamical, geometric, and topological features of the underlying signal's manifold, offering informative representations of the RF signals. Our approach is effective in handling the modulation classification task and is notably noise robust, outperforming the commonly used deep neural network approaches in mode classification. Moreover, our fusion of dynamical and topological information is able to attain similar performance to deep neural network architectures with significantly smaller training datasets.

[*]Equal contribution [1]Pacific Northwest National Laboratory [2]US Naval Research Laboratory [3]Colorado State University [4]University of Texas El Paso. Correspondence to: Audun Myers <audun.myers@pnnl.gov>.

Proceedings of the Geometry-grounded Representation Learning and Generative Modeling Workshop (GRaM) at the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 251, 2024.

1. Introduction

Radio frequency (RF) signals play a crucial role in wireless communication systems; classifying the modulation type used in an RF signal is essential for tasks like signal decoding, spectrum management, and interference mitigation. Traditional approaches for modulation classification often rely on handcrafted features which are difficult to develop for specific applications or deep neural network models that take a significant amount of signal type and environmental variety to generalize. Due to these limitations, our goal in this work is to explore the viability of data analysis method known as topological data analysis (TDA) in conjunction with the more traditional Takens' embedding as a dynamical representation of the data for studying RF signals with the specific application of classifying modulation type.

TDA is a field in data analysis that focuses on the *intrinsic geometric and topological properties* of data sets (Carlsson, 2009). These properties remain unchanged under continuous deformations, such as stretching or bending, but not tearing or gluing. This makes TDA particularly useful for analyzing noisy or high-dimensional data where small-scale features may be less important than the overall structure. As is the case with RF modulation classification where the objective is to recognize the global modulation without bias towards features induced by environmental variation.

A key concept in TDA is the *persistence diagram* (Edelsbrunner & Harer, 2022). It is a graphical representation that summarizes the topological features of a data set across different scales. Each point in the diagram corresponds to a topological feature, such as a connected component or a hole, and its coordinates represent the "birth" and"death" scales of that feature. Features that persist across a wider range of scales are considered more significant.

TDA has been widely used in machine learning-based data analysis. These approaches use a vectorization of the resulting persistence diagram as input for a neural network. Popular methods for vectorizing persistence diagrams (Barnes et al., 2021) include persistence images (Adams et al., 2017),

persistence landscapes (Bubenik et al., 2015), Adcock-Carlsson coordinates (Adcock et al., 2013), and template functions (Perea et al., 2023).

In the context of time series data, TDA offers a powerful framework for capturing the underlying dynamics and identifying hidden patterns. Common approaches for studying time series data using TDA are based on first constructing point cloud data through, for example, Takens' embedding (Takens, 2006). The standard method then uses the Vietoris-Rips filtration on the metric space to construct simplicial complexes from the embedded time series data for a given distance threshold. By incrementing the distance threshold we can capture the persistent homology through the distant filtration by studying the changing simplicial complex homology.

By applying these methods, TDA can be used to perform various analysis tasks on time series data, including anomaly detection (Umeda et al., 2019), change point detection (Zheng et al., 2023; Ravishanker & Chen, 2019; Guralnik & Srivastava, 1999; Islambekov et al., 2020; Myers et al., 2023b), and time series classification (Umeda, 2017; Karan & Kaygun, 2021; Majumdar & Laha, 2020).

Related work introduced the concept of topological fusion of persistence image (Myers et al., 2023a) or TopFusion. We build on this method in this work by fusing multiple persistent homology filtrations as well as a heat map representation of the Takens' embedding into a single multi-dimensional image that can be analyzed with convolutional neural network architectures designed for image analysis.

Organization This work is organized as follows. We begin by introducing the needed background information including RF modulation, sublevel set persistence, Vietoris-Rips filtration, persistence images, and the heat map of a Takens' embedding. Next we overview our method of fusing each of the signal processing methods into a single multi-dimensional image. Lastly we apply our method to two datasets (RadioML (2018) (O'Shea et al., 2018) and PanoRadio (Scholl, 2019)) and show that our method performs comparably to a pure deep learning approach with reduced data requirements.

2. Background

In this section we will introduce several commonly used tools from TDA for time series analysis. Additionally, we will pair these descriptions with simple examples.

TDA studies the shape of data to gain understanding of the underlying system. In this work we leverage two tools from TDA for time series analysis: persistent homology of a Vietoris-Rips filtration and sublevel set persistence. The output of these filtrations is a persistence diagram D_p that captures the topology of the data at dimension p over various scales which then also captures information about the underlying geometry. A more thorough background on TDA, and persistent homology specifically, can be found in (Edelsbrunner & Harer, 2008; Munch, 2017; Perea, 2018)

In section 2.1 we introduce RF modulation with an example, in section 2.1.1 we introduce sublevel set persistence as a method of applying persistent homology to time series, in Section 2.1.2 we overview persistent homology using Vietrois-Rips filtrations of point cloud data derived from time series, and in Section 2.1.3 we provide an overview of a common approach for vectorizing persistence diagrams into an image known as a persistence image.

2.1. Radio Frequency Modulation

In wireless communication systems, information cannot be directly transmitted through free space. Instead, radio frequency (RF) waves act as a carrier to convey the information signal. RF modulation refers to the process of imprinting the information signal onto the carrier wave. This is achieved by varying specific characteristics of the carrier wave in accordance with the information signal. The three primary characteristics of a carrier wave that can be modulated are:

- Amplitude: amplitude modulation varies the amplitude of the carrier wave in proportion to the information signal.
- Frequency: frequency modulation varies the frequency of the carrier wave in proportion to the information signal.
- Phase: phase modulation varies the phase of the carrier wave in proportion to the information signal.

The choice of modulation technique depends on factors such as the type of information being transmitted, desired signal robustness, and spectral efficiency.

As an example, On-Off Keying (OOK) is a simple form of amplitude modulation where the information signal is a binary stream of data (e.g., 0s and 1s). A logic "1" is represented by transmitting a carrier wave at a constant amplitude (carrier on), while a logic "0" is represented by completely suppressing the carrier (carrier off). Fig. 1 illustrates the concept of OOK modulation.

The figure shows a binary stream (top) and the corresponding OOK modulated RF signal (bottom). During a "1", the carrier wave is transmitted at a constant amplitude. During a "0", the carrier is completely suppressed. By analyzing the presence or absence of the carrier wave, the receiver can recover the original information signal. In this work we will attempt to distinguish between more complicated modulations that are listed in Section 4.

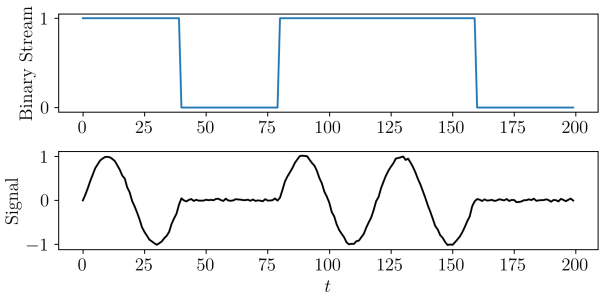

Figure 1. Example signal from OOK modulation.

2.1.1. Sublevel Set Persistence

We now provide an introduction to persistence, as it applies to computing persistence of the sublevel sets of a time series.

For time series data we assume a single variable function $f : \mathbb{R} \to \mathbb{R}$. Given $r \in \mathbb{R}$, we define the sublevel set below r as $f^{-1}(-\infty, r]$. As the filtration parameter r increases, we can tracking how the homology of the sublevel sets change. In the case of 1-D time series data these changes would only occur in D_0 since we are only interested in connected components and no changes would occur until reaching an extrema in the signal assuming the function f satisfies some general and standard conditions (e.g., q-tame (Chazal et al., 2016)).

For extrema being local minima, we add new connected component "born" at height r_B. For the case of local extrema being maxima two existing sets (or components) are combined, each of which were born at r_B and r'_B. Form this information we follow the Elder Rule (Edelsbrunner & Harer, 2010), assuming $r_B \leq r'_B$ and the maxima is at r_D, then we say that the component born at r'_B dies going into r_D and the resulting set assumes the label r_B. The pair $(r'_B, r_D) \in D_0$ is called a persistence pair in the zero-dimensional sublevel set persistence. We continue generating the persistence pairs as r increase from $-\infty$ to ∞. For the case of any unpaired births, we set a death coordinate of ∞ and label this persistence pair as an essential classes. As such, the resulting persistence diagram exists in the extended plane $\overline{\mathbb{R}}^2$.

The lifetime or persistence of a pair $(b_i, d_i) \in D_p(f)$ is defined as $\ell_i = d_i - b_i$. In this paper, our functions are only sampled on a finite domain, with the first sample at time t_a and the last sample at time t_b. We obtain a continuous function over $[t_a, t_b]$ by using a piece-wise linear interpolation between consecutive samples, and extending the function to $\pm\infty$ by extending the first (resp., last) edges to rays. Doing so allows us to define a persistence diagram that does not have critical points on the boundary of our time series. As such, we study the persistence points where both coordinates are finite, and omit persistence points that contain an unbounded coordinate. However, for demonstrative purposes we will still show the essential class persistence pair in our example in Fig. 2.

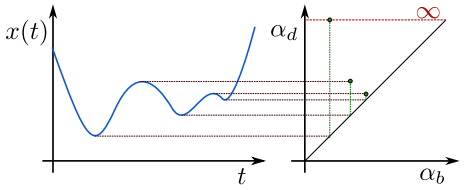

Figure 2. Demonstrative example of sublevel set persistence applied to time series data.

The example function shown in Fig. 2 has three local minima and two local maxima. The resulting persistence diagram captures how these minima and maxima are paired together through sublevel set persistence with one essential class that will be ignored when we generate our persistence images. The algorithm used to calculate the this persistence diagram is detailed in (Myers et al., 2022).

2.1.2. Vietoris-Rips Filtration

A simplicial complex is a generalization of a graph to higher dimensions, which are collections of simplices at various dimensions (e.g., points are zero-dimensional, edges are one-dimensional, and faces are two-dimensional simplices). These simplices are subsets of a vertex set $\sigma \subset V$, and we require for face closure such that if $\sigma \in K$ and $\tau \subseteq \sigma$, then τ is also in K. For data that is stored as vectors (point cloud χ) we can directly form a simplicial complex using the distance between points. For time series data we can generate point cloud data by performing Takens' embedding. Takens' embedding of a time series embeds a signal $x(t) \in \mathbb{R}$ to $\chi(t) \in \mathbb{R}^n$ by using a delay τ with $\chi(t) = [x(t), x(t+\tau), x(t+2\tau), \ldots, x(t+(n-1)\tau)]$. Applying this over the entire time series results in a point cloud χ.

To generate an abstract simplicial complex (ASC) from the point cloud, a Vietoris-Rips complex is used where we build the ASC K_r for any fixed parameter $r \geq 0$ by including all simplices with distance at most r; i.e. $K_r = \{\sigma \subseteq V \mid d(u, v) \leq r \text{ for all } u, v \in \sigma\}$, where d is a distance function. Zero-dimensional simplices, the vertices of the complex, are all added at $r = 0$. An edge uv, which is a 1-dimensional simplex, is present in K_r for any r value above $d(u, v)$. Higher dimensional simplices such as triangles are included when all subedges are present.

Generating simplicial complexes as r increases results in a sequence of ASCs that we can use to study the persistent homology of the point cloud data. Persistent homology (Zomorodian & Carlsson, 2004), a filtration tool from

the field of TDA (Edelsbrunner et al., 2002; Zomorodian & Carlsson, 2004), is used to gain a sense of the shape and size of a dataset at multiple dimensions and filtration values. For example, it can measure connected components (dimension zero), holes (dimension one), voids (dimension two), and higher dimensional analogues, as well as an idea of their general size or geometry. Persistent homology measures these shapes using a parameterized filtration to detect when homology groups are born (appear) and die (disappear). In this case the parameterization filtration uses the parameter r for point cloud data.

By incrementing r we create a nested sequence of ASCs

$$K_0 \subseteq K_1 \subseteq K_2 \subseteq \ldots \subseteq K_n. \tag{1}$$

We then calculate the homology of dimension p for each complex, $H_p(K_i)$, which is a vector space representing the p-dimensional structure of the space such as components, holes, voids, and higher dimensional features. However, this information does not yet yield how the homology of each ASC is related to the next ASC. To get this information, persistent homology uses the inclusions on the ASCs to induce linear maps on the vector spaces resulting in a construction called a persistence module $\mathcal{V}$:

$$H_p(K_{\alpha_0}) \hookrightarrow H_p(K_{\alpha_1}) \hookrightarrow H_p(K_{\alpha_2}) \hookrightarrow \ldots \hookrightarrow H_p(K_{\alpha_n}), \tag{2}$$

where $\hookrightarrow$ are the maps induced by the inclusion map between ASCs. It should be noted that in the sequence of ASCs, each vertex must be unique and consistently identified.

The appearance and disappearance of classes at various dimensions in this object can be tracked, resulting in a summary known as a persistence diagram $\mathcal{D} = \{D_0, D_1, \ldots, D_p\}$. For each homology generator which appears (born) at K_b and disappears (dies) at K_d, we add the persistence $[b, d]$ in the persistence diagram.

For example, consider Fig. 3 which shows point cloud data and the ASCs as $r \in [0.00, 0.43, 0.56, 0.83, 1.52]$. Additionally, this figure shows the corresponding persistence diagram for dimensions 0 and 1 as D_0 and D_1, respectively.

For D_0 all persistence pairs are born at 0 since these are the components which begin as all the points in the point cloud represented as vectices in the first simplicial complex. These components begin to connect where persistence pairs in D_0 die following the Elder Rule. At $r = 0.56$, all of the components are combined into one component.

For D_1 the loops or holes in the ASCs are tracked through the filtration. At $r = 0.43$ the first, smaller hole is formed as shown in the ASC. This hole persists until approximately $r = 0.83$ in which is fills in resulting in the persistence pair $[0.43, 0.83]$. The second, larger hole first forms when the point cloud is fully connected at $r = 0.56$. This hole

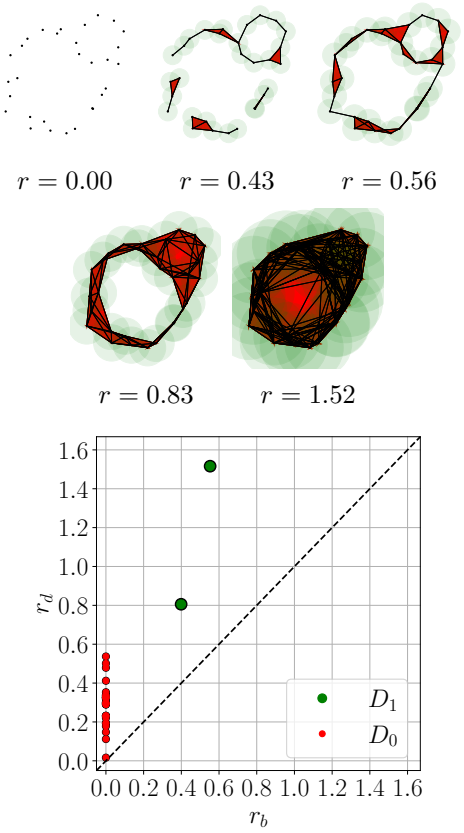

Figure 3. Example demonstrating persistent homology of point cloud data using the Vietoris-Rips complex filtration.

persists until $r = 1.52$ resulting in the persistence pair $[0.56, 1.52]$. We can see that the filtration of r results in two main persistence pairs in D_1 capturing the general shape and size of the data.

2.1.3. Persistence Images

Persistence images (PIs) (Adams et al., 2017) are a stable method for vectorizing a persistence diagram for applications in machine learning. The procedure with a toy example persistence diagram for calculating the PI for a persistence diagram D_p of dimension p is shown in Fig. 4.

The first step in the method is to get the birth-persistence representation of a persistence diagram $T : \mathbb{R}^2 \to \mathbb{R}^2$, where each persistence pair $(b_i, d_i) \in D_p$ is mapped to the pair $(b_i, d_i - b_i)$ as shown in Fig. 4 (a), where the persistence (or lifetime) is $\ell_i = d_i - b_i$. The second step is to represent $T(D_p)$ as a surface using each pair and a differentiable distribution function $g(x, y)$, which is the Gaussian kernel here or

$$g_G(x, y) = \frac{1}{2\pi\sigma^2} e^{((x--\mu_x)^2+(y-\mu_y)^2)/2\sigma^2}, \tag{3}$$

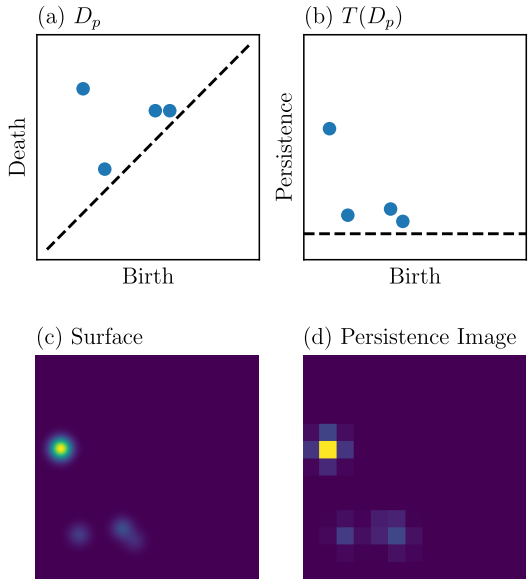

Figure 4. Persistence image pipeline.

where σ is the standard deviation and μ_x and μ_y are the mean or center point of the distribution. In this work we set $\sigma = 0.05p_{\max}$ for all experiments with $p_{\max}$ as the maximum persistence. We can now define our surface function of $T(D_p)$ as

$$S(x,y) = \sum_{(b_i,\ell_i)\in T(D_p)} w(b_i,\ell_i)g_G(x,y), \qquad (4)$$

where $\mu_x = b_i$, $\mu_y = \ell_i$ for g_G, and we use a linear weighting function on the persistence as $w(b_i,\ell_i) = \ell_i$. To practically apply the surface function as an impute to machine learning architectures we must create a finite sized representation of the surface. In this work we do this by creating a tessellation of squares (pixels) that cover the area occupied by $[b_{\min} - 3\sigma, b_{\min} + p_{\max} + 3\sigma] \times [-3\sigma, p_{\max} + 3\sigma]$ resulting in a square area, where $b_{\min}$ is the minimum birth value. These pixels have resolution $p \times p$ with the width of the square pixels w_p is chosen as $w_p = (p_{\max} + 6\sigma)/p$.

2.2. Heat Map of Takens' Embedding

Another approach to get an image from the Takens' embedding which captures the dynamics more directly is to use a heat map of the Takens' embedding. When the embedding dimension is two ($n = 2$), the Takens' embedding can be visualized as a heat map, where each embedded vector $\mathbf{x}_t \in \mathbb{R}^2$ is represented by increasing the pixel value of the corresponding location of the square cover of the image. We accomplish this by using a simple image-based approach to represent the density of points in this cover. Specifically, we divide the embedding space into a grid of pixels. For each pixel, we count the number of data points that fall within its boundaries. The color intensity of a pixel is then determined by this count. Brighter colors indicate regions with higher concentrations of data points, while darker colors represent sparser areas. This approach provides a quick visual interpretation of the data distribution in the embedding space.

3. Method

In this section we describe our method of topologically and dynamically representing time series data from RF data sources to classify and, in future work, study the data from this perspective. Specifically, we first describe how we fuse our multiple views of the signal into a single multi-dimensional image and then second how we train a Convolution Neural Network using these images.

We first introduced our method of fusing multiple topological views of a data source through our TopFusion Methodology (Myers et al., 2023a). Here we modify this procedure for the application of studying time series data by including two topological filtrations (Vietoris-Rips and sublevel set persistence) as well as the heat map of the Takens' embedding. This general pipeline is shown in Fig. 5 where there are three branches of studying the time series data that are fused together into a four-dimensional image (two dimensions from the Vietoris-Rips Filtration).

The first time series analysis branch begins with a two-dimensional Takens' embedding with the delay parameter τ chosen using the autocorrelation method (Box et al., 2015) due to modulation signals being compositions of sinusoidal functions making the linear autocorrelation method appropriate. We next create a heat map from the probability distribution of the embedding which captures the dynamics of the time series. This heat map image is chosen to be the same size as the resulting persistence images from the other branches.

The second branch takes the zero-dimensional sublevel set persistence in $O(n) = n\log(n)$ complexity with n here being the length of the signal using the algorithm provided in (Myers et al., 2020). This method captures the relative peaks and valleys (minima and maxima) in the time series data as described in Section 1. We then represent the persistence diagram as a persistence image of the same size as the heat map. In this work we exclusively used 20×20 pixel images.

The third branch follows the standard Vietoris-Rips filtration of the Takens' embedding of the time series. Again we chose the delay using the autocorrelation method and a dimension of three. We used both the zero and one dimensional persistence diagrams but only show the diagram for dimension one. We again vectorized both of these diagrams using persistence images.

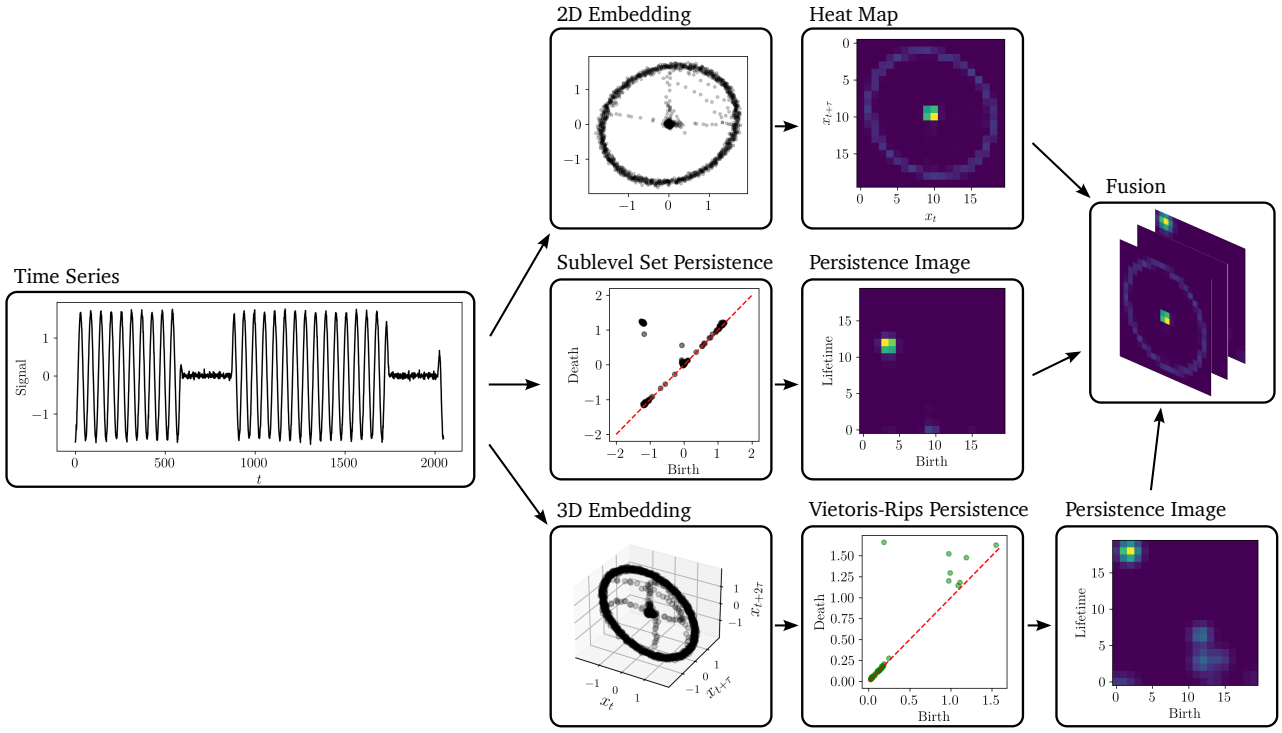

Figure 5. Method pipeline for studying RF time series using topological data analysis and fusion.

Lastly, we can combine these images into a fused image for analysis with a two-dimensional convolution neural network. Specifically, the simple CNN we used in this work has two convolutional layers, each with 5×5 filters, and each followed by maximum pooling with filter size 2×2, and one fully connected layer. For all of our experiments we trained the network for 30 epochs.

As an additional point of comparison we used spectrograms of the signal. Each spectrogram was 25×25 pixels. We constructed a spectrogram of this shape by using a sliding window over the signal with a length of a twentieth of the length of the signal with each window overlapping by 25% of the length of the window.

4. Data and Results

In this section we provide results from two benchmark datasets: Panoradio (Scholl, 2019) and RadioML (O'Shea et al., 2018). Both of these datasets are used for testing RF signal classification methods over various levels of additive Gaussian signal noise. We provide results comparing our method to the published results where we are only using a fraction of the training set and a significantly simpler convolutional neural network.

In addition to these results we also provide the training times in detail in the Appendix A.1, which shows that the overall lowest computation and training time was for both sublevel set persistence and heat maps. Additionally, the training time for the ResNet on the signal directly was the slowest.

We repeat our experiments for each dataset to test both the noise robustness and generalization of the models. For noise robustness we retrain the models at each Signal-to-Noise Ratio (SNR) and measure the classification accuracy, while for generalization we train a model using low-noise data and test at the higher-noise levels.

4.1. Panoradio

The Panoradio (Scholl, 2019) dataset is a collection of RF signals designed for training machine learning models in RF signal mode classification. The mode differs from the modulation in that it is both the modulation and baud rate or application pairs. The signals are synthetically generated signals representing 18 RF modes. A total of 120K signals were generated in this dataset for training; however, we only use 9K signals for our topological and dynamical representations. This dataset also incorporates three main impairments to simulate realistic radio signal reception conditions: Gaussian Noise to simulate the common random interference in RF signals, Watterson fading to simulate signal propagation through the ionosphere, and random frequency and phase offset simulating slight transmission shifts.

For the model retrained at each SNR we get comparable if

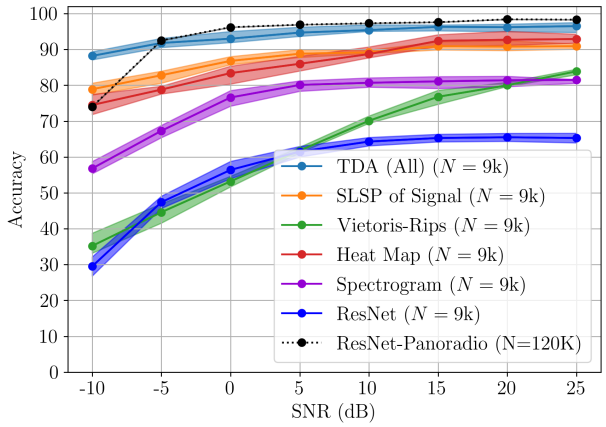

Figure 6. Classification accuracies for Panoradio dataset with models trained at each SNR from -10 to 25 dB.

not better results using our fused topological filtration and dynamical images using only 9K training signals, 500 representives of each class - only 7.5% of the available training data. We tested various combinations of our TDA feature extractors including: heat map, sublevel set persistence, and Vietoris-Rips. Additionally, we trained a ResNet described in (Ismail Fawaz et al., 2019), which was designed for time series classification, using only the 9K signals as an addition point of comparison. For all of the reported results we retrained each model 5 times providing a band of accuracy values. We did two training experiments: the first using a single model trained only on higher SNR (SNR = 25dB) data and tested on SNRs from -10 to 20 dB (see Fig. 7) as well as a new model trained and tested at each SNR (see Fig. 6). The results in (Scholl, 2019) were from retraining the model at each SNR as shown as ResNet-PanoRadio in Fig. 6.

These results show that using the fusion of the heat maps and the persistence images provides comparable results to the ResNet in (Scholl, 2019) (ResNet-PanoRadio in Fig. 6) for SNR values between -5 and 25 dB. However, at -10 dB we see that the our approach begins to outperform and is more noise robust than the ResNet. We also note that the Vietoris-Rips persistence image performs the worst compared to the sublevel set persistence and the heat map. We believe this sensitivity to noise is due to the additive noise having a more serious affect on the topology than for methods such as sublevel set persistence which still capture the structure of the signal even with significant amounts of noise. As such, we believe that using the combination of the sublevel set persistence and heat map provide enough information to accurately classify the RF modes with limited data. As an additional point of comparison, we also trained our CNN on spectrograms of the signals as this is a common method for RF signal analysis. Our results show that the spectrogram performs worse than both the heat map and sublevel set persistence for all SNRs.

We additionally ran a second experiment testing the performance of the CNN when a single model is trained at 25 dB SNR and then tested at the lower SNRs as shown in Fig. 7. These results show a faster loss in accuracy as the SNR decreases. We believe this is due to the learned topological and dynamical features not being significantly present or not at the same scale when the SNR is increased. However, we see the same relative performance between the various topological and dynamical data representations with sublevel set persistence and the heat map representations performing the best of the individual methods.

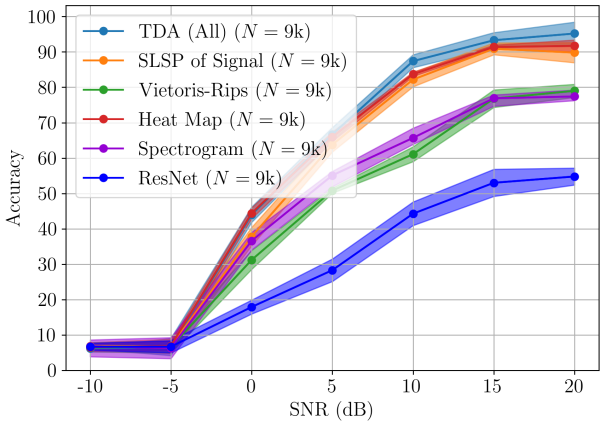

Figure 7. Classification accuracies for Panoradio dataset with model trained at 25 dB SNR and tested on SNR from -10 to 20 dB.

4.2. RadioML

The RadioML dataset (O'Shea et al., 2018) is a modulation classification dataset with 24 modulation schemes (classes). This dataset also has the same three different signal impairments that are found in the Panoradio dataset.

The original results (O'Shea et al., 2018) we compare to were trained using 240k (10k of each modulation type) signals. In comparison, for our various filtration combinations we only used 4.8k signals (200 of each modulation type) - 2% of the avaiable training data.

Our first experiment results comparing the classification accuracy against noise level are shown in Fig. 8, where we trained a single model at 30 dB SNR and tested at -20 to 20 dB SNR to follow the same training procedure as was published in (O'Shea et al., 2018).

These results show similar performance as was seen in the Panoradio dataset. Specifically, for higher SNR values (from 14 to 20 dB), the fused images perform comparable to the ResNet even with significantly less training data. Additionally, the Vietoris-Rips filtration does not perform well in

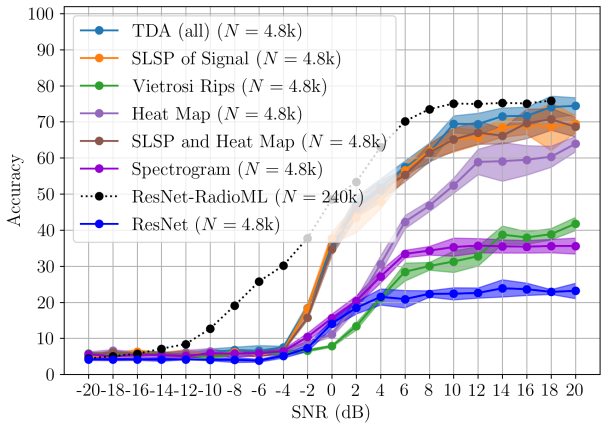

Figure 8. Classification accuracies for RadioML (2018) dataset for model trained at 30 dB SNR.

comparison to the other images. This strengthens the argument that just the heat map and sublevel set persistence are suitable for capturing the information needed to analyze RF signals. One key difference in this experiment is that the ResNet is more noise robust than the fused methods which is in disagreement with the findings using the Panoradio dataset.

Our second experiment for this dataset trained a new model at each SNR. Our results are shown in Fig. 9. We again see comparable results to that in our first experiment with the change of having the models be more noise robust than for the single model experiment. We hypothesise this is due to the significance of the dynamical and topological features being learned at each SNR in comparison to a single model not capturing the decreasing feature significance as the SNR decreases.

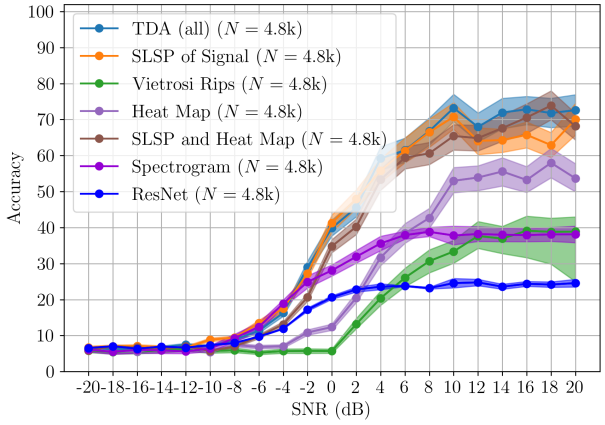

Figure 9. Classification accuracies for RadioML (2018) dataset for model trained at each SNR.

5. Conclusion

In this work we demonstrated that fusing both topological and dynamical information generated from time series data is useful for studying RF signal data. Specifically, we show comparable classification accuracy for both mode and modulations using two benchmark datasets. We also showed that these results could be achieved with a significantly smaller neural network (compared to the standard ResNet) and with a fraction of the required training data.The reduced demands on training data abundance suggest greater adaptability to real world scenarios that may include evolving classes and the need for efficient retraining.

In future work we plan to apply other time series analysis methods that can be vectorized into images. This will provide greater understanding of which time series analysis methods are most beneficial for studying RF signals. Additionally, we plan to study the effect of other signal interferences on classification accuracies to understand the limitations of the method.

References

Adams, H., Emerson, T., Kirby, M., Neville, R., Peterson, C., Shipman, P., Chepushtanova, S., Hanson, E., Motta, F., and Ziegelmeier, L. Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.

Adcock, A., Carlsson, E., and Carlsson, G. The ring of algebraic functions on persistence bar codes. *arXiv preprint arXiv:1304.0530*, 2013.

Barnes, D., Polanco, L., and Perea, J. A. A comparative study of machine learning methods for persistence diagrams. *Frontiers in Artificial Intelligence*, 4:681174, 2021.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

Bubenik, P. et al. Statistical topological data analysis using persistence landscapes. *J. Mach. Learn. Res.*, 16(1):77–102, 2015.

Carlsson, G. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009.

Chazal, F., De Silva, V., Glisse, M., and Oudot, S. *The Structure and Stability of Persistence Modules*. Springer, 2016.

Edelsbrunner, Letscher, and Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002. ISSN 0179-5376. doi:

10.1007/s00454-002-2885-2. URL http://dx.doi.org/10.1007/s00454-002-2885-2.

Edelsbrunner, H. and Harer, J. Persistent homology-a survey. *Contemporary mathematics*, 453:257–282, 2008.

Edelsbrunner, H. and Harer, J. *Computational Topology - an Introduction.* American Mathematical Society, 2010. ISBN 978-0-8218-4925-5.

Edelsbrunner, H. and Harer, J. L. *Computational topology: an introduction*. American Mathematical Society, 2022.

Guralnik, V. and Srivastava, J. Event detection from time series data. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 33–42, 1999.

Islambekov, U., Yuvaraj, M., and Gel, Y. R. Harnessing the power of topological data analysis to detect change points. *Environmetrics*, 31(1):e2612, 2020.

Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., and Muller, P.-A. Deep learning for time series classification: a review. *Data mining and knowledge discovery*, 33 (4):917–963, 2019.

Karan, A. and Kaygun, A. Time series classification via topological data analysis. *Expert Systems with Applications*, 183:115326, 2021.

Majumdar, S. and Laha, A. K. Clustering and classification of time series using topological data analysis with applications to finance. *Expert Systems with Applications*, 162: 113868, 2020.

Munch, E. A user's guide to topological data analysis. *Journal of Learning Analytics*, 4(2):47–61, July 2017. doi: 10.18608/jla.2017.42.6. URL https://doi.org/10.18608/jla.2017.42.6.

Myers, A., Kvinge, H., and Emerson, T. Topfusion: Using topological feature space for fusion and imputation in multi-modal data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 600–609, 2023a.

Myers, A., Muñoz, D., Khasawneh, F. A., and Munch, E. Temporal network analysis using zigzag persistence. *EPJ Data Science*, 12(1):6, 2023b.

Myers, A. D., Khasawneh, F. A., and Fasy, B. T. Anapt: Additive noise analysis for persistence thresholding. *arXiv preprint arXiv:2012.04039*, 2020.

Myers, A. D., Khasawneh, F. A., and Fasy, B. T. Anapt: Additive noise analysis for persistence thresholding. *Foundations of Data Science*, December 2022.

O'Shea, T. J., Roy, T., and Clancy, T. C. Over-the-air deep learning based radio signal classification. *IEEE Journal of Selected Topics in Signal Processing*, 12(1):168–179, 2018.

Perea, J. A. A brief history of persistence. September 2018.

Perea, J. A., Munch, E., and Khasawneh, F. A. Approximating continuous functions on persistence diagrams using template functions. *Foundations of Computational Mathematics*, 23(4):1215–1272, 2023.

Ravishanker, N. and Chen, R. Topological data analysis (tda) for time series. *arXiv preprint arXiv:1909.10604*, 2019.

Scholl, S. Classification of radio signals and hf transmission modes with deep learning. *arXiv preprint arXiv:1906.04459*, 2019.

Takens, F. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980: proceedings of a symposium held at the University of Warwick 1979/80*, pp. 366–381. Springer, 2006.

Umeda, Y. Time series classification via topological data analysis. *Information and Media Technologies*, 12:228–239, 2017.

Umeda, Y., Kaneko, J., and Kikuchi, H. Topological data analysis and its application to time-series data analysis. *Fujitsu Scientific & Technical Journal*, 55(2):65–71, 2019.

Zheng, X., Mak, S., Xie, L., and Xie, Y. Percept: a new online change-point detection method using topological data analysis. *Technometrics*, 65(2):162–178, 2023.

Zomorodian, A. and Carlsson, G. Computing persistent homology. *Discrete & Computational Geometry*, 33(2): 249–274, November 2004. ISSN 0179-5376. doi: 10.1007/s00454-004-1146-y. URL https://doi.org/10.1007/s00454-004-1146-y.

A. Appendix

A.1. Computational Time

To best understand which topological and/or dynamical data analysis approaches are most suitable for RF signal processing we also wanted to compare the computation time. For all of the computations we used a personal computer and trained the CNN using CPUs. We broke the computation times into two steps: pre-processing time which is the time to apply the topological or dynamical signal analysis methods and the training time. These results are shown in Table 1

For both experiments the computation time comparing methods are proportional with the VR filtration taking the most preprocessing time of all the methods we studied. The heat map and the sublevel set persistence methods were the two fastest preprocessing methods excluding the ResNet trained directly on the signal as this had no preprocessing. For the overall time, again, both the heat map and sublevel set persistence methods show a fast processing and training time. Additionally, these approaches showed close to or the highest accuracies for our noise sensitivity experiments.

Table 1. Computation time (personal computer trained using CPU) comparison between topological, dynamical, and time series approaches for data processing and training.

Dataset	**Model**	**Data**	**Processing Time (Minutes)**	**Training Time (Minutes)**	**Total Time (Minutes)**
Panoradio ($N = 9$k)	2D CNN	TDA (All)	32.4	2.6	35.0
		SLSP	9.6	2.2	11.8
		VR	22.8	2.3	25.1
		HM	1.6	1.4	3.0
	ResNet	Signal	0	36.3	36.3
RadioML ($N = 4.8$k)	2D CNN	TDA (All)	13.6	0.6	14.2
		SLSP	2.8	0.4	3.2
		VR	10.8	0.4	11.2
		HM	0.4	0.3	0.7
	ResNet	Signal	0	27.4	27.4